dsPIC33EPXXXGS70X/80X FAMILY

16-Bit Digital Signal Controllers for Digital Power Applications with Interconnected High-Speed PWM, ADC, PGA and Comparators

Operating Conditions

- 3.0V to 3.6V, -40°C to +85°C, DC to 70 MIPS
- 3.0V to 3.6V, -40°C to +125°C, DC to 60 MIPS

Flash Architecture

- Dual Partition Flash Program Memory with Live Update:
  - Supports programming while operating
  - Supports partition soft swap

Core: 16-Bit dsPIC33E CPU

- Code-Efficient (C and Assembly) Architecture
- Two 40-Bit Wide Accumulators
- Single-Cycle (MAC/MPY) with Dual Data Fetch
- Single-Cycle Mixed-Sign MUL plus Hardware Divide
- 32-Bit Multiply Support
- Four Additional Working Register Sets (reduces context switching)

Clock Management

- ±0.9% Internal Oscillator
- Programmable PLLs and Oscillator Clock Sources
- Fail-Safe Clock Monitor (FSCM)
- Independent Watchdog Timer (WDT)
- Fast Wake-up and Start-up

Power Management

- Low-Power Management modes (Sleep, Idle, Doze)
- Integrated Power-on Reset and Brown-out Reset
- 0.5 mA/MHz Dynamic Current (typical)
- 20 μA I_{PD} Current (typical)

High-Speed PWM

- Eight PWM Generators (two outputs per generator)
- Individual Time Base and Duty Cycle for each PWM
- 1.04 ns PWM Resolution (frequency, duty cycle, dead time and phase)
- Supports Center-Aligned, Redundant, Complementary and True Independent Output modes
- Independent Fault and Current-Limit Inputs
- Output Override Control
- PWM Support for AC/DC, DC/DC, Inverters, PFC and Lighting

Advanced Analog Features

- High-Speed ADC module:
  - 12-bit with 4 dedicated SAR ADC cores and one shared SAR ADC core
  - Configurable resolution (up to 12-bit) for each ADC core
  - Up to 3.25 Msps conversion rate per channel at 12-bit resolution
  - 11 to 22 single-ended inputs
  - Dedicated result buffer for each analog channel
  - Flexible and independent ADC trigger sources
  - Two digital comparators
  - Two oversampling filters for increased resolution
- Four Rail-to-Rail Comparators with Hysteresis:
  - Dedicated 12-bit Digital-to-Analog Converter (DAC) for each analog comparator
  - Up to two DAC reference outputs
  - Up to two external reference inputs
- Two Programmable Gain Amplifiers:
  - Single-ended or independent ground reference
  - Five selectable gains (4x, 8x, 16x, 32x and 64x)
  - 40 MHz gain bandwidth

Interconnected SMPS Peripherals

- Reduces CPU Interaction to Improve Performance
- Flexible PWM Trigger Options for ADC Conversions
- High-Speed Comparator Truncates PWM (15 ns typical):
  - Supports Cycle-by-Cycle Current mode control
  - Current Reset mode (variable frequency)

Timers/Output Compare/Input Capture

- Five 16-Bit and up to Two 32-Bit Timers/Counters
- Four Output Compare (OC) modules, Configurable as Timers/Counters
- Four Input Capture (IC) modules

Communication Interfaces

- Two UART modules (15 Mbps):
  - Supports LIN/J2602 protocols and IrDA®
- Three Variable Width SPI modules with Operating modes:
  - 3-wire SPI
  - 8x16 or 8x8 FIFO mode
  - I^2S mode
- Two I^2C modules (up to 1 Mbaud) with SMBus Support
- Up to Two CAN modules
- Four-Channel DMA

Input/Output

- Constant-Current Source (10 µA nominal)
- Sink/Source up to 12 mA/15 mA, respectively; Pin-Specific for Standard VOH/VOL
- 5V Tolerant Pins
- Selectable, Open-Drain Pull-ups and Pull-Downs
- External Interrupts on all I/O Pins
- Peripheral Pin Select (PPS) to allow Function Remap with Six Virtual I/Os

Qualification and Class B Support

- AEC-Q100 REVG (Grade 1, -40°C to +125°C)
- Class B Safety Library, IEC 60730
- The 6x6x0.55 mm UQFN Package is Designed and Optimized to ease IPC9592B 2nd Level Temperature Cycle Qualification

Debugger Development Support

- In-Circuit and In-Application Programming
- Five Program and Three Complex Data Breakpoints
- IEEE 1149.2 Compatible (JTAG) Boundary Scan
- Trace and Run-Time Watch

Digital Peripherals

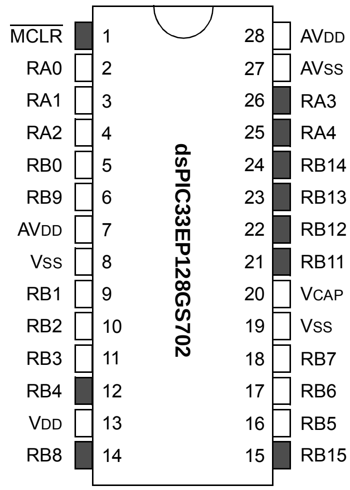
- Four Configurable Logic Cells
- Peripheral Trigger Generator

Device	Pins	Program Memory Bytes	RAM (Bytes)	General Purpose I/O (GPIO)	Remappable Peripherals									I^2C	CLC	PTG	12-Bit ADC		PGA	DMA	Analog Comparator	DAC Output	Constant-Current Source	Packages
					Timers(1)	Input Capture	Output Compare	UART	SPI	PWM(2)	External Interrupts(3)	CAN	Reference Clock				Analog Inputs	S&H Circuits						
dsPIC33EP128GS702	28	128K	8K	20	5	4	4	2	3	8x2	4	0	1	2	4	1	11	5	2	0	4	1	1	SOIC, QFN-S, UQFN
dsPIC33EP64GS804	44	64K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	QFN, TQFP
dsPIC33EP128GS704	44	128K	8K	33	5	4	4	2	3	8x2	4	0	1	2	4	1	17	5	2	0	4	1	1	
dsPIC33EP128GS804	44	128K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	
dsPIC33EP64GS805	48	64K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	TQFP
dsPIC33EP128GS705	48	128K	8K	33	5	4	4	2	3	8x2	4	0	1	2	4	1	17	5	2	0	4	1	1	
dsPIC33EP128GS805	48	128K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	
dsPIC33EP64GS806	64	64K	8K	51	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	TQFP
dsPIC33EP128GS706	64	128K	8K	51	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	
dsPIC33EP128GS806	64	128K	8K	51	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	
dsPIC33EP64GS708	80	64K	8K	67	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	TQFP
dsPIC33EP64GS808	80	64K	8K	67	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	
dsPIC33EP128GS708	80	128K	8K	67	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	
dsPIC33EP128GS808	80	128K	8K	67	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	

Note 1: The external clock for Timer1, Timer2 and Timer3 is remappable.
2: PWM4 through PWM8 are remappable on 28/44/48-pin devices; on 64-pin devices, only PWM7/PWM8 are remappable.
3: External interrupts, INT0 and INT4, are not remappable.

Pin Diagrams

28-Pin SOIC

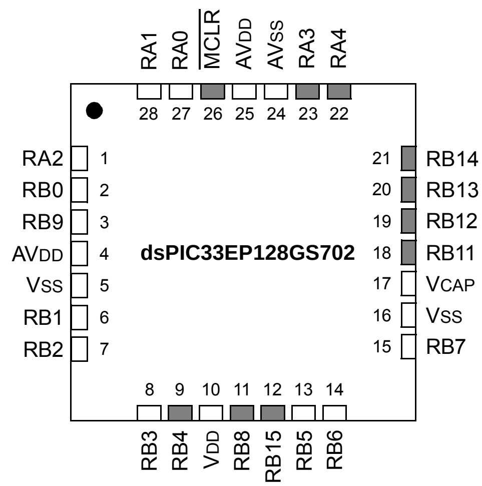


Pin	Pin Function	Pin	Pin Function
1	MCLR	15	PGEC3/SCL2/**RP47**/RB15
2	AN0/CMP1A/PGA1P1/**RP16**/RA0	16	TDO/AN19/PGA2N2/**RP37**/RB5
3	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1	17	PGED1/TDI/AN20/SCL1/**RP38**/RB6
4	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2	18	PGEC1/AN21/SDA1/**RP39**/RB7
5	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0	19	VSS
6	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9	20	VCAP
7	AVDD	21	TMS/PWM3H/**RP43**/RB11
8	VSS	22	TCK/PWM3L/**RP44**/RB12
9	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1	23	PWM2H/**RP45**/RB13
10	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2	24	PWM2L/**RP46**/RB14
11	PGED2/DACOUT1/AN18/INT0/**RP35**/RB3	25	PWM1H/**RP20**/RA4
12	PGEC2/ADTRG31/EXTREF1/**RP36**/RB4	26	PWM1L/**RP19**/RA3
13	VDD	27	AVSS
14	PGED3/SDA2/FLT31/**RP40**/RB8	28	AVDD

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Pin Diagrams (Continued)

28-Pin QFN-S, UQFN

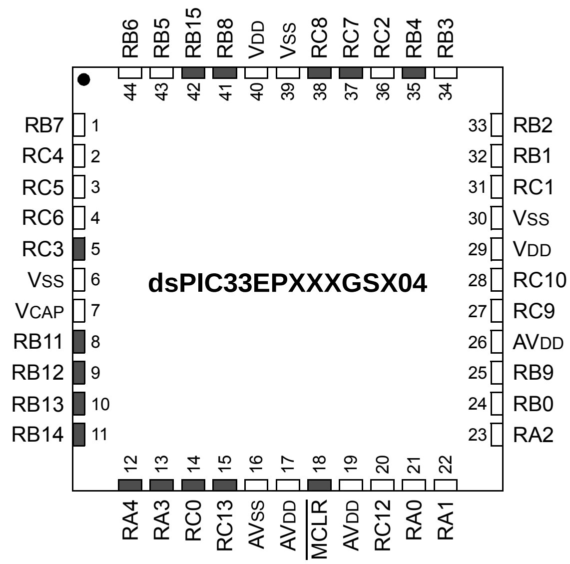


Pin	Pin Function	Pin	Pin Function
1	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2	15	PGEC1/AN21/SDA1/**RP39**/RB7
2	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0	16	VSS
3	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9	17	VCAP
4	AVDD	18	TMS/PWM3H/**RP46**/RB11
5	VSS	19	TCK/PWM3L/**RP44**/RB12
6	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1	20	PWM2H/**RP45**/RB13
7	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2	21	PWM2L/**RP46**/RB14
8	PGED2/DACOUT1/AN18/INT0/**RP35**/RB3	22	PWM1H/**RP20**/RA4
9	PGEC2/ADTRG31/EXTREF1/**RP36**/RB4	23	PWM1L/**RP19**/RA3
10	VDD	24	AVSS
11	PGED3/SDA2/FLT31/**RP40**/RB8	25	AVDD
12	PGEC3/SCL2/**RP47**/RB15	26	MCLR
13	TDO/AN19/PGA2N2/**RP37**/RB5	27	AN0/CMP1A/PGA1P1/**RP16**/RA0
14	PGED1/TDI/AN20/SCL1/**RP38**/RB6	28	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Pin Diagrams (Continued)

44-Pin QFN, TQFP

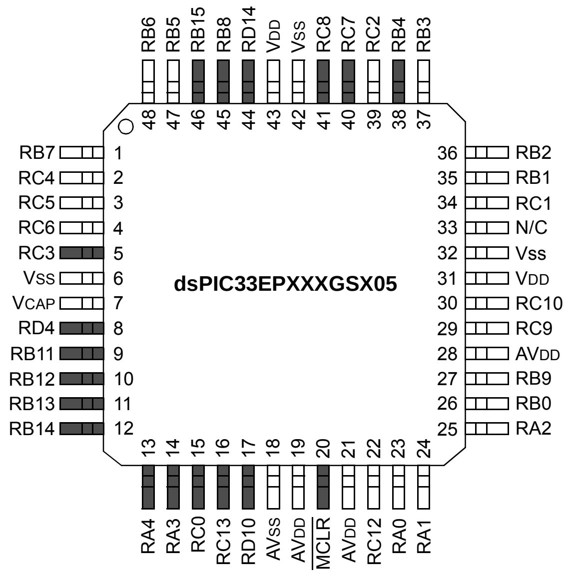


Pin	Pin Function	Pin	Pin Function
1	PGEC1/AN21/SDA1/**RP39**/RB7	23	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2
2	AN1ALT/**RP52**/RC4	24	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0
3	AN0ALT/**RP53**/RC5	25	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9
4	AN17/**RP54**/RC6	26	AVDD
5	**RP51**/RC3	27	AN11/PGA1N3/**RP57**/RC9
6	VSS	28	EXTREF2/AN10/PGA1P4/**RP58**/RC10
7	VCAP	29	VDD
8	TMS/PWM3H/**RP43**/RB11	30	VSS
9	TCK/PWM3L/**RP44**/RB12	31	AN8/CMP4C/PGA2P4/**RP49**/RC1
10	PWM2H/**RP45**/RB13	32	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1
11	PWM2L/**RP46**/RB14	33	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2
12	PWM1H/**RP20**/RA4	34	PGED2/DACOUT1/AN18/INT0/**RP35**/RB3
13	PWM1L/**RP19**/RA3	35	PGEC2/ADTRG31/**RP36**/RB4
14	FLT12/**RP48**/RC0	36	EXTREF1/AN9/CMP4D/**RP50**/RC2
15	FLT11/**RP61**/RC13	37	ASDA1/**RP55**/RC7
16	AVSS	38	ASCL1/**RP56**/RC8
17	AVDD	39	VSS
18	$\overline{\text{MCLR}}$	40	VDD
19	AVDD	41	PGED3/SDA2/FLT31/**RP40**/RB8
20	AN14/PGA2N3/**RP60**/RC12	42	PGEC3/SCL2/**RP47**/RB15
21	AN0/CMP1A/PGA1P1/**RP16**/RA0	43	TDO/AN19/PGA2N2/**RP37**/RB5
22	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1	44	PGED1/TDI/AN20/SCL1/**RP38**/RB6

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Pin Diagrams (Continued)

48-Pin TQFP

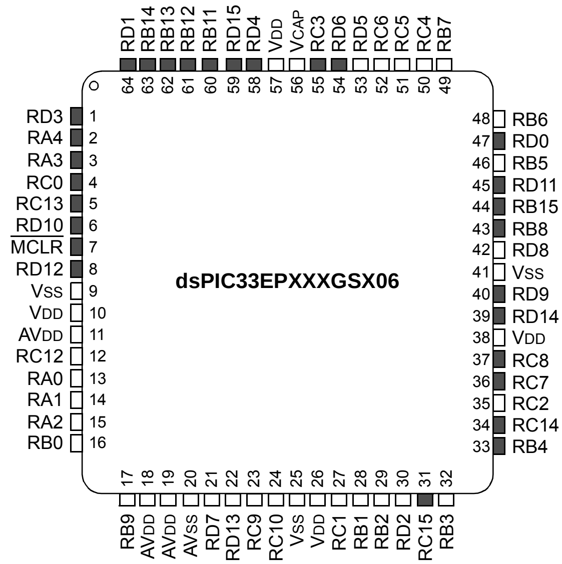


Pin	Pin Function	Pin	Pin Function
1	PGEC1/AN21/SDA1/**RP39**/RB7	25	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2
2	AN1ALT/**RP52**/RC4	26	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0
3	AN0ALT/**RP53**/RC5	27	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9
4	AN17/**RP54**/RC6	28	AVDD
5	**RP51**/RC3	29	AN11/PGA1N3/**RP57**/RC9
6	VSS	30	EXTREF2/AN10/PGA1P4/**RP58**/RC10
7	VCAP	31	VDD
8	**RP68**/RD4	32	VSS
9	TMS/PWM3H/**RP43**/RB11	33	N/C
10	TCK/PWM3L/**RP44**/RB12	34	AN8/CMP4C/PGA2P4/**RP49**/RC1
11	PWM2H/**RP45**/RB13	35	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1
12	PWM2L/**RP46**/RB14	36	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2
13	PWM1H/**RP20**/RA4	37	PGED2/DACOUT1/AN18/INT0/**RP35**/RB3
14	PWM1L/**RP19**/RA3	38	PGEC2/ADTRG31/**RP36**/RB4
15	FLT12/**RP48**/RC0	39	EXTREF1/AN9/CMP4D/**RP50**/RC2
16	FLT11/**RP61**/RC13	40	ASDA1/**RP55**/RC7
17	CLC4OUT/FLT10/**RP74**/RD10	41	ASCL1/**RP56**/RC8
18	AVSS	42	VSS
19	AVDD	43	VDD
20	MCLR	44	CLC3OUT/RD14
21	AVDD	45	PGED3/SDA2/FLT31/**RP40**/RB8
22	AN14/PGA2N3/**RP60**/RC12	46	PGEC3/SCL2/**RP47**/RB15
23	AN0/CMP1A/PGA1P1/**RP16**/RA0	47	TDO/AN19/PGA2N2/**RP37**/RB5
24	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1	48	PGED1/TDI/AN20/SCL1/**RP38**/RB6

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Pin Diagrams (Continued)

64-Pin TQFP

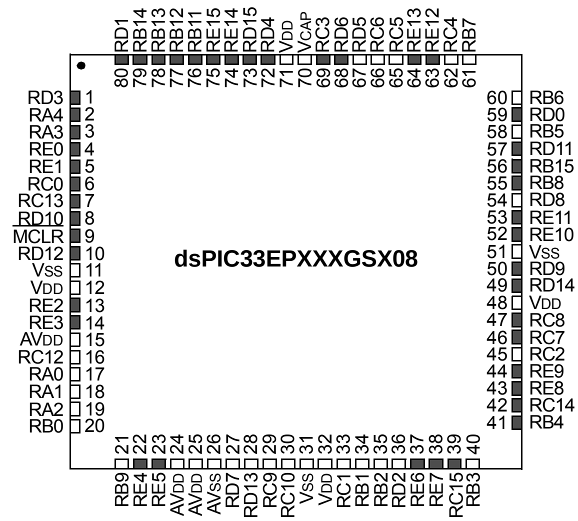


Pin	Pin Function	Pin	Pin Function
1	PWM4L/**RP67**/RD3	33	PGEC2/ADTRG31/**RP36**/RB4
2	PWM1H/**RP20**/RA4	34	**RP62**/RC14
3	PWM1L/**RP19**/RA3	35	EXTREF1/AN9/CMP4D/**RP50**/RC2
4	FLT12/**RP48**/RC0	36	ASDA1/**RP55**/RC7
5	FLT11/**RP61**/RC13	37	ASCL1/**RP56**/RC8
6	CLC4OUT/FLT10/**RP74**/RD10	38	VDD
7	MCLR	39	CLC3OUT/RD14
8	T5CK/FLT9/**RP76**/RD12	40	SCK3/**RP73**/RD9
9	VSS	41	VSS
10	VDD	42	AN5/CMP2D/CMP3B/ISRC3/**RP72**RD8
11	AVDD	43	PGED3/SDA2/FLT31/**RP40**/RB8
12	AN14/PGA2N3/**RP60**/RC12	44	PGEC3/SCL2/**RP47**/RB15
13	AN0/CMP1A/PGA1P1/**RP16**/RA0	45	INT4/**RP75**/RD11
14	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1	46	TDO/AN19/PGA2N2/**RP37**/RB5
15	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2	47	T4CK/**RP64**/RD0
16	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0	48	PGED1/TDI/AN20/SCL1/**RP38**/RB6
17	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9	49	PGEC1/AN21/SDA1/**RP39**/RB7
18	AVDD	50	AN1ALT/**RP52**/RC4
19	AVDD	51	AN0ALT/**RP53**/RC5
20	AVSS	52	AN17/**RP54**/RC6
21	AN15/**RP71**/RD7	53	AN12/ISRC1/**RP69**/RD5
22	DACOUT2/AN13/RD13	54	PWM5H/**RP70**/RD6
23	AN11/PGA1N3/**RP57**/RC9	55	PWM5L/**RP51**/RC3
24	EXTREF2/AN10/PGA1P4/**RP58**/RC10	56	VCAP
25	VSS	57	VDD
26	VDD	58	PWM6H/**RP68**/RD4
27	AN8/CMP4C/PGA2P4/**RP49**/RC1	59	PWM6L/RD15
28	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1	60	TMS/PWM3H/**RP43**/RB11
29	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2	61	TCK/PWM3L/**RP44**/RB12
30	AN16/**RP66**/RD2	62	PWM2H/**RP45**/RB13
31	ASDA2/**RP63**/RC15	63	PWM2L/**RP46**/RB14
32	PGED2/DACOUT1/AN18/ASCL2/INT0/**RP35**/RB3	64	PWM4H/**RP65**/RD1

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Pin Diagrams (Continued)

80-Pin TQFP



Pin	Pin Function	Pin	Pin Function
1	PWM4L/**RP67**/RD3	41	PGEC2/ADTRG31/**RP36**/RB4
2	PWM1H/**RP20**/RA4	42	**RP62**/RC14
3	PWM1L/**RP19**/RA3	43	RE8
4	PWM8L/RE0	44	RE9
5	PWM8H/RE1	45	EXTREF1/AN9/CMP4D/**RP50**/RC2
6	FLT12/**RP48**/RC0	46	ASDA1/**RP55**/RC7
7	FLT11/**RP61**/RC13	47	ASCL1/**RP56**/RC8
8	CLC4OUT/FLT10/**RP74**/RD10	48	VDD
9	MCLR	49	CLC3OUT/RD14
10	T5CK/FLT9/**RP76**/RD12	50	SCK3/**RP73**/RD9
11	VSS	51	VSS
12	VDD	52	FLT21/RE10
13	FLT17/RE2	53	FLT22/RE11
14	FLT18/RE3	54	AN5/CMP2D/CMP3B/ISRC3/**RP72**/RD8
15	AVDD	55	PGED3/SDA2/FLT31/**RP40**/RB8
16	AN14/PGA2N3/**RP60**/RC12	56	PGEC3/SCL2/**RP47**/RB15
17	AN0/CMP1A/PGA1P1/**RP16**/RA0	57	INT4/**RP75**/RD11
18	AN1/CMP1B/PGA1P2/PGA2P1/**RP17**/RA1	58	TD0/AN19/PGA2N2/**RP37**/RB5
19	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/**RP18**/RA2	59	T4CK/**RP64**/RD0
20	AN3/CMP1D/CMP2B/PGA2P3/**RP32**/RB0	60	PGED1/TDI/AN20/SCL1/**RP38**/RB6
21	AN4/CMP2C/CMP3A/ISRC4/**RP41**/RB9	61	PGEC1/AN21/SDA1/**RP39**/RB7
22	RE4	62	AN1ALT/**RP52**/RC4
23	RE5	63	RE12
24	AVDD	64	RE13
25	AVDD	65	AN0ALT/**RP53**/RC5
26	AVSS	66	AN17/**RP54**/RC6
27	AN15/**RP71**/RD7	67	AN12/ISRC1/**RP69**/RD5
28	DACOUT2/AN13/RD13	68	PWM5H/**RP70**/RD6
29	AN11/PGA1N3/**RP57**/RC9	69	PWM5L/**RP51**/RC3
30	EXTREF2/AN10/PGA1P4/**RP58**/RC10	70	VCAP
31	VSS	71	VDD
32	VDD	72	PWM6H/**RP68**/RD4
33	AN8/CMP4C/PGA2P4/**RP49**/RC1	73	PWM6L/RD15
34	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/**RP33**/RB1	74	PWM7L/RE14
35	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/**RP34**/RB2	75	PWM7H/RE15
36	AN16/**RP66**/RD2	76	TMS/PWM3H/**RP43**/RB11
37	FLT19/RE6	77	TCK/PWM3L/**RP44**/RB12
38	FLT20/RE7	78	PWM2H/**RP45**/RB13
39	ASDA2/**RP63**/RC15	79	PWM2L/**RP46**/RB14
40	PGED2/DACOUT1/AN18/ASCL2/INT0/**RP35**/RB3	80	PWM4H/**RP65**/RD1

Legend: **Shaded pins** are up to 5 VDC tolerant.
RPn represents remappable peripheral functions. See Table 11-12 and Table 11-13 for the complete list of remappable sources.

Table of Contents

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

1.0 DEVICE OVERVIEW

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive resource. To complement the information in this data sheet, refer to the related section of the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

This document contains device-specific information for the dsPIC33EPXXXGS70X/80X Digital Signal Controller (DSC) devices.

dsPIC33EPXXXGS70X/80X devices contain extensive Digital Signal Processor (DSP) functionality with a high-performance, 16-bit MCU architecture.

Figure 1-1 shows a general block diagram of the core and peripheral modules. Table 1-1 lists the functions of the various pins shown in the pinout diagrams.

FIGURE 1-1: dsPIC33EPXXXGS70X/80X FAMILY BLOCK DIAGRAM

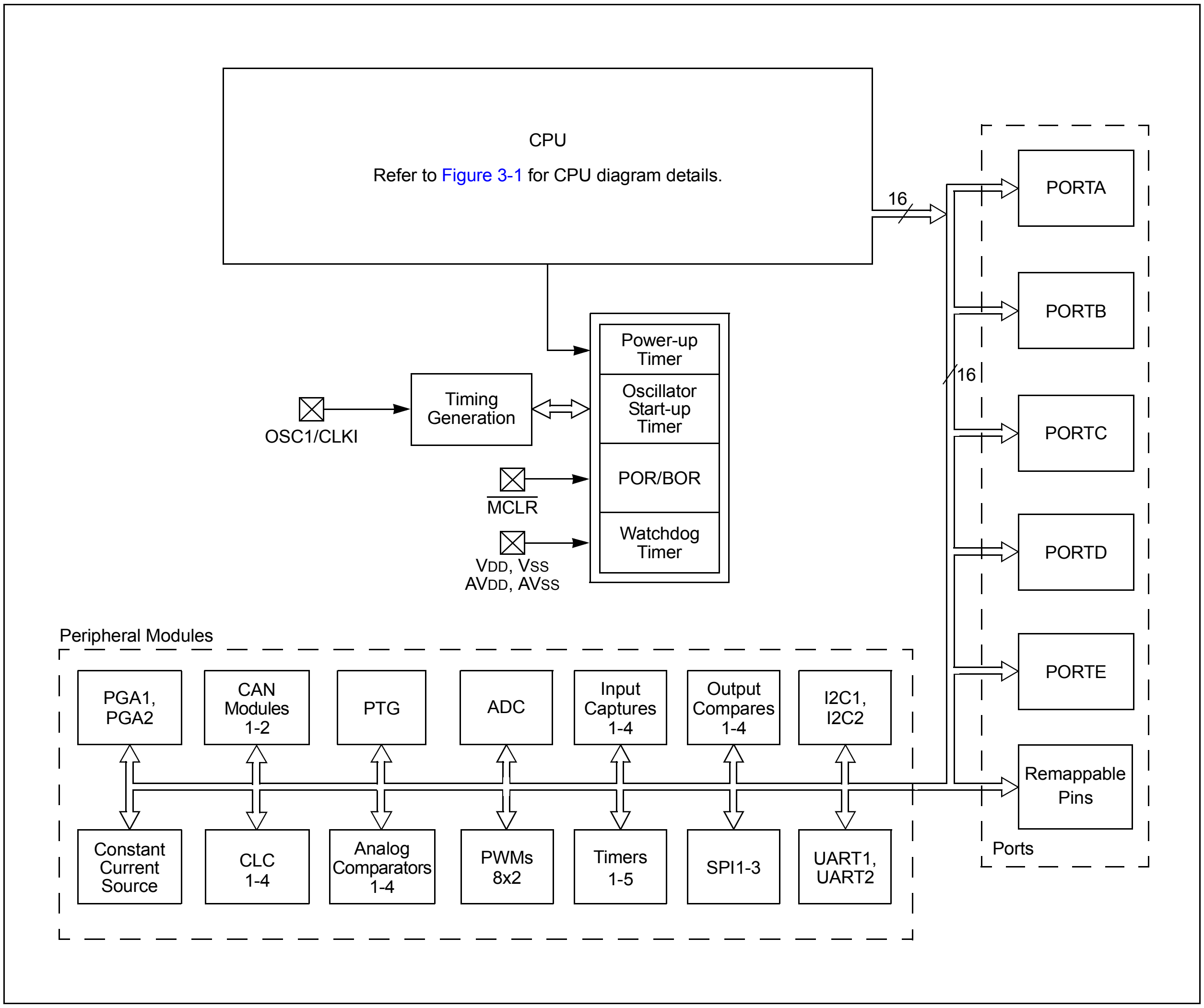


TABLE 1-1: PINOUT I/O DESCRIPTIONS

Pin Name(1)	Pin Type	Buffer Type	PPS	Description
AN0-AN21 AN0ALT-AN1ALT	I I	Analog Analog	No No	Analog input channels. Alternate analog input channels.
C1RXR C2RXR C1TX C2TX	I I O O	ST ST ST ST	Yes Yes Yes Yes	CAN1 receive. CAN2 receive. CAN1 transmit. CAN2 transmit.
CLKI CLKO	I O	ST/ CMOS —	No No	External clock source input. Always associated with OSC1 pin function. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. Always associated with OSC2 pin function.
OSC1 OSC2	I I/O	ST/ CMOS —	No No	Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes.
CLC1OUT CLC2OUT CLC3OUT CLC4OUT	O O O O	DIG DIG DIG DIG	Yes Yes No(4) No(4)	CLC1 output. CLC2 output. CLC3 output. CLC4 output.
REFCLKO	O	—	Yes	Reference clock output.
IC1-IC4	I	ST	Yes	Capture Inputs 1 through 4.
OCFA OC1-OC4	I O	ST —	Yes Yes	Compare Fault A input (for compare channels). Compare Outputs 1 through 4.
INT0 INT1 INT2 INT4	I I I I	ST ST ST ST	No Yes Yes Yes	External Interrupt 0. External Interrupt 1. External Interrupt 2. External Interrupt 4.
RA0-RA4	I/O	ST	No	PORTA is a bidirectional I/O port.
RB0-RB15	I/O	ST	No	PORTB is a bidirectional I/O port.
RC0-RC15	I/O	ST	No	PORTC is a bidirectional I/O port.
RD0-RD15	I/O	ST	No	PORTD is a bidirectional I/O port.
RE0-RE15	I/O	ST	No	PORTE is a bidirectional I/O port.
T1CK T2CK T3CK T4CK T5CK	I I I I I	ST ST ST ST ST	Yes Yes Yes No No	Timer1 external clock input. Timer2 external clock input. Timer3 external clock input. Timer4 external clock input. Timer5 external clock input.
$\overline{\text{U1CTS}}$ $\overline{\text{U1RTS}}$ U1RX U1TX BCLK1	I O I O O	ST — ST — ST	Yes Yes Yes Yes Yes	UART1 Clear-to-Send. UART1 Ready-to-Send. UART1 receive. UART1 transmit. UART1 IrDA® baud clock output.

Legend: CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; PPS = Peripheral Pin Select; Analog = Analog input; O = Output; TTL = TTL input buffer; P = Power; I = Input

1: Not all pins are available in all package variants. See the **"Pin Diagrams"** section for pin availability.

2: PWM4H/L through PWM8H/L are fixed on dsPIC33EPXXXGS708/808 devices. PWM4H/L through PWM6H/L are fixed on dsPIC33EPXXXGS706/806 devices.

3: The SCK3 pin is fixed on dsPIC33EPXXXGS706/806 and dsPIC33EPXXXGS708/808 devices.

4: PPS is available on dsPIC33EPXXXGS702 devices only.

TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name[1]	Pin Type	Buffer Type	PPS	Description
U2CTS	I	ST	Yes	UART2 Clear-to-Send.
U2RTS	O	—	Yes	UART2 Ready-to-Send.
U2RX	I	ST	Yes	UART2 receive.
U2TX	O	—	Yes	UART2 transmit.
BCLK2	O	ST	Yes	UART2 IrDA baud clock output.
SCK1	I/O	ST	Yes	Synchronous serial clock input/output for SPI1.
SDI1	I	ST	Yes	SPI1 data in.
SDO1	O	—	Yes	SPI1 data out.
SS1	I/O	ST	Yes	SPI1 slave synchronization or frame pulse I/O.
SCK2	I/O	ST	Yes	Synchronous serial clock input/output for SPI2.
SDI2	I	ST	Yes	SPI2 data in.
SDO2	O	—	Yes	SPI2 data out.
SS2	I/O	ST	Yes	SPI2 slave synchronization or frame pulse I/O.
SCK3	I/O	ST	Yes[3]	Synchronous serial clock input/output for SPI3.
SDI3	I	ST	Yes	SPI3 data in.
SDO3	O	—	Yes	SPI3 data out.
SS3	I/O	ST	Yes	SPI3 slave synchronization or frame pulse I/O.
SCL1	I/O	ST	No	Synchronous serial clock input/output for I2C1.
SDA1	I/O	ST	No	Synchronous serial data input/output for I2C1.
ASCL1	I/O	ST	No	Alternate synchronous serial clock input/output for I2C1.
ASDA1	I/O	ST	No	Alternate synchronous serial data input/output for I2C1.
SCL2	I/O	ST	No	Synchronous serial clock input/output for I2C2.
SDA2	I/O	ST	No	Synchronous serial data input/output for I2C2.
ASCL2	I/O	ST	No	Alternate synchronous serial clock input/output for I2C2.
ASDA2	I/O	ST	No	Alternate synchronous serial data input/output for I2C2.
TMS	I	ST	No	JTAG Test mode select pin.
TCK	I	ST	No	JTAG test clock input pin.
TDI	I	ST	No	JTAG test data input pin.
TDO	O	—	No	JTAG test data output pin.
FLT1-FLT8	I	ST	Yes	PWM Fault Inputs 1 through 8.
FLT9-FLT12	I	ST	No	PWM Fault Inputs 9 through 12.
PWM1L-PWM3L	O	—	No	PWM Low Outputs 1 through 3.
PWM1H-PWM3H	O	—	No	PWM High Outputs 1 through 3.
PWM4L-PWM8L[2]	O	—	Yes	PWM Low Outputs 4 through 8.
PWM4H-PWM8H[2]	O	—	Yes	PWM High Outputs 4 through 8.
SYNCI1, SYNCI2	I	ST	Yes	PWM Synchronization Inputs 1 and 2.
SYNCO1, SYNCO2	O	—	Yes	PWM Synchronization Outputs 1 and 2.

Legend: CMOS = CMOS compatible input or output; Analog = Analog input; P = Power
ST = Schmitt Trigger input with CMOS levels; O = Output; I = Input
PPS = Peripheral Pin Select; TTL = TTL input buffer

1: Not all pins are available in all package variants. See the **"Pin Diagrams"** section for pin availability.
2: PWM4H/L through PWM8H/L are fixed on dsPIC33EPXXXGS708/808 devices. PWM4H/L through PWM6H/L are fixed on dsPIC33EPXXXGS706/806 devices.
3: The SCK3 pin is fixed on dsPIC33EPXXXGS706/806 and dsPIC33EPXXXGS708/808 devices.
4: PPS is available on dsPIC33EPXXXGS702 devices only.

TABLE 1-1: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name[1]	Pin Type	Buffer Type	PPS	Description
CMP1A-CMP4A CMP1B-CMP4B CMP1C-CMP4C CMP1D-CMP4D	I I I I	Analog Analog Analog Analog	No No No No	Comparator Channels 1A through 4A inputs. Comparator Channels 1B through 4B inputs. Comparator Channels 1C through 4C inputs. Comparator Channels 1D through 4D inputs.
ACMP1-ACMP4	O	—	Yes	Analog Comparator Outputs 1-4.
DACOUT1, DACOUT2	O	—	No	DAC Output Voltages 1 and 2.
EXTREF1, EXTREF2	I	Analog	No	External Voltage Reference Inputs 1 and 2 for the Reference DACs.
PGA1P1-PGA1P4	I	Analog	No	PGA1 Positive Inputs 1 through 4.
PGA1N1-PGA1N3	I	Analog	No	PGA1 Negative Inputs 1 through 3.
PGA2P1-PGA2P4	I	Analog	No	PGA2 Positive Inputs 1 through 4.
PGA2N1-PGA2N3	I	Analog	No	PGA2 Negative Inputs 1 through 3.
ADTRG31	I	ST	No	External ADC trigger source.
PGED1 PGEC1 PGED2 PGEC2 PGED3 PGEC3	I/O I I/O I I/O I	ST ST ST ST ST ST	No No No No No No	Data I/O pin for Programming/Debugging Communication Channel 1. Clock input pin for Programming/Debugging Communication Channel 1. Data I/O pin for Programming/Debugging Communication Channel 2. Clock input pin for Programming/Debugging Communication Channel 2. Data I/O pin for Programming/Debugging Communication Channel 3. Clock input pin for Programming/Debugging Communication Channel 3.
$\overline{\text{MCLR}}$	I/P	ST	No	Master Clear (Reset) input. This pin is an active-low Reset to the device.
AVDD	P	P	No	Positive supply for analog modules. This pin must be connected at all times.
AVSS	P	P	No	Ground reference for analog modules. This pin must be connected at all times.
VDD	P	—	No	Positive supply for peripheral logic and I/O pins.
VCAP	P	—	No	CPU logic filter capacitor connection.
VSS	P	—	No	Ground reference for logic and I/O pins.

Legend: CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; PPS = Peripheral Pin Select; Analog = Analog input; O = Output; TTL = TTL input buffer; P = Power; I = Input

1: Not all pins are available in all package variants. See the **"Pin Diagrams"** section for pin availability.

2: PWM4H/L through PWM8H/L are fixed on dsPIC33EPXXXGS708/808 devices. PWM4H/L through PWM6H/L are fixed on dsPIC33EPXXXGS706/806 devices.

3: The SCK3 pin is fixed on dsPIC33EPXXXGS706/806 and dsPIC33EPXXXGS708/808 devices.

4: PPS is available on dsPIC33EPXXXGS702 devices only.

2.0 GUIDELINES FOR GETTING STARTED WITH 16-BIT DIGITAL SIGNAL CONTROLLERS

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the related section of the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

2.1 Basic Connection Requirements

Getting started with the dsPIC33EPXXXGS70X/80X family requires attention to a minimal set of device pin connections before proceeding with development. The following is a list of pin names which must always be connected:

- All VDD and VSS pins (see **Section 2.2 "Decoupling Capacitors"**)
- All AVDD and AVSS pins regardless if ADC module is not used (see **Section 2.2 "Decoupling Capacitors"**)
- VCAP (see **Section 2.3 "CPU Logic Filter Capacitor Connection (VCAP)"**)
- $\overline{\text{MCLR}}$ pin (see **Section 2.4 "Master Clear (MCLR) Pin"**)
- PGECx/PGEDx pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes (see **Section 2.5 "ICSP Pins"**)
- OSC1 and OSC2 pins when external oscillator source is used (see **Section 2.6 "External Oscillator Pins"**)

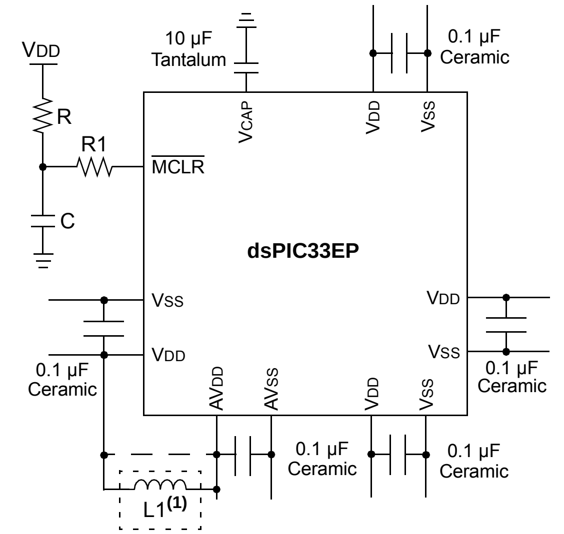
2.2 Decoupling Capacitors

The use of decoupling capacitors on every pair of power supply pins, such as VDD, VSS, AVDD and AVSS is required.

Consider the following criteria when using decoupling capacitors:

- **Value and type of capacitor:** Recommendation of 0.1 µF (100 nF), 10-20V. This capacitor should be a low-ESR and have resonance frequency in the range of 20 MHz and higher. It is recommended to use ceramic capacitors.
- **Placement on the printed circuit board:** The decoupling capacitors should be placed as close to the pins as possible. It is recommended to place the capacitors on the same side of the board as the device. If space is constricted, the capacitor can be placed on another layer on the PCB using a via; however, ensure that the trace length from the pin to the capacitor is within one-quarter inch (6 mm) in length.
- **Handling high-frequency noise:** If the board is experiencing high-frequency noise, above tens of MHz, add a second ceramic-type capacitor in parallel to the above described decoupling capacitor. The value of the second capacitor can be in the range of 0.01 µF to 0.001 µF. Place this second capacitor next to the primary decoupling capacitor. In high-speed circuit designs, consider implementing a decade pair of capacitances as close to the power and ground pins as possible. For example, 0.1 µF in parallel with 0.001 µF.
- **Maximizing performance:** On the board layout from the power supply circuit, run the power and return traces to the decoupling capacitors first, and then to the device pins. This ensures that the decoupling capacitors are first in the power chain. Equally important is to keep the trace length between the capacitor and the power pins to a minimum, thereby reducing PCB track inductance.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTION



Note 1: As an option, instead of a hard-wired connection, an inductor (L1) can be substituted between VDD and AVDD to improve ADC noise rejection. The inductor impedance should be less than 1Ω and the inductor capacity greater than 10 mA.

Where:

$$f = \frac{F_{CNV}}{2} \quad \text{(i.e., ADC Conversion Rate/2)}$$

$$f = \frac{1}{(2\pi\sqrt{LC})}$$

$$L = \left(\frac{1}{(2\pi f\sqrt{C})}\right)^2$$

2.2.1 TANK CAPACITORS

On boards with power traces running longer than six inches in length, it is suggested to use a tank capacitor for integrated circuits, including DSCs, to supply a local power source. The value of the tank capacitor should be determined based on the trace resistance that connects the power supply source to the device and the maximum current drawn by the device in the application. In other words, select the tank capacitor so that it meets the acceptable voltage sag at the device. Typical values range from 4.7 µF to 47 µF.

2.3 CPU Logic Filter Capacitor Connection (VCAP)

A low-ESR (< 0.5Ω) capacitor is required on the VCAP pin, which is used to stabilize the voltage regulator output voltage. The VCAP pin must not be connected to VDD and must have a capacitor greater than 4.7 µF (10 µF is recommended), 16V connected to ground. The type can be ceramic or tantalum. See **Section 30.0 "Electrical Characteristics"** for additional information.

The placement of this capacitor should be close to the VCAP pin. It is recommended that the trace length not exceeds one-quarter inch (6 mm). See **Section 27.4 "On-Chip Voltage Regulator"** for details.

2.4 Master Clear ($\overline{\text{MCLR}}$) Pin

The $\overline{\text{MCLR}}$ pin provides two specific device functions:

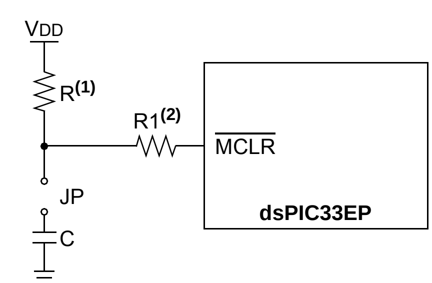
- Device Reset
- Device Programming and Debugging.

During device programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the $\overline{\text{MCLR}}$ pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R and C will need to be adjusted based on the application and PCB requirements.

For example, as shown in Figure 2-2, it is recommended that the capacitor, C, be isolated from the $\overline{\text{MCLR}}$ pin during programming and debugging operations.

Place the components as shown in Figure 2-2, within one-quarter inch (6 mm) from the $\overline{\text{MCLR}}$ pin.

FIGURE 2-2: EXAMPLE OF $\overline{\text{MCLR}}$ PIN CONNECTIONS



Note 1: R ≤ 10 kΩ is recommended. A suggested starting value is 10 kΩ. Ensure that the $\overline{\text{MCLR}}$ pin VIH and VIL specifications are met.

2: R1 ≤ 470Ω will limit any current flowing into $\overline{\text{MCLR}}$ from the external capacitor, C, in the event of $\overline{\text{MCLR}}$ pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the $\overline{\text{MCLR}}$ pin VIH and VIL specifications are met.

2.5 ICSP Pins

The PGECx and PGEDx pins are used for ICSP and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of Ohms, not to exceed 100 Ohms.

Pull-up resistors, series diodes and capacitors on the PGECx and PGEDx pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits and pin Voltage Input High (V_{IH}) and Voltage Input Low (V_{IL}) requirements.

Ensure that the "Communication Channel Select" (i.e., PGECx/PGEDx pins) programmed into the device matches the physical connections for the ICSP to MPLAB® PICkit™ 3, MPLAB ICD 3, or MPLAB REAL ICE™.

For more information on MPLAB ICD 2, MPLAB ICD 3 and REAL ICE connection requirements, refer to the following documents that are available on the Microchip web site.

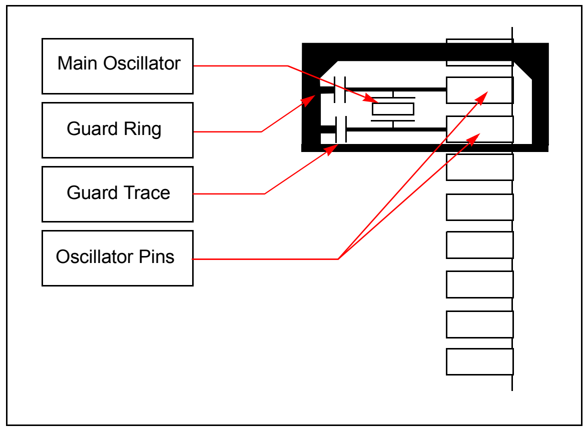
- *"Using MPLAB® ICD 3 In-Circuit Debugger"* (poster) (DS51765)
- *"Development Tools Design Advisory"* (DS51764)
- *"MPLAB® REAL ICE™ In-Circuit Emulator User's Guide"* (DS51616)
- *"Using MPLAB® REAL ICE™ In-Circuit Emulator"* (poster) (DS51749)

2.6 External Oscillator Pins

Many DSCs have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator. For details, see **Section 9.0 "Oscillator Configuration"** for details.

The oscillator circuit should be placed on the same side of the board as the device. Also, place the oscillator circuit close to the respective oscillator pins, not exceeding one-half inch (12 mm) distance between them. The load capacitors should be placed next to the oscillator itself, on the same side of the board. Use a grounded copper pour around the oscillator circuit to isolate them from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed. A suggested layout is shown in Figure 2-3.

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



2.7 Oscillator Value Conditions on Device Start-up

If the PLL of the target device is enabled and configured for the device start-up oscillator, the maximum oscillator source frequency must be limited to 3 MHz < FIN < 5.5 MHz to comply with device PLL start-up conditions. This means that if the external oscillator frequency is outside this range, the application must start up in the FRC mode first. The default PLL settings, after a POR with an oscillator frequency outside this range, will violate the device operating speed.

Once the device powers up, the application firmware can initialize the PLL SFRs, CLKDIV and PLLFBD, to a suitable value, and then perform a clock switch to the Oscillator + PLL clock source. Note that clock switching must be enabled in the device Configuration Word.

2.8 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state.

Alternatively, connect a 1k to 10k resistor between VSS and unused pins, and drive the output to logic low.

2.9 Targeted Applications

- Power Factor Correction (PFC)
  - Interleaved PFC
  - Critical Conduction PFC
  - Bridgeless PFC
- DC/DC Converters
  - Buck, Boost, Forward, Flyback, Push-Pull
  - Half/Full-Bridge
  - Phase-Shift Full-Bridge
  - Resonant Converters
- DC/AC
  - Half/Full-Bridge Inverter
  - Resonant Inverter

Examples of typical application connections are shown in Figure 2-4 through Figure 2-6.

FIGURE 2-4: INTERLEAVED PFC

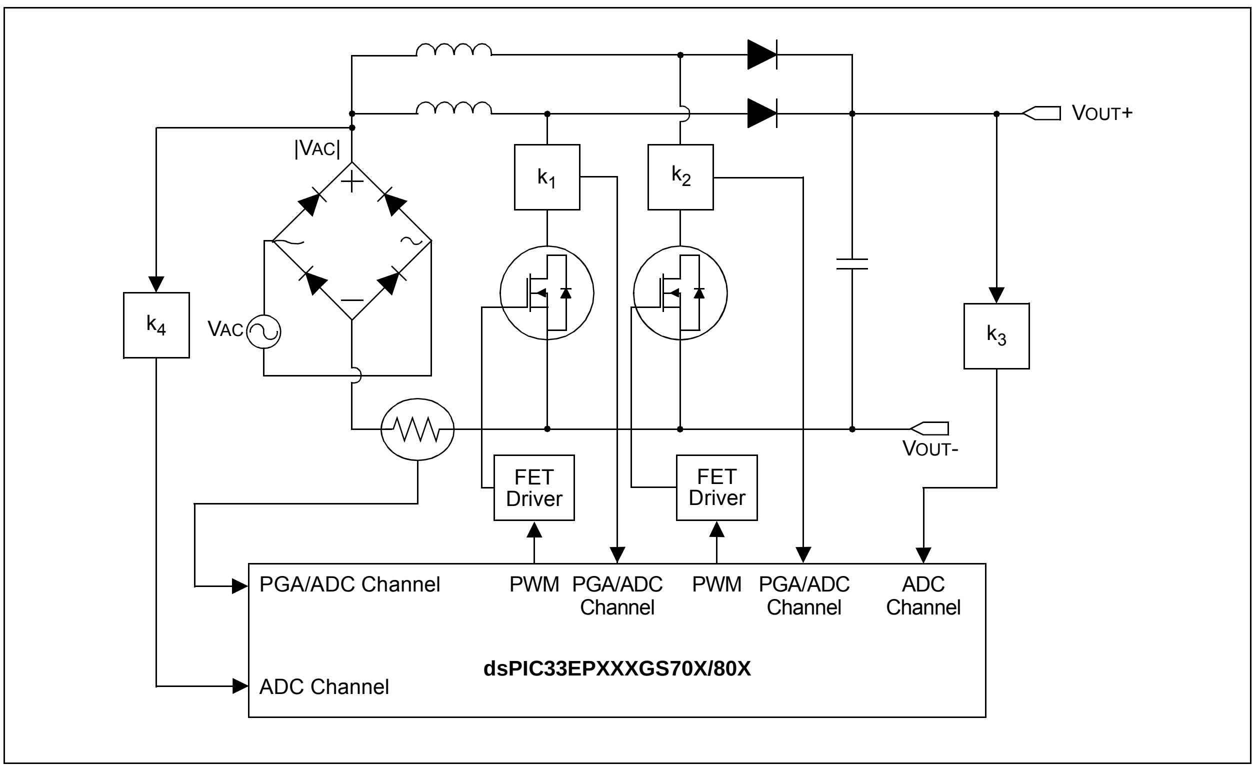


FIGURE 2-5: PHASE-SHIFTED FULL-BRIDGE CONVERTER

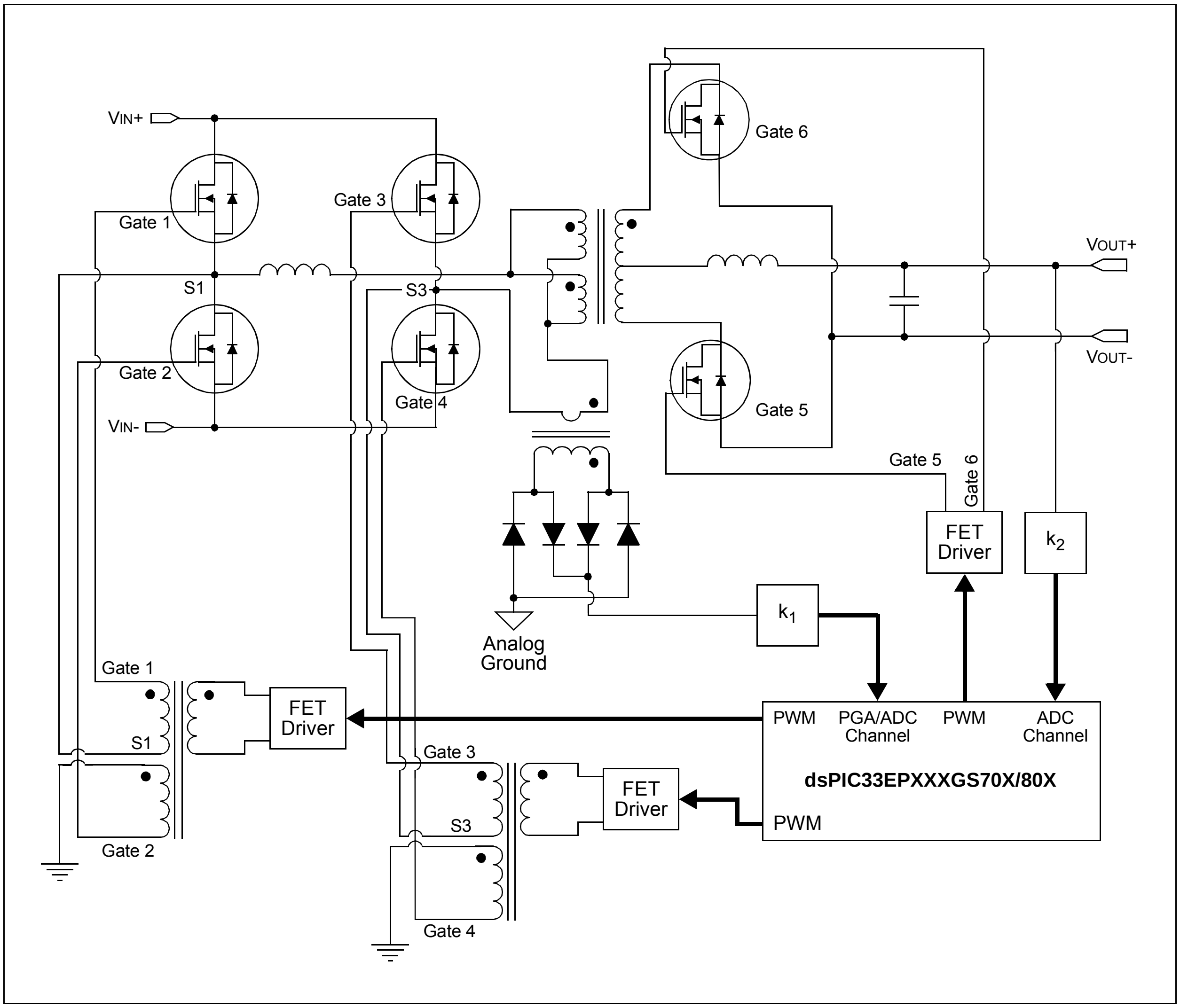
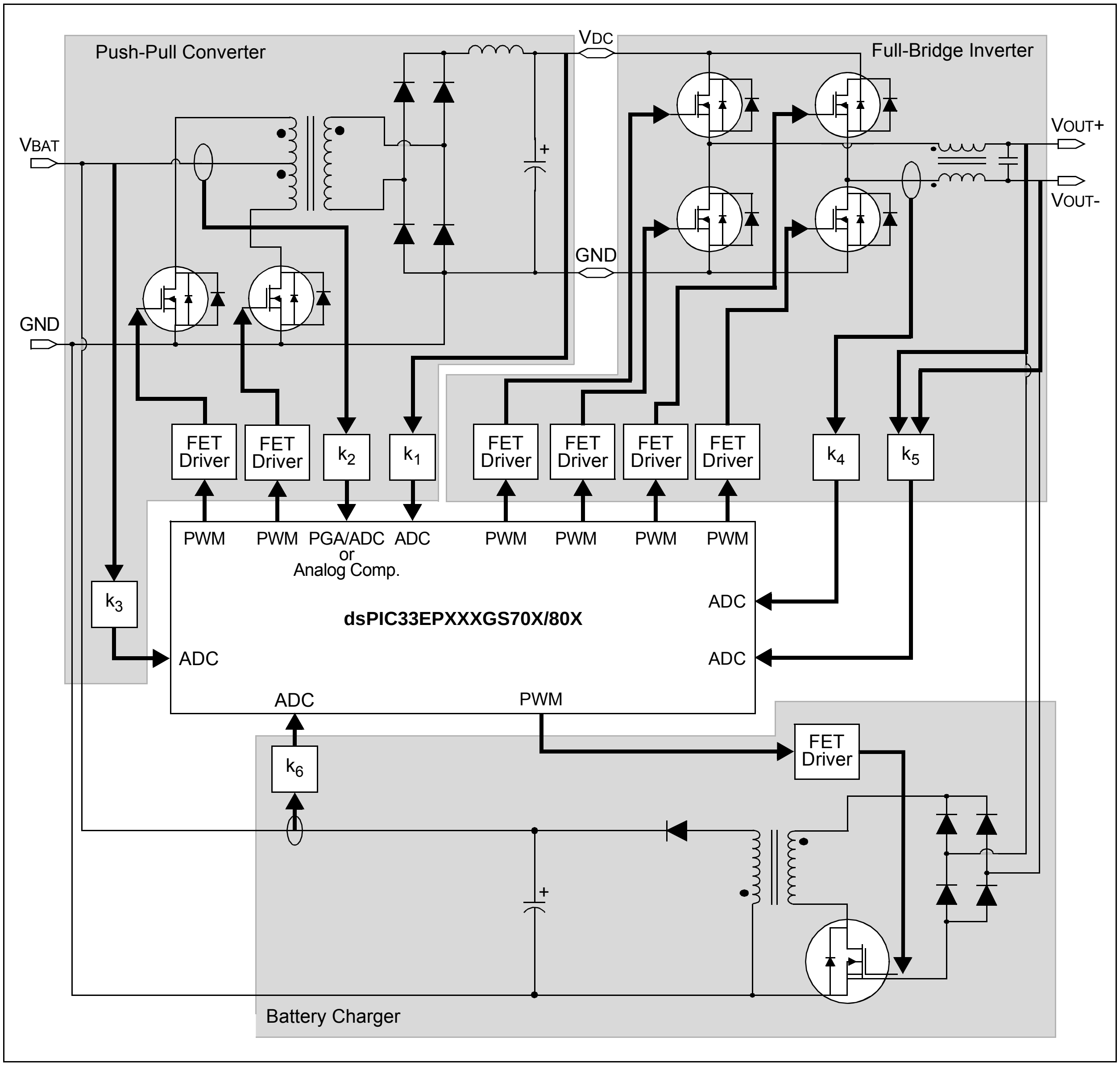


FIGURE 2-6: OFF-LINE UPS



3.0 CPU

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"dsPIC33E Enhanced CPU"** (DS70005158) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family CPU has a 16-bit (data) modified Harvard architecture with an enhanced instruction set, including significant support for Digital Signal Processing (DSP). The CPU has a 24-bit instruction word with a variable length opcode field. The Program Counter (PC) is 23 bits wide and addresses up to 4M x 24 bits of user program memory space.

An instruction prefetch mechanism helps maintain throughput and provides predictable execution. Most instructions execute in a single-cycle effective execution rate, with the exception of instructions that change the program flow, the double-word move (`MOV.D`) instruction, PSV accesses and the table instructions. Overhead-free program loop constructs are supported using the `DO` and `REPEAT` instructions, both of which are interruptible at any point.

3.1 Registers

The dsPIC33EPXXXGS70X/80X devices have sixteen, 16-bit Working registers in the programmer's model. Each of the Working registers can act as a Data, Address or Address Offset register. The 16th Working register (W15) operates as a Software Stack Pointer for interrupts and calls.

In addition, the dsPIC33EPXXXGS70X/80X devices include four Alternate Working register sets which consist of W0 through W14. The Alternate Working registers can be made persistent to help reduce the saving and restoring of register content during Interrupt Service Routines (ISRs). The Alternate Working registers can be assigned to a specific Interrupt Priority Level (IPL1 through IPL7) by configuring the CTXTx<2:0> bits in the FALTREG Configuration register. The Alternate Working registers can also be accessed manually by using the `CTXTSWP` instruction. The CCTXI<2:0> and MCTXI<2:0> bits in the CTXTSTAT register can be used to identify the current, and most recent, manually selected Working register sets.

3.2 Instruction Set

The instruction set for dsPIC33EPXXXGS70X/80X devices has two classes of instructions: the MCU class of instructions and the DSP class of instructions. These two instruction classes are seamlessly integrated into the architecture and execute from a single execution unit. The instruction set includes many addressing modes and was designed for optimum C compiler efficiency.

3.3 Data Space Addressing

The base Data Space can be addressed as up to 4K words or 8 Kbytes, and is split into two blocks, referred to as X and Y data memory. Each memory block has its own independent Address Generation Unit (AGU). The MCU class of instructions operates solely through the X memory AGU, which accesses the entire memory map as one linear Data Space. Certain DSP instructions operate through the X and Y AGUs to support dual operand reads, which splits the data address space into two parts. The X and Y Data Space boundary is device-specific.

The upper 32 Kbytes of the Data Space memory map can optionally be mapped into Program Space (PS) at any 16K program word boundary. The program-to-Data Space mapping feature, known as Program Space Visibility (PSV), lets any instruction access Program Space as if it were Data Space. Refer to **"Data Memory"** (DS70595) in the *"dsPIC33/PIC24 Family Reference Manual"* for more details on PSV and table accesses.

On dsPIC33EPXXXGS70X/80X devices, overhead-free circular buffers (Modulo Addressing) are supported in both X and Y address spaces. The Modulo Addressing removes the software boundary checking overhead for DSP algorithms. The X AGU Circular Addressing can be used with any of the MCU class of instructions. The X AGU also supports Bit-Reversed Addressing to greatly simplify input or output data re-ordering for radix-2 FFT algorithms.

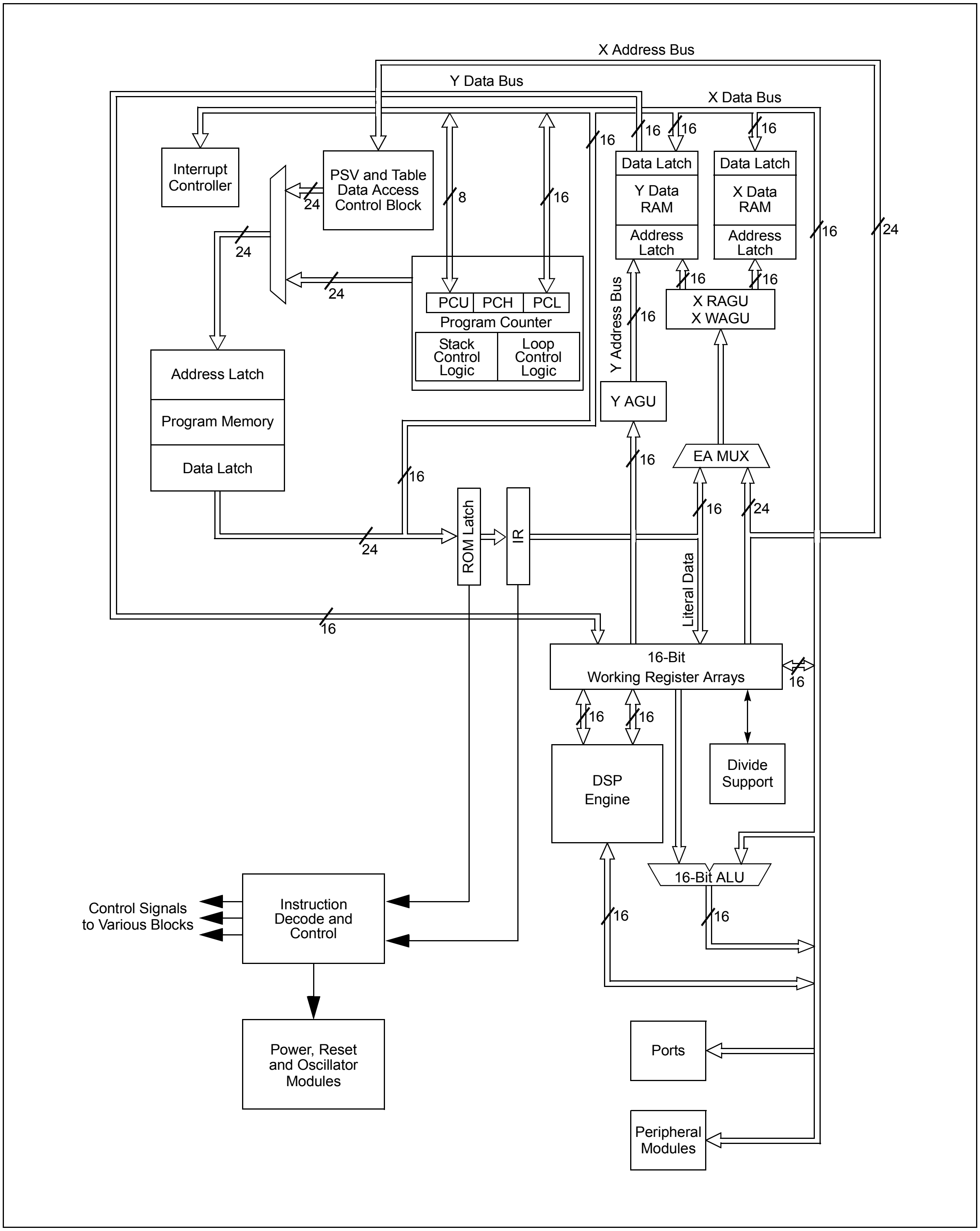
3.4 Addressing Modes

The CPU supports these addressing modes:

- Inherent (no operand)
- Relative
- Literal
- Memory Direct
- Register Direct
- Register Indirect

Each instruction is associated with a predefined addressing mode group, depending upon its functional requirements. As many as six addressing modes are supported for each instruction.

FIGURE 3-1: dsPIC33EPXXXGS70X/80X FAMILY CPU BLOCK DIAGRAM



3.5 Programmer's Model

The programmer's model for the dsPIC33EPXXXGS70X/80X family is shown in Figure 3-2. All registers in the programmer's model are memory-mapped and can be manipulated directly by instructions. Table 3-1 lists a description of each register.

In addition to the registers contained in the programmer's model, the dsPIC33EPXXXGS70X/80X devices contain control registers for Modulo Addressing, Bit-Reversed Addressing and interrupts. These registers are described in subsequent sections of this document.

All registers associated with the programmer's model are memory-mapped, as shown in Table 3-1.

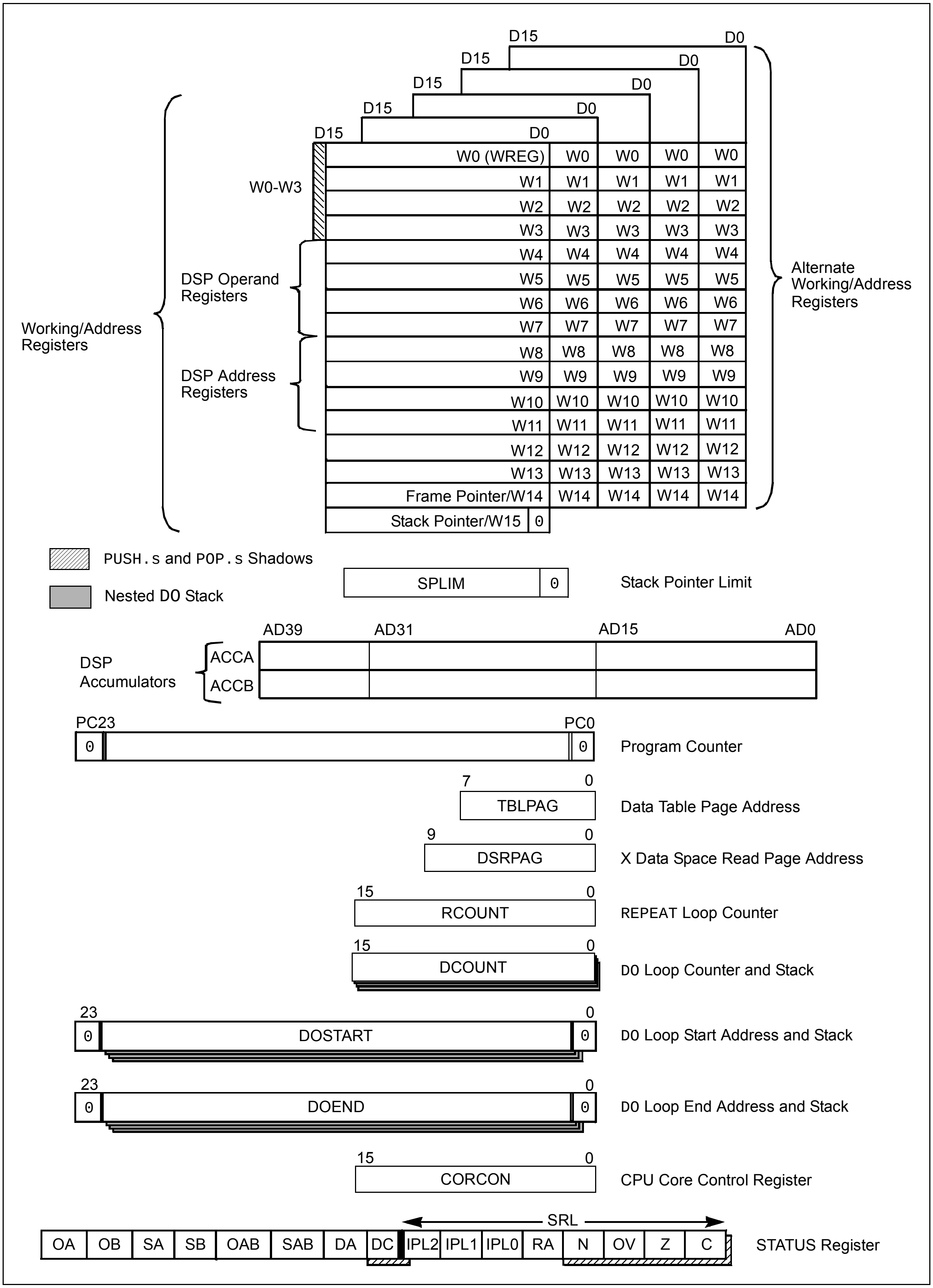
TABLE 3-1: PROGRAMMER'S MODEL REGISTER DESCRIPTIONS

Register(s) Name	Description
W0 through W15[1]	Working Register Array
W0 through W14[1]	Alternate 1 Working Register Array
W0 through W14[1]	Alternate 2 Working Register Array
W0 through W14[1]	Alternate 3 Working Register Array
W0 through W14[1]	Alternate 4 Working Register Array
ACCA, ACCB	40-Bit DSP Accumulators
PC	23-Bit Program Counter
SR	ALU and DSP Engine STATUS Register
SPLIM	Stack Pointer Limit Value Register
TBLPAG	Table Memory Page Address Register
DSRPAG	Extended Data Space (EDS) Read Page Register
RCOUNT	REPEAT Loop Counter Register
DCOUNT	DO Loop Counter Register
DOSTARTH[2], DOSTARTL[2]	DO Loop Start Address Register (High and Low)
DOENDH, DOENDL	DO Loop End Address Register (High and Low)
CORCON	Contains DSP Engine, DO Loop Control and Trap Status bits

Note 1: Memory-mapped W0 through W14 represent the value of the register in the currently active CPU context.

2: The DOSTARTH and DOSTARTL registers are read-only.

FIGURE 3-2: PROGRAMMER'S MODEL



3.6 CPU Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

3.6.1 KEY RESOURCES

- **“dsPIC33E Enhanced CPU”** (DS70005158) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

3.7 CPU Control Registers

REGISTER 3-1: SR: CPU STATUS REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/C-0	R/C-0	R-0	R/W-0
OA	OB	SA(3)	SB(3)	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0(2)	R/W-0(2)	R/W-0(2)	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2(1)	IPL1(1)	IPL0(1)	RA	N	OV	Z	C
bit 7							bit 0

Legend:	C = Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1'= Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **OA:** Accumulator A Overflow Status bit
1 = Accumulator A has overflowed
0 = Accumulator A has not overflowed

bit 14 **OB:** Accumulator B Overflow Status bit
1 = Accumulator B has overflowed
0 = Accumulator B has not overflowed

bit 13 **SA:** Accumulator A Saturation 'Sticky' Status bit(3)
1 = Accumulator A is saturated or has been saturated at some time
0 = Accumulator A is not saturated

bit 12 **SB:** Accumulator B Saturation 'Sticky' Status bit(3)
1 = Accumulator B is saturated or has been saturated at some time
0 = Accumulator B is not saturated

bit 11 **OAB:** OA || OB Combined Accumulator Overflow Status bit
1 = Accumulator A or B has overflowed
0 = Neither Accumulator A or B has overflowed

bit 10 **SAB:** SA || SB Combined Accumulator 'Sticky' Status bit
1 = Accumulator A or B is saturated or has been saturated at some time
0 = Neither Accumulator A or B is saturated

bit 9 **DA:** DO Loop Active bit
1 = DO loop is in progress
0 = DO loop is not in progress

bit 8 **DC:** MCU ALU Half Carry/$\overline{\text{Borrow}}$ bit
1 = A carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred
0 = No carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred

Note 1: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL, if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.

2: The IPL<2:0> Status bits are read-only when the NSTDIS bit (INTCON1<15>) = 1.

3: A data write to the SR register can modify the SA and SB bits by either a data write to SA and SB or by clearing the SAB bit. To avoid a possible SA or SB bit write race condition, the SA and SB bits should not be modified using bit operations.

REGISTER 3-1: SR: CPU STATUS REGISTER (CONTINUED)

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits[1,2]

111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

bit 4 **RA:** REPEAT Loop Active bit

1 = REPEAT loop is in progress
0 = REPEAT loop is not in progress

bit 3 **N:** MCU ALU Negative bit

1 = Result was negative
0 = Result was non-negative (zero or positive)

bit 2 **OV:** MCU ALU Overflow bit

This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude that causes the sign bit to change state.
1 = Overflow occurred for signed arithmetic (in this arithmetic operation)
0 = No overflow occurred

bit 1 **Z:** MCU ALU Zero bit

1 = An operation that affects the Z bit has set it at some time in the past
0 = The most recent operation that affects the Z bit has cleared it (i.e., a non-zero result)

bit 0 **C:** MCU ALU Carry/$\overline{\text{Borrow}}$ bit

1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note 1: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL, if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.

2: The IPL<2:0> Status bits are read-only when the NSTDIS bit (INTCON1<15>) = 1.

3: A data write to the SR register can modify the SA and SB bits by either a data write to SA and SB or by clearing the SAB bit. To avoid a possible SA or SB bit write race condition, the SA and SB bits should not be modified using bit operations.

REGISTER 3-2: CORCON: CORE CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-0
VAR	—	US1	US0	EDT(1)	DL2	DL1	DL0
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3(2)	SFA	RND	IF
bit 7							bit 0

Legend:	C = Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **VAR:** Variable Exception Processing Latency Control bit
1 = Variable exception processing is enabled
0 = Fixed exception processing is enabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13-12 **US<1:0>:** DSP Multiply Unsigned/Signed Control bits
11 = Reserved
10 = DSP engine multiplies are mixed-sign
01 = DSP engine multiplies are unsigned
00 = DSP engine multiplies are signed

bit 11 **EDT:** Early DO Loop Termination Control bit(1)
1 = Terminates executing DO loop at the end of current loop iteration
0 = No effect

bit 10-8 **DL<2:0>:** DO Loop Nesting Level Status bits
111 = 7 DO loops are active
•
•
•
001 = 1 DO loop is active
000 = 0 DO loops are active

bit 7 **SATA:** ACCA Saturation Enable bit
1 = Accumulator A saturation is enabled
0 = Accumulator A saturation is disabled

bit 6 **SATB:** ACCB Saturation Enable bit
1 = Accumulator B saturation is enabled
0 = Accumulator B saturation is disabled

bit 5 **SATDW:** Data Space Write from DSP Engine Saturation Enable bit
1 = Data Space write saturation is enabled
0 = Data Space write saturation is disabled

bit 4 **ACCSAT:** Accumulator Saturation Mode Select bit
1 = 9.31 saturation (super saturation)
0 = 1.31 saturation (normal saturation)

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit 3(2)
1 = CPU Interrupt Priority Level is greater than 7
0 = CPU Interrupt Priority Level is 7 or less

Note 1: This bit is always read as ‘0’.
2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.

REGISTER 3-2: CORCON: CORE CONTROL REGISTER (CONTINUED)

bit 2 **SFA:** Stack Frame Active Status bit
1 = Stack frame is active; W14 and W15 address 0x0000 to 0xFFFF, regardless of DSRPAG
0 = Stack frame is not active; W14 and W15 address the base Data Space

bit 1 **RND:** Rounding Mode Select bit
1 = Biased (conventional) rounding is enabled
0 = Unbiased (convergent) rounding is enabled

bit 0 **IF:** Integer or Fractional Multiplier Mode Select bit
1 = Integer mode is enabled for DSP multiply
0 = Fractional mode is enabled for DSP multiply

Note 1: This bit is always read as ‘0’.
2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.

REGISTER 3-3: CTXTSTAT: CPU W REGISTER CONTEXT STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	CCTXI2	CCTXI1	CCTXI0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	MCTXI2	MCTXI1	MCTXI0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-11 **Unimplemented:** Read as ‘0’

bit 10-8 **CCTXI<2:0>:** Current (W Register) Context Identifier bits
111 = Reserved
•
•
•
101 = Reserved
100 = Alternate Working Register Set 4 is currently in use
011 = Alternate Working Register Set 3 is currently in use
010 = Alternate Working Register Set 2 is currently in use
001 = Alternate Working Register Set 1 is currently in use
000 = Default register set is currently in use

bit 7-3 **Unimplemented:** Read as ‘0’

bit 2-0 **MCTXI<2:0>:** Manual (W Register) Context Identifier bits
111 = Reserved
•
•
•
101 = Reserved
100 = Alternate Working Register Set 4 was most recently manually selected
011 = Alternate Working Register Set 3 was most recently manually selected
010 = Alternate Working Register Set 2 was most recently manually selected
001 = Alternate Working Register Set 1 was most recently manually selected
000 = Default register set was most recently manually selected

3.8 Arithmetic Logic Unit (ALU)

The dsPIC33EPXXXGS70X/80X family ALU is 16 bits wide and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. Depending on the operation, the ALU can affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

The ALU can perform 8-bit or 16-bit operations, depending on the mode of the instruction that is used. Data for the ALU operation can come from the W register array or data memory, depending on the addressing mode of the instruction. Likewise, output data from the ALU can be written to the W register array or a data memory location.

Refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157) for information on the SR bits affected by each instruction.

The core CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

3.8.1 MULTIPLIER

Using the high-speed, 17-bit x 17-bit multiplier, the ALU supports unsigned, signed or mixed-sign operation in several MCU Multiplication modes:

- 16-bit x 16-bit signed
- 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 16-bit signed x 16-bit unsigned
- 16-bit unsigned x 5-bit (literal) unsigned
- 16-bit unsigned x 16-bit signed
- 8-bit unsigned x 8-bit unsigned

3.8.2 DIVIDER

The divide block supports 32-bit/16-bit and 16-bit/16-bit signed and unsigned integer divide operations with the following data sizes:

- 32-bit signed/16-bit signed divide
- 32-bit unsigned/16-bit unsigned divide
- 16-bit signed/16-bit signed divide
- 16-bit unsigned/16-bit unsigned divide

The quotient for all divide instructions ends up in W0 and the remainder in W1. 16-bit signed and unsigned `DIV` instructions can specify any W register for both the 16-bit divisor (Wn) and any W register (aligned) pair (W(m + 1):Wm) for the 32-bit dividend. The divide algorithm takes one cycle per bit of divisor, so both 32-bit/16-bit and 16-bit/16-bit instructions take the same number of cycles to execute.

3.9 DSP Engine

The DSP engine consists of a high-speed 17-bit x 17-bit multiplier, a 40-bit barrel shifter and a 40-bit adder/ subtracter (with two target accumulators, round and saturation logic).

The DSP engine can also perform inherent accumulator-to-accumulator operations that require no additional data. These instructions are, `ADD`, `SUB` and `NEG`.

The DSP engine has options selected through bits in the CPU Core Control register (CORCON), as listed below:

- Fractional or Integer DSP Multiply (IF)
- Signed, Unsigned or Mixed-Sign DSP Multiply (USx)
- Conventional or Convergent Rounding (RND)
- Automatic Saturation On/Off for ACCA (SATA)
- Automatic Saturation On/Off for ACCB (SATB)
- Automatic Saturation On/Off for Writes to Data Memory (SATDW)
- Accumulator Saturation mode Selection (ACCSAT)

TABLE 3-2: DSP INSTRUCTIONS SUMMARY

Instruction	Algebraic Operation	ACC Write-Back
`CLR`	$A = 0$	Yes
`ED`	$A = (x - y)^2$	No
`EDAC`	$A = A + (x - y)^2$	No
`MAC`	$A = A + (x \bullet y)$	Yes
`MAC`	$A = A + x^2$	No
`MOVSAC`	No change in A	Yes
`MPY`	$A = x \bullet y$	No
`MPY`	$A = x^2$	No
`MPY.N`	$A = - x \bullet y$	No
`MSC`	$A = A - x \bullet y$	Yes

4.0 MEMORY ORGANIZATION

Note:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"dsPIC33E/PIC24E Program Memory"** (DS70000613) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The dsPIC33EPXXXGS70X/80X family architecture features separate program and data memory spaces, and buses. This architecture also allows the direct access of program memory from the Data Space (DS) during code execution.

4.1 Program Address Space

The program address memory space of the dsPIC33EPXXXGS70X/80X family devices is 4M instructions. The space is addressable by a 24-bit value derived either from the 23-bit PC during program execution, or from table operation or Data Space remapping, as described in **Section 4.9 "Interfacing Program and Data Memory Spaces"**.

User application access to the program memory space is restricted to the lower half of the address range (0x000000 to 0x7FFFFF). The exception is the use of TBLRD operations, which use TBLPAG<7> to permit access to calibration data and Device ID sections of the configuration memory space.

The program memory maps for dsPIC33EPXXXGS70X/80X devices not operating in Dual Partition mode are shown in Figure 4-1 and Figure 4-2.

The dsPIC33EPXXXGS70X/80X devices can operate in a Dual Partition Flash Program Memory mode, where the user Program Flash Memory is arranged as two separate address spaces, one for each of the Flash partitions. The Active Partition always starts at address, 0x000000, and contains half of the available Flash memory (64k/128k, depends on device). The Inactive Partition always starts at address, 0x400000, and implements the remaining half of Flash memory. As shown in Figure 4-3 and Figure 4-4, the Active and Inactive Partitions are identical, and both contain unique copies of the Reset vector, Interrupt Vector Tables (IVT and AIVT if enabled) and the Flash Configuration Words.

4.2 Unique Device Identifier (UDID)

All dsPIC33EPXXXGS70X/80X family devices are individually encoded during final manufacturing with a Unique Device Identifier or UDID. This feature allows for manufacturing traceability of Microchip Technology devices in applications where this is a requirement. It may also be used by the application manufacturer for any number of things that may require unique identification, such as:

- Tracking the device
- Unique serial number
- Unique security key

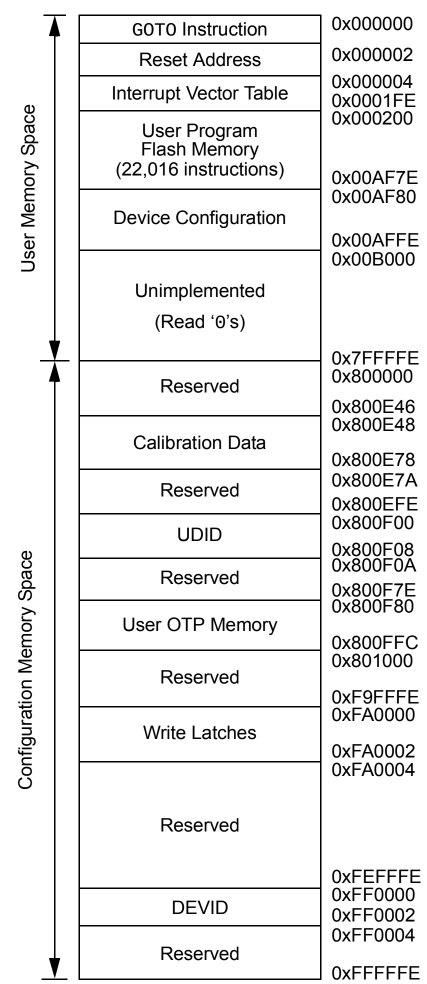
The UDID comprises five 24-bit program words. When taken together, these fields form a unique 120-bit identifier.

The UDID is stored in five read-only locations, located between 800F00h and 800F08h in the device configuration space. Table 4-1 lists the addresses of the identifier words and shows their contents.

TABLE 4-1: UDID ADDRESSES

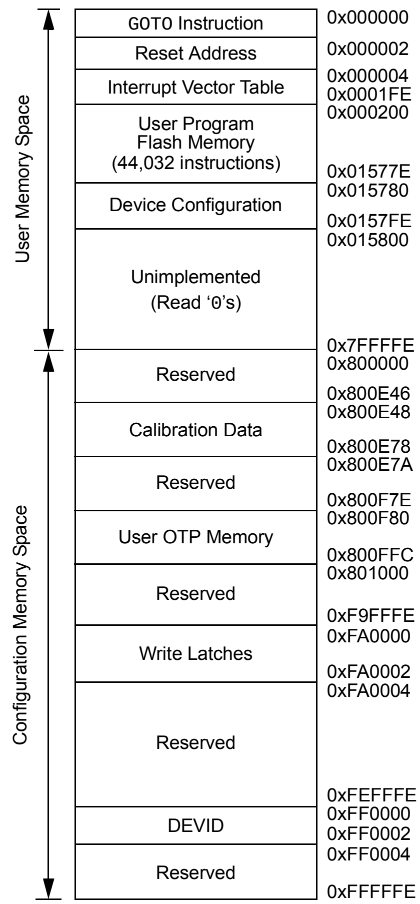
Name	Address	Bits 23:16	Bits 15:8	Bits 7:0
UDID1	800F00	UDID Word 1		
UDID2	800F02	UDID Word 2		
UDID3	800F04	UDID Word 3		
UDID4	800F06	UDID Word 4		
UDID5	800F08	UDID Word 5		

FIGURE 4-1: PROGRAM MEMORY MAP FOR dsPIC33EP64GS70X/80X DEVICES



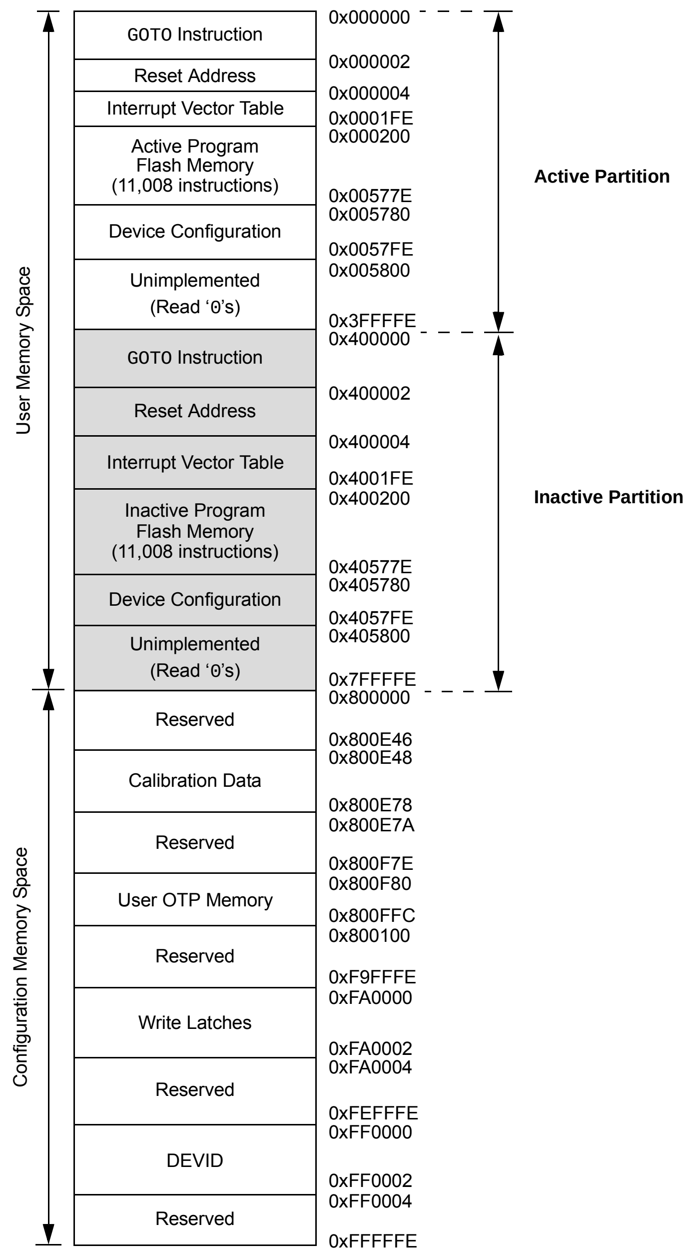
Note: Memory areas are not shown to scale.

FIGURE 4-2: PROGRAM MEMORY MAP FOR dsPIC33EP128GS70X/80X DEVICES



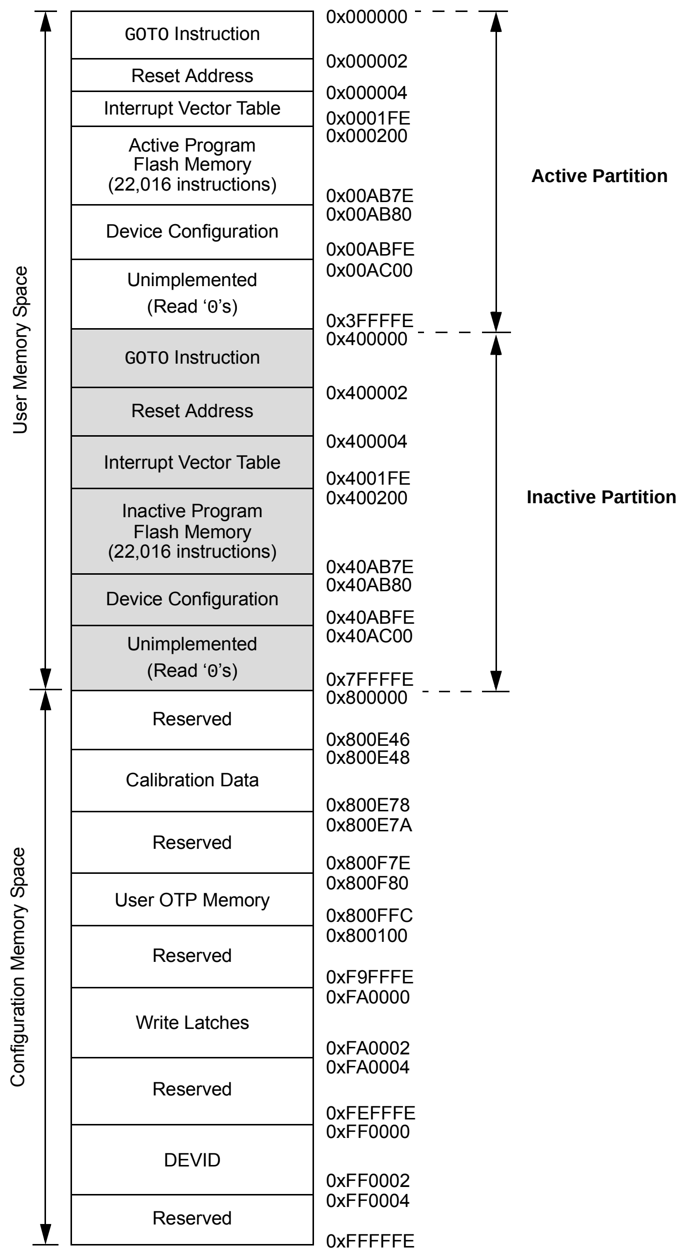
Note: Memory areas are not shown to scale.

FIGURE 4-3: PROGRAM MEMORY MAP FOR dsPIC33EP64GS70X/80X DEVICES (DUAL PARTITION)



Note: Memory areas are not shown to scale.

FIGURE 4-4: PROGRAM MEMORY MAP FOR dsPIC33EP128GS70X/80X DEVICES (DUAL PARTITION)



Note: Memory areas are not shown to scale.

4.2.1 PROGRAM MEMORY ORGANIZATION

The program memory space is organized in word-addressable blocks. Although it is treated as 24 bits wide, it is more appropriate to think of each address of the program memory as a lower and upper word, with the upper byte of the upper word being unimplemented. The lower word always has an even address, while the upper word has an odd address (Figure 4-5).

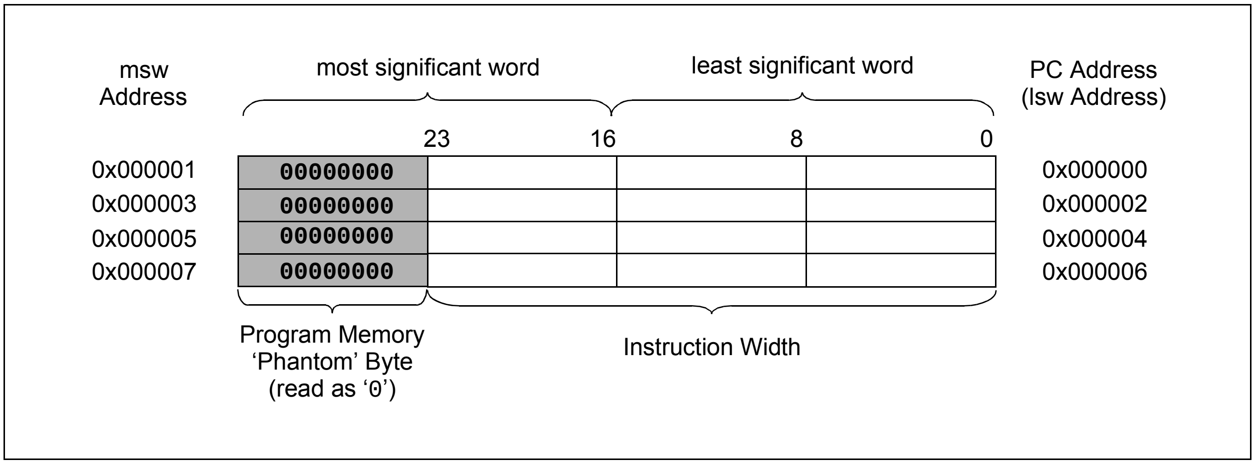
Program memory addresses are always word-aligned on the lower word, and addresses are incremented or decremented by two, during code execution. This arrangement provides compatibility with data memory space addressing and makes data in the program memory space accessible.

4.2.2 INTERRUPT AND TRAP VECTORS

All dsPIC33EPXXXGS70X/80X family devices reserve the addresses between 0x000000 and 0x000200 for hard-coded program execution vectors. A hardware Reset vector is provided to redirect code execution from the default value of the PC on device Reset to the actual start of code. A `GOTO` instruction is programmed by the user application at address, 0x000000, of Flash memory, with the actual address for the start of code at address, 0x000002, of Flash memory.

A more detailed discussion of the Interrupt Vector Tables (IVTs) is provided in **Section 7.1 "Interrupt Vector Table"**.

FIGURE 4-5: PROGRAM MEMORY ORGANIZATION



4.3 Data Address Space

The dsPIC33EPXXXGS70X/80X family CPU has a separate 16-bit wide data memory space. The Data Space is accessed using separate Address Generation Units (AGUs) for read and write operations. The data memory map is shown in Figure 4-6.

All Effective Addresses (EAs) in the data memory space are 16 bits wide and point to bytes within the Data Space. This arrangement gives a base Data Space address range of 64 Kbytes or 32K words.

The lower half of the data memory space (i.e., when EA<15> = 0) is used for implemented memory addresses, while the upper half (EA<15> = 1) is reserved for the Program Space Visibility (PSV).

dsPIC33EPXXXGS70X/80X family devices implement up to 12 Kbytes of data memory. If an EA points to a location outside of this area, an all-zero word or byte is returned.

4.3.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit wide blocks. Data is aligned in data memory and registers as 16-bit words, but all Data Space EAs resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

4.3.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® MCU devices and improve Data Space memory usage efficiency, the dsPIC33EPXXXGS70X/80X family instruction set supports both word and byte operations. As a consequence of byte accessibility, all Effective Address calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] results in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

A data byte read, reads the complete word that contains the byte, using the LSb of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, data memory and registers are organized as two parallel, byte-wide entities with shared (word) address decode, but separate write lines. Data byte writes only write to the corresponding side of the array or register that matches the byte address.

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported, so care must be taken when mixing byte and word operations, or translating from 8-bit MCU code. If a misaligned read or write is attempted, an address error trap is generated. If the error occurred on a read, the instruction underway is completed. If the error occurred on a write, the instruction is executed but the write does not occur. In either case, a trap is then executed, allowing the system and/or user application to examine the machine state prior to execution of the address Fault.

All byte loads into any W register are loaded into the LSB; the MSB is not modified.

A Sign-Extend (SE) instruction is provided to allow user applications to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, user applications can clear the MSB of any W register by executing a Zero-Extend (ZE) instruction on the appropriate address.

4.3.3 SFR SPACE

The first 4 Kbytes of the Near Data Space, from 0x0000 to 0x0FFF, is primarily occupied by Special Function Registers (SFRs). These are used by the dsPIC33EPXXXGS70X/80X family core and peripheral modules for controlling the operation of the device.

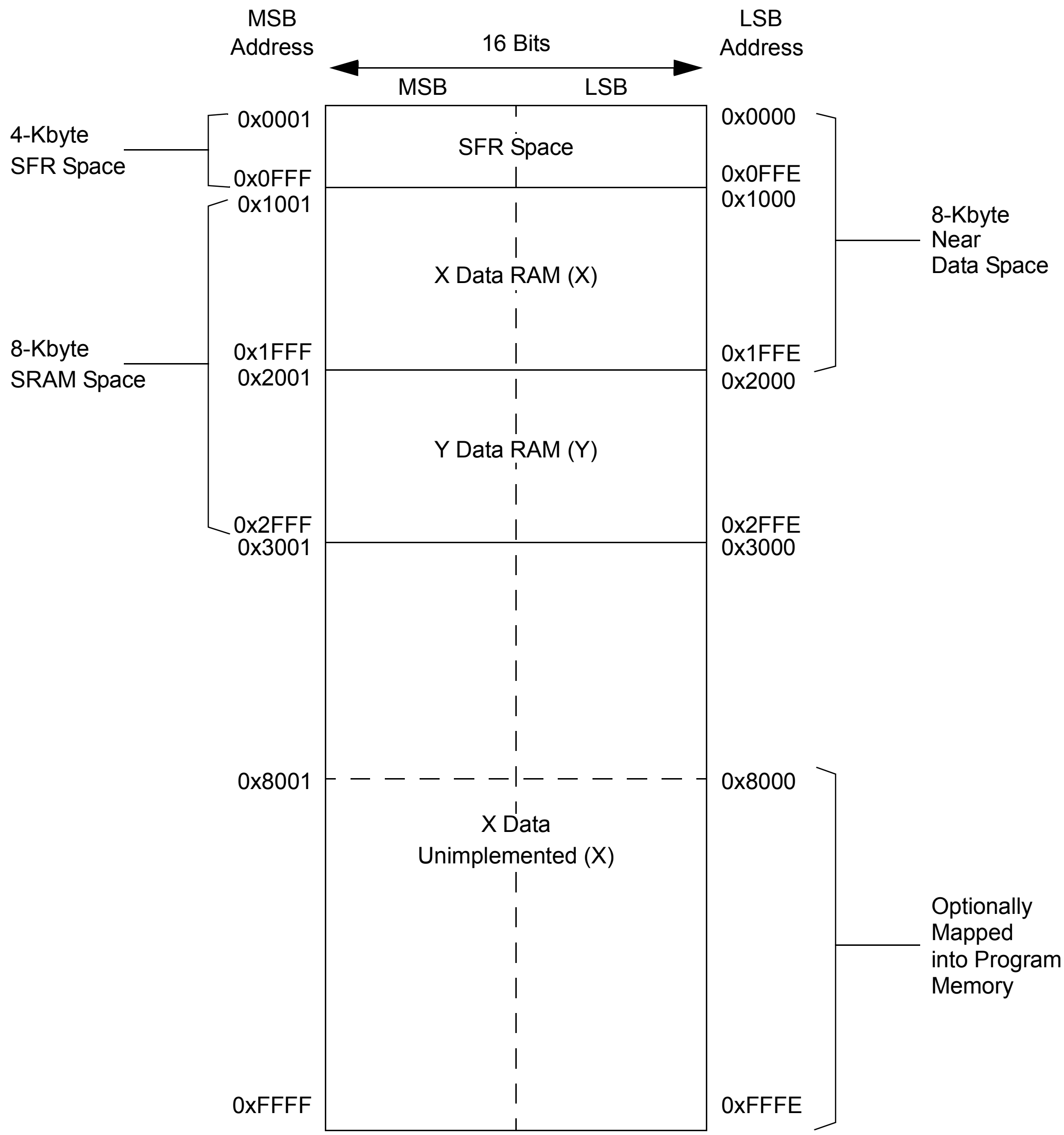
SFRs are distributed among the modules that they control and are generally grouped together by module. Much of the SFR space contains unused addresses; these are read as '0'.

Note:	The actual set of peripheral features and interrupts varies by the device. Refer to the corresponding device tables and pinout diagrams for device-specific information.

4.3.4 NEAR DATA SPACE

The 8-Kbyte area, between 0x0000 and 0x1FFF, is referred to as the Near Data Space. Locations in this space are directly addressable through a 13-bit absolute address field within all memory direct instructions. Additionally, the whole Data Space is addressable using MOV instructions, which support Memory Direct Addressing mode with a 16-bit address field, or by using Indirect Addressing mode using a Working register as an Address Pointer.

FIGURE 4-6: DATA MEMORY MAP FOR dsPIC33EP64GS70X/80X DEVICES



Note: Memory areas are not shown to scale.

4.3.5 X AND Y DATA SPACES

The dsPIC33EPXXXGS70X/80X core has two Data Spaces, X and Y. These Data Spaces can be considered either separate (for some DSP instructions) or as one unified linear address range (for MCU instructions). The Data Spaces are accessed using two Address Generation Units (AGUs) and separate data paths. This feature allows certain instructions to concurrently fetch two words from RAM, thereby enabling efficient execution of DSP algorithms, such as Finite Impulse Response (FIR) filtering and Fast Fourier Transform (FFT).

The X Data Space is used by all instructions and supports all addressing modes. X Data Space has separate read and write data buses. The X read data bus is the read data path for all instructions that view Data Space as combined X and Y address space. It is also the X data prefetch path for the dual operand DSP instructions (`MAC` class).

The Y Data Space is used in concert with the X Data Space by the `MAC` class of instructions (`CLR`, `ED`, `EDAC`, `MAC`, `MOVSAC`, `MPY`, `MPY.N` and `MSC`) to provide two concurrent data read paths.

Both the X and Y Data Spaces support Modulo Addressing mode for all instructions, subject to addressing mode restrictions. Bit-Reversed Addressing mode is only supported for writes to X Data Space.

All data memory writes, including in DSP instructions, view Data Space as combined X and Y address space. The boundary between the X and Y Data Spaces is device-dependent and is not user-programmable.

4.4 Memory Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

4.4.1 KEY RESOURCES

- **"dsPIC33E/PIC24E Program Memory"** (DS70000613) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

4.5 Special Function Register Maps

TABLE 4-2: SFR BLOCK 000h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
Core			WREG14	01C	0000000000000000	DOSTARTL	03A	xxxxxxxxxxxxxxx0
WREG0	000	0000000000000000	WREG15	01E	0000100000000000	DOSTARTH	03C	0000000000xxxxxx
WREG1	002	0000000000000000	SPLIM	020	xxxxxxxxxxxxxxx0	DOENDL	03E	xxxxxxxxxxxxxxx0
WREG2	004	0000000000000000	ACCAL	022	xxxxxxxxxxxxxxxx	DOENDH	040	0000000000xxxxxx
WREG3	006	0000000000000000	ACCAH	024	xxxxxxxxxxxxxxxx	SR	042	0000000000000000
WREG4	008	0000000000000000	ACCAU	026	00000000xxxxxxxx	CORCON	044	0000000000100000
WREG5	00A	0000000000000000	ACCBL	028	xxxxxxxxxxxxxxxx	MODCON	046	0000000000000000
WREG6	00C	0000000000000000	ACCBH	02A	xxxxxxxxxxxxxxxx	XMODSRT	048	xxxxxxxxxxxxxxx0
WREG7	00E	0000000000000000	ACCBU	02C	00000000xxxxxxxx	XMODEND	04A	xxxxxxxxxxxxxxx1
WREG8	010	0000000000000000	PCL	02E	0000000000000000	YMODSRT	04C	xxxxxxxxxxxxxxx0
WREG9	012	0000000000000000	PCH	030	0000000000000000	YMODEND	04E	xxxxxxxxxxxxxxx1
WREG10	014	0000000000000000	DSRPAG	032	0000000000000001	XBREV	050	xxxxxxxxxxxxxxxx
WREG11	016	0000000000000000	DSWPAG	034	0000000000000001	DISICNT	052	00xxxxxxxxxxxxxx
WREG12	018	0000000000000000	RCOUNT	036	xxxxxxxxxxxxxxxx	TBLPAG	054	00000000xxxxxxxx
WREG13	01A	0000000000000000	DCOUNT	038	xxxxxxxxxxxxxxxx	CTXTSTAT	05A	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-3: SFR BLOCK 100h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
Timers			TMR5HLD	116	xxxxxxxxxxxxxxxx	IC2CON2	14A	0000000000001101
TMR1	100	xxxxxxxxxxxxxxxx	TMR5	118	xxxxxxxxxxxxxxxx	IC2BUF	14C	xxxxxxxxxxxxxxxx
PR1	102	1111111111111111	PR4	11A	1111111111111111	IC2TMR	14E	0000000000000000
T1CON	104	0000000000000000	PR5	11C	1111111111111111	IC3CON1	150	0000000000000000
TMR2	106	xxxxxxxxxxxxxxxx	T4CON	11E	0000000000000000	IC3CON2	152	0000000000001101
TMR3HLD	108	xxxxxxxxxxxxxxxx	T5CON	120	0000000000000000	IC3BUF	154	xxxxxxxxxxxxxxxx
TMR3	10A	xxxxxxxxxxxxxxxx	**Input Capture**			IC3TMR	156	0000000000000000
PR2	10C	1111111111111111	IC1CON1	140	0000000000000000	IC4CON1	158	0000000000000000
PR3	10E	1111111111111111	IC1CON2	142	0000000000001101	IC4CON2	15A	0000000000001101
T2CON	110	0000000000000000	IC1BUF	144	xxxxxxxxxxxxxxxx	IC4BUF	15C	xxxxxxxxxxxxxxxx
T3CON	112	0000000000000000	IC1TMR	146	0000000000000000	IC4TMR	15E	0000000000000000
TMR4	114	xxxxxxxxxxxxxxxx	IC2CON1	148	0000000000000000			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-4: SFR BLOCK 200h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
I2C1 and I2C2			U1STA	222	0000000010010000	SPI1BRGH	252	0000000000000000
I2C1CONL	200	0001000000000000	U1TXREG	224	0000000xxxxxxxxx	SPI1IMSKL	254	0000000000000000
I2C1CONH	202	0000000000000000	U1RXREG	226	0000000000000000	SPI1IMSKH	256	0000000000000000
I2C1STAT	204	0000000000000000	U1BRG	228	0000000000000000	SPI1URDTL	258	0000000000000000
I2C1ADD	206	0000000000000000	U2MODE	230	0000000000000000	SPI1URDTH	25A	0000000000000000
I2C1MSK	208	0000000000000000	U2STA	232	0000000010010000	SPI2CON1L	260	0000000000000000
I2C1BRG	20A	0000000000000000	U2TXREG	234	0000000xxxxxxxxx	SPI2CON1H	262	0000000000000000
I2C1TRN	20C	0000000011111111	U2RXREG	236	0000000000000000	SPI2CON2L	264	0000000000000000
I2C1RCV	20E	0000000000000000	U2BRG	238	0000000000000000	SPI2CON2H	266	0000000000000000
I2C2CON1	210	0001000000000000	**SPI**			SPI2STATL	268	0000000000101000
I2C2CON2	212	0000000000000000	SPI1CON1L	240	0000000000000000	SPI2STATH	26A	0000000000000000
I2C2STAT	214	0000000000000000	SPI1CON1H	242	0000000000000000	SPI2BUFL	26C	0000000000000000
I2C2ADD	216	0000000000000000	SPI1CON2L	244	0000000000000000	SPI2BUFH	26E	0000000000000000
I2C2MSK	218	0000000000000000	SPI1CON2H	246	0000000000000000	SPI3STAT	270	000xxxxxxxxxxxxx
I2C2BRG	21A	0000000000000000	SPI1STATL	248	0000000000101000	SPI2BRGH	272	0000000000000000
I2C2TRN	21C	0000000011111111	SPI1STATH	24A	0000000000000000	SPI2IMSKL	274	0000000000000000
I2C2RCV	21E	0000000000000000	SPI1BUFL	24C	0000000000000000	SPI2IMSKH	276	0000000000000000
UART1 and UART2			SPI1BUFH	24E	0000000000000000	SPI2URDTL	278	0000000000000000
U1MODE	220	0000000000000000	SPI1BRGL	250	000xxxxxxxxxxxxx	SPI2URDTH	27A	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-5: SFR BLOCK 300h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
ADC			ADCMP0ENH	33A	0000000000000000	ADTRIG4L	390	0000000000000000
ADCON1L	300	0000000000000000	ADCMP0LO	33C	0000000000000000	ADTRIG4H	392	0000000000000000
ADCON1H	302	0000000001100000	ADCMP0HI	33E	0000000000000000	ADCMP0CON	3A0	0000000000000000
ADCON2L	304	0000000000000000	ADCMP1ENL	340	0000000000000000	ADCMP1CON	3A4	0000000000000000
ADCON2H	306	0000000000000000	ADCMP1ENH	342	0000000000000000	ADBASE	3C0	0000000000000000
ADCON3L	308	0000000000000000	ADCMP1LO	344	0000000000000000	ADLVLTRGL	3D0	0000000000000000
ADCON3H	30A	0000000000000000	ADCMP1HI	346	0000000000000000	ADLVLTRGH	3D2	0000000000000000
ADCON4L	30C	0000000000000000	ADFL0DAT	368	0000000000000000	ADCORE0L	3D4	0000000000000000
ADCON4H	30E	0000000000000000	ADFL0CON	36A	0000000000000000	ADCORE0H	3D6	0000001100000000
ADMOD0L	310	0000000000000000	ADFL1DAT	36C	0000000000000000	ADCORE1L	3D8	0000000000000000
ADMOD0H	312	0000000000000000	ADFL1CON	36E	0000000000000000	ADCORE1H	3DA	0000001100000000
ADMOD1L	314	0000000000000000	ADTRIG0L	380	0000000000000000	ADCORE2L	3DC	0000000000000000
ADIEL	320	0000000000000000	ADTRIG0H	382	0000000000000000	ADCORE2H	3DE	0000001100000000
ADIEH	322	0000000000000000	ADTRIG1L	384	0000000000000000	ADCORE3L	3E0	0000000000000000
ADCSS1L	328	0000000000000000	ADTRIG1H	386	0000000000000000	ADCORE3H	3E2	0000001100000000
ADCSS1H	32A	0000000000000000	ADTRIG2L	388	0000000000000000	ADEIEL	3F0	0000000000000000
ADSTATL	330	0000000000000000	ADTRIG2H	38A	0000000000000000	ADEIEH	3F2	0000000000000000
ADSTATH	332	0000000000000000	ADTRIG3L	38C	0000000000000000	ADEISTATL	3F8	0000000000000000
ADCMP0ENL	338	0000000000000000	ADTRIG3H	38E	0000000000000000	ADEISTATH	3FA	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-6: SFR BLOCK 400h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
ADC			C1FCTRL	486	0000000000000000	C1RXM2EID	4BA	xxxxxxxxxxxxxxxx
ADCON5L	400	0000000000000000	C1FIFO	488	0000000000000000	C1RXF1SID	4C4	xxxxxxxxxxxxxxxx
ADCON5H	402	0000000000000000	C1INTF	48A	0000000000000000	C1RXF1EID	4C6	xxxxxxxxxxxxxxxx
ADCAL0L	404	0000000000000000	C1INTE	48C	0000000000000000	C1RXF2SID	4C8	xxxxxxxxxxxxxxxx
ADCAL0H	406	0000000000000000	C1EC	48E	0000000000000000	C1RXF2EID	4CA	xxxxxxxxxxxxxxxx
ADCAL1H	40A	0000000000000000	C1CFG1	490	0000000000000000	C1RXF3SID	4CC	xxxxxxxxxxxxxxxx
ADCBUF0	40C	0000000000000000	C1CFG2	492	0x000xxxxxxxxxxx	C1RXF3EID	4CE	xxxxxxxxxxxxxxxx
ADCBUF1	40E	0000000000000000	C1FEN1	494	1111111111111111	C1RXF4SID	4D0	xxxxxxxxxxxxxxxx
ADCBUF2	410	0000000000000000	C1FMSKSEL1	498	0000000000000000	C1RXF4EID	4D2	xxxxxxxxxxxxxxxx
ADCBUF3	412	0000000000000000	C1FMSKSEL2	49A	0000000000000000	C1RXF5SID	4D4	xxxxxxxxxxxxxxxx
ADCBUF4	414	0000000000000000	**CAN (WIN (C1CTRL<0>) = 0)**			C1RXF5EID	4D6	xxxxxxxxxxxxxxxx
ADCBUF5	416	0000000000000000	C1RXFUL1	4A0	0000000000000000	C1RXF6SID	4D8	xxxxxxxxxxxxxxxx
ADCBUF6	418	0000000000000000	C1RXFUL2	4A2	0000000000000000	C1RXF6EID	4DA	xxxxxxxxxxxxxxxx
ADCBUF7	41A	0000000000000000	C1RXOVF1	4A8	0000000000000000	C1RXF7SID	4DC	xxxxxxxxxxxxxxxx
ADCBUF8	41C	0000000000000000	C1RXOVF2	4AA	0000000000000000	C1RXF7EID	4DE	xxxxxxxxxxxxxxxx
ADCBUF9	41E	0000000000000000	C1TR01CON	4B0	0000000000000000	C1RXF8SID	4E0	xxxxxxxxxxxxxxxx
ADCBUF10	420	0000000000000000	C1TR23CON	4B2	0000000000000000	C1RXF8EID	4E2	xxxxxxxxxxxxxxxx
ADCBUF11	422	0000000000000000	C1TR45CON	4B4	0000000000000000	C1RXF9SID	4E4	xxxxxxxxxxxxxxxx
ADCBUF12	424	0000000000000000	C1TR67CON	4B6	xxxxxxxxxxxxxxxx	C1RXF9EID	4E6	xxxxxxxxxxxxxxxx
ADCBUF13	426	0000000000000000	C1RXD	4C0	xxxxxxxxxxxxxxxx	C1RXF10SID	4E8	xxxxxxxxxxxxxxxx
ADCBUF14	428	0000000000000000	C1TXD	4C2	xxxxxxxxxxxxxxxx	C1RXF10EID	4EA	xxxxxxxxxxxxxxxx
ADCBUF15	42A	0000000000000000	**CAN (WIN (C1CTR1<0>) = 1)**			C1RXF11SID	4EC	xxxxxxxxxxxxxxxx
ADCBUF16	42C	0000000000000000	C1BUFPNT1	4A0	0000000000000000	C1RXF11EID	4EE	xxxxxxxxxxxxxxxx
ADCBUF17	42E	0000000000000000	C1BUFPNT2	4A2	0000000000000000	C1RXF12SID	4F0	xxxxxxxxxxxxxxxx
ADCBUF18	430	0000000000000000	C1BUFPNT3	4A4	0000000000000000	C1RXF12EID	4F2	xxxxxxxxxxxxxxxx
ADCBUF19	432	0000000000000000	C1BUFPNT4	4A6	0000000000000000	C1RXF13SID	4F4	xxxxxxxxxxxxxxxx
ADCBUF20	434	0000000000000000	C1RXM0SID	4B0	xxxxxxxxxxxxxxxx	C1RXF13EID	4F6	xxxxxxxxxxxxxxxx
ADCBUF21	436	0000000000000000	C1RXM0EID	4B2	xxxxxxxxxxxxxxxx	C1RXF14SID	4F8	xxxxxxxxxxxxxxxx
CAN (WIN (C1CTRL<0>) = 0 OR 1)			C1RXM1SID	4B4	xxxxxxxxxxxxxxxx	C1RXF14EID	4FA	xxxxxxxxxxxxxxxx
C1CTRL1	480	000010010000000	C1RXM1EID	4B6	xxxxxxxxxxxxxxxx	C1RXF15SID	4FC	xxxxxxxxxxxxxxxx
C1CTRL2	482	000000000000000	**CAN**			C1RXF15EID	4FE	xxxxxxxxxxxxxxxx
C1VEC	484	000000001000000	C1RXM2SID	4B8	xxxxxxxxxxxxxxxx			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-7: SFR BLOCK 500h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
PGA			PGA2CAL	50A	0000000000000000	CMP2DAC	546	0000000000000000
ISRCCON	500	0000000000000000	**Comparators**			CMP3CON	548	0000000000000000
PGA1CON	504	0000000000000000	CMP1CON	540	0000000000000000	CMP3DAC	54A	0000000000000000
PGA1CAL	506	0000000000000000	CMP1DAC	542	0000000000000000	CMP4CON	54C	0000000000000000
PGA2CON	508	0000000000000000	CMP2CON	544	0000000000000000	CMP4DAC	54E	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-8: SFR BLOCK 600h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
SPI			RPOR8	678	0000000000000000	RPINR7	6AE	0000000000000000
SPI3CON1L	600	0000000000000000	RPOR9	67A	0000000000000000	RPINR8	6B0	0000000000000000
SPI3CON1H	602	0000000000000000	RPOR10	67C	0000000000000000	RPINR11	6B6	0000000000000000
SPI3CON2L	604	0000000000000000	RPOR11	67E	0000000000000000	RPINR12	6B8	0000000000000000
SPI3CON2H	606	0000000000000000	RPOR12	680	0000000000000000	RPINR13	6BA	0000000000000000
SPI3STATL	608	0000000000101000	RPOR13	682	0000000000000000	RPINR18	6C4	0000000000000000
SPI3STATH	60A	0000000000000000	RPOR14	684	0000000000000000	RPINR19	6C6	0000000000000000
SPI3BUFL	60C	0000000000000000	RPOR15	686	0000000000000000	RPINR20	6C8	0000000000000000
SPI3BUFH	60E	0000000000000000	RPOR17	68A	0000000000000000	RPINR21	6CA	0000000000000000
SPI3BRGL	610	000xxxxxxxxxxxxx	RPOR18	68C	0000000000000000	RPINR22	6CC	0000000000000000
SPI3BRGH	612	0000000000000000	RPOR19	68E	0000000000000000	RPINR23	6CE	0000000000000000
SPI3IMSKL	614	0000000000000000	RPOR20	690	0000000000000000	RPINR26	6D4	0000000000000000
SPI3IMSKH	616	0000000000000000	RPOR21	692	0000000000000000	RPINR29	6DA	0000000000000000
SPI3URDTL	618	0000000000000000	RPOR22	694	0000000000000000	RPINR30	6DC	0000000000000000
SPI3URDTH	61A	0000000000000000	RPOR23	696	0000000000000000	RPINR37	6EA	0000000000000000
RPOR0	668	0000000000000000	RPOR24	698	0000000000000000	RPINR38	6EC	0000000000000000
RPOR1	66A	0000000000000000	RPOR25	69A	0000000000000000	RPINR42	6F4	0000000000000000
RPOR2	66C	0000000000000000	RPOR26	69C	0000000000000000	RPINR43	6F6	0000000000000000
RPOR3	66E	0000000000000000	RPINR0	6A0	0000000000000000	RPINR45	6FA	0000000000000000
RPOR4	670	0000000000000000	RPINR1	6A2	0000000000000000	RPINR46	6FC	0000000000000000
RPOR5	672	0000000000000000	RPINR2	6A4	0000000000000000			
RPOR6	674	0000000000000000	RPINR3	6A6	0000000000000000			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-9: SFR BLOCK 700h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
NVM			C2INTF	78A	0000000000000000	C2RXF1SID	7C4	xxxxxxxxxxxxxxxx
NVMCON	728	0000000000000000	C2INTE	78C	0000000000000000	C2RXF1EID	7C6	xxxxxxxxxxxxxxxx
NVMADR	72A	0000000000000000	C2EC	78E	0000000000000000	C2RXF2SID	7C8	xxxxxxxxxxxxxxxx
NVMADRU	72C	0000000000000000	C2CFG1	790	0000000000000000	C2RXF2EID	7CA	xxxxxxxxxxxxxxxx
NVMKEY	72E	0000000000000000	C2CFG2	792	0x000xxxxxxxxxxx	C2RXF3SID	7CC	xxxxxxxxxxxxxxxx
NVMSRCADR	730	0000000000000000	C2FEN1	794	1111111111111111	C2RXF3EID	7CE	xxxxxxxxxxxxxxxx
NVMSRCADRH	732	0000000000000000	C2FMSKSEL1	798	0000000000000000	C2RXF4SID	7D0	xxxxxxxxxxxxxxxx
System Control			C2FMSKSEL2	79A	0000000000000000	C2RXF4EID	7D2	xxxxxxxxxxxxxxxx
RCON	740	0x00x0x01x0xxxxx	**CAN (WIN (C1CTR1<0>) = 0)**			C2RXF5SID	7D4	xxxxxxxxxxxxxxxx
OSCCON	742	0000000000000000	C2RXFUL1	7A0	0000000000000000	C2RXF5EID	7D6	xxxxxxxxxxxxxxxx
CLKDIV	744	0000000000000000	C2RXFUL2	7A2	0000000000000000	C2RXF6SID	7D8	xxxxxxxxxxxxxxxx
PLLFBD	746	0000000000000000	C2RXOVF1	7A8	0000000000000000	C2RXF6EID	7DA	xxxxxxxxxxxxxxxx
OSCTUN	748	0000000000000000	C2RXOVF2	7AA	0000000000000000	C2RXF7SID	7DC	xxxxxxxxxxxxxxxx
LFSR	74C	0000000000000000	C2TR01CON	7B0	0000000000000000	C2RXF7EID	7DE	xxxxxxxxxxxxxxxx
REFOCON	74E	0000000000000000	C2TR23CON	7B2	0000000000000000	C2RXF8SID	7E0	xxxxxxxxxxxxxxxx
ACLKCON	750	0000000000000000	C2TR45CON	7B4	0000000000000000	C2RXF8EID	7E2	xxxxxxxxxxxxxxxx
PMD			C2TR67CON	7B6	xxxxxxxxxxxxxxxx	C2RXF9SID	7E4	xxxxxxxxxxxxxxxx
PMD1	760	0000000000000000	C2RXD	7C0	xxxxxxxxxxxxxxxx	C2RXF9EID	7E6	xxxxxxxxxxxxxxxx
PMD2	762	0000000000000000	C2TXD	7C2	xxxxxxxxxxxxxxxx	C2RXF10SID	7E8	xxxxxxxxxxxxxxxx
PMD3	764	0000000000000000	**CAN (WIN (C1CTR1<0>) = 1)**			C2RXF10EID	7EA	xxxxxxxxxxxxxxxx
PMD4	766	0000000000000000	C2BUFPNT1	7A0	0000000000000000	C2RXF11SID	7EC	xxxxxxxxxxxxxxxx
PMD6	76A	0000000000000000	C2BUFPNT2	7A2	0000000000000000	C2RXF11EID	7EE	xxxxxxxxxxxxxxxx
PMD7	76C	0000000000000000	C2BUFPNT3	7A4	0000000000000000	C2RXF12SID	7F0	xxxxxxxxxxxxxxxx
PMD8	76E	0000000000000000	C2BUFPNT4	7A6	0000000000000000	C2RXF12EID	7F2	xxxxxxxxxxxxxxxx
CAN (WIN (C1CTR1<0>) = 0 or 1)			C2RXM0SID	7B0	xxxxxxxxxxxxxxxx	C2RXF13SID	7F4	xxxxxxxxxxxxxxxx
C2CTRL1	780	0000010010000000	C2RXM0EID	7B2	xxxxxxxxxxxxxxxx	C2RXF13EID	7F6	xxxxxxxxxxxxxxxx
C2CTRL2	782	0000000000000000	C2RXM1SID	7B4	xxxxxxxxxxxxxxxx	C2RXF14SID	7F8	xxxxxxxxxxxxxxxx
C2VEC	784	0000000001000000	C2RXM1EID	7B6	xxxxxxxxxxxxxxxx	C2RXF14EID	7FA	xxxxxxxxxxxxxxxx
C2FCTRL	786	0000000000000000	C2RXM2SID	7B8	xxxxxxxxxxxxxxxx	C2RXF15SID	7FC	xxxxxxxxxxxxxxxx
C2FIFO	788	0000000000000000	C2RXM2EID	7BA	xxxxxxxxxxxxxxxx	C2RXF15EID	7FE	xxxxxxxxxxxxxxxx

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-10: SFR BLOCK 800h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
Interrupt Controller			IEC9	832	0000000000000000	IPC26	874	0000000001000100
IFS0	800	0000000000000000	IEC10	834	0000000000000000	IPC27	876	0100010000000000
IFS1	802	0000000000000000	IEC11	836	0000000000000000	IPC28	878	0100010001000100
IFS2	804	0000000000000000	IPC0	840	0100010001000100	IPC29	87A	0000000001000100
IFS3	806	0000000000000000	IPC1	842	0100010001000000	IPC35	886	0100010000000000
IFS4	808	0000000000000000	IPC2	844	0100010001000100	IPC36	888	0000000000000000
IFS5	80A	0000000000000000	IPC3	846	0100000001000100	IPC37	88A	0100000000000000
IFS6	80C	0000000000000000	IPC4	848	0100010001000100	IPC38	88C	0100010001000100
IFS7	80E	0000000000000000	IPC5	84A	0000000000000100	IPC39	88E	0100010001000100
IFS8	810	0000000000000000	IPC6	84C	0100010001000000	IPC40	890	0100010001000100
IFS9	812	0000000000000000	IPC7	84E	0100010001000100	IPC41	892	0100010001000100
IFS10	814	0000000000000000	IPC8	850	0000000001000100	IPC42	894	0000000001000100
IFS11	816	0000000000000000	IPC9	852	0000010001000000	IPC43	896	0000010001000000
IEC0	820	0000000000000000	IPC11	856	0000000000000000	IPC44	898	0100010001000000
IEC1	822	0000000000000000	IPC12	858	0000010001000000	IPC45	89A	0000000000000100
IEC2	824	0000000000000000	IPC13	85A	0000010000000000	IPC46	89C	0100010000000000
IEC3	826	0000000000000000	IPC14	85C	0000000001000000	IPC47	89E	0000010001000100
IEC4	828	0000000000000000	IPC16	860	0000010001000000	INTCON1	8C0	0000000000000000
IEC5	82A	0000000000000000	IPC18	864	0000000001000000	INTCON2	8C2	0000000000000000
IEC6	82C	0000000000000000	IPC23	86E	0100010000000000	INTCON3	8C4	0000000000000000
IEC7	82E	0000000000000000	IPC24	870	0000010001000100	INTCON4	8C6	0000000000000000
IEC8	830	0000000000000000	IPC25	872	0100000000000000	INTTREG	8C8	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-11: SFR BLOCK 900h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
Output Compare			OC3R	91A	xxxxxxxxxxxxxxxx	CLC2CONH	9CE	0000000000000000
OC1CON1	900	0000000000000000	OC3TMR	91C	0000000000000000	CLC2SEL	9D0	0000000000000000
OC1CON2	902	0000000000001100	OC4CON1	91E	0000000000000000	CLC2GLSL	9D4	0000000000000000
OC1RS	904	xxxxxxxxxxxxxxxx	OC4CON2	920	0000000000001100	CLC2GLSH	9D6	0000000000000000
OC1R	906	xxxxxxxxxxxxxxxx	OC4RS	922	xxxxxxxxxxxxxxxx	CLC3CONL	9D8	0000000000000000
OC1TMR	908	0000000000000000	OC4R	924	xxxxxxxxxxxxxxxx	CLC3CONH	9DA	0000000000000000
OC2CON1	90A	0000000000000000	OC4TMR	926	0000000000000000	CLC3SEL	9DC	0000000000000000
OC2CON2	90C	0000000000001100	**CLC**			CLC3GLSL	9E0	0000000000000000
OC2RS	90E	xxxxxxxxxxxxxxxx	CLC1CONL	9C0	0000000000000000	CLC3GLSH	9E2	0000000000000000
OC2R	910	xxxxxxxxxxxxxxxx	CLC1CONH	9C2	0000000000000000	CLC4CONL	9E4	0000000000000000
OC2TMR	912	0000000000000000	CLC1SEL	9C4	0000000000000000	CLC4CONH	9E6	0000000000000000
OC3CON1	914	0000000000000000	CLC1GLSL	9C8	0000000000000000	CLC4SEL	9E8	0000000000000000
OC3CON2	916	0000000000001100	CLC1GLSH	9CA	0000000000000000	CLC4GLSL	9EC	0000000000000000
OC3RS	918	xxxxxxxxxxxxxxxx	CLC2CONL	9CC	0000000000000000	CLC4GLSH	9EE	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-12: SFR BLOCK A00h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
PTG			PTGADJ	AD2	0000000000000000	PTGQUE7	AE6	xxxxxxxxxxxxxxxx
PTGCST	AC0	0000000000000000	PTGL0	AD4	0000000000000000	PTGQUE8	AE8	xxxxxxxxxxxxxxxx
PTGCON	AC2	0000000000000000	PTGQPTR	AD6	0000000000000000	PTGQUE9	AEA	xxxxxxxxxxxxxxxx
PTGBTE	AC4	0000000000000000	PTGQUE0	AD8	xxxxxxxxxxxxxxxx	PTGQUE10	AEC	xxxxxxxxxxxxxxxx
PTGHOLD	AC6	0000000000000000	PTGQUE1	ADA	xxxxxxxxxxxxxxxx	PTGQUE11	AEE	xxxxxxxxxxxxxxxx
PTGT0LIM	AC8	0000000000000000	PTGQUE2	ADC	xxxxxxxxxxxxxxxx	PTGQUE12	AF0	xxxxxxxxxxxxxxxx
PTGT1LIM	ACA	0000000000000000	PTGQUE3	ADE	xxxxxxxxxxxxxxxx	PTGQUE13	AF2	xxxxxxxxxxxxxxxx
PTGSDLIM	ACC	0000000000000000	PTGQUE4	AE0	xxxxxxxxxxxxxxxx	PTGQUE14	AF4	xxxxxxxxxxxxxxxx
PTGC0LIM	ACE	0000000000000000	PTGQUE5	AE2	xxxxxxxxxxxxxxxx	PTGQUE15	AF6	xxxxxxxxxxxxxxxx
PTGC1LIM	AD0	0000000000000000	PTGQUE6	AE4	xxxxxxxxxxxxxxxx			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-13: SFR BLOCK B00h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
DMA			DMA1STBL	B18	0000000000000000	DMA3REQ	B32	0000000000000000
DMA0CON	B00	0000000000000000	DMA1STBH	B1A	0000000000000000	DMA3STAL	B34	0000000000000000
DMA0REQ	B02	0000000000000000	DMA1PAD	B1C	0000000000000000	DMA3STAH	B36	0000000000000000
DMA0STAL	B04	0000000000000000	DMA1CNT	B1E	0000000000000000	DMA3STBL	B38	0000000000000000
DMA0STAH	B06	0000000000000000	DMA2CON	B20	0000000000000000	DMA3STBH	B3A	0000000000000000
DMA0STBL	B08	0000000000000000	DMA2REQ	B22	0000000000000000	DMA3PAD	B3C	0000000000000000
DMA0STBH	B0A	0000000000000000	DMA2STAL	B24	0000000000000000	DMA3CNT	B3E	0000000000000000
DMA0PAD	B0C	0000000000000000	DMA2STAH	B26	0000000000000000	DMAPWC	BF0	0000000000000000
DMA0CNT	B0E	0000000000000000	DMA2STBL	B28	0000000000000000	DMARQC	BF2	0000000000000000
DMA1CON	B10	0000000000000000	DMA2STBH	B2A	0000000000000000	DMAPPS	BF4	0000000000000000
DMA1REQ	B12	0000000000000000	DMA2PAD	B2C	0000000000000000	DMALCA	BF6	0000000000001111
DMA1STAL	B14	0000000000000000	DMA2CNT	B2E	0000000000000000	DSADRL	BF8	0000000000000000
DMA1STAH	B16	0000000000000000	DMA3CON	B30	0000000000000000	DSADRH	BFA	0000000000000000

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-14: SFR BLOCK C00h-D00h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
PWM			FCLCON3	C64	0000000000000000	IOCON6	CC2	1100000000000000
PTCON	C00	0000000000000000	PDC3	C66	0000000000000000	FCLCON6	CC4	0000000000000000
PTCON2	C02	0000000000000000	PHASE3	C68	0000000000000000	PDC6	CC6	0000000000000000
PTPER	C04	1111111111111000	DTR3	C6A	0000000000000000	PHASE6	CC8	0000000000000000
SEVTCMP	C06	0000000000000000	ALTDTR3	C6C	0000000000000000	DTR6	CCA	0000000000000000
MDC	C0A	0000000000000000	SDC3	C6E	0000000000000000	ALTDTR6	CCC	0000000000000000
STCON	C0E	0000000000000000	SPHASE3	C70	0000000000000000	SDC6	CCE	0000000000000000
STCON2	C10	0000000000000000	TRIG3	C72	0000000000000000	SPHASE6	CD0	0000000000000000
STPER	C12	1111111111111000	TRGCON3	C74	0000000000000000	TRIG6	CD2	0000000000000000
SSEVTCMP	C14	0000000000000000	STRIG3	C76	0000000000000000	TRGCON6	CD4	0000000000000000
CHOP	C1A	0000000000000000	PWMCAP3	C78	0000000000000000	STRIG6	CD6	0000000000000000
PWMKEY	C1E	xxxxxxxxxxxxxxxx	LEBCON3	C7A	0000000000000000	PWMCAP6	CD8	0000000000000000
PWM Generator			LEBDLY3	C7C	0000000000000000	LEBCON6	CDA	0000000000000000
PWMCON1	C20	0000000000000000	AUXCON3	C7E	0000000000000000	LEBDLY6	CDC	0000000000000000
IOCON1	C22	1100000000000000	PWMCON4	C80	0000000000000000	AUXCON6	CDE	0000000000000000
FCLCON1	C24	0000000000000000	IOCON4	C82	1100000000000000	PWMCON7	CE0	0000000000000000
PDC1	C26	0000000000000000	FCLCON4	C84	0000000000000000	IOCON7	CE2	1100000000000000
PHASE1	C28	0000000000000000	PDC4	C86	0000000000000000	FCLCON7	CE4	0000000000000000
DTR1	C2A	0000000000000000	PHASE4	C88	0000000000000000	PDC7	CE6	0000000000000000
ALTDTR1	C2C	0000000000000000	DTR4	C8A	0000000000000000	PHASE7	CE8	0000000000000000
SDC1	C2E	0000000000000000	ALTDTR4	C8C	0000000000000000	DTR7	CEA	0000000000000000
SPHASE1	C30	0000000000000000	SDC4	C8E	0000000000000000	ALTDTR7	CEC	0000000000000000
TRIG1	C32	0000000000000000	SPHASE4	C90	0000000000000000	SDC7	CEE	0000000000000000
TRGCON1	C34	0000000000000000	TRIG4	C92	0000000000000000	SPHASE7	CF0	0000000000000000
STRIG1	C36	0000000000000000	TRGCON4	C94	0000000000000000	TRIG7	CF2	0000000000000000
PWMCAP1	C38	0000000000000000	STRIG4	C96	0000000000000000	TRGCON7	CF4	0000000000000000
LEBCON1	C3A	0000000000000000	PWMCAP4	C98	0000000000000000	STRIG7	CF6	0000000000000000
LEBDLY1	C3C	0000000000000000	LEBCON4	C9A	0000000000000000	PWMCAP7	CF8	0000000000000000
AUXCON1	C3E	0000000000000000	LEBDLY4	C9C	0000000000000000	LEBCON7	CFA	0000000000000000
PWMCON2	C40	0000000000000000	AUXCON4	C9E	0000000000000000	LEBDLY7	CFC	0000000000000000
IOCON2	C42	1100000000000000	PWMCON5	CA0	0000000000000000	AUXCON7	CFE	0000000000000000
FCLCON2	C44	0000000000000000	IOCON5	CA2	1100000000000000	PWMCON8	D00	0000000000000000
PDC2	C46	0000000000000000	FCLCON5	CA4	0000000000000000	IOCON8	D02	1100000000000000
PHASE2	C48	0000000000000000	PDC5	CA6	0000000000000000	FCLCON8	D04	0000000000000000
DTR2	C4A	0000000000000000	PHASE5	CA8	0000000000000000	PDC8	D06	0000000000000000
ALTDTR2	C4C	0000000000000000	DTR5	CAA	0000000000000000	PHASE8	D08	0000000000000000
SDC2	C4E	0000000000000000	ALTDTR5	CAC	0000000000000000	ALTDTR8	D0C	0000000000000000
SPHASE2	C50	0000000000000000	SDC5	CAE	0000000000000000	SDC8	D0E	0000000000000000
TRIG2	C52	0000000000000000	SPHASE5	CB0	0000000000000000	SPHASE8	D10	0000000000000000
TRGCON2	C54	0000000000000000	TRIG5	CB2	0000000000000000	TRIG8	D12	0000000000000000
STRIG2	C56	0000000000000000	TRGCON5	CB4	0000000000000000	TRGCON8	D14	0000000000000000
PWMCAP2	C58	0000000000000000	STRIG5	CB6	0000000000000000	STRIG8	D16	0000000000000000
LEBCON2	C5A	0000000000000000	PWMCAP5	CB8	0000000000000000	PWMCAP8	D18	0000000000000000
LEBDLY2	C5C	0000000000000000	LEBCON5	CBA	0000000000000000	LEBCON8	D1A	0000000000000000
AUXCON2	C5E	0000000000000000	LEBDLY5	CBC	0000000000000000	LEBDLY8	D1C	0000000000000000
PWMCON3	C60	0000000000000000	AUXCON5	CBE	0000000000000000	AUXCON8	D1E	0000000000000000
IOCON3	C62	1100000000000000	PWMCON6	CC0	0000000000000000			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

TABLE 4-15: SFR BLOCK E00h-F00h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
PORTA			ANSELB	E1E	0000001011101111	CNPDD	E3C	0000000000000000
TRISA	E00	0000000000011111	**PORTC**			ANSELD	E3E	0110000110100000
PORTA	E02	0000000000000000	TRISC	E20	0111011111111111	**PORTE**		
LATA	E04	0000000000000000	PORTC	E22	0000000000000000	TRISE	E40	1111111111111111
ODCA	E06	0000000000000000	LATC	E24	0000000000000000	PORTE	E42	0000000000000000
CNENA	E08	0000000000000000	ODCC	E26	0000000000000000	LATE	E44	0000000000000000
CNPUA	E0A	0000000000000000	CNENC	E28	0000000000000000	ODCE	E46	0000000000000000
CNPDA	E0C	0000000000000000	CNPUC	E2A	0000000000000000	CNENE	E48	0000000000000000
ANSELA	E0E	0000000000000111	CNPDC	E2C	0000000000000000	CNPUE	E4A	0000000000000000
PORTB			ANSELC	E2E	0001011001110111	CNPDE	E4C	0000000000000000
TRISB	E10	0111101111111111	**PORTD**			ANSELE	E4E	1100000100000000
PORTB	E12	0000000000000000	TRISD	E30	1111111111111111	**CPU**		
LATB	E14	0000000000000000	PORTD	E32	0000000000000000	VISI	F88	0000000000000000
ODCB	E16	0000000000000000	LATD	E34	0000000000000000	**JTAG**		
CNENB	E18	0000000000000000	ODCD	E36	0000000000000000	JDATAH	FF0	0000000000000000
CNPUB	E1A	0000000000000000	CNEND	E38	0000000000000000	JDATAL	FF2	0000000000000000
CNPDB	E1C	0000000000000000	CNPUD	E3A	0000000000000000			

Legend: x = unknown or indeterminate value. Address values are in hexadecimal. Reset values are in binary.

4.5.1 PAGED MEMORY SCHEME

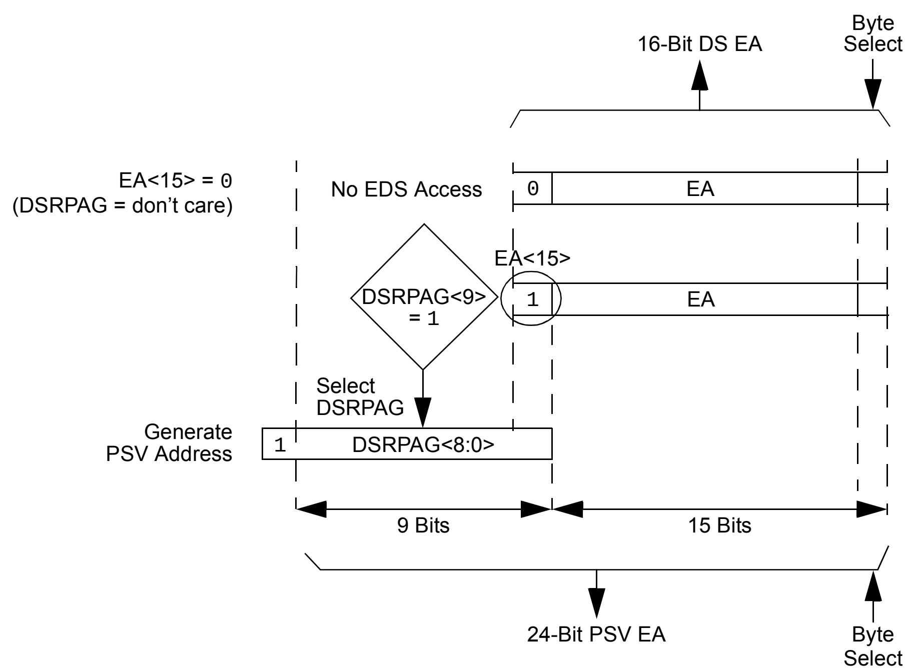
The dsPIC33EPXXXGS70X/80X family architecture extends the available Data Space through a paging scheme, which allows the available Data Space to be accessed using MOV instructions in a linear fashion for pre- and post-modified Effective Addresses (EAs). The upper half of the base Data Space address is used in conjunction with the Data Space Read Page (DSRPAG) register to form the Program Space Visibility (PSV) address.

The Data Space Read Page (DSRPAG) register is located in the SFR space. Construction of the PSV address is shown in Figure 4-7. When DSRPAG<9> = 1 and the base address bit, EA<15> = 1, the DSRPAG<8:0> bits are concatenated onto EA<14:0> to form the 24-bit PSV read address.

The paged memory scheme provides access to multiple 32-Kbyte windows in the PSV memory. The Data Space Read Page (DSRPAG) register, in combination with the upper half of the Data Space address, can provide up to 8 Mbytes of PSV address space. The paged data memory space is shown in Figure 4-8.

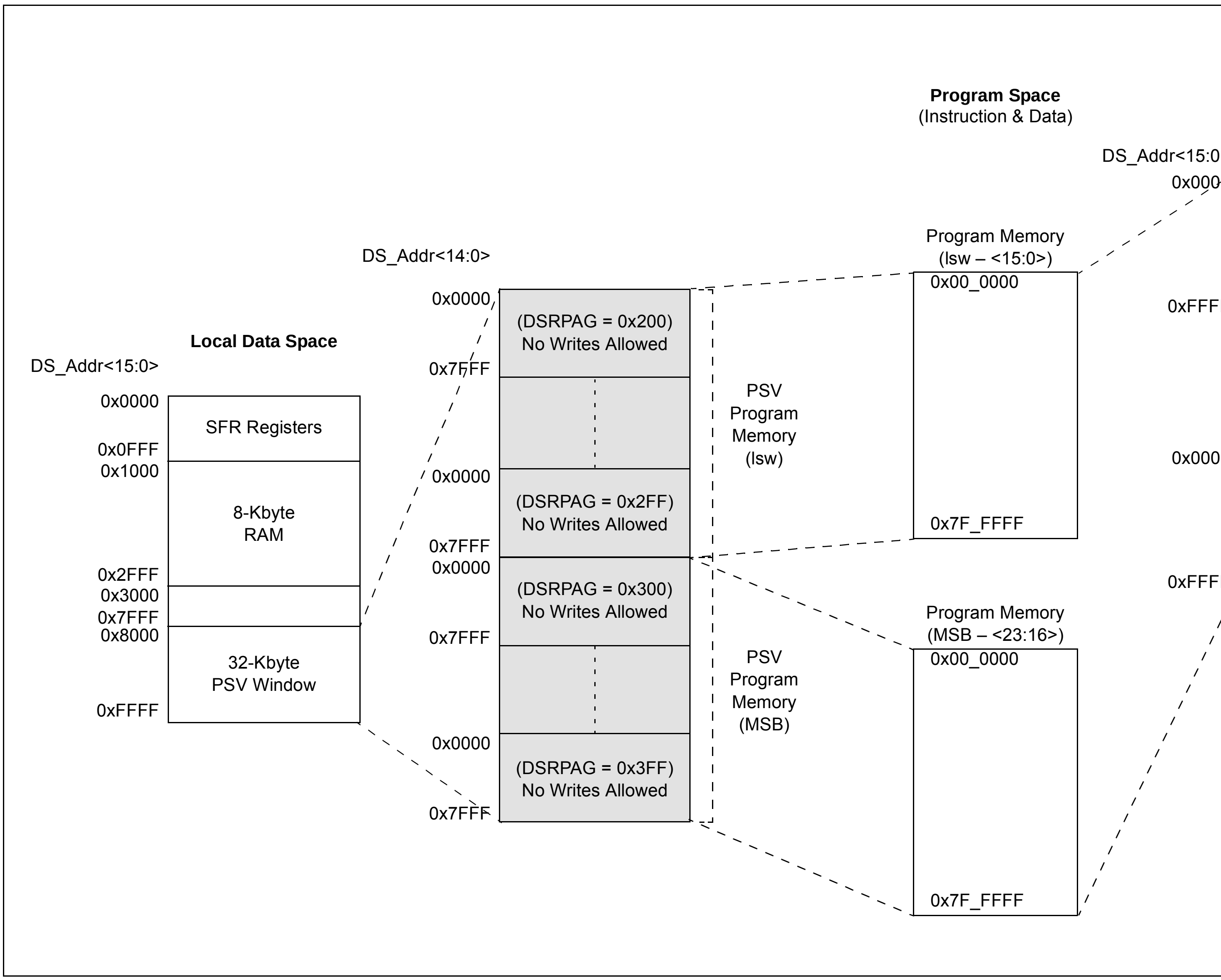
The Program Space (PS) can be accessed with a DSRPAG of 0x200 or greater. Only reads from PS are supported using the DSRPAG register.

FIGURE 4-7: PROGRAM SPACE VISIBILITY (PSV) READ ADDRESS GENERATION



Note: DS read access when DSRPAG = 0x000 will force an address error trap.

FIGURE 4-8: PAGED DATA MEMORY SPACE



When a PSV page overflow or underflow occurs, EA<15> is cleared as a result of the register indirect EA calculation. An overflow or underflow of the EA in the PSV pages can occur at the page boundaries when:

- The initial address, prior to modification, addresses the PSV page
- The EA calculation uses Pre- or Post-Modified Register Indirect Addressing; however, this does not include Register Offset Addressing

In general, when an overflow is detected, the DSRPAG register is incremented and the EA<15> bit is set to keep the base address within the PSV window. When an underflow is detected, the DSRPAG register is decremented and the EA<15> bit is set to keep the base address within the PSV window. This creates a linear PSV address space, but only when using Register Indirect Addressing modes.

Exceptions to the operation described above arise when entering and exiting the boundaries of Page 0 and PSV spaces. Table 4-16 lists the effects of overflow and underflow scenarios at different boundaries.

In the following cases, when overflow or underflow occurs, the EA<15> bit is set and the DSRPAG is not modified; therefore, the EA will wrap to the beginning of the current page:

- Register Indirect with Register Offset Addressing
- Modulo Addressing
- Bit-Reversed Addressing

TABLE 4-16: OVERFLOW AND UNDERFLOW SCENARIOS AT PAGE 0 AND PSV SPACE BOUNDARIES(2,3,4)

O/U, R/W	Operation	Before			After		
		DSxPAG	DS EA<15>	Page Description	DSxPAG	DS EA<15>	Page Description
O, Read	`[++Wn]` or `[Wn++]`	DSRPAG = 0x2FF	1	PSV: Last lsw page	DSRPAG = 0x300	1	PSV: First MSB page
O, Read		DSRPAG = 0x3FF	1	PSV: Last MSB page	DSRPAG = 0x3FF	0	See Note 1
U, Read	`[--Wn]` or `[Wn--]`	DSRPAG = 0x001	1	PSV page	DSRPAG = 0x001	0	See Note 1
U, Read		DSRPAG = 0x200	1	PSV: First lsw page	DSRPAG = 0x200	0	See Note 1
U, Read		DSRPAG = 0x300	1	PSV: First MSB page	DSRPAG = 0x2FF	1	PSV: Last lsw page

Legend: O = Overflow, U = Underflow, R = Read, W = Write

Note 1: The Register Indirect Addressing now addresses a location in the base Data Space (0x0000-0x7FFF).

2: An EDS access, with DSRPAG = 0x000, will generate an address error trap.

3: Only reads from PS are supported using DSRPAG.

4: Pseudolinear Addressing is not supported for large offsets.

4.5.2 EXTENDED X DATA SPACE

The lower portion of the base address space range, between 0x0000 and 0x7FFF, is always accessible, regardless of the contents of the Data Space Read Page register. It is indirectly addressable through the register indirect instructions. It can be regarded as being located in the default EDS Page 0 (i.e., EDS address range of 0x000000 to 0x007FFF with the base address bit, EA<15> = `0`, for this address range). However, Page 0 cannot be accessed through the upper 32 Kbytes, 0x8000 to 0xFFFF, of base Data Space in combination with DSRPAG = 0x00. Consequently, DSRPAG is initialized to 0x001 at Reset.

> **Note 1:** DSRPAG should not be used to access Page 0. An EDS access with DSRPAG set to 0x000 will generate an address error trap.
>
> **2:** Clearing the DSRPAG in software has no effect.

The remaining PSV pages are only accessible using the DSRPAG register in combination with the upper 32 Kbytes, 0x8000 to 0xFFFF, of the base address, where base address bit, EA<15> = `1`.

4.5.3 SOFTWARE STACK

The W15 register serves as a dedicated Software Stack Pointer (SSP), and is automatically modified by exception processing, subroutine calls and returns; however, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies reading, writing and manipulating the Stack Pointer (for example, creating stack frames).

> **Note:** To protect against misaligned stack accesses, W15<0> is fixed to '`0`' by the hardware.

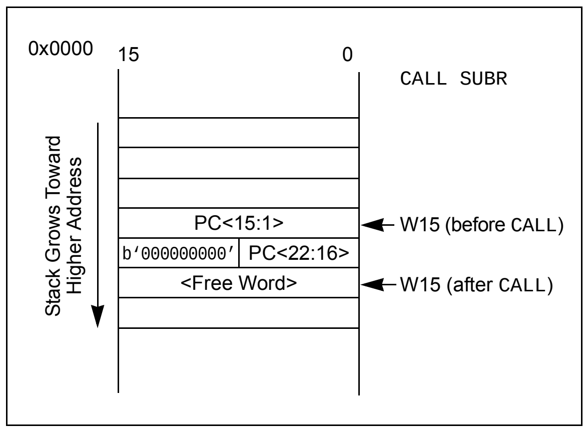
W15 is initialized to 0x1000 during all Resets. This address ensures that the SSP points to valid RAM in all dsPIC33EPXXXGS70X/80X devices and permits stack availability for non-maskable trap exceptions. These can occur before the SSP is initialized by the user software. You can reprogram the SSP during initialization to any location within Data Space.

The Software Stack Pointer always points to the first available free word and fills the software stack, working from lower toward higher addresses. Figure 4-9 illustrates how it pre-decrements for a stack pop (read) and post-increments for a stack push (writes).

When the PC is pushed onto the stack, PC<15:0> are pushed onto the first available stack word, then PC<22:16> are pushed into the second available stack location. For a PC push during any `CALL` instruction, the MSB of the PC is zero-extended before the push, as shown in Figure 4-9. During exception processing, the MSB of the PC is concatenated with the lower 8 bits of the CPU STATUS Register, SR. This allows the contents of SRL to be preserved automatically during interrupt processing.

> **Note 1:** To maintain system Stack Pointer (W15) coherency, W15 is never subject to (EDS) paging, and is therefore, restricted to an address range of 0x0000 to 0xFFFF. The same applies to the W14 when used as a Stack Frame Pointer (SFA = `1`).
>
> **2:** As the stack can be placed in, and can access X and Y spaces, care must be taken regarding its use, particularly with regard to local automatic variables in a C development environment

FIGURE 4-9: CALL STACK FRAME



4.6 Instruction Addressing Modes

The addressing modes shown in Table 4-17 form the basis of the addressing modes optimized to support the specific features of individual instructions. The addressing modes provided in the MAC class of instructions differ from those in the other instruction types.

4.6.1 FILE REGISTER INSTRUCTIONS

Most file register instructions use a 13-bit address field (f) to directly address data present in the first 8192 bytes of data memory (Near Data Space). Most file register instructions employ a Working register, W0, which is denoted as WREG in these instructions. The destination is typically either the same file register or WREG (with the exception of the MUL instruction), which writes the result to a register or register pair. The MOV instruction allows additional flexibility and can access the entire Data Space.

4.6.2 MCU INSTRUCTIONS

The three-operand MCU instructions are of the form:

```
Operand 3 = Operand 1 <function> Operand 2
```

where Operand 1 is always a Working register (that is, the addressing mode can only be Register Direct), which is referred to as Wb. Operand 2 can be a W register fetched from data memory or a 5-bit literal. The result location can either be a W register or a data memory location. The following addressing modes are supported by MCU instructions:

- Register Direct
- Register Indirect
- Register Indirect Post-Modified
- Register Indirect Pre-Modified
- 5-Bit or 10-Bit Literal

Note: Not all instructions support all the addressing modes given above. Individual instructions can support different subsets of these addressing modes.

TABLE 4-17: FUNDAMENTAL ADDRESSING MODES SUPPORTED

Addressing Mode	Description
File Register Direct	The address of the file register is specified explicitly.
Register Direct	The contents of a register are accessed directly.
Register Indirect	The contents of Wn form the Effective Address (EA).
Register Indirect Post-Modified	The contents of Wn form the EA. Wn is post-modified (incremented or decremented) by a constant value.
Register Indirect Pre-Modified	Wn is pre-modified (incremented or decremented) by a signed constant value to form the EA.
Register Indirect with Register Offset (Register Indexed)	The sum of Wn and Wb forms the EA.
Register Indirect with Literal Offset	The sum of Wn and a literal forms the EA.

4.6.3 MOVE AND ACCUMULATOR INSTRUCTIONS

Move instructions, and the DSP accumulator class of instructions, provide a greater degree of addressing flexibility than other instructions. In addition to the addressing modes supported by most MCU instructions, move and accumulator instructions also support Register Indirect with Register Offset Addressing mode, also referred to as Register Indexed mode.

Note:	For the MOV instructions, the addressing mode specified in the instruction can differ for the source and destination EA. However, the 4-bit Wb (Register Offset) field is shared by both source and destination (but typically only used by one).

In summary, the following addressing modes are supported by move and accumulator instructions:

- Register Direct
- Register Indirect
- Register Indirect Post-modified
- Register Indirect Pre-modified
- Register Indirect with Register Offset (Indexed)
- Register Indirect with Literal Offset
- 8-Bit Literal
- 16-Bit Literal

Note:	Not all instructions support all the addressing modes given above. Individual instructions may support different subsets of these addressing modes.

4.6.4 MAC INSTRUCTIONS

The dual source operand DSP instructions (CLR, ED, EDAC, MAC, MPY, MPY.N, MOVSAC and MSC), also referred to as MAC instructions, use a simplified set of addressing modes to allow the user application to effectively manipulate the Data Pointers through register indirect tables.

The two-source operand prefetch registers must be members of the set {W8, W9, W10, W11}. For data reads, W8 and W9 are always directed to the X RAGU, and W10 and W11 are always directed to the Y AGU. The Effective Addresses generated (before and after modification) must therefore, be valid addresses within X Data Space for W8 and W9, and Y Data Space for W10 and W11.

Note:	Register Indirect with Register Offset Addressing mode is available only for W9 (in X space) and W11 (in Y space).

In summary, the following addressing modes are supported by the MAC class of instructions:

- Register Indirect
- Register Indirect Post-Modified by 2
- Register Indirect Post-Modified by 4
- Register Indirect Post-Modified by 6
- Register Indirect with Register Offset (Indexed)

4.6.5 OTHER INSTRUCTIONS

Besides the addressing modes outlined previously, some instructions use literal constants of various sizes. For example, BRA (branch) instructions use 16-bit signed literals to specify the branch destination directly, whereas the DISI instruction uses a 14-bit unsigned literal field. In some instructions, such as ULNK, the source of an operand or result is implied by the opcode itself. Certain operations, such as a NOP, do not have any operands.

4.7 Modulo Addressing

Modulo Addressing mode is a method of providing an automated means to support circular data buffers using hardware. The objective is to remove the need for software to perform data address boundary checks when executing tightly looped code, as is typical in many DSP algorithms.

Modulo Addressing can operate in either Data or Program Space (since the Data Pointer mechanism is essentially the same for both). One circular buffer can be supported in each of the X (which also provides the pointers into Program Space) and Y Data Spaces. Modulo Addressing can operate on any W Register Pointer. However, it is not advisable to use W14 or W15 for Modulo Addressing since these two registers are used as the Stack Frame Pointer and Stack Pointer, respectively.

In general, any particular circular buffer can be configured to operate in only one direction, as there are certain restrictions on the buffer start address (for incrementing buffers) or end address (for decrementing buffers), based upon the direction of the buffer.

The only exception to the usage restrictions is for buffers that have a power-of-two length. As these buffers satisfy the start and end address criteria, they can operate in a Bidirectional mode (that is, address boundary checks are performed on both the lower and upper address boundaries).

4.7.1 START AND END ADDRESS

The Modulo Addressing scheme requires that a starting and ending address be specified and loaded into the 16-bit Modulo Buffer Address registers: XMODSRT, XMODEND, YMODSRT and YMODEND (see Table 4-2).

Note:	Y space Modulo Addressing EA calculations assume word-sized data (LSb of every EA is always clear).

The length of a circular buffer is not directly specified. It is determined by the difference between the corresponding start and end addresses. The maximum possible length of the circular buffer is 32K words (64 Kbytes).

4.7.2 W ADDRESS REGISTER SELECTION

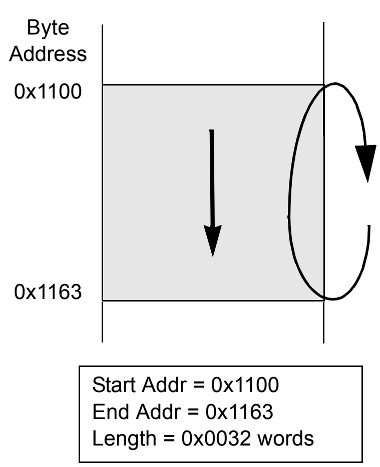
The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags, as well as a W register field to specify the W Address registers. The XWM and YWM fields select the registers that operate with Modulo Addressing:

- If XWM = `1111`, X RAGU and X WAGU Modulo Addressing is disabled
- If YWM = `1111`, Y AGU Modulo Addressing is disabled

The X Address Space Pointer W (XWM) register, to which Modulo Addressing is to be applied, is stored in MODCON<3:0> (see Table 4-2). Modulo Addressing is enabled for X Data Space when XWM is set to any value other than ‘`1111`’ and the XMODEN bit is set (MODCON<15>).

The Y Address Space Pointer W (YWM) register, to which Modulo Addressing is to be applied, is stored in MODCON<7:4>. Modulo Addressing is enabled for Y Data Space when YWM is set to any value other than ‘`1111`’ and the YMODEN bit (MODCON<14>) is set.

FIGURE 4-10: MODULO ADDRESSING OPERATION EXAMPLE




```
MOV     #0x1100, W0
MOV     W0, XMODSRT     ;set modulo start address
MOV     #0x1163, W0
MOV     W0, MODEND      ;set modulo end address
MOV     #0x8001, W0
MOV     W0, MODCON      ;enable W1, X AGU for modulo

MOV     #0x0000, W0     ;W0 holds buffer fill value

MOV     #0x1110, W1     ;point W1 to buffer

DO      AGAIN, #0x31    ;fill the 50 buffer locations
MOV     W0, [W1++]      ;fill the next location
AGAIN:  INC W0, W0      ;increment the fill value
```

4.7.3 MODULO ADDRESSING APPLICABILITY

Modulo Addressing can be applied to the Effective Address (EA) calculation associated with any W register. Address boundaries check for addresses equal to:

- The upper boundary addresses for incrementing buffers
- The lower boundary addresses for decrementing buffers

It is important to realize that the address boundaries check for addresses less than or greater than the upper (for incrementing buffers) and lower (for decrementing buffers) boundary addresses (not just equal to). Address changes can, therefore, jump beyond boundaries and still be adjusted correctly.

Note:	The modulo corrected Effective Address is written back to the register only when Pre-Modify or Post-Modify Addressing mode is used to compute the Effective Address. When an address offset (such as [W7 + W2]) is used, Modulo Addressing correction is performed, but the contents of the register remain unchanged.

4.8 Bit-Reversed Addressing

Bit-Reversed Addressing mode is intended to simplify data reordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which can be a constant value or register contents, is regarded as having its bit order reversed. The address source and destination are kept in normal order. Thus, the only operand requiring reversal is the modifier.

4.8.1 BIT-REVERSED ADDRESSING IMPLEMENTATION

Bit-Reversed Addressing mode is enabled when all of these situations are met:

- BWMx bits (W register selection) in the MODCON register are any value other than ‘1111’ (the stack cannot be accessed using Bit-Reversed Addressing)
- The BREN bit is set in the XBREV register
- The addressing mode used is Register Indirect with Pre-Increment or Post-Increment

If the length of a bit-reversed buffer is $M = 2^N$ bytes, the last ‘N’ bits of the data buffer start address must be zeros.

XB<14:0> is the Bit-Reversed Addressing modifier, or ‘pivot point’, which is typically a constant. In the case of an FFT computation, its value is equal to half of the FFT data buffer size.

Note:	All bit-reversed EA calculations assume word-sized data (LSb of every EA is always clear). The XB value is scaled accordingly to generate compatible (byte) addresses.

When enabled, Bit-Reversed Addressing is executed only for Register Indirect with Pre-Increment or Post-Increment Addressing and word-sized data writes. It does not function for any other addressing mode or for byte-sized data and normal addresses are generated instead. When Bit-Reversed Addressing is active, the W Address Pointer is always added to the address modifier (XB) and the offset associated with the Register Indirect Addressing mode is ignored. In addition, as word-sized data is a requirement, the LSb of the EA is ignored (and always clear).

Note:	Modulo Addressing and Bit-Reversed Addressing can be enabled simultaneously using the same W register, but Bit-Reversed Addressing operation will always take precedence for data writes when enabled.

If Bit-Reversed Addressing has already been enabled by setting the BREN (XBREV<15>) bit, a write to the XBREV register should not be immediately followed by an indirect read operation using the W register that has been designated as the Bit-Reversed Pointer.

FIGURE 4-11: BIT-REVERSED ADDRESSING EXAMPLE

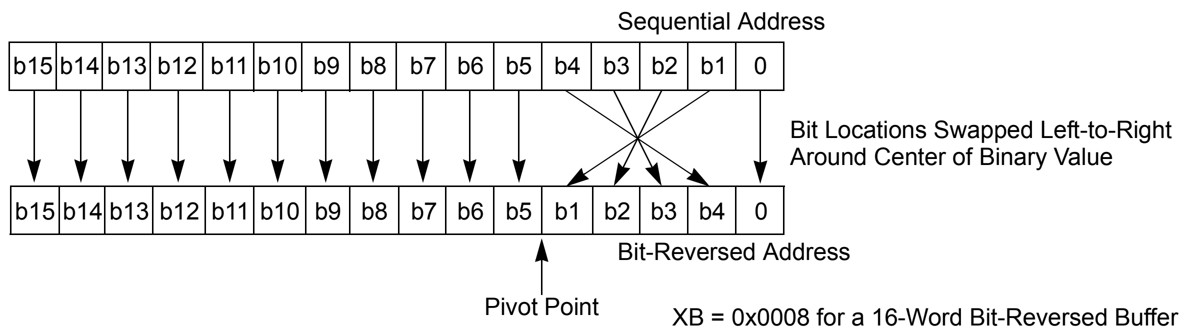


TABLE 4-18: BIT-REVERSED ADDRESSING SEQUENCE (16-ENTRY)

Normal Address					Bit-Reversed Address				
A3	**A2**	**A1**	**A0**	**Decimal**	**A3**	**A2**	**A1**	**A0**	**Decimal**
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	8
0	0	1	0	2	0	1	0	0	4
0	0	1	1	3	1	1	0	0	12
0	1	0	0	4	0	0	1	0	2
0	1	0	1	5	1	0	1	0	10
0	1	1	0	6	0	1	1	0	6
0	1	1	1	7	1	1	1	0	14
1	0	0	0	8	0	0	0	1	1
1	0	0	1	9	1	0	0	1	9
1	0	1	0	10	0	1	0	1	5
1	0	1	1	11	1	1	0	1	13
1	1	0	0	12	0	0	1	1	3
1	1	0	1	13	1	0	1	1	11
1	1	1	0	14	0	1	1	1	7
1	1	1	1	15	1	1	1	1	15

4.9 Interfacing Program and Data Memory Spaces

The dsPIC33EPXXXGS70X/80X family architecture uses a 24-bit wide Program Space (PS) and a 16-bit wide Data Space (DS). The architecture is also a modified Harvard scheme, meaning that data can also be present in the Program Space. To use this data successfully, it must be accessed in a way that preserves the alignment of information in both spaces.

Aside from normal execution, the architecture of the dsPIC33EPXXXGS70X/80X family devices provides two methods by which Program Space can be accessed during operation:

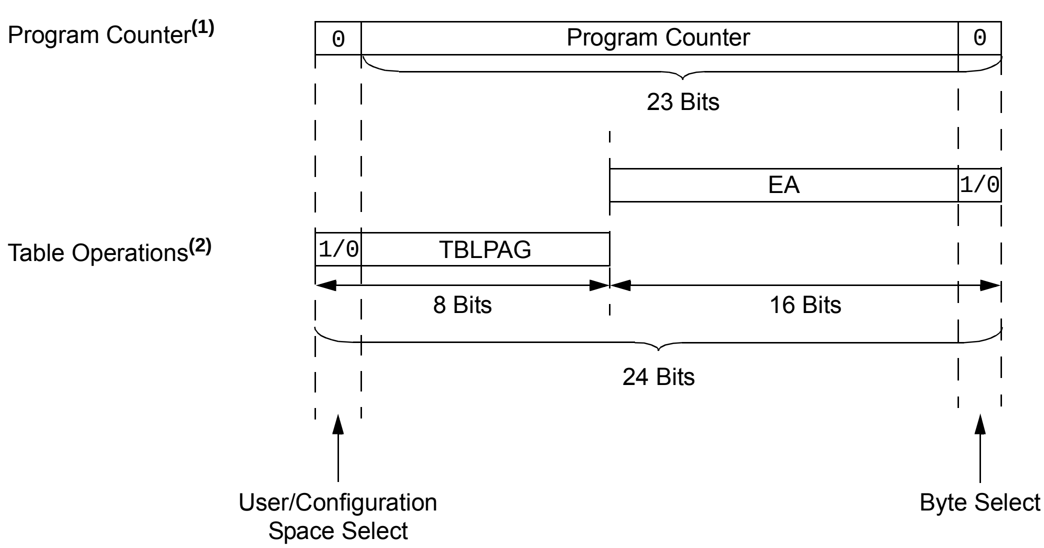
- Using table instructions to access individual bytes or words anywhere in the Program Space
- Remapping a portion of the Program Space into the Data Space (Program Space Visibility)

Table instructions allow an application to read or write to small areas of the program memory. This capability makes the method ideal for accessing data tables that need to be updated periodically. It also allows access to all bytes of the program word. The remapping method allows an application to access a large block of data on a read-only basis, which is ideal for look-ups from a large table of static data. The application can only access the least significant word of the program word.

TABLE 4-19: PROGRAM SPACE ADDRESS CONSTRUCTION

Access Type	Access Space	Program Space Address				
		<23>	<22:16>	<15>	<14:1>	<0>
Instruction Access (Code Execution)	User	0	PC<22:1>			0
		0xxx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (Byte/Word Read/Write)	User	TBLPAG<7:0>		Data EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	Configuration	TBLPAG<7:0>		Data EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		

FIGURE 4-12: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



Note 1: The Least Significant bit (LSb) of Program Space addresses is always fixed as '0' to maintain word alignment of data in the Program and Data Spaces.

2: Table operations are not required to be word-aligned. Table Read operations are permitted in the configuration memory space.

4.9.1 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

The `TBLRDL` and `TBLWTL` instructions offer a direct method of reading or writing the lower word of any address within the Program Space without going through Data Space. The `TBLRDH` and `TBLWTH` instructions are the only method to read or write the upper 8 bits of a Program Space word as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to Data Space addresses. Program memory can thus be regarded as two 16-bit wide word address spaces, residing side by side, each with the same address range. `TBLRDL` and `TBLWTL` access the space that contains the least significant data word. `TBLRDH` and `TBLWTH` access the space that contains the upper data byte.

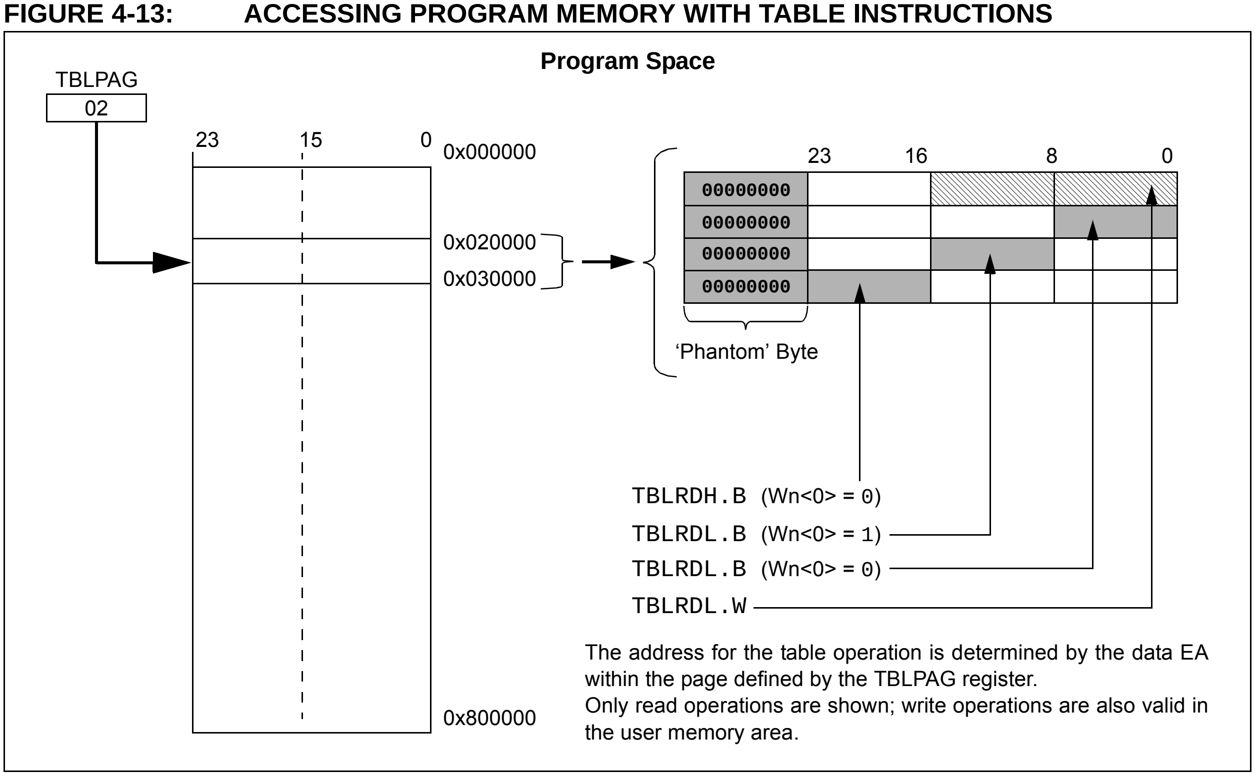
Two table instructions are provided to move byte or word-sized (16-bit) data to and from Program Space. Both function as either byte or word operations.

- `TBLRDL` (Table Read Low):
  - In Word mode, this instruction maps the lower word of the Program Space location (P<15:0>) to a data address (D<15:0>)
  - In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when Byte Select is '`1`'; the lower byte is selected when it is '`0`'.
- `TBLRDH` (Table Read High):
  - In Word mode, this instruction maps the entire upper word of a program address (P<23:16>) to a data address. The 'phantom' byte (D<15:8>) is always '`0`'.
  - In Byte mode, this instruction maps the upper or lower byte of the program word to D<7:0> of the data address in the `TBLRDL` instruction. The data is always '`0`' when the upper 'phantom' byte is selected (Byte Select = `1`).

In a similar fashion, two table instructions, `TBLWTH` and `TBLWTL`, are used to write individual bytes or words to a Program Space address. The details of their operation are explained in **Section 5.0 "Flash Program Memory"**.

For all table operations, the area of program memory space to be accessed is determined by the Table Page register (TBLPAG). TBLPAG covers the entire program memory space of the device, including user application and configuration spaces. When TBLPAG<7> = `0`, the table page is located in the user memory space. When TBLPAG<7> = `1`, the page is located in configuration space.

FIGURE 4-13: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



NOTES:

5.0 FLASH PROGRAM MEMORY

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Dual Partition Flash Program Memory"** (DS70005156) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com)
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family devices contain internal Program Flash Memory for storing and executing application code. The memory is readable, writable and erasable during normal operation over the entire VDD range.

Flash memory can be programmed in three ways:

- In-Circuit Serial Programming™ (ICSP™) programming capability
- Enhanced In-Circuit Serial Programming (Enhanced ICSP)
- Run-Time Self-Programming (RTSP)

ICSP allows for a dsPIC33EPXXXGS70X/80X family device to be serially programmed while in the end application circuit. This is done with a programming clock and programming data (PGECx/PGEDx) line, and three other lines for power (VDD), ground (VSS) and Master Clear ($\overline{MCLR}$). This allows customers to manufacture boards with unprogrammed devices and then program the device just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

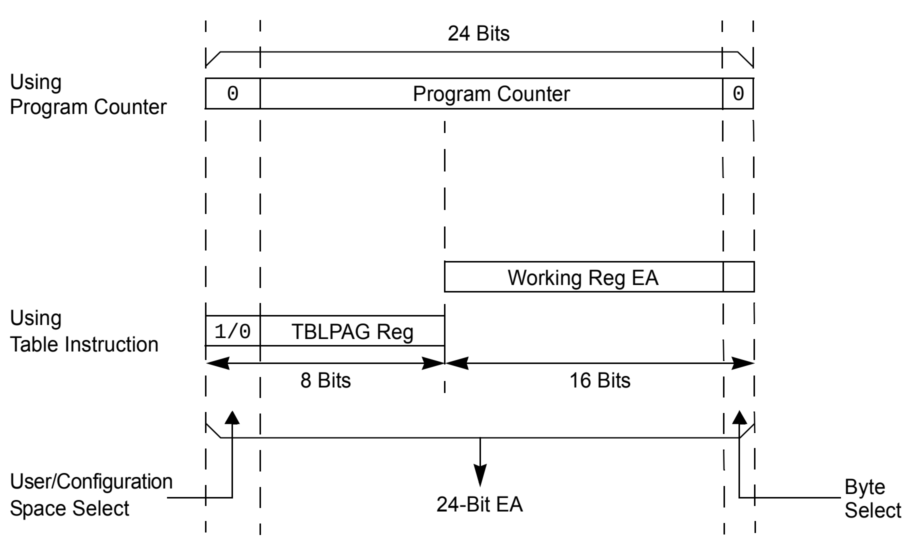
Enhanced In-Circuit Serial Programming uses an on-board bootloader, known as the Program Executive, to manage the programming process. Using an SPI data frame format, the Program Executive can erase, program and verify program memory. For more information on Enhanced ICSP, see the device programming specification.

RTSP is accomplished using `TBLRD` (Table Read) and `TBLWT` (Table Write) instructions. With RTSP, the user application can write program memory data with a single program memory word and erase program memory in blocks or 'pages' of 512 instructions (1536 bytes) at a time.

5.1 Table Instructions and Flash Programming

Regardless of the method used, all programming of Flash memory is done with the Table Read and Table Write instructions. These instructions allow direct read and write access to the program memory space, from the data memory, while the device is in normal operating mode. The 24-bit target address in the program memory is formed using bits<7:0> of the TBLPAG register and the Effective Address (EA) from a W register, specified in the table instruction, as shown in Figure 5-1. The `TBLRDL` and the `TBLWTL` instructions are used to read or write to bits<15:0> of program memory. `TBLRDL` and `TBLWTL` can access program memory in both Word and Byte modes. The `TBLRDH` and `TBLWTH` instructions are used to read or write to bits<23:16> of program memory. `TBLRDH` and `TBLWTH` can also access program memory in Word or Byte mode.

FIGURE 5-1: ADDRESSING FOR TABLE REGISTERS



5.2 RTSP Operation

The dsPIC33EPXXXGS70X/80X family Flash program memory array is organized into rows of 64 instructions or 192 bytes. RTSP allows the user application to erase a single page (8 rows or 512 instructions) of memory at a time and to program one row at a time. It is possible to program two instructions at a time as well.

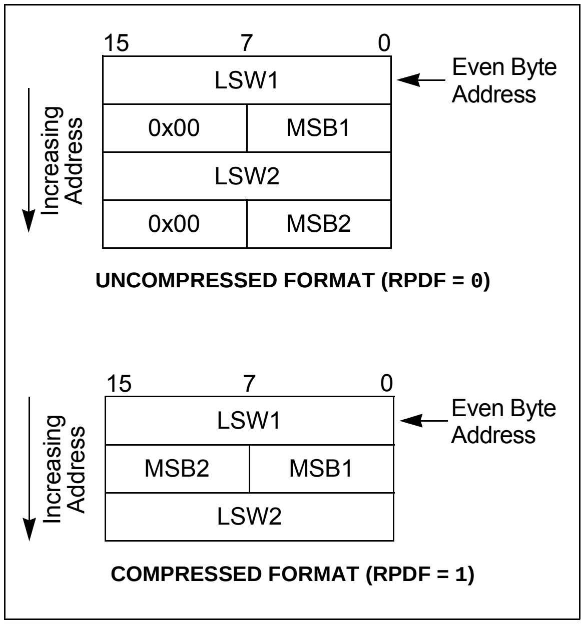
The page erase and single row write blocks are edge-aligned, from the beginning of program memory, on boundaries of 1536 bytes and 192 bytes, respectively. Figure 30-14 in **Section 30.0 "Electrical Characteristics"** lists the typical erase and programming times.

Row programming is performed by loading 192 bytes into data memory and then loading the address of the first byte in that row into the NVMSRCADR register. Once the write has been initiated, the device will automatically load the write latches and increment the NVMSRCADR and the NVMADR(U) registers until all bytes have been programmed. The RPDF bit (NVMCON<9>) selects the format of the stored data in RAM to be either compressed or uncompressed. See Figure 5-2 for data formatting. Compressed data helps to reduce the amount of required RAM by using the upper byte of the second word for the MSB of the second instruction.

The basic sequence for RTSP word programming is to use the `TBLWTL` and `TBLWTH` instructions to load two of the 24-bit instructions into the write latches found in configuration memory space. Refer to Figure 4-1 through Figure 4-4 for write latch addresses. Programming is performed by unlocking and setting the control bits in the NVMCON register.

All erase and program operations may optionally use the NVM interrupt to signal the successful completion of the operation. For example, when performing Flash write operations on the Inactive Partition in Dual Partition mode, where the CPU remains running, it is necessary to wait for the NVM interrupt before programming the next block of Flash program memory.

FIGURE 5-2: UNCOMPRESSED/ COMPRESSED FORMAT



5.3 Programming Operations

A complete programming sequence is necessary for programming or erasing the internal Flash in RTSP mode. The processor stalls (waits) until the programming operation is finished. Setting the WR bit (NVMCON<15>) starts the operation and the WR bit is automatically cleared when the operation is finished.

5.3.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

Programmers can program two adjacent words (24 bits x 2) of Program Flash Memory at a time on every other word address boundary (0x000000, 0x000004, 0x000008, etc.). To do this, it is necessary to erase the page that contains the desired address of the location the user wants to change. For protection against accidental operations, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user application must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be `NOP`s.

5.4 Dual Partition Flash Configuration

For dsPIC33EPXXXGS70X/80X devices operating in Dual Partition Flash Program Memory modes, the Inactive Partition can be erased and programmed without stalling the processor. The same programming algorithms are used for programming and erasing the Flash in the Inactive Partition, as described in **Section 5.2 "RTSP Operation"**. On top of the page erase option, the entire Flash memory of the Inactive Partition can be erased by configuring the NVMOP<3:0> bits in the NVMCON register.

> **Note 1:** The application software to be loaded into the Inactive Partition will have the address of the Active Partition. The bootloader firmware will need to offset the address by 0x400000 in order to write to the Inactive Partition.

5.4.1 FLASH PARTITION SWAPPING

The Boot Sequence Number is used for determining the Active Partition at start-up and is encoded within the FBTSEQ Configuration register bits. Unlike most Configuration registers, which only utilize the lower 16 bits of the program memory, FBTSEQ is a 24-bit Configuration Word. The Boot Sequence Number (BSEQ) is a 12-bit value and is stored in FBTSEQ twice. The true value is stored in bits, FBTSEQ<11:0>, and its complement is stored in bits, FBTSEQ<23:12>. At device Reset, the sequence numbers are read and the partition with the lowest sequence number becomes the Active Partition. If one of the Boot Sequence Numbers is invalid, the device will select the partition with the valid Boot Sequence Number, or default to Partition 1 if both sequence numbers are invalid. See **Section 27.0 "Special Features"** for more information.

The `BOOTSWP` instruction provides an alternative means of swapping the Active and Inactive Partitions (soft swap) without the need for a device Reset. The `BOOTSWP` must always be followed by a `GOTO` instruction. The `BOOTSWP` instruction swaps the Active and Inactive Partitions, and the PC vectors to the location specified by the `GOTO` instruction in the newly Active Partition.

It is important to note that interrupts should temporarily be disabled while performing the soft swap sequence and that after the partition swap, all peripherals and interrupts which were enabled remain enabled. Additionally, the RAM and stack will maintain state after the switch. As a result, it is recommended that applications using soft swaps jump to a routine that will reinitialize the device in order to ensure the firmware runs as expected. The Configuration registers will have no effect during a soft swap.

For robustness of operation, in order to execute the `BOOTSWP` instruction, it is necessary to execute the NVM unlocking sequence as follows:

1. Write 0x55 to NVMKEY.
2. Write 0xAA to NVMKEY.
3. Execute the `BOOTSWP` instruction.

If the unlocking sequence is not performed, the `BOOTSWP` instruction will be executed as a forced `NOP` and a `GOTO` instruction, following the `BOOTSWP` instruction, will be executed, causing the PC to jump to that location in the current operating partition.

The SFTSWP and P2ACTIV bits in the NVMCON register are used to determine a successful swap of the Active and Inactive Partitions, as well as which partition is active. After the `BOOTSWP` and `GOTO` instructions, the SFTSWP bit should be polled to verify the partition swap has occurred and then cleared for the next panel swap event.

5.4.2 DUAL PARTITION MODES

While operating in Dual Partition mode, the dsPIC33EPXXXGS70X/80X family devices have the option for both partitions to have their own defined security segments, as shown in Figure 27-4. Alternatively, the device can operate in Protected Dual Partition mode, where Partition 1 becomes permanently erase/write-protected. Protected Dual Partition mode allows for a "Factory Default" mode, which provides a fail-safe backup image to be stored in Partition 1.

dsPIC33EPXXXGS70X/80X family devices can also operate in Privileged Dual Partition mode, where additional security protections are implemented to allow for protection of intellectual property when multiple parties have software within the device. In Privileged Dual Partition mode, both partitions place additional restrictions on the FBSLIM register. These prevent changes to the size of the Boot Segment and General Segment, ensuring that neither segment will be altered.

5.5 Flash Memory Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

5.5.1 KEY RESOURCES

- **"Dual Partition Flash Program Memory"** (DS70005156) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

5.6 Control Registers

Five SFRs are used to write and erase the Program Flash Memory: NVMCON, NVMKEY, NVMADR, NVMADRU and NVMSRCADR/H.

The NVMCON register (Register 5-1) selects the operation to be performed (page erase, word/row program, Inactive Partition erase), initiates the program or erase cycle and is used to determine the Active Partition in Dual Partition modes.

NVMKEY (Register 5-4) is a write-only register that is used for write protection. To start a programming or erase sequence, the user application must consecutively write 0x55 and 0xAA to the NVMKEY register.

There are two NVM Address registers: NVMADRU and NVMADR. These two registers, when concatenated, form the 24-bit Effective Address (EA) of the selected word/row for programming operations, or the selected page for erase operations. The NVMADRU register is used to hold the upper 8 bits of the EA, while the NVMADR register is used to hold the lower 16 bits of the EA.

For row programming operation, data to be written to Program Flash Memory is written into data memory space (RAM) at an address defined by the NVMSRCADR register (location of first element in row programming data).

REGISTER 5-1: NVMCON: NONVOLATILE MEMORY (NVM) CONTROL REGISTER

R/SO-0[1]	R/W-0[1]	R/W-0[1]	R/W-0	R/C-0	R-0	R/W-0	R/C-0
WR	WREN	WRERR	NVMSIDL[2]	SFTSWP[6]	P2ACTIV[6]	RPDF	URERR
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0[1]	R/W-0[1]	R/W-0[1]	R/W-0[1]
—	—	—	—	NVMOP3[3,4]	NVMOP2[3,4]	NVMOP1[3,4]	NVMOP0[3,4]
bit 7							bit 0

Legend:	C = Clearable bit	SO = Settable Only bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **WR:** Write Control bit[1]
1 = Initiates a Program Flash Memory or erase operation; the operation is self-timed and the bit is cleared by hardware once the operation is complete
0 = Program or erase operation is complete and inactive

bit 14 **WREN:** Write Enable bit[1]
1 = Enables Flash program/erase operations
0 = Inhibits Flash program/erase operations

bit 13 **WRERR:** Write Sequence Error Flag bit[1]
1 = An improper program or erase sequence attempt, or termination has occurred (bit is set automatically on any set attempt of the WR bit)
0 = The program or erase operation completed normally

bit 12 **NVMSIDL:** NVM Stop in Idle Control bit[2]
1 = Flash voltage regulator goes into Standby mode during Idle mode
0 = Flash voltage regulator is active during Idle mode

bit 11 **SFTSWP:** Partition Soft Swap Status bit[6]
1 = Partitions have been successfully swapped using the BOOTSWP instruction (soft swap)
0 = Awaiting successful partition swap using the BOOTSWP instruction or a device Reset will determine the Active Partition based on the FBTSEQ register

bit 10 **P2ACTIV:** Partition 2 Active Status bit[6]
1 = Partition 2 Flash is mapped into the active region
0 = Partition 1 Flash is mapped into the active region

bit 9 **RPDF:** Row Programming Data Format bit
1 = Row data to be stored in RAM is in compressed format
0 = Row data to be stored in RAM is in uncompressed format

bit 8 **URERR:** Row Programming Data Underrun Error bit
1 = Indicates row programming operation has been terminated
0 = No data underrun error is detected

bit 7-4 **Unimplemented:** Read as ‘0’

Note 1: These bits can only be reset on a POR.
2: If this bit is set, power consumption will be further reduced (I_{IDLE}) and upon exiting Idle mode, there is a delay (T_{VREG}) before Flash memory becomes operational.
3: All other combinations of NVMOP<3:0> are unimplemented.
4: Execution of the PWRSAV instruction is ignored while any of the NVM operations are in progress.
5: Two adjacent words on a 4-word boundary are programmed during execution of this operation.
6: Only applicable when operating in Dual Partition mode.

REGISTER 5-1: NVMCON: NONVOLATILE MEMORY (NVM) CONTROL REGISTER (CONTINUED)

bit 3-0 **NVMOP<3:0>:** NVM Operation Select bits[1,3,4]
1111 = Reserved
•
•
•
0101 = Reserved
0100 = Inactive Partition memory erase operation
0011 = Memory page erase operation
0010 = Memory row program operation
0001 = Memory double-word program operation[5]
0000 = Reserved

Note 1: These bits can only be reset on a POR.

2: If this bit is set, power consumption will be further reduced (I_{IDLE}) and upon exiting Idle mode, there is a delay (T_{VREG}) before Flash memory becomes operational.

3: All other combinations of NVMOP<3:0> are unimplemented.

4: Execution of the PWRSAV instruction is ignored while any of the NVM operations are in progress.

5: Two adjacent words on a 4-word boundary are programmed during execution of this operation.

6: Only applicable when operating in Dual Partition mode.

REGISTER 5-2: NVMADR: NONVOLATILE MEMORY LOWER ADDRESS REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADR<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADR<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **NVMADR<15:0>:** Nonvolatile Memory Lower Write Address bits

Selects the lower 16 bits of the location to program or erase in Program Flash Memory. This register may be read or written to by the user application.

REGISTER 5-3: NVMADRU: NONVOLATILE MEMORY UPPER ADDRESS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADRU<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **NVMADRU<23:16>:** Nonvolatile Memory Upper Write Address bits

Selects the upper 8 bits of the location to program or erase in Program Flash Memory. This register may be read or written to by the user application.

REGISTER 5-4: NVMKEY: NONVOLATILE MEMORY KEY REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as '0'

bit 7-0 **NVMKEY<7:0>:** NVM Key Register bits (write-only)

REGISTER 5-5: NVMSRCADR: NVM SOURCE DATA ADDRESS REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADR<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADR<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **NVMSRCADR<15:0>:** NVM Source Data Address bits

The RAM address of the data to be programmed into Flash when the NVMOP<3:0> bits are set to row programming.

6.0 RESETS

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Reset"** (DS70602) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com)
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Reset module combines all Reset sources and controls the device Master Reset Signal, $\overline{\text{SYSRST}}$. The following is a list of device Reset sources:

- POR: Power-on Reset
- BOR: Brown-out Reset
- $\overline{\text{MCLR}}$: Master Clear Pin Reset
- SWR: RESET Instruction
- WDTO: Watchdog Timer Time-out Reset
- CM: Configuration Mismatch Reset
- TRAPR: Trap Conflict Reset
- IOPUWR: Illegal Condition Device Reset
  - Illegal Opcode Reset
  - Uninitialized W Register Reset
  - Security Reset

A simplified block diagram of the Reset module is shown in Figure 6-1.

Any active source of Reset will make the $\overline{\text{SYSRST}}$ signal active. On system Reset, some of the registers associated with the CPU and peripherals are forced to a known Reset state, and some are unaffected.

> **Note:** Refer to the specific peripheral section or **Section 4.0 "Memory Organization"** of this data sheet for register Reset states.

All types of device Reset set a corresponding status bit in the RCON register to indicate the type of Reset (see Register 6-1).

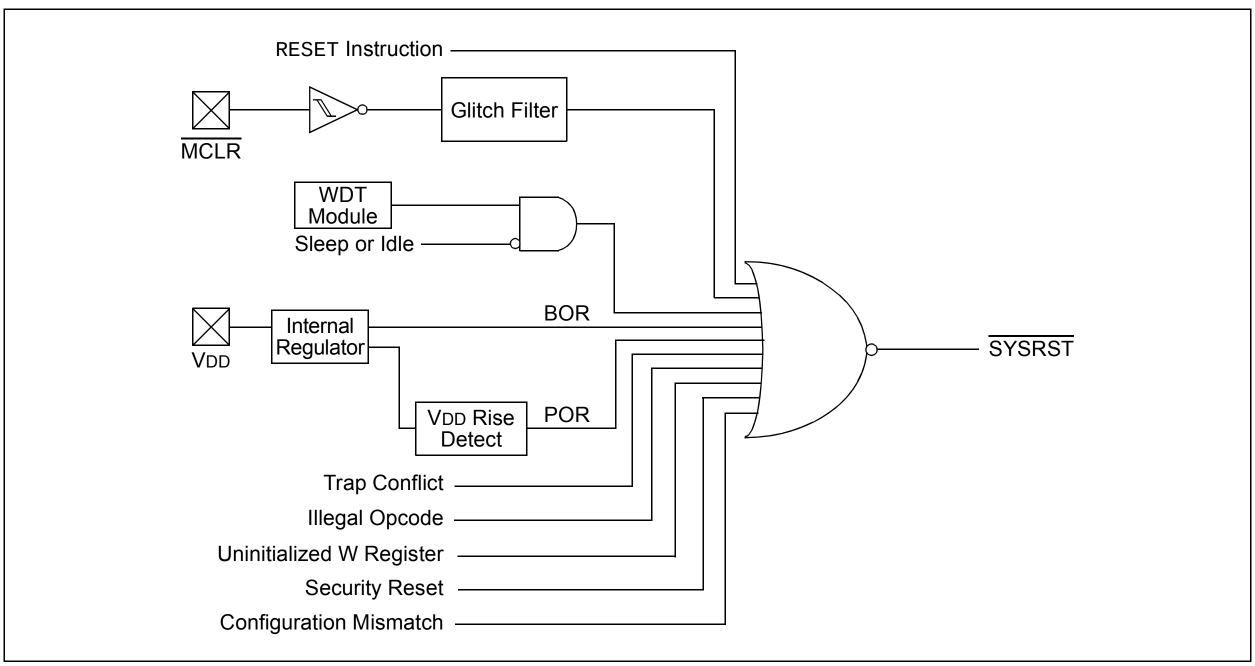
A POR clears all the bits, except for the BOR and POR bits (RCON<1:0>) that are set. The user application can set or clear any bit, at any time, during code execution. The RCON bits only serve as status bits. Setting a particular Reset status bit in software does not cause a device Reset to occur.

The RCON register also has other bits associated with the Watchdog Timer and device power-saving states. The function of these bits is discussed in other sections of this manual.

> **Note:** The status bits in the RCON register should be cleared after they are read so that the next RCON register value after a device Reset is meaningful.

For all Resets, the default clock source is determined by the FNOSC<2:0> bits in the FOSCSEL Configuration register. The value of the FNOSCx bits is loaded into the NOSC<2:0> (OSCCON<10:8>) bits on Reset, which in turn, initializes the system clock.

FIGURE 6-1: RESET SYSTEM BLOCK DIAGRAM



6.1 Reset Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

6.1.1 KEY RESOURCES

- **"Reset"** (DS70602) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

REGISTER 6-1: RCON: RESET CONTROL REGISTER[1]

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0
TRAPR	IOPUWR	—	—	VREGSF	—	CM	VREGS
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR	SWR	SWDTEN[2]	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **TRAPR:** Trap Reset Flag bit
1 = A Trap Conflict Reset has occurred
0 = A Trap Conflict Reset has not occurred

bit 14 **IOPUWR:** Illegal Opcode or Uninitialized W Register Access Reset Flag bit
1 = An illegal opcode detection, an illegal address mode or Uninitialized W register used as an Address Pointer caused a Reset
0 = An illegal opcode or Uninitialized W register Reset has not occurred

bit 13-12 **Unimplemented:** Read as ‘0’

bit 11 **VREGSF:** Flash Voltage Regulator Standby During Sleep bit
1 = Flash voltage regulator is active during Sleep
0 = Flash voltage regulator goes into Standby mode during Sleep

bit 10 **Unimplemented:** Read as ‘0’

bit 9 **CM:** Configuration Mismatch Flag bit
1 = A Configuration Mismatch Reset has occurred.
0 = A Configuration Mismatch Reset has not occurred

bit 8 **VREGS:** Voltage Regulator Standby During Sleep bit
1 = Voltage regulator is active during Sleep
0 = Voltage regulator goes into Standby mode during Sleep

bit 7 **EXTR:** External Reset ($\overline{\text{MCLR}}$) Pin bit
1 = A Master Clear (pin) Reset has occurred
0 = A Master Clear (pin) Reset has not occurred

bit 6 **SWR:** Software RESET (Instruction) Flag bit
1 = A RESET instruction has been executed
0 = A RESET instruction has not been executed

bit 5 **SWDTEN:** Software Enable/Disable of WDT bit[2]
1 = WDT is enabled
0 = WDT is disabled

bit 4 **WDTO:** Watchdog Timer Time-out Flag bit
1 = WDT time-out has occurred
0 = WDT time-out has not occurred

Note 1: All of the Reset status bits can be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

2: If the WDTEN<1:0> Configuration bits are ‘11’ (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

REGISTER 6-1: RCON: RESET CONTROL REGISTER[1] (CONTINUED)

bit 3	**SLEEP:** Wake-up from Sleep Flag bit 1 = Device has been in Sleep mode 0 = Device has not been in Sleep mode
bit 2	**IDLE:** Wake-up from Idle Flag bit 1 = Device has been in Idle mode 0 = Device has not been in Idle mode
bit 1	**BOR:** Brown-out Reset Flag bit 1 = A Brown-out Reset has occurred 0 = A Brown-out Reset has not occurred
bit 0	**POR:** Power-on Reset Flag bit 1 = A Power-on Reset has occurred 0 = A Power-on Reset has not occurred

Note 1: All of the Reset status bits can be set or cleared in software. Setting one of these bits in software does not cause a device Reset.

2: If the WDTEN<1:0> Configuration bits are '11' (unprogrammed), the WDT is always enabled, regardless of the SWDTEN bit setting.

7.0 INTERRUPT CONTROLLER

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Interrupts"** (DS70000600) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33EPXXXGS70X/80X family CPU.

The interrupt controller has the following features:

- Six Processor Exceptions and Software Traps
- Seven User-Selectable Priority Levels
- Interrupt Vector Table (IVT) with a Unique Vector for each Interrupt or Exception Source
- Fixed Priority within a Specified User Priority Level
- Fixed Interrupt Entry and Return Latencies
- Alternate Interrupt Vector Table (AIVT) for Debug Support

7.1 Interrupt Vector Table

The dsPIC33EPXXXGS70X/80X family Interrupt Vector Table (IVT), shown in Figure 7-1, resides in program memory, starting at location, 000004h. The IVT contains six non-maskable trap vectors and up to 246 sources of interrupts. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority. This priority is linked to their position in the vector table. Lower addresses generally have a higher natural priority. For example, the interrupt associated with Vector 0 takes priority over interrupts at any other vector address.

7.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT), shown in Figure 7-2, is available only when the Boot Segment is defined and the AIVT has been enabled. To enable the Alternate Interrupt Vector Table, the Configuration bit, AIVTDIS in the FSEC register, must be programmed and the AIVTEN bit must be set (INTCON2<8> = `1`). When the AIVT is enabled, all interrupt and exception processes use the alternate vectors instead of the default vectors. The AIVT begins at the start of the last page of the Boot Segment, defined by BSLIM<12:0>. The second half of the page is no longer usable space. The Boot Segment must be at least 2 pages to enable the AIVT.

> **Note:** Although the Boot Segment must be enabled in order to enable the AIVT, application code does not need to be present inside of the Boot Segment. The AIVT (and IVT) will inherit the Boot Segment code protection.

The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time.

7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33EPXXXGS70X/80X family devices clear their registers in response to a Reset, which forces the PC to zero. The device then begins program execution at location, 0x000000. A `GOTO` instruction at the Reset address can redirect program execution to the appropriate start-up routine.

> **Note:** Any unimplemented or unused vector locations in the IVT should be programmed with the address of a default interrupt handler routine that contains a `RESET` instruction.

FIGURE 7-1: dsPIC33EPXXXGS70X/80X FAMILY INTERRUPT VECTOR TABLE

Reset – GOTO Instruction	0x000000
Reset – GOTO Address	0x000002
Oscillator Fail Trap Vector	0x000004
Address Error Trap Vector	0x000006
Generic Hard Trap Vector	0x000008
Stack Error Trap Vector	0x00000A
Math Error Trap Vector	0x00000C
Reserved	0x00000E
Generic Soft Trap Vector	0x000010
Reserved	0x000012
Interrupt Vector 0	0x000014
Interrupt Vector 1	0x000016
:	:
:	:
:	:
Interrupt Vector 52	0x00007C
Interrupt Vector 53	0x00007E
Interrupt Vector 54	0x000080
:	:
:	:
:	:
Interrupt Vector 116	0x0000FC
Interrupt Vector 117	0x0000FE
Interrupt Vector 118	0x000100
Interrupt Vector 119	0x000102
Interrupt Vector 120	0x000104
:	:
:	:
:	:
Interrupt Vector 244	0x0001FC
Interrupt Vector 245	0x0001FE
START OF CODE	0x000200

Decreasing Natural Order Priority

IVT

See Table 7-1 for Interrupt Vector Details

Note: In Dual Partition Flash modes, each partition has a dedicated Interrupt Vector Table.

FIGURE 7-2: dsPIC33EPXXXGS70X/80X ALTERNATE INTERRUPT VECTOR TABLE[2]

Decreasing Natural Order Priority

AIVT

Reserved	BSLIM<12:0>[1] + 0x000000
Reserved	BSLIM<12:0>[1] + 0x000002
Oscillator Fail Trap Vector	BSLIM<12:0>[1] + 0x000004
Address Error Trap Vector	BSLIM<12:0>[1] + 0x000006
Generic Hard Trap Vector	BSLIM<12:0>[1] + 0x000008
Stack Error Trap Vector	BSLIM<12:0>[1] + 0x00000A
Math Error Trap Vector	BSLIM<12:0>[1] + 0x00000C
Reserved	BSLIM<12:0>[1] + 0x00000E
Generic Soft Trap Vector	BSLIM<12:0>[1] + 0x000010
Reserved	BSLIM<12:0>[1] + 0x000012
Interrupt Vector 0	BSLIM<12:0>[1] + 0x000014
Interrupt Vector 1	BSLIM<12:0>[1] + 0x000016
:	:
:	:
:	:
Interrupt Vector 52	BSLIM<12:0>[1] + 0x00007C
Interrupt Vector 53	BSLIM<12:0>[1] + 0x00007E
Interrupt Vector 54	BSLIM<12:0>[1] + 0x000080
:	:
:	:
:	:
Interrupt Vector 116	BSLIM<12:0>[1] + 0x0000FC
Interrupt Vector 117	BSLIM<12:0>[1] + 0x0000FE
Interrupt Vector 118	BSLIM<12:0>[1] + 0x000100
Interrupt Vector 119	BSLIM<12:0>[1] + 0x000102
Interrupt Vector 120	BSLIM<12:0>[1] + 0x000104
:	:
:	:
:	:
Interrupt Vector 244	BSLIM<12:0>[1] + 0x0001FC
Interrupt Vector 245	BSLIM<12:0>[1] + 0x0001FE

See Table 7-1 for Interrupt Vector Details

Note 1: The address depends on the size of the Boot Segment defined by BSLIM<12:0>. [($\overline{\text{BSLIM<12:0>}}$ – 1) x 0x400] + Offset.

2: In Dual Partition Flash modes, each partition has a dedicated Alternate Interrupt Vector Table (if enabled).

TABLE 7-1: INTERRUPT VECTOR DETAILS

Interrupt Source	Vector #	IRQ #	IVT Address	Interrupt Bit Location		
				Flag	Enable	Priority
Highest Natural Order Priority						
INT0 – External Interrupt 0	8	0	0x000014	IFS0<0> INT0IF	IEC0<0> INT0IE	IPC0<2:0> INT0IP<2:0>
IC1 – Input Capture 1	9	1	0x000016	IFS0<1> IC1IF	IEC0<1> IC1IE	IPC0<6:4> IC1IP<2:0>
OC1 – Output Compare 1	10	2	0x000018	IFS0<2> OC1IF	IEC0<2> OC1IE	IPC0<10:8> OC1IP<2:0>
T1 – Timer1	11	3	0x00001A	IFS0<3> T1IF	IEC0<3> T1IE	IPC0<14:12> T1IP<2:0>
DMA0 – DMA Channel 0	12	4	0x00001C	IFS0<4> DMA0IF	IEC0<4> DMA0IE	IPC1<2:0> DMA0IP<2:0>
IC2 – Input Capture 2	13	5	0x00001E	IFS0<5> IC2IF	IEC0<5> IC2IE	IPC1<6:4> IC2IP<2:0>
OC2 – Output Compare 2	14	6	0x000020	IFS0<6> OC2IF	IEC0<6> OC2IE	IPC1<10:8> OC2IP<2:0>
T2 – Timer2	15	7	0x000022	IFS0<7> T2IF	IEC0<7> T2IE	IPC1<14:12> T2IP<2:0>
T3 – Timer3	16	8	0x000024	IFS0<8> T3IF	IEC0<8> T3IE	IPC2<2:0> T3IP<2:0>
SPI1TX – SPI1 Transfer Done	17	9	0x000026	IFS0<9> SPI1TXIF	IEC0<9> SPI1TXIE	IPC2<6:4> SPI1TXIP<2:0>
SPI1RX – SPI1 Receive Done	18	10	0x000028	IFS0<10> SPI1RXIF	IEC0<10> SPI1RXIE	IPC2<10:8> SPI1RXIP<2:0>
U1RX – UART1 Receiver	19	11	0x00002A	IFS0<11> U1RXIF	IEC0<11> U1RXIE	IPC2<14:12> U1RXIP<2:0>
U1TX – UART1 Transmitter	20	12	0x00002C	IFS0<12> U1TXIF	IEC0<12> U1TXIE	IPC3<2:0> U1TXIP<2:0>
ADC – ADC Global Convert Done	21	13	0x00002E	IFS0<13> ADCIF	IEC0<13> ADCIE	IPC3<6:4> ADCIP<2:0>
DMA1 – DMA Channel 1	22	14	0x000030	IFS0<14> DMA1IF	IEC0<14> DMA1IE	IPC3<10:8> DMA1IP<2:0>
NVM – NVM Write Complete	23	15	0x000032	IFS0<15> NVMIF	IEC0<15> NVMIE	IPC3<14:12> NVMIP<2:0>
SI2C1 – I2C1 Slave Event	24	16	0x000034	IFS1<0> SI2C1IF	IEC1<0> SI2C1IE	IPC4<2:0> SI2C1IP<2:0>
MI2C1 – I2C1 Master Event	25	17	0x000036	IFS1<1> MI2C1IF	IEC1<1> MI2C1IE	IPC4<6:4> MI2C1IP<2:0>
AC1 – Analog Comparator 1 Interrupt	26	18	0x000038	IFS1<2> AC1IF	IEC1<2> AC1IE	IPC4<10:8> AC1IP<2:0>
CN – Input Change Interrupt	27	19	0x00003A	IFS1<3> CNIF	IEC1<3> CNIE	IPC4<14:12> CNIP<2:0>
INT1 – External Interrupt 1	28	20	0x00003C	IFS1<4> INT1IF	IEC1<4> INT1IE	IPC5<2:0> INT1IP<2:0>
Reserved	29-31	21-23	0x00003E-0x000043	—	—	—
DMA2 – DMA Channel 2	32	24	0x00044	IFS1<8> DMA2IF	IEC1<8> DMA2IE	IPC6<2:0> DMA2IP<2:0>
OC3 – Output Compare 3	33	25	0x000046	IFS1<9> OC3IF	IEC1<9> OC3IE	IPC6<6:4> OC3IP<2:0>
OC4 – Output Compare 4	34	26	0x000048	IFS1<10> OC4IF	IEC1<10> OC4IE	IPC6<10:8> OC4IP<2:0>

TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)

Interrupt Source	Vector #	IRQ #	IVT Address	Interrupt Bit Location		
				Flag	Enable	Priority
T4 – Timer4	35	27	0x00004A	IFS1<11> T4IF	IEC1<11> T4IE	IPC6<14:12> T4IP<2:0>
T5 – Timer5	36	28	0x00004C	IFS1<12> T5IF	IEC1<12> T5IE	IPC7<2:0> T5IP<2:0>
INT2 – External Interrupt 2	37	29	0x00004E	IFS1<13> INT2IF	IEC1<13> INT2IE	IPC7<6:4> INT2IP<2:0>
U2RX – UART2 Receiver	38	30	0x000050	IFS1<14> U2RXIF	IEC1<14> U2RXIE	IPC7<10:8> U2RXIP<2:0>
U2TX – UART2 Transmitter	39	31	0x000052	IFS1<15> U2TXIF	IEC1<15> U2TXIE	IPC7<14:12> U2TXIP<2:0>
SPI2TX – SPI2 Transfer Done	40	32	0x000054	IFS2<0> SPI2TXIF	IEC2<0> SPI2TXIE	IPC8<2:0> SPI2TXIP<2:0>
SPI2RX – SPI2 Receive Done	41	33	0x000056	IFS2<1> SPI2RXIF	IEC2<1> SPI2RXIE	IPC8<6:4> SPI2RXIP<2:0>
C1RX – CAN1 RX Data Ready	42	34	0x000058	IFS2<2> C1RXIF	IEC2<2> C1RXIE	IPC8<10:8> C1RXIP<2:0>
C1 – CAN1 Combined Error	43	35	0x000059	IFS2<3> C1IF	IEC2<3> C1IE	IPC8<14:12> C1IP<2:0>
DMA3 – DMA Channel 3	44	36	0x00005A	IFS2<4> DMA3IF	IEC2<4> DMA3IE	IPC9<2:0> DMA3IP<2:0>
IC3 – Input Capture 3	45	37	0x00005E	IFS2<5> IC3IF	IEC2<5> IC3IE	IPC9<6:4> IC3IP<2:0>
IC4 – Input Capture 4	46	38	0x000060	IFS2<6> IC4IF	IEC2<6> IC4IE	IPC9<10:8> IC4IP<2:0>
Reserved	47-56	39-48	0x000062-0x000074	—	—	—
SI2C2 – I2C2 Slave Event	57	49	0x000076	IFS3<1> SI2C2IF	IEC3<1> SI2C2IE	IPC12<6:4> SI2C2IP<2:0>
MI2C2 – I2C2 Master Event	58	50	0x000078	IFS3<2> MI2C2IF	IEC3<2> MI2C2IE	IPC12<10:8> MI2C2IP<2:0>
Reserved	59-61	51-53	0x00007A-0x00007E	—	—	—
INT4 – External Interrupt 4	62	54	0x000080	IFS3<6> INT4IF	IEC3<6> INT4IE	IPC13<10:8> INT4IP<2:0>
C2RX – CAN2 RX Data Ready	63	55	0x000082	IFS3<7> C2RXIF	IEC3<7> C2RXIE	IPC13<14:12> C2RXIP<2:0>
C2 – CAN 2 Combined Error	64	56	0x000083	IFS3<8> C2IF	IEC3<8> C2IE	IPC14<2:0> C2IP<2:0>
PSEM – PWM Special Event Match	65	57	0x000086	IFS3<9> PSEMIF	IEC3<9> PSEMIE	IPC14<6:4> PSEMIP<2:0>
Reserved	66-72	58-64	0x000088-0x000094	—	—	—
U1E – UART1 Error Interrupt	73	65	0x000096	IFS4<1> U1EIF	IEC4<1> U1EIE	IPC16<6:4> U1EIP<2:0>
U2E – UART2 Error Interrupt	74	66	0x000098	IFS4<2> U2EIF	IEC4<2> U2EIE	IPC16<10:8> U2EIP<2:0>
Reserved	75-77	67-69	0x00009A-0x0000A2	—	—	—
C1TX – CAN1 TX Data Request	78	70	0x0000A0	IFS4<6> C1TXIF	IEC4<6> C1TXIE	IPC17<10:8> C1TXIP<2:0>
C2TX – CAN2 TX Data Request	79	71	0x0000A	IFS4<7> C2TXIF	IEC4<7> C2TXIE	IPC17<14:12> C2TXIP<2:0>
Reserved	80	72	0x0000A4	—	—	—

TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)

Interrupt Source	Vector #	IRQ #	IVT Address	Interrupt Bit Location		
				Flag	Enable	Priority
PSES – PWM Secondary Special Event Match	81	73	0x0000A6	IFS4<9> PSESIF	IEC4<9> PSESIE	IPC18<6:4> PSESIP<2:0>
Reserved	82-97	74-89	0x0000A8-0x0000C6	—	—	—
SPI3TX – SPI3 Transfer Done	98	90	0x0000C8	IFS5<10> SPI3TXIF	IEC5<10> SPI3TXIE	IPC22<10:8> SPI3TXIP<2:0>
SPI3RX – SPI3 Receive Done	99	91	0x0000CA	IFS5<10> SPI3RXIF	IEC5<11> SPI3RXIE	IPC22<14:12> SPI3RXIP<2:0>
Reserved	100-101	92-93	0x0000CC-0x0000CE	—	—	—
PWM1 – PWM1 Interrupt	102	94	0x0000D0	IFS5<14> PWM1IF	IEC5<14> PWM1IE	IPC23<10:8> PWM1IP<2:0>
PWM2 – PWM2 Interrupt	103	95	0x0000D2	IFS5<15> PWM2IF	IEC5<15> PWM2IE	IPC23<14:12> PWM2IP<2:0>
PWM3 – PWM3 Interrupt	104	96	0x0000D4	IFS6<0> PWM3IF	IEC6<0> PWM3IE	IPC24<2:0> PWM3IP<2:0>
PWM4 – PWM4 Interrupt	105	97	0x0000D6	IFS6<1> PWM4IF	IEC6<1> PWM4IE	IPC24<6:4> PWM4IP<2:0>
PWM5 – PWM5 Interrupt	106	98	0x0000D8	IFS6<2> PWM5IF	IEC6<2> PWM5IE	IPC24<10:8> PWM5IP<2:0>
PWM6 – PWM6 Interrupt	107	99	0x0000DA	IFS6<3> PWM6IF	IEC6<3> PWM6IE	IPC24<14:12> PWM6IP<2:0>
PWM7 – PWM7 Interrupt	108	100	0x0000DC	IFS6<4> PWM7IF	IEC6<4> PWM7IE	IPC25<2:0> PWM7IP<2:0>
PWM8 – PWM8 Interrupt	109	101	0x0000DE	IFS6<5> PWM8IF	IEC6<5> PWM8IE	IPC25<6:4> PWM8IP<2:0>
Reserved	110	102	0x0000E0	—	—	—
AC2 – Analog Comparator 2 Interrupt	111	103	0x0000E2	IFS6<7> AC2IF	IEC6<7> AC2IE	IPC25<14:12> AC2IP<2:0>
AC3 – Analog Comparator 3 Interrupt	112	104	0x0000E4	IFS6<8> AC3IF	IEC6<8> AC3IE	IPC26<2:0> AC3IP<2:0>
AC4 – Analog Comparator 4 Interrupt	113	105	0x0000E6	IFS6<9> AC4IF	IEC6<9> AC4IE	IPC26<6:4> AC4IP<2:0>
Reserved	114-117	106-109	0x0000E8-0x0000EE	—	—	—
AN0 Conversion Done	118	110	0x0000F0	IFS6<14> AN0IF	IEC6<14> AN0IE	IPC27<10:8> AN0IP<2:0>
AN1 Conversion Done	119	111	0x0000F2	IFS6<15> AN1IF	IEC6<15> AN1IE	IPC27<14:12> AN1IP<2:0>
AN2 Conversion Done	120	112	0x0000F4	IFS7<0> AN2IF	IEC7<0> AN2IE	IPC28<2:0> AN2IP<2:0>
AN3 Conversion Done	121	113	0x0000F6	IFS7<1> AN3IF	IEC7<1> AN3IE	IPC28<6:4> AN3IP<2:0>
AN4 Conversion Done	122	114	0x0000F8	IFS7<2> AN4IF	IEC7<2> AN4IE	IPC28<10:8> AN4IP<2:0>
AN5 Conversion Done	123	115	0x0000FA	IFS7<3> AN5IF	IEC7<3> AN5IE	IPC28<14:12> AN5IP<2:0>
AN6 Conversion Done	124	116	0x0000FC	IFS7<4> AN6IF	IEC7<4> AN6IE	IPC29<2:0> AN6IP<2:0>
AN7 Conversion Done	125	117	0x0000FE	IFS7<5> AN7IF	IEC7<5> AN7IE	IPC29<6:4> AN7IP<2:0>
Reserved	126-131	118-123	0x000100-0x00010A	—	—	—

TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)

Interrupt Source	Vector #	IRQ #	IVT Address	Interrupt Bit Location		
				Flag	Enable	Priority
SPI1 Error Interrupt	132	124	0x00010C	IFS7<12> SPI1IF	IEC7<12> SPI1IE	IPC31<2:0> SPI1IP<2:0>
SPI2 Error Interrupt	133	125	0x00010E	IFS7<13> SPI2IF	IEC7<13> SPI2IE	IPC31<6:4> SPI2IP<2:0>
SPI3 Error Interrupt	134	126	0x000110	IFS7<13> SPI3IF	IEC7<13> SPI3IE	IPC31<10:8> SPI3IP<2:0>
Reserved	135-145	127-137	0x000112-0x000126	—	—	—
CLC1 Interrupt	146	138	0x000128	IFS8<10> CLC1IF	IEC8<10> CLC1IE	IPC34<10:8> CLC1IP<2:0>
CLC2 Interrupt	147	139	0x00012A	IFS8<11> CLC2IF	IEC8<11> CLC2IE	IPC34<14:12> CLC2IP<2:0>
CLC3 Interrupt	148	140	0x00012C	IFS8<12> CLC3IF	IEC8<12> CLC3IE	IPC35<2:0> CLC3IP<2:0>
CLC4 Interrupt	149	141	0x00012E	IFS8<13> CLC4IF	IEC8<13> CLC4IE	IPC35<6:4> CLC4IP<2:0>
ICD – ICD Application	150	142	0x000130	IFS8<14> ICDIF	IEC8<14> ICDIE	IPC35<10:8> ICDIP<2:0>
JTAG – JTAG Programming	151	143	0x000132	IFS8<15> JTAGIF	IEC8<15> JTAGIE	IPC35<14:12> JTAGIP<2:0>
Reserved	152	144	0x000134	—	—	—
PTGSTEP – PTG Step	153	145	0x000136	IFS9<1> PTGSTEPIF	IEC9<1> PTGSTEPIE	IPC36<6:4> PTGSTEP<2:0>
PTGWDT – PTG WDT Time-out	154	146	0x000138	IFS9<2> PTGWDTIF	IEC9<2> PTGWDTIE	IPC36<10:8> PTGWDT<2:0>
PTG0 – PTG Interrupt Trigger 0	155	147	0x00013A	IFS9<3> PTG0IF	IEC9<3> PTG0IE	IPC36<14:12> PTG0IP<2:0>
PTG1 – PTG Interrupt Trigger 1	156	148	0x00013C	IFS9<4> PTG1IF	IEC9<4> PTG1IE	IPC37<2:0> PTG1IP<2:0>
PTG2 – PTG Interrupt Trigger 2	157	149	0x00013E	IFS9<5> PTG2IF	IEC9<5> PTG2IE	IPC37<6:4> PTG2IP<2:0>
PTG3 – PTG Interrupt Trigger 3	158	150	0x000140	IFS9<6> PTG3IF	IEC9<6> PTG3IE	IPC37<10:8> PTG3IP<2:0>
AN8 Conversion Done	159	151	0x000142	IFS9<7> AN8IF	IEC9<7> AN8IE	IPC37<14:12> AN8IP<2:0>
AN9 Conversion Done	160	152	0x000144	IFS9<8> AN9IF	IEC9<8> AN9IE	IPC38<2:0> AN9IP<2:0>
AN10 Conversion Done	161	153	0x000146	IFS9<9> AN10IF	IEC9<9> AN10IE	IPC38<6:4> AN10IP<2:0>
AN11 Conversion Done	162	154	0x000148	IFS9<10> AN11IF	IEC9<10> AN11IE	IPC38<10:8> AN11IP<2:0>
AN12 Conversion Done	163	155	0x00014A	IFS9<11> AN12IF	IEC9<11> AN12IE	IPC38<14:12> AN12IP<2:0>
AN13 Conversion Done	164	156	0x00014C	IFS9<12> AN13IF	IEC9<12> AN13IE	IPC39<2:0> AN13IP<2:0>
AN14 Conversion Done	165	157	0x00014E	IFS9<13> AN14IF	IEC9<13> AN14IE	IPC39<6:4> AN14IP<2:0>
AN15 Conversion Done	166	158	0x000150	IFS9<14> AN15IF	IEC9<14> AN15IE	IPC39<10:8> AN15IP<2:0>
AN16 Conversion Done	167	159	0x000152	IFS9<15> AN16IF	IEC9<15> AN16IE	IPC39<14:12> AN16IP<2:0>

TABLE 7-1: INTERRUPT VECTOR DETAILS (CONTINUED)

Interrupt Source	Vector #	IRQ #	IVT Address	Interrupt Bit Location		
				Flag	Enable	Priority
AN17 Conversion Done	168	160	0x000154	IFS10<0> AN17IF	IEC10<0> AN17IE	IPC40<2:0> AN17IP<2:0>
AN18 Conversion Done	169	161	0x000156	IFS10<1> AN18IF	IEC10<1> AN18IE	IPC40<6:4> AN18IP<2:0>
AN19 Conversion Done	170	162	0x000158	IFS10<2> AN19IF	IEC10<2> AN19IE	IPC40<10:8> AN19IP<2:0>
AN20 Conversion Done	171	163	0x00015A	IFS10<3> AN20IF	IEC10<3> AN20IE	IPC40<14:12> AN20IP<2:0>
AN21 Conversion Done	172	164	0x00015C	IFS10<4> AN21IF	IEC10<4> AN21IE	IPC41<2:0> AN21IP<2:0>
Reserved	173-180	165-172	0x00015C-0x00016C	—	—	—
I2C1 – I2C1 Bus Collision	181	173	0x00016E	IFS10<13> I2C1IF	IEC10<13> I2C1IE	IPC43<6:4> I2C1IP<2:0>
I2C2 – I2C2 Bus Collision	182	174	0x000170	IFS10<14> I2C2IF	IEC10<14> I2C2IE	IPC43<10:8> I2C2IP<2:0>
Reserved	183-184	175-176	0x000172-0x000174	—	—	—
ADCMP0 – ADC Digital Comparator 0	185	177	0x000176	IFS11<1> ADCMP0IF	IEC11<1> ADCMP0IE	IPC44<6:4> ADCMP0IP<2:0>
ADCMP1 – ADC Digital Comparator 1	186	178	0x000178	IFS11<2> ADCMP1IF	IEC11<2> ADCMP1IE	IPC44<10:8> ADCMP1IP<2:0>
ADFLTR0 – ADC Filter 0	187	179	0x00017A	IFS11<3> ADFLTR0IF	IEC11<3> ADFLTR0IE	IPC44<14:12> ADFLTR0IP<2:0>
ADFLTR1 – ADC Filter 1	188	180	0x00017C	IFS11<4> ADFLTR1IF	IEC11<4> ADFLTR1IE	IPC45<2:0> ADFLTR1IP<2:0>
Reserved	189-253	181-245	0x00017E-0x000192	—	—	—

7.3 Interrupt Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

7.3.1 KEY RESOURCES

- **“Interrupts”** (DS70000600) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

7.4 Interrupt Control and Status Registers

dsPIC33EPXXXGS70X/80X family devices implement the following registers for the interrupt controller:

- INTCON1
- INTCON2
- INTCON3
- INTCON4
- INTTREG

7.4.1 INTCON1 THROUGH INTCON4

Global interrupt control functions are controlled from INTCON1, INTCON2, INTCON3 and INTCON4.

INTCON1 contains the Interrupt Nesting Disable bit (NSTDIS), as well as the control and status flags for the processor trap sources.

The INTCON2 register controls external interrupt request signal behavior, contains the Global Interrupt Enable bit (GIE) and the Alternate Interrupt Vector Table Enable bit (AIVTEN).

INTCON3 contains the status flags for the Auxiliary PLL and DO stack overflow status trap sources.

The INTCON4 register contains the Software Generated Hard Trap Status bit (SGHT).

7.4.2 IFSx

The IFSx registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals or external signal and is cleared via software.

7.4.3 IECx

The IECx registers maintain all of the interrupt enable bits. These control bits are used to individually enable interrupts from the peripherals or external signals.

7.4.4 IPCx

The IPCx registers are used to set the Interrupt Priority Level (IPL) for each source of interrupt. Each user interrupt source can be assigned to one of seven priority levels.

7.4.5 INTTREG

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt Priority Level, which are latched into the Vector Number of Pending Interrupt bits (VECNUM<7:0>) and New CPU Interrupt Priority Level bits (ILR<3:0>) fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the same sequence as they are listed in Table 7-1. For example, the INT0 (External Interrupt 0) is shown as having Vector Number 8 and a natural order priority of 0. Thus, the INT0IF bit is found in IFS0<0>, the INT0IE bit in IEC0<0> and the INT0IP<2:0> bits in the first position of IPC0 (IPC0<2:0>).

7.4.6 STATUS/CONTROL REGISTERS

Although these registers are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality. For more information on these registers, refer to **“dsPIC33E Enhanced CPU”** (DS70005158) in the *“dsPIC33/PIC24 Family Reference Manual”*.

- The CPU STATUS Register, SR, contains the IPL<2:0> bits (SR<7:5>). These bits indicate the current CPU Interrupt Priority Level. The user software can change the current CPU Interrupt Priority Level by writing to the IPLx bits.
- The CORCON register contains the IPL3 bit which, together with IPL<2:0>, also indicates the current CPU priority level. IPL3 is a read-only bit so that trap events cannot be masked by the user software.

All Interrupt registers are described in Register 7-3 through Register 7-7 in the following pages.

REGISTER 7-1: SR: CPU STATUS REGISTER[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/C-0	R/C-0	R-0	R/W-0
OA	OB	SA	SB	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0[3]	R/W-0[3]	R/W-0[3]	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2[2]	IPL1[2]	IPL0[2]	RA	N	OV	Z	C
bit 7							bit 0

Legend:	C = Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1'= Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits[2,3]

111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled
110 = CPU Interrupt Priority Level is 6 (14)
101 = CPU Interrupt Priority Level is 5 (13)
100 = CPU Interrupt Priority Level is 4 (12)
011 = CPU Interrupt Priority Level is 3 (11)
010 = CPU Interrupt Priority Level is 2 (10)
001 = CPU Interrupt Priority Level is 1 (9)
000 = CPU Interrupt Priority Level is 0 (8)

Note 1: For complete register details, see Register 3-1.

2: The IPL<2:0> bits are concatenated with the IPL<3> bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL, if IPL<3> = 1. User interrupts are disabled when IPL<3> = 1.

3: The IPL<2:0> Status bits are read-only when the NSTDIS bit (INTCON1<15>) = 1.

REGISTER 7-2: CORCON: CORE CONTROL REGISTER[1]

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-0
VAR	—	US1	US0	EDT	DL2	DL1	DL0
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3[2]	SFA	RND	IF
bit 7							bit 0

Legend:	C = Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’= Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **VAR:** Variable Exception Processing Latency Control bit
1 = Variable exception processing is enabled
0 = Fixed exception processing is enabled

bit 3 **IPL3:** CPU Interrupt Priority Level Status bit 3[2]
1 = CPU Interrupt Priority Level is greater than 7
0 = CPU Interrupt Priority Level is 7 or less

Note 1: For complete register details, see Register 3-2.
2: The IPL3 bit is concatenated with the IPL<2:0> bits (SR<7:5>) to form the CPU Interrupt Priority Level.

REGISTER 7-3: INTCON1: INTERRUPT CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE
bit 15							bit 8

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **NSTDIS:** Interrupt Nesting Disable bit
1 = Interrupt nesting is disabled
0 = Interrupt nesting is enabled

bit 14 **OVAERR:** Accumulator A Overflow Trap Flag bit
1 = Trap was caused by overflow of Accumulator A
0 = Trap was not caused by overflow of Accumulator A

bit 13 **OVBERR:** Accumulator B Overflow Trap Flag bit
1 = Trap was caused by overflow of Accumulator B
0 = Trap was not caused by overflow of Accumulator B

bit 12 **COVAERR:** Accumulator A Catastrophic Overflow Trap Flag bit
1 = Trap was caused by catastrophic overflow of Accumulator A
0 = Trap was not caused by catastrophic overflow of Accumulator A

bit 11 **COVBERR:** Accumulator B Catastrophic Overflow Trap Flag bit
1 = Trap was caused by catastrophic overflow of Accumulator B
0 = Trap was not caused by catastrophic overflow of Accumulator B

bit 10 **OVATE:** Accumulator A Overflow Trap Enable bit
1 = Trap overflow of Accumulator A
0 = Trap is disabled

bit 9 **OVBTE:** Accumulator B Overflow Trap Enable bit
1 = Trap overflow of Accumulator B
0 = Trap is disabled

bit 8 **COVTE:** Catastrophic Overflow Trap Enable bit
1 = Trap on catastrophic overflow of Accumulator A or B is enabled
0 = Trap is disabled

bit 7 **SFTACERR:** Shift Accumulator Error Status bit
1 = Math error trap was caused by an invalid accumulator shift
0 = Math error trap was not caused by an invalid accumulator shift

bit 6 **DIV0ERR:** Divide-by-Zero Error Status bit
1 = Math error trap was caused by a divide-by-zero
0 = Math error trap was not caused by a divide-by-zero

bit 5 **Unimplemented:** Read as ‘0’

bit 4 **MATHERR:** Math Error Status bit
1 = Math error trap has occurred
0 = Math error trap has not occurred

bit 3 **ADDRERR:** Address Error Trap Status bit
1 = Address error trap has occurred
0 = Address error trap has not occurred

REGISTER 7-3: INTCON1: INTERRUPT CONTROL REGISTER 1 (CONTINUED)

bit 2	**STKERR:** Stack Error Trap Status bit 1 = Stack error trap has occurred 0 = Stack error trap has not occurred
bit 1	**OSCFAIL:** Oscillator Failure Trap Status bit 1 = Oscillator failure trap has occurred 0 = Oscillator failure trap has not occurred
bit 0	**Unimplemented:** Read as ‘0’

REGISTER 7-4: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
GIE	DISI	SWTRAP	—	—	—	—	AIVTEN
bit 15							bit 8

U-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	INT4EP	—	INT2EP	INT1EP	INT0EP
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **GIE:** Global Interrupt Enable bit
1 = Interrupts and associated IE bits are enabled
0 = Interrupts are disabled, but traps are still enabled

bit 14 **DISI:** DISI Instruction Status bit
1 = DISI instruction is active
0 = DISI instruction is not active

bit 13 **SWTRAP:** Software Trap Status bit
1 = Software trap is enabled
0 = Software trap is disabled

bit 12-9 **Unimplemented:** Read as ‘0’

bit 8 **AIVTEN:** Alternate Interrupt Vector Table Enable
1 = Uses Alternate Interrupt Vector Table
0 = Uses standard Interrupt Vector Table

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4 **INT4EP:** External Interrupt 4 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 3 **Unimplemented:** Read as ‘0’

bit 2 **INT2EP:** External Interrupt 2 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 1 **INT1EP:** External Interrupt 1 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

bit 0 **INT0EP:** External Interrupt 0 Edge Detect Polarity Select bit
1 = Interrupt on negative edge
0 = Interrupt on positive edge

REGISTER 7-5: INTCON3: INTERRUPT CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	NAE
bit 15							bit 8

U-0	U-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0
—	—	—	DOOVR	—	—	—	APLL
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-9 **Unimplemented:** Read as ‘0’

bit 8 **NAE:** NVM Address Error Soft Trap Status bit
1 = NVM address error soft trap has occurred
0 = NVM address error soft trap has not occurred

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4 **DOOVR:** DO Stack Overflow Soft Trap Status bit
1 = DO stack overflow soft trap has occurred
0 = DO stack overflow soft trap has not occurred

bit 3-1 **Unimplemented:** Read as ‘0’

bit 0 **APLL:** Auxiliary PLL Loss of Lock Soft Trap Status bit
1 = APLL lock soft trap has occurred
0 = APLL lock soft trap has not occurred

REGISTER 7-6: INTCON4: INTERRUPT CONTROL REGISTER 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	SGHT
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-1 **Unimplemented:** Read as ‘0’

bit 0 **SGHT:** Software Generated Hard Trap Status bit
1 = Software generated hard trap has occurred
0 = Software generated hard trap has not occurred

REGISTER 7-7: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR3	ILR2	ILR1	ILR0
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
VECNUM7	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-8 **ILR<3:0>:** New CPU Interrupt Priority Level bits
1111 = CPU Interrupt Priority Level is 15
•
•
•
0001 = CPU Interrupt Priority Level is 1
0000 = CPU Interrupt Priority Level is 0

bit 7-0 **VECNUM<7:0>:** Vector Number of Pending Interrupt bits
11111111 = 255, Reserved; do not use
•
•
•
00001001 = 9, IC1 – Input Capture 1
00001000 = 8, INT0 – External Interrupt 0
00000111 = 7, Reserved; do not use
00000110 = 6, Generic soft error trap
00000101 = 5, Reserved; do not use
00000100 = 4, Math error trap
00000011 = 3, Stack error trap
00000010 = 2, Generic hard trap
00000001 = 1, Address error trap
00000000 = 0, Oscillator fail trap

8.0 DIRECT MEMORY ACCESS (DMA)

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Direct Memory Access (DMA)"** (DS70348) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The DMA Controller transfers data between Peripheral Data registers and Data Space SRAM

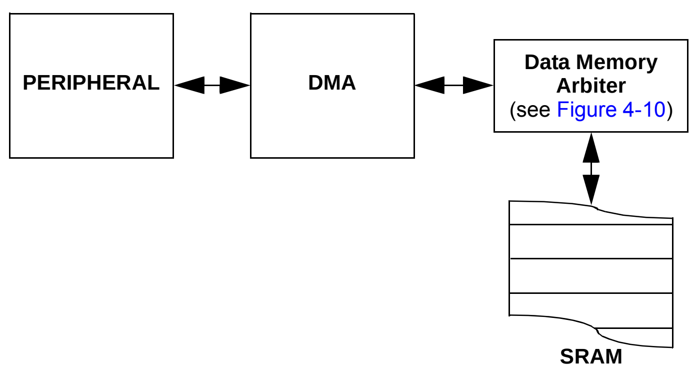
In addition, DMA can access the entire data memory space. The data memory bus arbiter is utilized when either the CPU or DMA attempts to access SRAM, resulting in potential DMA or CPU Stalls.

The DMA Controller supports 4 independent channels. Each channel can be configured for transfers to or from selected peripherals. The peripherals supported by the DMA Controller include:

- CAN
- UART
- Input Capture
- Output Compare
- Timers

Refer to Table 8-1 for a complete list of supported peripherals.

FIGURE 8-1: PERIPHERAL TO DMA CONTROLLER



In addition, DMA transfers can be triggered by timers as well as external interrupts. Each DMA channel is unidirectional. Two DMA channels must be allocated to read and write to a peripheral. If more than one channel receives a request to transfer data, a simple fixed priority scheme, based on channel number, dictates which channel completes the transfer and which channel, or channels, are left pending. Each DMA channel moves a block of data, after which, it generates an interrupt to the CPU to indicate that the block is available for processing.

The DMA Controller provides these functional capabilities:

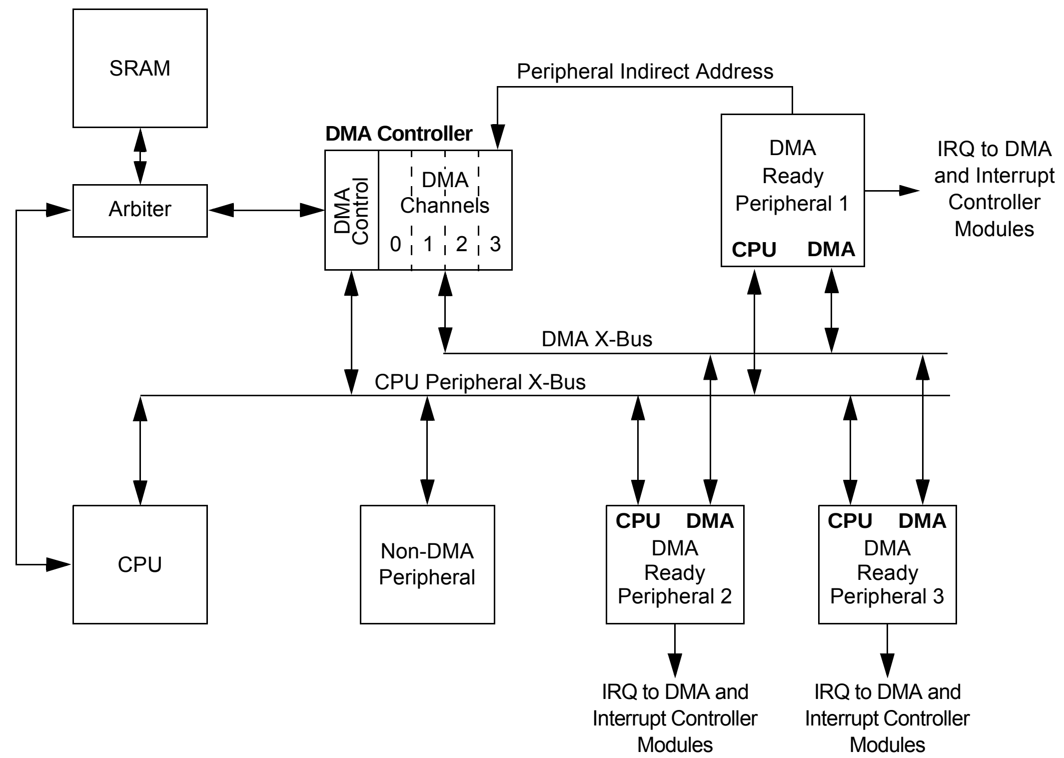
- Four DMA Channels
- Register Indirect with Post-Increment Addressing mode
- Register Indirect without Post-Increment Addressing mode
- Peripheral Indirect Addressing mode (peripheral generates destination address)
- CPU Interrupt after Half or Full Block Transfer Complete
- Byte or Word Transfers
- Fixed Priority Channel Arbitration
- Manual (software) or Automatic (peripheral DMA requests) Transfer Initiation
- One-Shot or Auto-Repeat Block Transfer modes
- Ping-Pong mode (automatic switch between two SRAM Start addresses after each block transfer complete)
- DMA Request for each Channel can be Selected from any Supported Interrupt Source
- Debug Support Features

The peripherals that can utilize DMA are listed in Table 8-1.

TABLE 8-1: DMA CHANNEL TO PERIPHERAL ASSOCIATIONS

Peripheral to DMA Association	DMAxREQ Register IRQSEL<7:0> Bits	DMAxPAD Register (Values to Read from Peripheral)	DMAxPAD Register (Values to Write to Peripheral)
INT0 – External Interrupt 0	00000000	—	—
IC1 – Input Capture 1	00000001	0x0144 (IC1BUF)	—
IC2 – Input Capture 2	00000101	0x014C (IC2BUF)	—
IC3 – Input Capture 3	00100101	0x0154 (IC3BUF)	—
IC4 – Input Capture 4	00100110	0x015C (IC4BUF)	—
OC1 – Output Compare 1	00000010	—	0x0906 (OC1R) 0x0904 (OC1RS)
OC2 – Output Compare 2	00000110	—	0x0910 (OC2R) 0x090E (OC2RS)
OC3 – Output Compare 3	00011001	—	0x091A (OC3R) 0x0918 (OC3RS)
OC4 – Output Compare 4	00011010	—	0x0924 (OC4R) 0x0922 (OC4RS)
TMR2 – Timer2	00000111	—	—
TMR3 – Timer3	00001000	—	—
TMR4 – Timer4	00011011	—	—
TMR5 – Timer5	00011100	—	—
UART1RX – UART1 Receiver	00001011	0x0226 (U1RXREG)	—
UART1TX – UART1 Transmitter	00001100	—	0x0224 (U1TXREG)
UART2RX – UART2 Receiver	00011110	0x0236 (U2RXREG)	—
UART2TX – UART2 Transmitter	00011111	—	0x0234 (U2TXREG)
CAN1 – RX Data Ready	00100010	0x0440 (C1RXD)	—
CAN1 – TX Data Request	01000110	—	0x0442 (C1TXD)
CAN2 – RX Data Ready	00110111	0X0540(C2RXD)	—
CAN2 – TX Data Request	01000111	—	0X0542(C2TXD)

FIGURE 8-2: DMA CONTROLLER BLOCK DIAGRAM



Note: CPU and DMA address buses are not shown for clarity.

8.1 DMA Controller Registers

Each DMA Controller Channel x (where x = 0 through 3) contains the following registers:

- 16-Bit DMA Channel x Control Register (DMAxCON)
- 16-Bit DMA Channel x IRQ Select Register (DMAxREQ)
- 32-Bit DMA Channel x Start Address Register A (DMAxSTAL/H)
- 32-Bit DMA Channel x Start Address Register B (DMAxSTBL/H)
- 16-Bit DMA Channel x Peripheral Address Register (DMAxPAD)
- 14-Bit DMA Channel x Transfer Count Register (DMAxCNT)

Additional status registers (DMAPWC, DMARQC, DMAPPS, DMALCA and DSADRL/H) are common to all DMA Controller channels. These status registers provide information on write and request collisions, as well as on last address and channel access information.

The interrupt flags (DMAxIF) are located in an IFSx register in the interrupt controller. The corresponding interrupt enable control bits (DMAxIE) are located in an IECx register in the interrupt controller and the corresponding interrupt priority control bits (DMAxIP) are located in an IPCx register in the interrupt controller.

REGISTER 8-1: DMAxCON: DMA CHANNEL x CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHEN	SIZE	DIR	HALF	NULLW	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	AMODE1	AMODE0	—	—	MODE1	MODE0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CHEN:** DMA Channel Enable bit
1 = Channel is enabled
0 = Channel is disabled

bit 14 **SIZE:** DMA Data Transfer Size bit
1 = Byte
0 = Word

bit 13 **DIR:** Transfer Direction bit (source/destination bus select)
1 = Reads from RAM address, writes to peripheral address
0 = Reads from peripheral address, writes to RAM address

bit 12 **HALF:** Block Transfer Interrupt Select bit
1 = Initiates interrupt when half of the data has been moved
0 = Initiates interrupt when all of the data has been moved

bit 11 **NULLW:** Null Data Peripheral Write Mode Select bit
1 = Null data write to peripheral in addition to RAM write (DIR bit must also be clear)
0 = Normal operation

bit 10-6 **Unimplemented:** Read as ‘0’

bit 5-4 **AMODE<1:0>:** DMA Channel Addressing Mode Select bits
11 = Reserved
10 = Peripheral Indirect mode
01 = Register Indirect without Post-Increment mode
00 = Register Indirect with Post-Increment mode

bit 3-2 **Unimplemented:** Read as ‘0’

bit 1-0 **MODE<1:0>:** DMA Channel Operating Mode Select bits
11 = One-Shot, Ping-Pong modes are enabled (one block transfer from/to each DMA buffer)
10 = Continuous, Ping-Pong modes are enabled
01 = One-Shot, Ping-Pong modes are disabled
00 = Continuous, Ping-Pong modes are disabled

REGISTER 8-2: DMAxREQ: DMA CHANNEL x IRQ SELECT REGISTER

R/S-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
FORCE[1]	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQSEL7	IRQSEL6	IRQSEL5	IRQSEL4	IRQSEL3	IRQSEL2	IRQSEL1	IRQSEL0
bit 7							bit 0

Legend:	S = Settable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **FORCE:** Force DMA Transfer bit[1]
1 = Forces a single DMA transfer (Manual mode)
0 = Automatic DMA transfer initiation by DMA request

bit 14-8 **Unimplemented:** Read as ‘0’

bit 7-0 **IRQSEL<7:0>:** DMA Peripheral IRQ Number Select bits
01000111 = CAN2 – TX data request
01000110 = CAN1 – TX data request
00110111 = CAN2 – RX data ready
00100110 = IC4 – Input Capture 4
00100101 = IC3 – Input Capture 3
00100010 = CAN1 – RX data ready
00011111 = UART2TX – UART2 transmitter
00011110 = UART2RX – UART2 receiver
00011100 = TMR5 – Timer5
00011011 = TMR4 – Timer4
00011010 = OC4 – Output Compare 4
00011001 = OC3 – Output Compare 3
00001100 = UART1TX – UART1 transmitter
00001011 = UART1RX – UART1 receiver
00001000 = TMR3 – Timer3
00000111 = TMR2 – Timer2
00000110 = OC2 – Output Compare 2
00000101 = IC2 – Input Capture 2
00000010 = OC1 – Output Compare 1
00000001 = IC1 – Input Capture 1
00000000 = INT0 – External Interrupt 0

Note 1: The FORCE bit cannot be cleared by user software. The FORCE bit is cleared by hardware when the forced DMA transfer is complete or the channel is disabled (CHEN = 0).

REGISTER 8-3: DMAxSTAH: DMA CHANNEL x START ADDRESS REGISTER A (HIGH)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **STA<23:16>:** DMA Primary Start Address bits (source or destination)

REGISTER 8-4: DMAxSTAL: DMA CHANNEL x START ADDRESS REGISTER A (LOW)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **STA<15:0>:** DMA Primary Start Address bits (source or destination)

REGISTER 8-5: DMAxSTBH: DMA CHANNEL x START ADDRESS REGISTER B (HIGH)

U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **STB<23:16>:** DMA Secondary Start Address bits (source or destination)

REGISTER 8-6: DMAxSTBL: DMA CHANNEL x START ADDRESS REGISTER B (LOW)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **STB<15:0>:** DMA Secondary Start Address bits (source or destination)

REGISTER 8-7: DMAxPAD: DMA CHANNEL x PERIPHERAL ADDRESS REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PAD<15:0>:** DMA Peripheral Address Register bits

Note 1: If the channel is enabled (i.e., active), writes to this register may result in unpredictable behavior of the DMA channel and should be avoided.

REGISTER 8-8: DMAxCNT: DMA CHANNEL x TRANSFER COUNT REGISTER[(1)]

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CNT<13:8>[(2)]					
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNT<7:0>[(2)]							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-0 **CNT<13:0>:** DMA Transfer Count Register bits[(2)]

Note 1: If the channel is enabled (i.e., active), writes to this register may result in unpredictable behavior of the DMA channel and should be avoided.

2: The number of DMA transfers = CNT<13:0> + 1.

REGISTER 8-9: DSADRH: DMA MOST RECENT RAM HIGH ADDRESS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **DSADR<23:16>:** Most Recent DMA Address Accessed by DMA bits

REGISTER 8-10: DSADRL: DMA MOST RECENT RAM LOW ADDRESS REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<15:8>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **DSADR<15:0>:** Most Recent DMA Address Accessed by DMA bits

REGISTER 8-11: DMAPWC: DMA PERIPHERAL WRITE COLLISION STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	PWCOL3	PWCOL2	PWCOL1	PWCOL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **PWCOL3:** Channel 3 Peripheral Write Collision Flag bit
1 = Write collision is detected
0 = No write collision is detected

bit 2 **PWCOL2:** Channel 2 Peripheral Write Collision Flag bit
1 = Write collision is detected
0 = No write collision is detected

bit 1 **PWCOL1:** Channel 1 Peripheral Write Collision Flag bit
1 = Write collision is detected
0 = No write collision is detected

bit 0 **PWCOL0:** Channel 0 Peripheral Write Collision Flag bit
1 = Write collision is detected
0 = No write collision is detected

REGISTER 8-12: DMARQC: DMA REQUEST COLLISION STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	RQCOL3	RQCOL2	RQCOL1	RQCOL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **RQCOL3:** Channel 3 Transfer Request Collision Flag bit
1 = User FORCE and interrupt-based request collision are detected
0 = No request collision is detected

bit 2 **RQCOL2:** Channel 2 Transfer Request Collision Flag bit
1 = User FORCE and interrupt-based request collision are detected
0 = No request collision is detected

bit 1 **RQCOL1:** Channel 1 Transfer Request Collision Flag bit
1 = User FORCE and interrupt-based request collision are detected
0 = No request collision is detected

bit 0 **RQCOL0:** Channel 0 Transfer Request Collision Flag bit
1 = User FORCE and interrupt-based request collision are detected
0 = No request collision is detected

REGISTER 8-13: DMALCA: DMA LAST CHANNEL ACTIVE STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R-1	R-1	R-1	R-1
—	—	—	—	LSTCH<3:0>			
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3-0 **LSTCH<3:0>:** Last DMA Controller Channel Active Status bits

1111 = No DMA transfer has occurred since system Reset
1110 = Reserved
•
•
•
0100 = Reserved
0011 = Last data transfer was handled by Channel 3
0010 = Last data transfer was handled by Channel 2
0001 = Last data transfer was handled by Channel 1
0000 = Last data transfer was handled by Channel 0

REGISTER 8-14: DMAPPS: DMA PING-PONG STATUS REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	PPST3	PPST2	PPST1	PPST0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **PPST3:** Channel 3 Ping-Pong Mode Status Flag bit
1 = DMA3STB register is selected
0 = DMA3STA register is selected

bit 2 **PPST2:** Channel 2 Ping-Pong Mode Status Flag bit
1 = DMA2STB register is selected
0 = DMA2STA register is selected

bit 1 **PPST1:** Channel 1 Ping-Pong Mode Status Flag bit
1 = DMA1STB register is selected
0 = DMA1STA register is selected

bit 0 **PPST0:** Channel 0 Ping-Pong Mode Status Flag bit
1 = DMA0STB register is selected
0 = DMA0STA register is selected

NOTES:

9.0 OSCILLATOR CONFIGURATION

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Oscillator Module"** (DS70005131) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

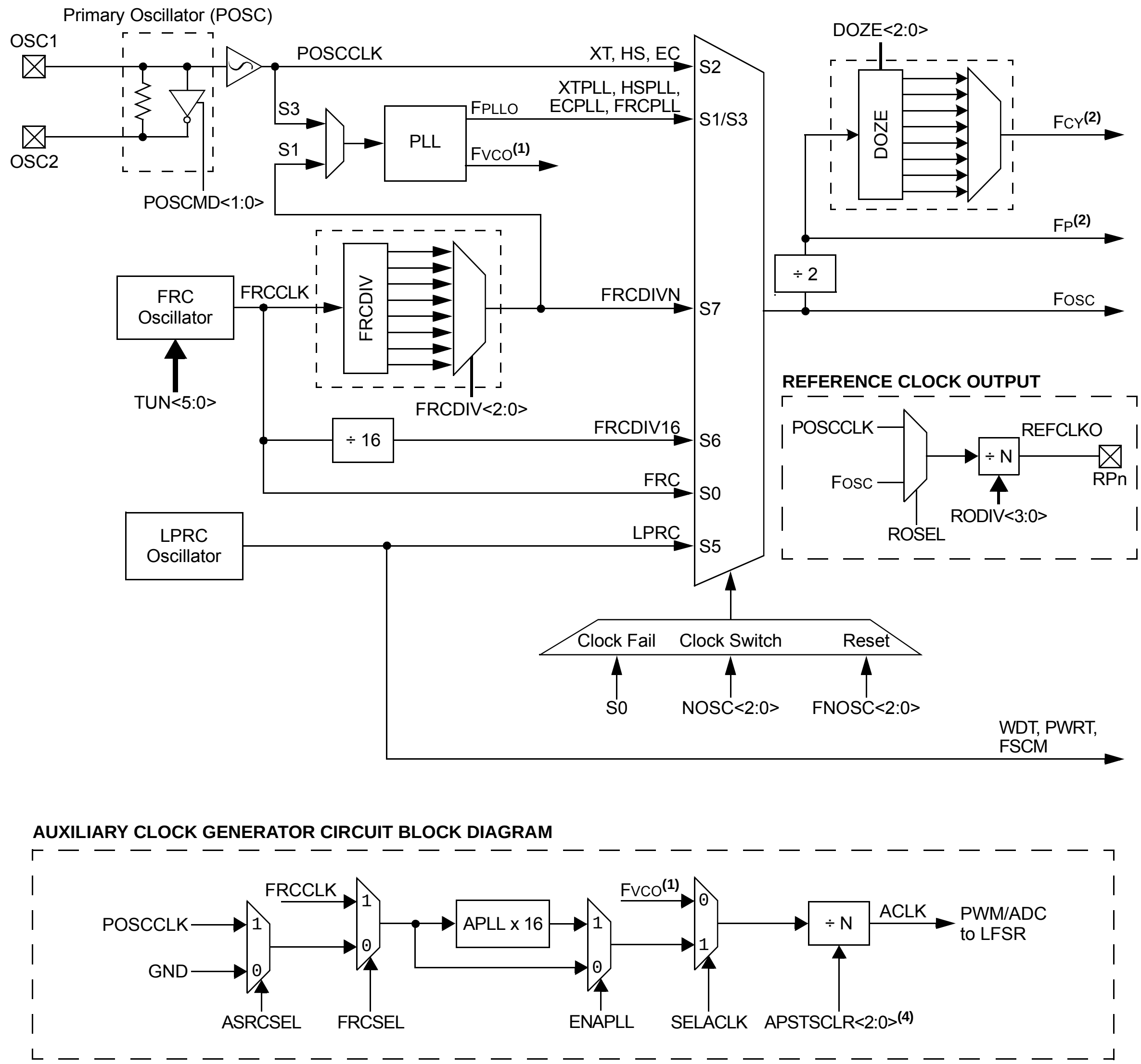
2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family oscillator system provides:

- On-Chip Phase-Locked Loop (PLL) to Boost Internal Operating Frequency on Select Internal and External Oscillator Sources
- On-the-Fly Clock Switching between Various Clock Sources
- Doze mode for System Power Savings
- Fail-Safe Clock Monitor (FSCM) that Detects Clock Failure and Permits Safe Application Recovery or Shutdown
- Configuration Bits for Clock Source Selection
- Auxiliary PLL for ADC and PWM

A simplified diagram of the oscillator system is shown in Figure 9-1.

FIGURE 9-1: OSCILLATOR SYSTEM DIAGRAM



Note 1: See Figure 9-2 for the source of the FVCO signal.

2: FP refers to the clock source for all the peripherals, while FCY (or MIPS) refers to the clock source for the CPU. Throughout this document, FCY and FP are used interchangeably, except in the case of Doze mode. FP and FCY will be different when Doze mode is used in any ratio other than 1:1.

3: The auxiliary clock postscaler must be configured to divide-by-1 (APSTSCLR<2:0> = `111`) for proper operation of the PWM and ADC modules.

9.1 CPU Clocking System

The dsPIC33EPXXXGS70X/80X family of devices provides six system clock options:

- Fast RC (FRC) Oscillator
- FRC Oscillator with Phase-Locked Loop (FRCPLL)
- FRC Oscillator with Postscaler
- Primary (XT, HS or EC) Oscillator
- Primary Oscillator with PLL (XTPLL, HSPLL, ECPLL)
- Low-Power RC (LPRC) Oscillator

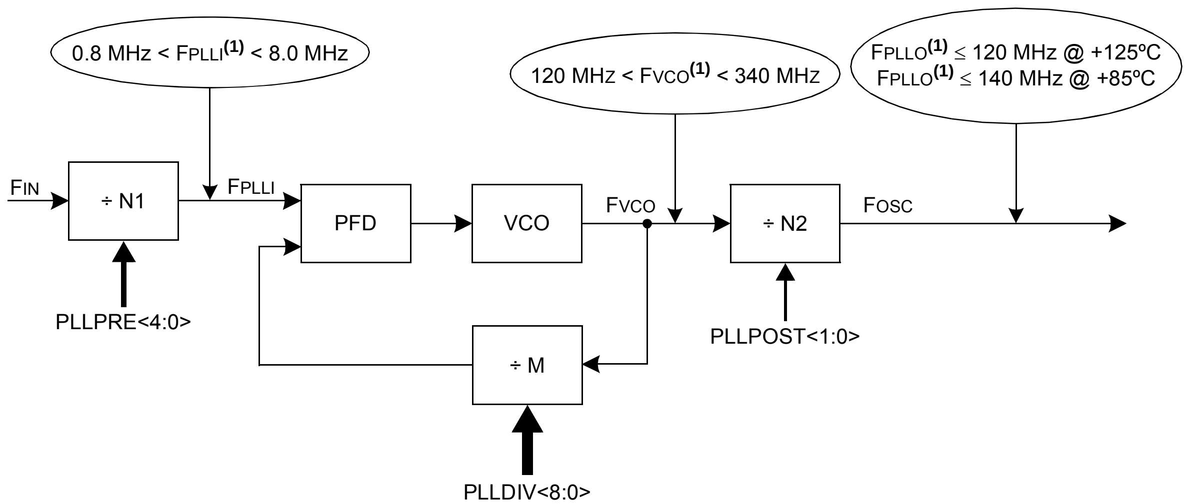
Instruction execution speed or device operating frequency, F_{CY}, is given by Equation 9-1.

EQUATION 9-1: DEVICE OPERATING FREQUENCY

$$F_{CY} = F_{OSC}/2$$

Figure 9-2 is a block diagram of the PLL module. Equation 9-2 provides the relationship between Input Frequency (F_{IN}) and Output Frequency (F_{PLLO}). Equation 9-3 provides the relationship between Input Frequency (F_{IN}) and VCO Frequency (F_{VCO}).

FIGURE 9-2: PLL BLOCK DIAGRAM



Note 1: This frequency range must be met at all times.

EQUATION 9-2: FPLLO CALCULATION

$$F_{PLLO} = F_{IN} \times \left(\frac{M}{N1 \times N2}\right) = F_{IN} \times \left(\frac{PLLDIV\langle 8:0\rangle + 2}{(PLLPRE\langle 4:0\rangle + 2) \times 2(PLLPOST\langle 1:0\rangle + 1)}\right)$$

Where:

$N1 = PLLPRE\langle 4:0\rangle + 2$

$N2 = 2 \text{ x } (PLLPOST\langle 1:0\rangle + 1)$

$M = PLLDIV\langle 8:0\rangle + 2$

EQUATION 9-3: FVCO CALCULATION

$$F_{VCO} = F_{IN} \times \left(\frac{M}{N1}\right) = F_{IN} \times \left(\frac{PLLDIV\langle 8:0\rangle + 2}{(PLLPRE\langle 4:0\rangle + 2)}\right)$$

TABLE 9-1: CONFIGURATION BIT VALUES FOR CLOCK SELECTION

Oscillator Mode	Oscillator Source	POSCMD<1:0>	FNOSC<2:0>	See Notes
Fast RC Oscillator with Divide-by-n (FRCDIVN)	Internal	xx	111	1, 2
Fast RC Oscillator with Divide-by-16	Internal	xx	110	1
Low-Power RC Oscillator (LPRC)	Internal	xx	101	1
Primary Oscillator (HS) with PLL (HSPLL)	Primary	10	011	
Primary Oscillator (XT) with PLL (XTPLL)	Primary	01	011	
Primary Oscillator (EC) with PLL (ECPLL)	Primary	00	011	1
Primary Oscillator (HS)	Primary	10	010	
Primary Oscillator (XT)	Primary	01	010	
Primary Oscillator (EC)	Primary	00	010	1
Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCPLL)	Internal	xx	001	1
Fast RC Oscillator (FRC)	Internal	xx	000	1

Note 1: OSC2 pin function is determined by the OSCIOFNC Configuration bit.
2: This is the default Oscillator mode for an unprogrammed (erased) device.

9.2 Auxiliary Clock Generation

The auxiliary clock generation is used for peripherals that need to operate at a frequency unrelated to the system clock, such as PWM or ADC.

The primary oscillator and internal FRC oscillator sources can be used with an Auxiliary PLL (APLL) to obtain the auxiliary clock. The Auxiliary PLL has a fixed 16x multiplication factor.

The auxiliary clock has the following configuration restrictions:

- For proper PWM operation, auxiliary clock generation must be configured for 120 MHz (see Parameter OS56 in **Section 30.0 “Electrical Characteristics”**). If a slower frequency is desired, the PWM Input Clock Prescaler (Divider) Select bits (PCLKDIV<2:0>) should be used.
- To achieve 1.04 ns PWM resolution, the auxiliary clock must use the 16x Auxiliary PLL (APLL). All other clock sources will have a minimum PWM resolution of 8 ns.
- If the primary PLL is used as a source for the auxiliary clock, the primary PLL should be configured up to a maximum operation of 30 MIPS or less.

9.3 Reference Clock Generation

The reference clock output logic provides the user with the ability to output a clock signal based on the system clock or the crystal oscillator on a device pin. The user application can specify a wide range of clock scaling prior to outputting the reference clock.

9.4 Oscillator Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

9.4.1 KEY RESOURCES

- **“Oscillator Module”** (DS70005131) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

9.5 Oscillator Control Registers

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER[1]

U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y
—	COSC2	COSC1	COSC0	—	NOSC2[2]	NOSC1[2]	NOSC0[2]
bit 15							bit 8

R/W-0	R/W-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
CLKLOCK	IOLOCK	LOCK	—	CF[3]	—	—	OSWEN
bit 7							bit 0

Legend:	y = Value set from Configuration bits on POR		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-12 **COSC<2:0>:** Current Oscillator Selection bits (read-only)
111 = Fast RC Oscillator (FRC) with Divide-by-n
110 = Fast RC Oscillator (FRC) with Divide-by-16
101 = Low-Power RC Oscillator (LPRC)
100 = Reserved
011 = Primary Oscillator (XT, HS, EC) with PLL
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCPLL)
000 = Fast RC Oscillator (FRC)

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **NOSC<2:0>:** New Oscillator Selection bits[2]
111 = Fast RC Oscillator (FRC) with Divide-by-n
110 = Fast RC Oscillator (FRC) with Divide-by-16
101 = Low-Power RC Oscillator (LPRC)
100 = Reserved
011 = Primary Oscillator (XT, HS, EC) with PLL
010 = Primary Oscillator (XT, HS, EC)
001 = Fast RC Oscillator (FRC) with Divide-by-N and PLL (FRCPLL)
000 = Fast RC Oscillator (FRC)

bit 7 **CLKLOCK:** Clock Lock Enable bit
1 = If (FCKSM0 = 1), then clock and PLL configurations are locked; if (FCKSM0 = 0), then clock and PLL configurations may be modified
0 = Clock and PLL selections are not locked, configurations may be modified

bit 6 **IOLOCK:** I/O Lock Enable bit
1 = I/O lock is active
0 = I/O lock is not active

bit 5 **LOCK:** PLL Lock Status bit (read-only)
1 = Indicates that PLL is in lock or PLL start-up timer is satisfied
0 = Indicates that PLL is out of lock, start-up timer is in progress or PLL is disabled

Note 1: Writes to this register require an unlock sequence.

2: Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.

3: This bit should only be cleared in software. Setting the bit in software (= 1) will have the same effect as an actual oscillator failure and will trigger an oscillator failure trap.

REGISTER 9-1: OSCCON: OSCILLATOR CONTROL REGISTER[1] (CONTINUED)

bit 4	**Unimplemented:** Read as '0'
bit 3	**CF:** Clock Fail Detect bit[3] 1 = FSCM has detected a clock failure 0 = FSCM has not detected a clock failure
bit 2-1	**Unimplemented:** Read as '0'
bit 0	**OSWEN:** Oscillator Switch Enable bit 1 = Requests oscillator switch to the selection specified by the NOSC<2:0> bits 0 = Oscillator switch is complete

Note 1: Writes to this register require an unlock sequence.

2: Direct clock switches between any Primary Oscillator mode with PLL and FRCPLL mode are not permitted. This applies to clock switches in either direction. In these instances, the application must switch to FRC mode as a transitional clock source between the two PLL modes.

3: This bit should only be cleared in software. Setting the bit in software (= 1) will have the same effect as an actual oscillator failure and will trigger an oscillator failure trap.

REGISTER 9-2: CLKDIV: CLOCK DIVISOR REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE2[(1)]	DOZE1[(1)]	DOZE0[(1)]	DOZEN[(2,3)]	FRCDIV2	FRCDIV1	FRCDIV0
bit 15							bit 8

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST1	PLLPOST0	—	PLLPRE4	PLLPRE3	PLLPRE2	PLLPRE1	PLLPRE0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ROI:** Recover on Interrupt bit
1 = Interrupts will clear the DOZEN bit and the processor clock, and the peripheral clock ratio is set to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits[(1)]
111 = FCY divided by 128
110 = FCY divided by 64
101 = FCY divided by 32
100 = FCY divided by 16
011 = FCY divided by 8 (default)
010 = FCY divided by 4
001 = FCY divided by 2
000 = FCY divided by 1

bit 11 **DOZEN:** Doze Mode Enable bit[(2,3)]
1 = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks
0 = Processor clock and peripheral clock ratio is forced to 1:1

bit 10-8 **FRCDIV<2:0>:** Internal Fast RC Oscillator Postscaler bits
111 = FRC divided by 256
110 = FRC divided by 64
101 = FRC divided by 32
100 = FRC divided by 16
011 = FRC divided by 8
010 = FRC divided by 4
001 = FRC divided by 2
000 = FRC divided by 1 (default)

bit 7-6 **PLLPOST<1:0>:** PLL VCO Output Divider Select bits (also denoted as ‘N2’, PLL postscaler)
11 = Output divided by 8
10 = Reserved
01 = Output divided by 4 (default)
00 = Output divided by 2

bit 5 **Unimplemented:** Read as ‘0’

Note 1: The DOZE<2:0> bits can only be written to when the DOZEN bit is clear. If DOZEN = 1, any writes to DOZE<2:0> are ignored.
2: This bit is cleared when the ROI bit is set and an interrupt occurs.
3: The DOZEN bit cannot be set if DOZE<2:0> = 000. If DOZE<2:0> = 000, any attempt by user software to set the DOZEN bit is ignored.

REGISTER 9-2: CLKDIV: CLOCK DIVISOR REGISTER (CONTINUED)

bit 4-0 **PLLPRE<4:0>:** PLL Phase Detector Input Divider Select bits (also denoted as 'N1', PLL prescaler)
11111 = Input divided by 33
•
•
•
00001 = Input divided by 3
00000 = Input divided by 2 (default)

Note 1: The DOZE<2:0> bits can only be written to when the DOZEN bit is clear. If DOZEN = 1, any writes to DOZE<2:0> are ignored.
2: This bit is cleared when the ROI bit is set and an interrupt occurs.
3: The DOZEN bit cannot be set if DOZE<2:0> = 000. If DOZE<2:0> = 000, any attempt by user software to set the DOZEN bit is ignored.

REGISTER 9-3: PLLFBD: PLL FEEDBACK DIVISOR REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV8
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-9 **Unimplemented:** Read as '0'

bit 8-0 **PLLDIV<8:0>:** PLL Feedback Divisor bits (also denoted as 'M', PLL multiplier)
111111111 = 513
•
•
•
000110000 = 50 (default)
•
•
•
000000010 = 4
000000001 = 3
000000000 = 2

REGISTER 9-4: OSCTUN: FRC OSCILLATOR TUNING REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN<5:0>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **TUN<5:0>:** FRC Oscillator Tuning bits

011111 = Maximum frequency deviation of 1.457% (7.477 MHz)
011110 = Center frequency + 1.41% (7.474 MHz)
•
•
•
000001 = Center frequency + 0.047% (7.373 MHz)
000000 = Center frequency (7.37 MHz nominal)
111111 = Center frequency – 0.047% (7.367 MHz)
•
•
•
100001 = Center frequency – 1.457% (7.263 MHz)
100000 = Minimum frequency deviation of -1.5% (7.259 MHz)

REGISTER 9-5: ACLKCON: AUXILIARY CLOCK DIVISOR CONTROL REGISTER

R/W-0	R-0	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-1
ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR2	APSTSCLR1	APSTSCLR0
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
ASRCSEL	FRCSEL	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ENAPLL:** Auxiliary PLL Enable bit
1 = APLL is enabled
0 = APLL is disabled

bit 14 **APLLCK:** APLL Locked Status bit (read-only)
1 = Indicates that Auxiliary PLL is in lock
0 = Indicates that Auxiliary PLL is not in lock

bit 13 **SELACLK:** Select Auxiliary Clock Source for Auxiliary Clock Divider bit
1 = Auxiliary oscillators provide the source clock for the auxiliary clock divider
0 = Primary PLL (F_{VCO}) provides the source clock for the auxiliary clock divider

bit 12-11 **Unimplemented:** Read as ‘0’

bit 10-8 **APSTSCLR<2:0>:** Auxiliary Clock Output Divider bits
111 = Divided by 1
110 = Divided by 2
101 = Divided by 4
100 = Divided by 8
011 = Divided by 16
010 = Divided by 32
001 = Divided by 64
000 = Divided by 256

bit 7 **ASRCSEL:** Select Reference Clock Source for Auxiliary Clock bit
1 = Primary oscillator is the clock source
0 = No clock input is selected

bit 6 **FRCSEL:** Select Reference Clock Source for Auxiliary PLL bit
1 = Selects the FRC clock for Auxiliary PLL
0 = Input clock source is determined by the ASRCSEL bit setting

bit 5-0 **Unimplemented:** Read as ‘0’

REGISTER 9-6: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL	RODIV3[(1)]	RODIV2[(1)]	RODIV1[(1)]	RODIV0[(1)]
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ROON:** Reference Oscillator Output Enable bit
1 = Reference oscillator output is enabled on the RPn pin[(2)]
0 = Reference oscillator output is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **ROSSLP:** Reference Oscillator Run in Sleep bit
1 = Reference oscillator output continues to run in Sleep
0 = Reference oscillator output is disabled in Sleep

bit 12 **ROSEL:** Reference Oscillator Source Select bit
1 = Oscillator crystal is used as the reference clock
0 = System clock is used as the reference clock

bit 11-8 **RODIV<3:0>:** Reference Oscillator Divider bits[(1)]
1111 = Reference clock divided by 32,768
1110 = Reference clock divided by 16,384
1101 = Reference clock divided by 8,192
1100 = Reference clock divided by 4,096
1011 = Reference clock divided by 2,048
1010 = Reference clock divided by 1,024
1001 = Reference clock divided by 512
1000 = Reference clock divided by 256
0111 = Reference clock divided by 128
0110 = Reference clock divided by 64
0101 = Reference clock divided by 32
0100 = Reference clock divided by 16
0011 = Reference clock divided by 8
0010 = Reference clock divided by 4
0001 = Reference clock divided by 2
0000 = Reference clock

bit 7-0 **Unimplemented:** Read as ‘0’

Note 1: The reference oscillator output must be disabled (ROON = 0) before writing to these bits.
2: This pin is remappable. See **Section 11.6 “Peripheral Pin Select (PPS)”** for more information.

REGISTER 9-7: LFSR: LINEAR FEEDBACK SHIFT REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	LFSR<14:8>						
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LFSR<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-0 **LFSR<14:0>:** Pseudorandom Data bits

10.0 POWER-SAVING FEATURES

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Watchdog Timer and Power-Saving Modes"** (DS70615) in the "*dsPIC33/PIC24 Family Reference Manual*", which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family devices provide the ability to manage power consumption by selectively managing clocking to the CPU and the peripherals. In general, a lower clock frequency and a reduction in the number of peripherals being clocked constitutes lower consumed power.

dsPIC33EPXXXGS70X/80X family devices can manage power consumption in four ways:

- Clock Frequency
- Instruction-Based Sleep and Idle modes
- Software-Controlled Doze mode
- Selective Peripheral Control in Software

Combinations of these methods can be used to selectively tailor an application's power consumption while still maintaining critical application features, such as timing-sensitive communications.

10.1 Clock Frequency and Clock Switching

The dsPIC33EPXXXGS70X/80X family devices allow a wide range of clock frequencies to be selected under application control. If the system clock configuration is not locked, users can choose low-power or high-precision oscillators by simply changing the NOSCx bits (OSCCON<10:8>). The process of changing a system clock during operation, as well as limitations to the process, are discussed in more detail in **Section 9.0 "Oscillator Configuration"**.

10.2 Instruction-Based Power-Saving Modes

The dsPIC33EPXXXGS70X/80X family devices have two special power-saving modes that are entered through the execution of a special `PWRSAV` instruction. Sleep mode stops clock operation and halts all code execution. Idle mode halts the CPU and code execution, but allows peripheral modules to continue operation. The assembler syntax of the `PWRSAV` instruction is shown in Example 10-1.

> **Note:** `SLEEP_MODE` and `IDLE_MODE` are constants defined in the assembler include file for the selected device.

Sleep and Idle modes can be exited as a result of an enabled interrupt, WDT time-out or a device Reset. When the device exits these modes, it is said to "wake-up".

EXAMPLE 10-1: PWRSAV INSTRUCTION SYNTAX

```
PWRSAV  #SLEEP_MODE    ; Put the device into Sleep mode
PWRSAV  #IDLE_MODE     ; Put the device into Idle mode
```

10.2.1 SLEEP MODE

The following occurs in Sleep mode:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption is reduced to a minimum, provided that no I/O pin is sourcing current.
- The Fail-Safe Clock Monitor does not operate, since the system clock source is disabled.
- The LPRC clock continues to run in Sleep mode if the WDT is enabled.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some device features or peripherals can continue to operate. This includes items such as the Input Change Notification on the I/O ports or peripherals that use an external clock input.
- Any peripheral that requires the system clock source for its operation is disabled.

The device wakes up from Sleep mode on any of the these events:

- Any interrupt source that is individually enabled
- Any form of device Reset
- A WDT time-out

On wake-up from Sleep mode, the processor restarts with the same clock source that was active when Sleep mode was entered.

For optimal power savings, the internal regulator and the Flash regulator can be configured to go into standby when Sleep mode is entered by clearing the VREGS (RCON<8>) and VREGSF (RCON<11>) bits (default configuration).

If the application requires a faster wake-up time, and can accept higher current requirements, the VREGS (RCON<8>) and VREGSF (RCON<11>) bits can be set to keep the internal regulator and the Flash regulator active during Sleep mode.

10.2.2 IDLE MODE

The following occurs in Idle mode:

- The CPU stops executing instructions.
- The WDT is automatically cleared.
- The system clock source remains active. By default, all peripheral modules continue to operate normally from the system clock source, but can also be selectively disabled (see **Section 10.4 “Peripheral Module Disable”**).
- If the WDT or FSCM is enabled, the LPRC also remains active.

The device wakes from Idle mode on any of these events:

- Any interrupt that is individually enabled
- Any device Reset
- A WDT time-out

On wake-up from Idle mode, the clock is reapplied to the CPU and instruction execution will begin (2-4 clock cycles later), starting with the instruction following the `PWRSAV` instruction or the first instruction in the ISR.

All peripherals also have the option to discontinue operation when Idle mode is entered to allow for increased power savings. This option is selectable in the control register of each peripheral (for example, the TSIDL bit in the Timer1 Control register (T1CON<13>).

10.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

Any interrupt that coincides with the execution of a `PWRSAV` instruction is held off until entry into Sleep or Idle mode has completed. The device then wakes up from Sleep or Idle mode.

10.3 Doze Mode

The preferred strategies for reducing power consumption are changing clock speed and invoking one of the power-saving modes. In some circumstances, this cannot be practical. For example, it may be necessary for an application to maintain uninterrupted synchronous communication, even while it is doing nothing else. Reducing system clock speed can introduce communication errors, while using a power-saving mode can stop communications completely.

Doze mode is a simple and effective alternative method to reduce power consumption while the device is still executing code. In this mode, the system clock continues to operate from the same source and at the same speed. Peripheral modules continue to be clocked at the same speed, while the CPU clock speed is reduced. Synchronization between the two clock domains is maintained, allowing the peripherals to access the SFRs while the CPU executes code at a slower rate.

Doze mode is enabled by setting the DOZEN bit (CLKDIV<11>). The ratio between peripheral and core clock speed is determined by the DOZE<2:0> bits (CLKDIV<14:12>). There are eight possible configurations, from 1:1 to 1:128, with 1:1 being the default setting.

Programs can use Doze mode to selectively reduce power consumption in event-driven applications. This allows clock-sensitive functions, such as synchronous communications, to continue without interruption while the CPU Idles, waiting for something to invoke an interrupt routine. An automatic return to full-speed CPU operation on interrupts can be enabled by setting the ROI bit (CLKDIV<15>). By default, interrupt events have no effect on Doze mode operation.

10.4 Peripheral Module Disable

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled using the appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral are also disabled, so writes to those registers do not have any effect and read values are invalid.

A peripheral module is enabled only if both the associated bit in the PMD register is cleared and the peripheral is supported by the specific dsPIC® DSC variant. If the peripheral is present in the device, it is enabled in the PMD register by default.

Note:	If a PMD bit is set, the corresponding module is disabled after a delay of one instruction cycle. Similarly, if a PMD bit is cleared, the corresponding module is enabled after a delay of one instruction cycle (assuming the module control registers are already configured to enable module operation).

10.5 Power-Saving Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

10.5.1 KEY RESOURCES

- **"Watchdog Timer and Power-Saving Modes"** (DS70615) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
T5MD	T4MD	T3MD	T2MD	T1MD	—	PWMMD	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	ADCMD
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **T5MD:** Timer5 Module Disable bit
1 = Timer5 module is disabled
0 = Timer5 module is enabled

bit 14 **T4MD:** Timer4 Module Disable bit
1 = Timer4 module is disabled
0 = Timer4 module is enabled

bit 13 **T3MD:** Timer3 Module Disable bit
1 = Timer3 module is disabled
0 = Timer3 module is enabled

bit 12 **T2MD:** Timer2 Module Disable bit
1 = Timer2 module is disabled
0 = Timer2 module is enabled

bit 11 **T1MD:** Timer1 Module Disable bit
1 = Timer1 module is disabled
0 = Timer1 module is enabled

bit 10 **Unimplemented:** Read as ‘0’

bit 9 **PWMMD:** PWMx Module Disable bit
1 = PWMx module is disabled
0 = PWMx module is enabled

bit 8 **Unimplemented:** Read as ‘0’

bit 7 **I2C1MD:** I2C1 Module Disable bit
1 = I2C1 module is disabled
0 = I2C1 module is enabled

bit 6 **U2MD:** UART2 Module Disable bit
1 = UART2 module is disabled
0 = UART2 module is enabled

bit 5 **U1MD:** UART1 Module Disable bit
1 = UART1 module is disabled
0 = UART1 module is enabled

bit 4 **SPI2MD:** SPI2 Module Disable bit
1 = SPI2 module is disabled
0 = SPI2 module is enabled

bit 3 **SPI1MD:** SPI1 Module Disable bit
1 = SPI1 module is disabled
0 = SPI1 module is enabled

bit 2 **C2MD:** CAN2 Module Disable bit
1 = CAN2 module is disabled
0 = CAN2 module is enabled

REGISTER 10-1: PMD1: PERIPHERAL MODULE DISABLE CONTROL REGISTER 1 (CONTINUED)

bit 1	**C1MD:** CAN1 Module Disable bit 1 = CAN1 module is disabled 0 = CAN1 module is enabled
bit 0	**ADCMD:** ADC Module Disable bit 1 = ADC module is disabled 0 = ADC module is enabled

REGISTER 10-2: PMD2: PERIPHERAL MODULE DISABLE CONTROL REGISTER 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	OC4MD	OC3MD	OC2MD	OC1MD
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11 **IC4MD:** Input Capture 4 Module Disable bit
1 = Input Capture 4 module is disabled
0 = Input Capture 4 module is enabled

bit 10 **IC3MD:** Input Capture 3 Module Disable bit
1 = Input Capture 3 module is disabled
0 = Input Capture 3 module is enabled

bit 9 **IC2MD:** Input Capture 2 Module Disable bit
1 = Input Capture 2 module is disabled
0 = Input Capture 2 module is enabled

bit 8 **IC1MD:** Input Capture 1 Module Disable bit
1 = Input Capture 1 module is disabled
0 = Input Capture 1 module is enabled

bit 7-4 **Unimplemented:** Read as ‘0’

bit 3 **OC4MD:** Output Compare 4 Module Disable bit
1 = Output Compare 4 module is disabled
0 = Output Compare 4 module is enabled

bit 2 **OC3MD:** Output Compare 3 Module Disable bit
1 = Output Compare 3 module is disabled
0 = Output Compare 3 module is enabled

bit 1 **OC2MD:** Output Compare 2 Module Disable bit
1 = Output Compare 2 module is disabled
0 = Output Compare 2 module is enabled

bit 0 **OC1MD:** Output Compare 1 Module Disable bit
1 = Output Compare 1 module is disabled
0 = Output Compare 1 module is enabled

REGISTER 10-3: PMD3: PERIPHERAL MODULE DISABLE CONTROL REGISTER 3

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	CMPMD	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	I2C2MD	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-11 **Unimplemented:** Read as ‘0’

bit 10 **CMPMD:** Comparator Module Disable bit
1 = Comparator module is disabled
0 = Comparator module is enabled

bit 9-2 **Unimplemented:** Read as ‘0’

bit 1 **I2C2MD:** I2C2 Module Disable bit
1 = I2C2 module is disabled
0 = I2C2 module is enabled

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 10-4: PMD4: PERIPHERAL MODULE DISABLE CONTROL REGISTER 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
—	—	—	—	REFOMD	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **REFOMD:** Reference Clock Module Disable bit
1 = Reference clock module is disabled
0 = Reference clock module is enabled

bit 2-0 **Unimplemented:** Read as ‘0’

REGISTER 10-5: PMD6: PERIPHERAL MODULE DISABLE CONTROL REGISTER 6

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	SPI3MD
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **PWM8MD:** PWM8 Module Disable bit
1 = PWM8 module is disabled
0 = PWM8 module is enabled

bit 14 **PWM7MD:** PWM7 Module Disable bit
1 = PWM7 module is disabled
0 = PWM7 module is enabled

bit 13 **PWM6MD:** PWM6 Module Disable bit
1 = PWM6 module is disabled
0 = PWM6 module is enabled

bit 12 **PWM5MD:** PWM5 Module Disable bit
1 = PWM5 module is disabled
0 = PWM5 module is enabled

bit 11 **PWM4MD:** PWM4 Module Disable bit
1 = PWM4 module is disabled
0 = PWM4 module is enabled

bit 10 **PWM3MD:** PWM3 Module Disable bit
1 = PWM3 module is disabled
0 = PWM3 module is enabled

bit 9 **PWM2MD:** PWM2 Module Disable bit
1 = PWM2 module is disabled
0 = PWM2 module is enabled

bit 8 **PWM1MD:** PWM1 Module Disable bit
1 = PWM1 module is disabled
0 = PWM1 module is enabled

bit 7-1 **Unimplemented:** Read as ‘0’

bit 0 **SPI3MD:** SPI3 Module Disable bit
1 = SPI3 module is disabled
0 = SPI3 module is enabled

REGISTER 10-6: PMD7: PERIPHERAL MODULE DISABLE CONTROL REGISTER 7

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	—	—	DMAMD	PTGMD	—	PGA1MD	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11 **CMP4MD:** CMP4 Module Disable bit
1 = CMP4 module is disabled
0 = CMP4 module is enabled

bit 10 **CMP3MD:** CMP3 Module Disable bit
1 = CMP3 module is disabled
0 = CMP3 module is enabled

bit 9 **CMP2MD:** CMP2 Module Disable bit
1 = CMP2 module is disabled
0 = CMP2 module is enabled

bit 8 **CMP1MD:** CMP1 Module Disable bit
1 = CMP1 module is disabled
0 = CMP1 module is enabled

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4 **DMAMD:** DMA Module Disable bit
1 = DMA module is disabled
0 = DMA module is enabled

bit 3 **PTGMD:** PTG Module Disable bit
1 = PTG module is disabled
0 = PTG module is enabled

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **PGA1MD:** PGA1 Module Disable bit
1 = PGA1 module is disabled
0 = PGA1 module is enabled

bit 0 **Unimplemented:** Read as ‘0’

REGISTER 10-7: PMD8: PERIPHERAL MODULE DISABLE CONTROL REGISTER 8

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	PGA2MD	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	CLC4MD	CLC3MD	CLC2MD	CLC1MD	CCSMD	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-11 **Unimplemented:** Read as ‘0’

bit 10 **PGA2MD:** PGA2 Module Disable bit
1 = PGA2 module is disabled
0 = PGA2 module is enabled

bit 9-6 **Unimplemented:** Read as ‘0’

bit 5 **CLC4MD:** CLC4 Module Disable bit
1 = CLC4 module is disabled
0 = CLC4 module is enabled

bit 4 **CLC3MD:** CLC3 Module Disable bit
1 = CLC3 module is disabled
0 = CLC3 module is enabled

bit 3 **CLC2MD:** CLC2 Module Disable bit
1 = CLC2 module is disabled
0 = CLC2 module is enabled

bit 2 **CLC1MD:** CLC1 Module Disable bit
1 = CLC1 module is disabled
0 = CLC1 module is enabled

bit 1 **CCSMD:** Constant-Current Source Module Disable bit
1 = Constant-current source module is disabled
0 = Constant-current source module is enabled

bit 0 **Unimplemented:** Read as ‘0’

11.0 I/O PORTS

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"I/O Ports"** (DS70000598) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

Many of the device pins are shared among the peripherals and the Parallel I/O ports. All I/O input ports feature Schmitt Trigger inputs for improved noise immunity.

11.1 Parallel I/O (PIO) Ports

Generally, a Parallel I/O port that shares a pin with a peripheral is subservient to the peripheral. The peripheral's output buffer data and control signals are provided to a pair of multiplexers. The multiplexers select whether the peripheral or the associated port has ownership of the output data and control signals of the I/O pin. The logic also prevents "loop through", in which a port's digital output can drive the input of a peripheral that shares the same pin. Figure 11-1 illustrates how ports are shared with other peripherals and the associated I/O pin to which they are connected.

When a peripheral is enabled and the peripheral is actively driving an associated pin, the use of the pin as a general purpose output pin is disabled. The I/O pin can be read, but the output driver for the parallel port bit is disabled. If a peripheral is enabled, but the peripheral is not actively driving a pin, that pin can be driven by a port.

All port pins have eight registers directly associated with their operation as digital I/Os. The Data Direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Reads from the latch (LATx), read the latch. Writes to the latch, write the latch. Reads from the port (PORTx), read the port pins, while writes to the port pins, write the latch.

Any bit and its associated data and control registers that are not valid for a particular device are disabled. This means the corresponding LATx and TRISx registers, and the port pin are read as zeros. Table 11-1 through Table 11-5 show ANSELx bits' availability for device variants.

When a pin is shared with another peripheral or function that is defined as an input only, it is nevertheless regarded as a dedicated port because there is no other competing source of outputs.

FIGURE 11-1: BLOCK DIAGRAM OF A TYPICAL SHARED PORT STRUCTURE

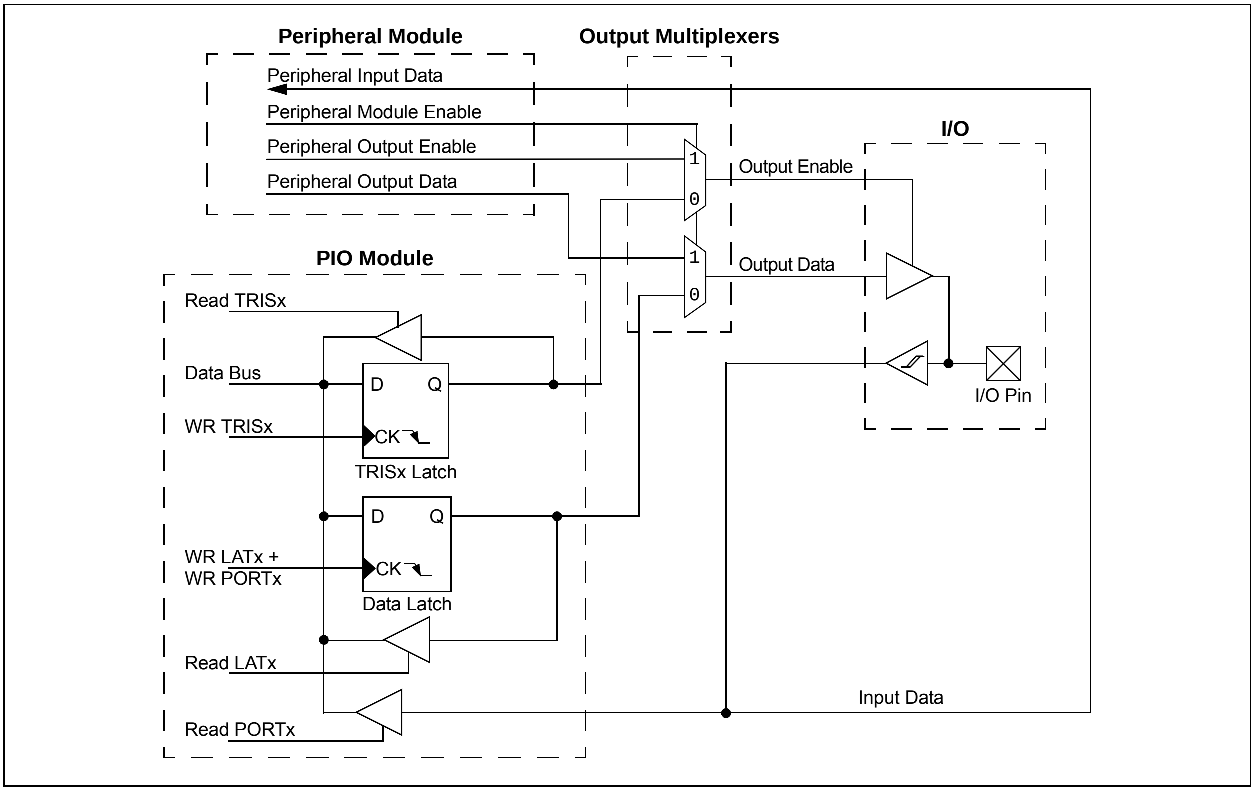


TABLE 11-1: PORTA PIN AND ANSELA AVAILABILITY

Device	PORTA I/O Pins															
	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
dsPIC33EPXXXGSX08	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXXGSX06	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXXGSX05	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXXGSX04	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXXGS702	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
ANSELA Bit Present	—	—	—	—	—	—	—	—	—	—	—	—	—	X	X	X

TABLE 11-2: PORTB PIN AND ANSELB AVAILABILITY

Device	PORTB I/O Pins															
	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
dsPIC33EPXXXGSX08	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX06	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX05	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX04	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGS702	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
ANSELB Bit Present	—	—	—	—	—	—	X	—	X	X	X	—	X	X	X	X

TABLE 11-3: PORTC PIN AND ANSELC AVAILABILITY

Device	PORTC I/O Pins															
	RC15	RC14	RC13	RC12	RC11	RC10	RC9	RC8	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
dsPIC33EPXXXGSX08	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX06	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX05	—	—	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX04	—	—	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSELC Bit Present	—	—	—	X	—	X	X	—	—	X	X	X	—	X	X	—

TABLE 11-4: PORTD PIN AND ANSELD AVAILABILITY

Device	PORTD I/O Pins															
	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
dsPIC33EPXXXGSX08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX06	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX05	—	X	—	—	—	X	—	—	—	—	—	X	—	—	—	—
dsPIC33EPXXXGSX04	—	X	—	—	—	X	—	—	—	—	—	X	—	—	—	—
dsPIC33EPXXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSELD Bit Present	—	—	X	—	—	—	—	X	X	—	X	—	—	X	—	—

TABLE 11-5: PORTE PIN AND ANSELE AVAILABILITY

Device	PORTE I/O Pins															
	RE15	RE14	RE13	RE12	RE11	RE10	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0
dsPIC33EPXXXGSX08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXXGSX06	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXXGSX05	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXXGSX04	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSELE Bit Present	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

11.2 I/O Port Control Register Maps

TABLE 11-6: PORTA REGISTER MAP[1]

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
TRISA	—	—	—	—	—	—	—	—	—	—	—	TRIS	
PORTA	—	—	—	—	—	—	—	—	—	—	—	RA	
LATA	—	—	—	—	—	—	—	—	—	—	—	LAT	
ODCA	—	—	—	—	—	—	—	—	—	—	—	ODC	
CNENA	—	—	—	—	—	—	—	—	—	—	—	CNIE	
CNPUA	—	—	—	—	—	—	—	—	—	—	—	CNP	
CNPDA	—	—	—	—	—	—	—	—	—	—	—	CNP	
ANSELA	—	—	—	—	—	—	—	—	—	—	—	—	—

Legend: — = unimplemented, read as ‘0’.
Note 1: Refer to Table 11-1 for bit availability on each pin count variant.

TABLE 11-7: PORTB REGISTER MAP[1]

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
TRISB	TRISB<15:11>					—	TRISB<9:0>						
PORTB	RB<15:11>					—	RB<9:0>						
LATB	LATB<15:11>					—	LATB<9:0>						
ODCB	ODCB<15:11>					—	ODCB<9:0>						
CNENB	CNIEB<15:11>					—	CNIEB<9:0>						
CNPUB	CNPUB<15:11>					—	CNPUB<9:0>						
CNPDB	CNPDB<15:11>					—	CNPDB<9:0>						
ANSELB	—	—	—	—	—	—	ANSB9	—	ANSB<7:5>			—	

Legend: — = unimplemented, read as ‘0’.
Note 1: Refer to Table 11-2 for bit availability on each pin count variant.

TABLE 11-8: PORTC REGISTER MAP[1]

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
TRISC	TRISC<15:12>				—	TRISC<10:0>							
PORTC	RC<15:12>				—	RC<10:0>							
LATC	LATC<15:12>				—	LATC<10:0>							
ODCC	ODCC<15:12>				—	ODCC<10:0>							
CNENC	CNIEC<15:12>				—	CNIEC<10:0>							
CNPUC	CNPUC<15:12>				—	CNPUC<10:0>							
CNPDC	CNPDC<15:12>				—	CNPDC<10:0>							
ANSELC	—	—	—	ANSC12	—	ANSC<10:9>		—	—	ANSC<6:4>			—

Legend: — = unimplemented, read as '0'.
Note 1: Refer to Table 11-3 for bit availability on each pin count variant.

TABLE 11-9: PORTD REGISTER MAP[1]

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
TRISD	TRISD<15:0>												
PORTD	RD<15:0>												
LATD	LATD<15:0>												
ODCD	ODCD<15:0>												
CNEND	CNIED<15:0>												
CNPUD	CNPUD<15:0>												
CNPDD	CNPDD<15:0>												
ANSELD	—	—	ANSD13	—	—	—	—	ANSD<8:7>		—	ANSD5	—	—

Legend: — = unimplemented, read as '0'.
Note 1: Refer to Table 11-4 for bit availability on each pin count variant.

TABLE 11-10: PORTE REGISTER MAP[1]

File Name	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
TRISE	TRISE<15:0>												
PORTE	RE<15:0>												
LATE	LATE<15:0>												
ODCE	ODCE<15:0>												
CNENE	CNIEE<15:0>												
CNPUE	CNPUE<15:0>												
CNPDE	CNPDE<15:0>												
ANSELE	—	—	—	—	—	—	—	—	—	—	—	—	—

Legend: — = unimplemented, read as '0'.

Note 1: Refer to Table 11-5 for bit availability on each pin count variant.

11.2.1 OPEN-DRAIN CONFIGURATION

In addition to the PORTx, LATx and TRISx registers for data control, port pins can also be individually configured for either digital or open-drain output. This is controlled by the Open-Drain Control x register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs other than VDD by using external pull-up resistors. The maximum open-drain voltage allowed on any pin is the same as the maximum VIH specification for that particular pin. See the **"Pin Diagrams"** section for the available 5V tolerant pins and Table 30-11 for the maximum VIH specification for each pin.

11.3 Configuring Analog and Digital Port Pins

The ANSELx register controls the operation of the analog port pins. The port pins that are to function as analog inputs or outputs must have their corresponding ANSELx and TRISx bits set. In order to use port pins for I/O functionality with digital modules, such as timers, UARTs, etc., the corresponding ANSELx bit must be cleared.

The ANSELx register has a default value of 0xFFFF; therefore, all pins that share analog functions are analog (not digital) by default.

Pins with analog functions affected by the ANSELx registers are listed with a buffer type of analog in the Pinout I/O Descriptions (see Table 1-1). Table 11-1 through Table 11-5 show ANSELx bits' availability for device variants.

If the TRISx bit is cleared (output) while the ANSELx bit is set, the digital output level (VOH or VOL) is converted by an analog peripheral, such as the ADC module or comparator module.

When the PORTx register is read, all pins configured as analog input channels are read as cleared (a low level).

Pins configured as digital inputs do not convert an analog input. Analog levels on any pin, defined as a digital input (including the ANx pins), can cause the input buffer to consume current that exceeds the device specifications.

11.3.1 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically, this instruction would be a NOP, as shown in Example 11-1.

11.4 Input Change Notification (ICN)

The Input Change Notification function of the I/O ports allows devices to generate interrupt requests to the processor in response to a Change-of-State (COS) on selected input pins. This feature can detect input Change-of-States, even in Sleep mode, when the clocks are disabled. Every I/O port pin can be selected (enabled) for generating an interrupt request on a Change-of-State.

Three control registers are associated with the ICN functionality of each I/O port. The CNENx registers contain the ICN interrupt enable control bits for each of the input pins. Setting any of these bits enables an ICN interrupt for the corresponding pins.

Each I/O pin also has a weak pull-up and a weak pull-down connected to it. The pull-ups and pull-downs act as a current source, or sink source, connected to the pin, and eliminate the need for external resistors when push button or keypad devices are connected. The pull-ups and pull-downs are enabled separately, using the CNPUx and the CNPDx registers, which contain the control bits for each of the pins. Setting any of the control bits enables the weak pull-ups and/or pull-downs for the corresponding pins.

> **Note:** Pull-ups and pull-downs on Input Change Notification pins should always be disabled when the port pin is configured as a digital output.

EXAMPLE 11-1: PORT WRITE/READ EXAMPLE

```
MOV   0xFF00, W0    ; Configure PORTB<15:8>
                    ; as inputs
MOV   W0, TRISB     ; and PORTB<7:0>
                    ; as outputs
NOP                 ; Delay 1 cycle
BTSS  PORTB, #13    ; Next Instruction
```

11.5 I/O Port Control Registers

REGISTER 11-1: TRISx: PORTx DATA DIRECTION CONTROL REGISTER[1]

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISx<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **TRISx<15:0>:** PORTx Data Direction Control bits
1 = The pin is an input
0 = The pin is an output

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-2: PORTx: I/O PORTx REGISTER[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PORTx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PORTx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **PORTx<15:0>:** I/O PORTx bits
1 = The pin data is ‘1’
0 = The pin data is ‘0’

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-3: LATx: PORTx DATA LATCH REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LATx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LATx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **LATx<15:0>:** PORTx Data Latch bits
1 = The latch content is ‘1’
0 = The latch content is ‘0’

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-4: ODCx: PORTx OPEN-DRAIN CONTROL REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ODCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ODCx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **PORTx<15:0>:** PORTx Open-Drain Control bits
1 = The pin acts as an open-drain output pin if TRISx is ‘0’
0 = The pin acts as a normal pin

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-5: CNENx: INPUT CHANGE NOTIFICATION INTERRUPT ENABLE x REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNIEx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNIEx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **CNIEx<15:0>:** Input Change Notification Interrupt Enable x bits
1 = Enables interrupt on input change
0 = Disables interrupt on input change

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-6: CNPUx: INPUT CHANGE NOTIFICATION PULL-UP ENABLE x REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPUx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPUx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **CNPUx<15:0>:** Input Change Notification Pull-up Enable bits
1 = Enables pull-up on PORTx pin
0 = Disables pull-up on PORTx pin

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-7: CNPDx: INPUT CHANGE NOTIFICATION PULL-DOWN ENABLE x REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPDx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPDx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **CNPDx<15:0>:** Input Change Notification Pull-Down Enable x bits
1 = Enables pull-down on PORTx pin
0 = Disables pull-down on PORTx pin

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

REGISTER 11-8: ANSELx: ANALOG SELECT CONTROL x REGISTER[(1)]

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSx<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **ANSx<15:0>:** Analog PORTx Enable bits
1 = Enables analog PORTx pin
0 = Disables digital PORTx pin

Note 1: See Table 11-1, Table 11-2, Table 11-3, Table 11-4 and Table 11-5 for individual bit availability in this register.

11.6 Peripheral Pin Select (PPS)

A major challenge in general purpose devices is providing the largest possible set of peripheral features, while minimizing the conflict of features on I/O pins. The challenge is even greater on low pin count devices. In an application where more than one peripheral needs to be assigned to a single pin, inconvenient work arounds in application code, or a complete redesign, may be the only option.

Peripheral Pin Select configuration provides an alternative to these choices by enabling peripheral set selection and placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, users can better tailor the device to their entire application, rather than trimming the application to fit the device.

The Peripheral Pin Select configuration feature operates over a fixed subset of digital I/O pins. Users may independently map the input and/or output of most digital peripherals to any one of these I/O pins. Hardware safeguards are included that prevent accidental or spurious changes to the peripheral mapping once it has been established.

11.6.1 AVAILABLE PINS

The number of available pins is dependent on the particular device and its pin count. Pins that support the Peripheral Pin Select feature include the label, “RPn”, in their full pin designation, where “n” is the remappable pin number. “RP” is used to designate pins that support both remappable input and output functions.

11.6.2 AVAILABLE PERIPHERALS

The peripherals managed by the Peripheral Pin Select are all digital only peripherals. These include general serial communications (UART and SPI), general purpose timer clock inputs, timer-related peripherals (input capture and output compare) and interrupt-on-change inputs.

In comparison, some digital only peripheral modules are never included in the Peripheral Pin Select feature. This is because the peripheral’s function requires special I/O circuitry on a specific port and cannot be easily connected to multiple pins. One example includes I^2C modules. A similar requirement excludes all modules with analog inputs, such as the ADC Converter.

A key difference between remappable and non-remappable peripherals is that remappable peripherals are not associated with a default I/O pin. The peripheral must always be assigned to a specific I/O pin before it can be used. In contrast, non-remappable peripherals are always available on a default pin, assuming that the peripheral is active and not conflicting with another peripheral.

When a remappable peripheral is active on a given I/O pin, it takes priority over all other digital I/Os and digital communication peripherals associated with the pin. Priority is given regardless of the type of peripheral that is mapped. Remappable peripherals never take priority over any analog functions associated with the pin.

11.6.3 CONTROLLING PERIPHERAL PIN SELECT

Peripheral Pin Select features are controlled through two sets of SFRs: one to map peripheral inputs and one to map outputs. Because they are separately controlled, a particular peripheral’s input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

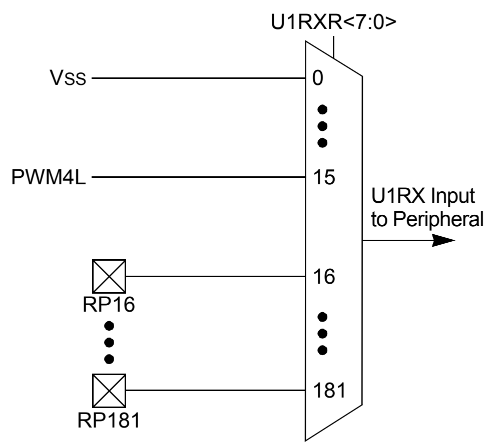
The association of a peripheral to a peripheral-selectable pin is handled in two different ways, depending on whether an input or output is being mapped.

11.6.4 INPUT MAPPING

The inputs of the Peripheral Pin Select options are mapped on the basis of the peripheral. That is, a control register associated with a peripheral dictates the pin it will be mapped to. The RPINRx registers are used to configure peripheral input mapping (see Register 11-9 through Register 11-32). Each register contains sets of 8-bit fields, with each set associated with one of the remappable peripherals. Programming a given peripheral's bit field with an appropriate 8-bit index value maps the RPn pin with the corresponding value, or internal signal, to that peripheral. See Table 11-11 for a list of available inputs.

For example, Figure 11-2 illustrates remappable pin selection for the U1RX input.

FIGURE 11-2: REMAPPABLE INPUT FOR U1RX



Note: For input only, Peripheral Pin Select functionality does not have priority over TRISx settings. Therefore, when configuring an RPn pin for input, the corresponding bit in the TRISx register must also be configured for input (set to '1').

11.6.4.1 Virtual Connections

The dsPIC33EPXXXGS70X/80X devices support six virtual RPn pins (RP176-RP181), which are identical in functionality to all other RPn pins, with the exception of pinouts. These six pins are internal to the devices and are not connected to a physical device pin.

These pins provide a simple way for inter-peripheral connection without utilizing a physical pin. For example, the output of the analog comparator can be connected to RP176 and the PWM Fault input can be configured for RP176 as well. This configuration allows the analog comparator to trigger PWM Faults without the use of an actual physical pin on the device.

TABLE 11-11: REMAPPABLE SOURCES

Remap Index	Output Function
0	Vss
1	CMP1
2	CMP2
3	CMP3
4	CMP4
5	PWM4H
6	PTGO30
7	PTGO31
8-11	Reserved
12	REFO
13	SYNCO1
14	SYNCO2
15	PWM4L
16-20	RP16-RP20
21-31	Reserved
32-41	RP32-RP41
42	Reserved
43-58	RP43-RP58
59	Reserved
60-76	RP60-RP76
77-175	Reserved
176-181	RP176-RP181

TABLE 11-12: SELECTABLE INPUT SOURCES (MAPS INPUT TO FUNCTION)

Input Name[1]	Function Name	Register	Configuration Bits
External Interrupt 1	INT1	RPINR0	INT1R<7:0>
External Interrupt 2	INT2	RPINR1	INT2R<7:0>
Timer1 External Clock	T1CK	RPINR2	T1CKR<7:0>
Timer2 External Clock	T2CK	RPINR3	T2CKR<7:0>
Timer3 External Clock	T3CK	RPINR3	T3CKR<7:0>
Input Capture 1	IC1	RPINR7	IC1R<7:0>
Input Capture 2	IC2	RPINR7	IC2R<7:0>
Input Capture 3	IC3	RPINR8	IC3R<7:0>
Input Capture 4	IC4	RPINR8	IC4R<7:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<7:0>
PWM Fault 1	FLT1	RPINR12	FLT1R<7:0>
PWM Fault 2	FLT2	RPINR12	FLT2R<7:0>
PWM Fault 3	FLT3	RPINR13	FLT3R<7:0>
PWM Fault 4	FLT4	RPINR13	FLT4R<7:0>
UART1 Receive	U1RX	RPINR18	U1RXR<7:0>
UART1 Clear-to-Send	$\overline{\text{U1CTS}}$	RPINR18	U1CTSR<7:0>
UART2 Receive	U2RX	RPINR19	U2RXR<7:0>
UART2 Clear-to-Send	$\overline{\text{U2CTS}}$	RPINR19	U2CTSR<7:0>
SPI1 Data Input	SDI1	RPINR20	SDI1R<7:0>
SPI1 Clock Input	SCK1	RPINR20	SCK1R<7:0>
SPI1 Slave Select	$\overline{\text{SS1}}$	RPINR21	SS1R<7:0>
CAN1 Receive	C1RX	PRINR26	C1RXR<7:0>
CAN2 Receive	C2RX	PRINR26	C2RXR<7:0>
SPI3 Data Input	SDI3	RPINR29	SDI3R<7:0>
SPI3 Clock Input	SCK3	RPINR29	SCK3R<7:0>
SPI3 Slave Select	$\overline{\text{SS3}}$	RPINR30	SS3R<7:0>
SPI2 Data Input	SDI2	RPINR22	SDI2R<7:0>
SPI2 Clock Input	SCK2	RPINR22	SCK2R<7:0>
SPI2 Slave Select	$\overline{\text{SS2}}$	RPINR23	SS2R<7:0>
PWM Synchronous Input 1	SYNCI1	RPINR37	SYNCI1R<7:0>
PWM Synchronous Input 2	SYNCI2	RPINR38	SYNCI2R<7:0>
PWM Fault 5	FLT5	RPINR42	FLT5R<7:0>
PWM Fault 6	FLT6	RPINR42	FLT6R<7:0>
PWM Fault 7	FLT7	RPINR43	FLT7R<7:0>
PWM Fault 8	FLT8	RPINR43	FLT8R<7:0>
CLC Input A	CLCINA	RPINR45	CLCINA<7:0>
CLC Input B	CLCINB	RPINR46	CLCINB<7:0>

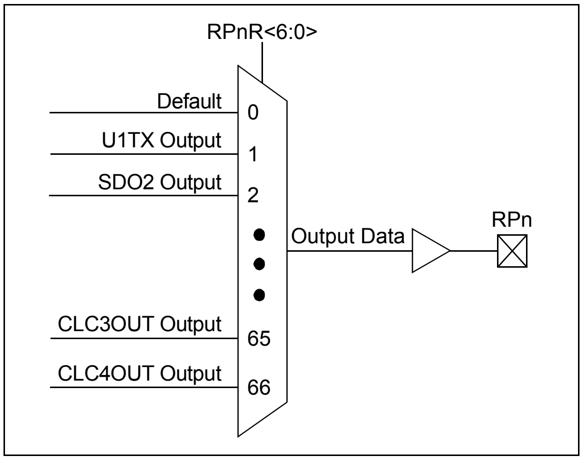
Note 1: Unless otherwise noted, all inputs use the Schmitt Trigger input buffers.

11.6.5 OUTPUT MAPPING

In contrast to inputs, the outputs of the Peripheral Pin Select options are mapped on the basis of the pin. In this case, a control register associated with a particular pin dictates the peripheral output to be mapped. The RPORx registers are used to control output mapping. Each register contains sets of 6-bit fields, with each set associated with one RPn pin (see Register 11-33 through Register 11-56). The value of the bit field corresponds to one of the peripherals and that peripheral's output is mapped to the pin (see Table 11-13 and Figure 11-3).

A null output is associated with the output register Reset value of '0'. This is done to ensure that remappable outputs remain disconnected from all output pins by default.

FIGURE 11-3: MULTIPLEXING REMAPPABLE OUTPUTS FOR RPn



11.6.5.1 Mapping Limitations

The control schema of the peripheral select pins is not limited to a small range of fixed peripheral configurations. There are no mutual or hardware-enforced lockouts between any of the peripheral mapping SFRs. Literally any combination of peripheral mappings, across any or all of the RPn pins, is possible. This includes both many-to-one and one-to-many mappings of peripheral inputs, and outputs to pins. While such mappings may be technically possible from a configuration point of view, they may not be supportable from an electrical point of view.

TABLE 11-13: OUTPUT SELECTION FOR REMAPPABLE PINS (RPn)

Function	RPnR<6:0>	Output Name
Default PORT	0000000	RPn tied to Default Pin
U1TX	0000001	RPn tied to UART1 Transmit
$\overline{U1RTS}$	0000010	RPn tied to UART1 Request-to-Send
U2TX	0000011	RPn tied to UART2 Transmit
$\overline{U2RTS}$	0000100	RPn tied to UART2 Request-to-Send
SDO1	0000101	RPn tied to SPI1 Data Output
SCK1	0000110	RPn tied to SPI1 Clock Output
$\overline{SS1}$	0000111	RPn tied to SPI1 Slave Select
SDO2	0001000	RPn tied to SPI2 Data Output
SCK2	0001001	RPn tied to SPI2 Clock Output
$\overline{SS2}$	0001010	RPn tied to SPI2 Slave Select
C1TX	0001110	RPn tied to CAN1 Transmit
C2TX	0001111	RPn tied to CAN2 Transmit
OC1	0010000	RPn tied to Output Compare 1 Output
OC2	0010001	RPn tied to Output Compare 2 Output
OC3	0010010	RPn tied to Output Compare 3 Output
OC4	0010011	RPn tied to Output Compare 4 Output
ACMP1	0011000	RPn tied to Analog Comparator 1 Output
ACMP2	0011001	RPn tied to Analog Comparator 2 Output
ACMP3	0011010	RPn tied to Analog Comparator 3 Output
SDO3	0011111	RPn tied to SPI3 Data Output
SCK3	0100000	RPn tied to SPI3 Clock Output
$\overline{SS3}$	0100001	RPn tied to SPI3 Slave Select
SYNCO1	0101101	RPn tied to PWM Primary Master Time Base Sync Output
SYNCO2	0101110	RPn tied to PWM Secondary Master Time Base Sync Output
REFCLKO	0110001	RPn tied to Reference Clock Output
ACMP4	0110010	RPn tied to Analog Comparator 4 Output
PWM4H	0110011	RPn tied to PWM Output Pins Associated with PWM Generator 4
PWM4L	0110100	RPn tied to PWM Output Pins Associated with PWM Generator 4
PWM5H	0110101	RPn tied to PWM Output Pins Associated with PWM Generator 5
PWM5L	0110110	RPn tied to PWM Output Pins Associated with PWM Generator 5
PWM6H	0111001	RPn tied to PWM Output Pins Associated with PWM Generator 6
PWM6L	0111010	RPn tied to PWM Output Pins Associated with PWM Generator 6
PWM7H	0111011	RPn tied to PWM Output Pins Associated with PWM Generator 7
PWM7L	0111100	RPn tied to PWM Output Pins Associated with PWM Generator 7
PWM8H	0111101	RPn tied to PWM Output Pins Associated with PWM Generator 8
PWM8L	0111110	RPn tied to PWM Output Pins Associated with PWM Generator 8
CLC1OUT	0111111	RPn tied to CLC1 Output
CLC2OUT	1000000	RPn tied to CLC2 Output
CLC3OUT[1]	1000001	RPn tied to CLC3 Output
CLC4OUT[1]	1000010	RPn tied to CLC4 Output

Note 1: PPS outputs are only available on dsPIC33EPXXXGS702 (28-pin) devices.

11.7 I/O Helpful Tips

1. In some cases, certain pins, as defined in Table 30-11 under "Injection Current", have internal protection diodes to V_{DD} and V_{SS}. The term, "Injection Current", is also referred to as "Clamp Current". On designated pins, with sufficient external current-limiting precautions by the user, I/O pin input voltages are allowed to be greater or less than the data sheet absolute maximum ratings, with respect to the V_{SS} and V_{DD} supplies. Note that when the user application forward biases either of the high or low-side internal input clamp diodes, that the resulting current being injected into the device, that is clamped internally by the V_{DD} and V_{SS} power rails, may affect the ADC accuracy by four to six counts.
2. I/O pins that are shared with any analog input pin (i.e., ANx) are always analog pins, by default, after any Reset. Consequently, configuring a pin as an analog input pin automatically disables the digital input pin buffer and any attempt to read the digital input level by reading PORTx or LATx will always return a '0', regardless of the digital logic level on the pin. To use a pin as a digital I/O pin on a shared ANx pin, the user application needs to configure the Analog Pin Configuration registers (i.e., ANSELx) in the I/O ports module by setting the appropriate bit that corresponds to that I/O port pin to a '0'.

Note:	Although it is not possible to use a digital input pin when its analog function is enabled, it is possible to use the digital I/O output function, TRISx = 0x0, while the analog function is also enabled. However, this is not recommended, particularly if the analog input is connected to an external analog voltage source, which would create signal contention between the analog signal and the output pin driver.

3. Most I/O pins have multiple functions. Referring to the device pin diagrams in this data sheet, the priorities of the functions allocated to any pins are indicated by reading the pin name from left-to-right. The left most function name takes precedence over any function to its right in the naming convention. For example: AN16/T2CK/T7CK/RC1; this indicates that AN16 is the highest priority in this example and will supersede all other functions to its right in the list. Those other functions to its right, even if enabled, would not work as long as any other function to its left was enabled. This rule applies to all of the functions listed for a given pin.
4. Each pin has an internal weak pull-up resistor and pull-down resistor that can be configured using the CNPUx and CNPDx registers, respectively. These resistors eliminate the need for external resistors in certain applications. The internal pull-up is up to ~(V_{DD} – 0.8), not V_{DD}. This value is still above the minimum V_{IH} of CMOS and TTL devices.
5. When driving LEDs directly, the I/O pin can source or sink more current than what is specified in the V_{OH}/I_{OH} and V_{OL}/I_{OL} DC characteristics specification. The respective I_{OH} and I_{OL} current rating only applies to maintaining the corresponding output at or above the V_{OH}, and at or below the V_{OL} levels. However, for LEDs, unlike digital inputs of an externally connected device, they are not governed by the same minimum V_{IH}/V_{IL} levels. An I/O pin output can safely sink or source any current less than that listed in the Absolute Maximum Ratings in **Section 30.0 "Electrical Characteristics"** of this data sheet. For example:

   V_{OH} = 2.4v @ I_{OH} = -8 mA and V_{DD} = 3.3V

   The maximum output current sourced by any 8 mA I/O pin = 12 mA.

   LED source current < 12 mA is technically permitted. Refer to the V_{OH}/I_{OH} graphs in **Section 31.0 "DC and AC Device Characteristics Graphs"** for additional information.

6. The Peripheral Pin Select (PPS) pin mapping rules are as follows:
   a) Only one “output” function can be active on a given pin at any time, regardless if it is a dedicated or remappable function (one pin, one output).
   b) It is possible to assign a “remappable output” function to multiple pins and externally short or tie them together for increased current drive.
   c) If any “dedicated output” function is enabled on a pin, it will take precedence over any remappable “output” function.
   d) If any “dedicated digital” (input or output) function is enabled on a pin, any number of “input” remappable functions can be mapped to the same pin.
   e) If any “dedicated analog” function(s) are enabled on a given pin, “digital input(s)” of any kind will all be disabled, although a single “digital output”, at the user’s cautionary discretion, can be enabled and active as long as there is no signal contention with an external analog input signal. For example, it is possible for the ADC to convert the digital output logic level, or to toggle a digital output on a comparator or ADC input, provided there is no external analog input, such as for a built-in self-test.
   f) Any number of “input” remappable functions can be mapped to the same pin(s) at the same time, including to any pin with a single output from either a dedicated or remappable “output”.
   g) The TRISx registers control *only* the digital I/O output buffer. Any other dedicated or remappable active “output” will automatically override the TRISx setting. The TRISx register *does not* control the digital logic “input” buffer. Remappable digital “inputs” do not automatically override TRISx settings, which means that the TRISx bit must be set to input for pins with only remappable input function(s) assigned.
   h) All analog pins are enabled by default after any Reset and the corresponding digital input buffer on the pin has been disabled. Only the Analog Pin Select x (ANSELx) registers control the digital input buffer, *not* the TRISx register. The user must disable the analog function on a pin using the Analog Pin Select x registers in order to use any “digital input(s)” on a corresponding pin, no exceptions.

11.8 I/O Ports Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

11.8.1 KEY RESOURCES

- **“I/O Ports”** (DS70000598) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

11.9 Peripheral Pin Select Registers

REGISTER 11-9: RPINR0: PERIPHERAL PIN SELECT INPUT REGISTER 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT1R7	INT1R6	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **INT1R<7:0>:** Assign External Interrupt 1 (INT1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 11-10: RPINR1: PERIPHERAL PIN SELECT INPUT REGISTER 1

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT2R7	INT2R6	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **INT2R<7:0>:** Assign External Interrupt 2 (INT2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-11: RPINR2: PERIPHERAL PIN SELECT INPUT REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T1CKR7	T1CKR6	T1CKR5	T1CKR4	T1CKR3	T1CKR2	T1CKR1	T1CKR0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **T1CKR<7:0>:** Assign Timer1 External Clock (T1CK) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 11-12: RPINR3: PERIPHERAL PIN SELECT INPUT REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T3CKR7	T3CKR6	T3CKR5	T3CKR4	T3CKR3	T3CKR2	T3CKR1	T3CKR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T2CKR7	T2CKR6	T2CKR5	T2CKR4	T2CKR3	T2CKR2	T2CKR1	T2CKR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **T3CKR<7:0>:** Assign Timer3 External Clock (T3CK) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **T2CKR<7:0>:** Assign Timer2 External Clock (T2CK) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-13: RPINR7: PERIPHERAL PIN SELECT INPUT REGISTER 7

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC2R7	IC2R6	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC1R7	IC1R6	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **IC2R<7:0>:** Assign Input Capture 2 (IC2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **IC1R<7:0>:** Assign Input Capture 1 (IC1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-14: RPINR8: PERIPHERAL PIN SELECT INPUT REGISTER 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC4R7	IC4R6	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC3R7	IC3R6	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **IC4R<7:0>:** Assign Input Capture 4 (IC4) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **IC3R<7:0>:** Assign Input Capture 3 (IC3) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-15: RPINR11: PERIPHERAL PIN SELECT INPUT REGISTER 11

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OCFAR7	OCFAR6	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **OCFAR<7:0>:** Assign Output Compare Fault A (OCFA) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-16: RPINR12: PERIPHERAL PIN SELECT INPUT REGISTER 12

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT2R7	FLT2R6	FLT2R5	FLT2R4	FLT2R3	FLT2R2	FLT2R1	FLT2R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT1R7	FLT1R6	FLT1R5	FLT1R4	FLT1R3	FLT1R2	FLT1R1	FLT1R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **FLT2R<7:0>:** Assign PWM Fault 2 (FLT2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **FLT1R<7:0>:** Assign PWM Fault 1 (FLT1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-17: RPINR13: PERIPHERAL PIN SELECT INPUT REGISTER 13

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT4R7	FLT4R6	FLT4R5	FLT4R4	FLT4R3	FLT4R2	FLT4R1	FLT4R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT3R7	FLT3R6	FLT3R5	FLT3R4	FLT3R3	FLT3R2	FLT3R1	FLT3R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **FLT4R<7:0>:** Assign PWM Fault 4 (FLT4) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **FLT3R<7:0>:** Assign PWM Fault 3 (FLT3) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-18: RPINR18: PERIPHERAL PIN SELECT INPUT REGISTER 18

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U1CTSR7	U1CTSR6	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U1RXR7	U1RXR6	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **U1CTSR<7:0>:** Assign UART1 Clear-to-Send ($\overline{\text{U1CTS}}$) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **U1RXR<7:0>:** Assign UART1 Receive (U1RX) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-19: RPINR19: PERIPHERAL PIN SELECT INPUT REGISTER 19

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2CTSR7	U2CTSR6	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2RXR7	U2RXR6	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **U2CTSR<7:0>:** Assign UART2 Clear-to-Send ($\overline{\text{U2CTS}}$) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **U2RXR<7:0>:** Assign UART2 Receive (U2RX) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-20: RPINR20: PERIPHERAL PIN SELECT INPUT REGISTER 20

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK1INR7	SCK1INR6	SCK1INR5	SCK1INR4	SCK1INR3	SCK1INR2	SCK1INR1	SCK1INR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI1R7	SDI1R6	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **SCK1INR<7:0>:** Assign SPI1 Clock Input (SCK1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **SDI1R<7:0>:** Assign SPI1 Data Input (SDI1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-21: RPINR21: PERIPHERAL PIN SELECT INPUT REGISTER 21

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS1R7	SS1R6	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **SS1R<7:0>:** Assign SPI1 Slave Select ($\overline{SS1}$) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-22: RPINR22: PERIPHERAL PIN SELECT INPUT REGISTER 22

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK2INR7	SCK2INR6	SCK2INR5	SCK2INR4	SCK2INR3	SCK2INR2	SCK2INR1	SCK2INR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI2R7	SDI2R6	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **SCK2INR<7:0>:** Assign SPI2 Clock Input (SCK2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **SDI2R<7:0>:** Assign SPI2 Data Input (SDI2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-23: RPINR23: PERIPHERAL PIN SELECT INPUT REGISTER 23

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS2R7	SS2R6	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **SS2R<7:0>:** Assign SPI2 Slave Select ($\overline{SS2}$) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-24: RPINR26: PERIPHERAL PIN SELECT INPUT REGISTER 26

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2RXR7	C2RXR6	C2RXR5	C2RXR4	C2RXR3	C2RXR2	C2RXR1	C2RXR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C1RXR7	C1RXR6	C1RXR5	C1RXR4	C1RXR3	C1RXR2	C1RXR1	C1RXR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **C2RXR<7:0>:** Assign CAN2 Receive (C2RX) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **C1RXR<7:0>:** Assign CAN1 Receive (C1RX) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-25: RPINR29: PERIPHERAL PIN SELECT INPUT REGISTER 29

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK3R7	SCK3R6	SCK3R5	SCK3R4	SCK3R3	SCK3R2	SCK3R1	SCK3R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI3R7	SDI3R6	SDI3R5	SDI3R4	SDI3R3	SDI3R2	SDI3R1	SDI3R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **SCK3R<7:0>:** Assign SPI3 Clock Input (SCK3) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **SDI3R<7:0>:** Assign SPI3 Data Input (SDI3) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-26: RPINR30: PERIPHERAL PIN SELECT INPUT REGISTER 30

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS3R7	SS3R6	SS3R5	SS3R4	SS3R3	SS3R2	SS3R1	SS3R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **SS3R<7:0>:** Assign SPI3 Slave Select ($\overline{SS3}$) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-27: RPINR37: PERIPHERAL PIN SELECT INPUT REGISTER 37

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCI1R7	SYNCI1R6	SYNCI1R5	SYNCI1R4	SYNCI1R3	SYNCI1R2	SYNCI1R1	SYNCI1R0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **SYNCI1R<7:0>:** Assign PWM Synchronization Input 1 (SYNCI1) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 11-28: RPINR38: PERIPHERAL PIN SELECT INPUT REGISTER 38

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCI2R7	SYNCI2R6	SYNCI2R5	SYNCI2R4	SYNCI2R3	SYNCI2R2	SYNCI2R1	SYNCI2R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **SYNCI2R<7:0>:** Assign PWM Synchronization Input 2 (SYNCI2) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-29: RPINR42: PERIPHERAL PIN SELECT INPUT REGISTER 42

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT6R7	FLT6R6	FLT6R5	FLT6R4	FLT6R3	FLT6R2	FLT6R1	FLT6R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT5R7	FLT5R6	FLT5R5	FLT5R4	FLT5R3	FLT5R2	FLT5R1	FLT5R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **FLT6R<7:0>:** Assign PWM Fault 6 (FLT6) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **FLT5R<7:0>:** Assign PWM Fault 5 (FLT5) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-30: RPINR43: PERIPHERAL PIN SELECT INPUT REGISTER 43

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT8R7	FLT8R6	FLT8R5	FLT8R4	FLT8R3	FLT8R2	FLT8R1	FLT8R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT7R7	FLT7R6	FLT7R5	FLT7R4	FLT7R3	FLT7R2	FLT7R1	FLT7R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **FLT8R<7:0>:** Assign PWM Fault 8 (FLT8) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **FLT7R<7:0>:** Assign PWM Fault 7 (FLT7) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-31: RPINR45: PERIPHERAL PIN SELECT INPUT REGISTER 45

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLCINAR7	CLCINAR6	CLCINAR5	CLCINAR4	CLCINAR3	CLCINAR2	CLCINAR1	CLCINAR0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **CLCINAR<7:0>:** Assign CLC Input A (CLCINA) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 11-32: RPINR46: PERIPHERAL PIN SELECT INPUT REGISTER 46

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLCINBR7	CLCINBR6	CLCINBR5	CLCINBR4	CLCINBR3	CLCINBR2	CLCINBR1	CLCINBR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **CLCINBR<7:0>:** Assign CLC Input B (CLCINB) to the Corresponding RPn Pin bits
See Table 11-11 which contains a list of remappable inputs for the index value.

REGISTER 11-33: RPOR0: PERIPHERAL PIN SELECT OUTPUT REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP17R6	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP16R6	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP17R<6:0>:** Peripheral Output Function is Assigned to RP17 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP16R<6:0>:** Peripheral Output Function is Assigned to RP16 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-34: RPOR1: PERIPHERAL PIN SELECT OUTPUT REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP19R6	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP18R6	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP19R<6:0>:** Peripheral Output Function is Assigned to RP19 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP18R<6:0>:** Peripheral Output Function is Assigned to RP18 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-35: RPOR2: PERIPHERAL PIN SELECT OUTPUT REGISTER 2

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP32R6	RP32R5	RP32R4	RP32R3	RP32R2	RP32R1	RP32R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP20R6	RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP32R<6:0>:** Peripheral Output Function is Assigned to RP32 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP20R<6:0>:** Peripheral Output Function is Assigned to RP20 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-36: RPOR3: PERIPHERAL PIN SELECT OUTPUT REGISTER 3

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP34R6	RP34R5	RP34R4	RP34R3	RP34R2	RP34R1	RP34R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP33R6	RP33R5	RP33R4	RP33R3	RP33R2	RP33R1	RP33R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP34R<6:0>:** Peripheral Output Function is Assigned to RP34 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP33R<6:0>:** Peripheral Output Function is Assigned to RP33 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-37: RPOR4: PERIPHERAL PIN SELECT OUTPUT REGISTER 4

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP36R6	RP36R5	RP36R4	RP36R3	RP36R2	RP36R1	RP36R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP35R6	RP35R5	RP35R4	RP35R3	RP35R2	RP35R1	RP35R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP36R<6:0>:** Peripheral Output Function is Assigned to RP36 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP35R<6:0>:** Peripheral Output Function is Assigned to RP35 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-38: RPOR5: PERIPHERAL PIN SELECT OUTPUT REGISTER 5

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP38R6	RP38R5	RP38R4	RP38R3	RP38R2	RP38R1	RP38R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP37R6	RP37R5	RP37R4	RP37R3	RP37R2	RP37R1	RP37R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP38R<6:0>:** Peripheral Output Function is Assigned to RP38 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP37R<6:0>:** Peripheral Output Function is Assigned to RP37 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-39: RPOR6: PERIPHERAL PIN SELECT OUTPUT REGISTER 6

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP40R6	RP40R5	RP40R4	RP40R3	RP40R2	RP40R1	RP40R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP39R6	RP39R5	RP39R4	RP39R3	RP39R2	RP39R1	RP39R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP40R<6:0>:** Peripheral Output Function is Assigned to RP40 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP39R<6:0>:** Peripheral Output Function is Assigned to RP39 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-40: RPOR7: PERIPHERAL PIN SELECT OUTPUT REGISTER 7

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP43R6	RP43R5	RP43R4	RP43R3	RP43R2	RP43R1	RP43R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP41R6	RP41R5	RP41R4	RP41R3	RP41R2	RP41R1	RP41R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP43R<6:0>:** Peripheral Output Function is Assigned to RP43 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP41R<6:0>:** Peripheral Output Function is Assigned to RP41 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-41: RPOR8: PERIPHERAL PIN SELECT OUTPUT REGISTER 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP45R6	RP45R5	RP45R4	RP45R3	RP45R2	RP45R1	RP45R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP44R6	RP44R5	RP44R4	RP44R3	RP44R2	RP44R1	RP44R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP45R<6:0>:** Peripheral Output Function is Assigned to RP45 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP44R<6:0>:** Peripheral Output Function is Assigned to RP44 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-42: RPOR9: PERIPHERAL PIN SELECT OUTPUT REGISTER 9

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP47R6	RP47R5	RP47R4	RP47R3	RP47R2	RP47R1	RP47R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP46R6	RP46R5	RP46R4	RP46R3	RP46R2	RP46R1	RP46R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP47R<6:0>:** Peripheral Output Function is Assigned to RP47 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP46R<6:0>:** Peripheral Output Function is Assigned to RP46 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-43: RPOR10: PERIPHERAL PIN SELECT OUTPUT REGISTER 10

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP49R6	RP49R5	RP49R4	RP49R3	RP49R2	RP49R1	RP49R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP48R6	RP48R5	RP48R4	RP48R3	RP48R2	RP48R1	RP48R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP49R<6:0>:** Peripheral Output Function is Assigned to RP49 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP48R<6:0>:** Peripheral Output Function is Assigned to RP48 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-44: RPOR11: PERIPHERAL PIN SELECT OUTPUT REGISTER 11

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP51R6	RP51R5	RP51R4	RP51R3	RP51R2	RP51R1	RP51R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP50R6	RP50R5	RP50R4	RP50R3	RP50R2	RP50R1	RP50R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP51R<6:0>:** Peripheral Output Function is Assigned to RP51 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP50R<6:0>:** Peripheral Output Function is Assigned to RP50 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-45: RPOR12: PERIPHERAL PIN SELECT OUTPUT REGISTER 12

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP53R6	RP53R5	RP53R4	RP53R3	RP53R2	RP53R1	RP53R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP52R6	RP52R5	RP52R4	RP52R3	RP52R2	RP52R1	RP52R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP53R<6:0>:** Peripheral Output Function is Assigned to RP53 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP52R<6:0>:** Peripheral Output Function is Assigned to RP52 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-46: RPOR13: PERIPHERAL PIN SELECT OUTPUT REGISTER 13

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP55R6	RP55R5	RP55R4	RP55R3	RP55R2	RP55R1	RP55R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP54R6	RP54R5	RP54R4	RP54R3	RP54R2	RP54R1	RP54R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP55R<6:0>:** Peripheral Output Function is Assigned to RP55 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP54R<6:0>:** Peripheral Output Function is Assigned to RP54 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-47: RPOR14: PERIPHERAL PIN SELECT OUTPUT REGISTER 14

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP57R6	RP57R5	RP57R4	RP57R3	RP57R2	RP57R1	RP57R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP56R6	RP56R5	RP56R4	RP56R3	RP56R2	RP56R1	RP56R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP57R<6:0>:** Peripheral Output Function is Assigned to RP57 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP56R<6:0>:** Peripheral Output Function is Assigned to RP56 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-48: RPOR15: PERIPHERAL PIN SELECT OUTPUT REGISTER 15

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP60R6	RP60R5	RP60R4	RP60R3	RP60R2	RP60R1	RP60R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP58R6	RP58R5	RP58R4	RP58R3	RP58R2	RP58R1	RP58R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP60R<6:0>:** Peripheral Output Function is Assigned to RP60 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP58R<6:0>:** Peripheral Output Function is Assigned to RP58 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-49: RPOR16: PERIPHERAL PIN SELECT OUTPUT REGISTER 16

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP62R6	RP62R5	RP62R4	RP62R3	RP62R2	RP62R1	RP62R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP61R6	RP61R5	RP61R4	RP61R3	RP61R2	RP61R1	RP61R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP62R<6:0>:** Peripheral Output Function is Assigned to RP62 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP61R<6:0>:** Peripheral Output Function is Assigned to RP61 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-50: RPOR17: PERIPHERAL PIN SELECT OUTPUT REGISTER 17

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP64R6	RP64R5	RP64R4	RP64R3	RP64R2	RP64R1	RP64R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP63R6	RP63R5	RP63R4	RP63R3	RP63R2	RP63R1	RP63R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP64R<6:0>:** Peripheral Output Function is Assigned to RP64 Output Pin bits
(see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP63R<6:0>:** Peripheral Output Function is Assigned to RP63 Output Pin bits
(see Table 11-13 for peripheral function numbers)

REGISTER 11-51: RPOR18: PERIPHERAL PIN SELECT OUTPUT REGISTER 18

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP66R6	RP66R5	RP66R4	RP66R3	RP66R2	RP66R1	RP66R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP65R6	RP65R5	RP65R4	RP65R3	RP65R2	RP65R1	RP65R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP66R<6:0>:** Peripheral Output Function is Assigned to RP66 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP65R<6:0>:** Peripheral Output Function is Assigned to RP65 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-52: RPOR19: PERIPHERAL PIN SELECT OUTPUT REGISTER 19

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP68R6	RP68R5	RP68R4	RP68R3	RP68R2	RP68R1	RP68R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP67R6	RP67R5	RP67R4	RP67R3	RP67R2	RP67R1	RP67R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP68R<6:0>:** Peripheral Output Function is Assigned to RP68 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP67R<6:0>:** Peripheral Output Function is Assigned to RP67 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-53: RPOR20: PERIPHERAL PIN SELECT OUTPUT REGISTER 20

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP70R6	RP70R5	RP70R4	RP70R3	RP70R2	RP70R1	RP70R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP69R6	RP69R5	RP69R4	RP69R3	RP69R2	RP69R1	RP69R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP70R<6:0>:** Peripheral Output Function is Assigned to RP70 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP69R<6:0>:** Peripheral Output Function is Assigned to RP69 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-54: RPOR21: PERIPHERAL PIN SELECT OUTPUT REGISTER 21

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP72R6	RP72R5	RP72R4	RP72R3	RP72R2	RP72R1	RP72R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP71R6	RP71R5	RP71R4	RP71R3	RP71R2	RP71R1	RP71R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP72R<6:0>:** Peripheral Output Function is Assigned to RP72 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP71R<6:0>:** Peripheral Output Function is Assigned to RP71 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-55: RPOR22: PERIPHERAL PIN SELECT OUTPUT REGISTER 22

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP74R6	RP74R5	RP74R4	RP74R3	RP74R2	RP74R1	RP74R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP73R6	RP73R5	RP73R4	RP73R3	RP73R2	RP73R1	RP73R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP74R<6:0>:** Peripheral Output Function is Assigned to RP74 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP73R<6:0>:** Peripheral Output Function is Assigned to RP73 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-56: RPOR23: PERIPHERAL PIN SELECT OUTPUT REGISTER 23

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP76R6	RP76R5	RP76R4	RP76R3	RP76R2	RP76R1	RP76R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP75R6	RP75R5	RP75R4	RP75R3	RP75R2	RP75R1	RP75R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP76R<6:0>:** Peripheral Output Function is Assigned to RP76 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP75R<6:0>:** Peripheral Output Function is Assigned to RP75 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-57: RPOR24: PERIPHERAL PIN SELECT OUTPUT REGISTER 24

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP177R6	RP177R5	RP177R4	RP177R3	RP177R2	RP177R1	RP177R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP176R6	RP176R5	RP176R4	RP176R3	RP176R2	RP176R1	RP176R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP177R<6:0>:** Peripheral Output Function is Assigned to RP177 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP176R<6:0>:** Peripheral Output Function is Assigned to RP176 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-58: RPOR25: PERIPHERAL PIN SELECT OUTPUT REGISTER 25

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP179R6	RP179R5	RP179R4	RP179R3	RP179R2	RP179R1	RP179R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP178R6	RP178R5	RP178R4	RP178R3	RP178R2	RP178R1	RP178R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-8	**RP179R<6:0>:** Peripheral Output Function is Assigned to RP179 Output Pin bits (see Table 11-13 for peripheral function numbers)
bit 7	**Unimplemented:** Read as ‘0’
bit 6-0	**RP178R<6:0>:** Peripheral Output Function is Assigned to RP178 Output Pin bits (see Table 11-13 for peripheral function numbers)

REGISTER 11-59: RPOR26: PERIPHERAL PIN SELECT OUTPUT REGISTER 26

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP181R6	RP181R5	RP181R4	RP181R3	RP181R2	RP181R1	RP181R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP180R6	RP180R5	RP180R4	RP180R3	RP180R2	RP180R1	RP180R0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14-8 **RP181R<6:0>:** Peripheral Output Function is Assigned to RP181 Output Pin bits (see Table 11-13 for peripheral function numbers)

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **RP180R<6:0>:** Peripheral Output Function is Assigned to RP180 Output Pin bits (see Table 11-13 for peripheral function numbers)

NOTES:

12.0 TIMER1

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Timers"** (DS70362) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Timer1 module is a 16-bit timer that can operate as a free-running interval timer/counter.

The Timer1 module has the following unique features over other timers:

- Can be Operated in Asynchronous Counter mode from an External Clock Source
- The External Clock Input (T1CK) can Optionally be Synchronized to the Internal Device Clock and the Clock Synchronization is Performed after the prescaler

A block diagram of Timer1 is shown in Figure 12-1.

The Timer1 module can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode
- Asynchronous Counter mode

In Timer and Gated Timer modes, the input clock is derived from the internal instruction cycle clock (FCY). In Synchronous and Asynchronous Counter modes, the input clock is derived from the external clock input at the T1CK pin.

The Timer modes are determined by the following bits:

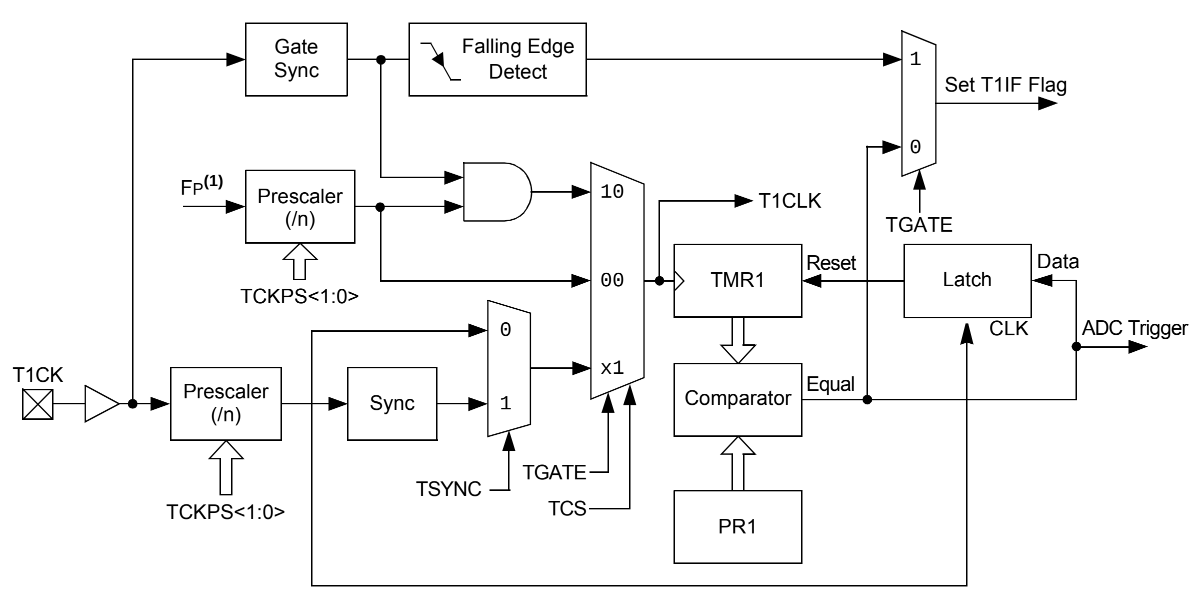
- Timer1 Clock Source Select bit (TCS): T1CON<1>
- Timer1 External Clock Input Synchronization Select bit (TSYNC): T1CON<2>
- Timer1 Gated Time Accumulation Enable bit (TGATE): T1CON<6>

Timer control bit settings for different operating modes are provided in Table 12-1.

TABLE 12-1: TIMER1 MODE SETTINGS

Mode	TCS	TGATE	TSYNC
Timer	0	0	x
Gated Timer	0	1	x
Synchronous Counter	1	x	1
Asynchronous Counter	1	x	0

FIGURE 12-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



Note 1: FP is the peripheral clock.

12.1 Timer1 Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

12.1.1 KEY RESOURCES

- **“Timers”** (DS70362) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

12.2 Timer1 Control Register

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON(1)	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	—	TSYNC(1)	TCS(1)	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **TON:** Timer1 On bit(1)
1 = Starts 16-bit Timer1
0 = Stops 16-bit Timer1

bit 14 **Unimplemented:** Read as '0'

bit 13 **TSIDL:** Timer1 Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **TGATE:** Timer1 Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3 **Unimplemented:** Read as '0'

bit 2 **TSYNC:** Timer1 External Clock Input Synchronization Select bit(1)
When TCS = 1:
1 = Synchronizes external clock input
0 = Does not synchronize external clock input
When TCS = 0:
This bit is ignored.

bit 1 **TCS:** Timer1 Clock Source Select bit(1)
1 = External clock is from pin, T1CK (on the rising edge)
0 = Internal clock (FP)

bit 0 **Unimplemented:** Read as '0'

Note 1: When Timer1 is enabled in External Synchronous Counter mode (TCS = 1, TSYNC = 1, TON = 1), any attempts by user software to write to the TMR1 register are ignored.

NOTES:

13.0 TIMER2/3 AND TIMER4/5

Note 1:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Timers"** (DS70362) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
2:	Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Timer2/3 and Timer4/5 modules are 32-bit timers, which can also be configured as four independent 16-bit timers with selectable operating modes.

As 32-bit timers, Timer2/3 and Timer4/5 operate in three modes:

- Two Independent 16-Bit Timers (e.g., Timer2 and Timer3) with all 16-Bit Operating modes (except Asynchronous Counter mode)
- Single 32-Bit Timer
- Single 32-Bit Synchronous Counter

They also support these features:

- Timer Gate Operation
- Selectable Prescaler Settings
- Timer Operation during Idle and Sleep modes
- Interrupt on a 32-Bit Period Register Match
- Time Base for Input Capture and Output Compare modules (Timer2 and Timer3 only)

Individually, all four of the 16-bit timers can function as synchronous timers or counters. They also offer the features listed previously, except for the event trigger; this is implemented only with Timer2/3. The operating modes and enabled features are determined by setting the appropriate bit(s) in the T2CON, T3CON, T4CON and T5CON registers. T2CON and T4CON are shown in generic form in Register 13-1. T3CON and T5CON are shown in Register 13-2.

For 32-bit timer/counter operation, Timer2 and Timer4 are the least significant word (lsw); Timer3 and Timer5 are the most significant word (msw) of the 32-bit timers.

Note:	For 32-bit operation, T3CON and T5CON control bits are ignored. Only T2CON and T4CON control bits are used for setup and control. Timer2 and Timer4 clock and gate inputs are utilized for the 32-bit timer modules, but an interrupt is generated with the Timer3 and Timer5 interrupt flags.

A block diagram for an example 32-bit timer pair (Timer2/3 and Timer4/5) is shown in Figure 13-2.

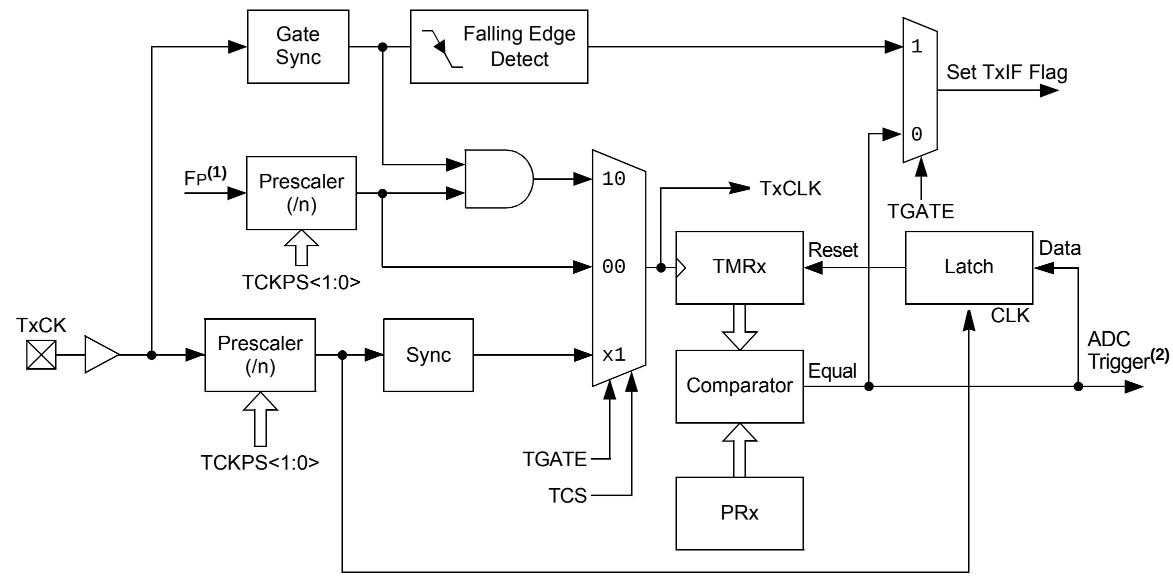
13.1 Timer Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

13.1.1 KEY RESOURCES

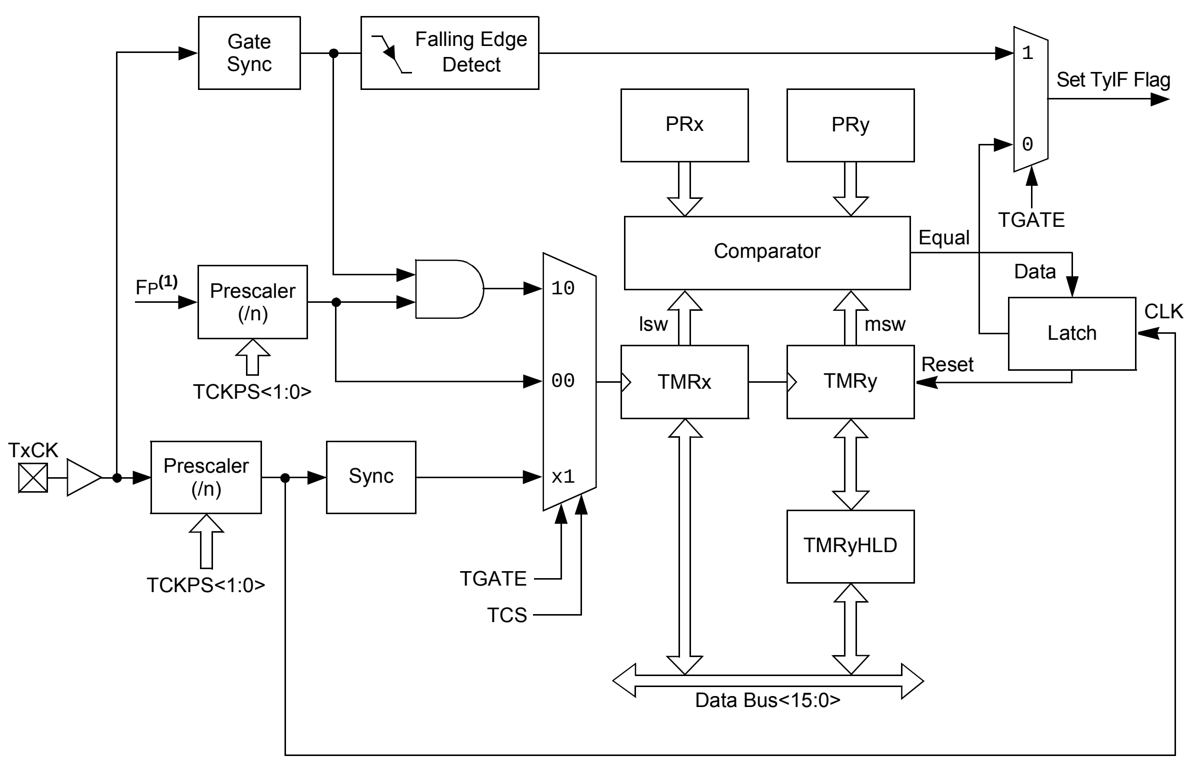
- **"Timers"** (DS70362) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

FIGURE 13-1: TIMERx BLOCK DIAGRAM (x = 2 THROUGH 5)



Note 1: FP is the peripheral clock.
2: The ADC trigger is only available on TMR2.

FIGURE 13-2: TYPE B/TYPE C TIMER PAIR BLOCK DIAGRAM (32-BIT TIMER)



Note 1: Timerx is a Type B timer (x = 2 and 4).
2: Timery is a Type C timer (y = 3 and 5).

13.2 Timer2/3 and Timer4/5 Control Registers

REGISTER 13-1: TxCON: (TIMER2 AND TIMER4) CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	T32	—	TCS[1]	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **TON:** Timerx On bit
When T32 = 1:
1 = Starts 32-bit Timerx/y
0 = Stops 32-bit Timerx/y
When T32 = 0:
1 = Starts 16-bit Timerx
0 = Stops 16-bit Timerx

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **TSIDL:** Timerx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-7 **Unimplemented:** Read as ‘0’

bit 6 **TGATE:** Timerx Gated Time Accumulation Enable bit
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timerx Input Clock Prescale Select bits
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3 **T32:** 32-Bit Timer Mode Select bit
1 = Timerx and Timery form a single 32-bit timer
0 = Timerx and Timery act as two 16-bit timers

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **TCS:** Timerx Clock Source Select bit[1]
1 = External clock is from pin, TxCK (on the rising edge)
0 = Internal clock (FP)

bit 0 **Unimplemented:** Read as ‘0’

Note 1: The TxCK pin is not available on all devices. Refer to the **“Pin Diagrams”** section for the available pins.

REGISTER 13-2: TyCON: (TIMER3 AND TIMER5) CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON[1]	—	TSIDL[2]	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE[1]	TCKPS1[1]	TCKPS0[1]	—	—	TCS[1,3]	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **TON:** Timery On bit[1]
1 = Starts 16-bit Timery
0 = Stops 16-bit Timery

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **TSIDL:** Timery Stop in Idle Mode bit[2]
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-7 **Unimplemented:** Read as ‘0’

bit 6 **TGATE:** Timery Gated Time Accumulation Enable bit[1]
When TCS = 1:
This bit is ignored.
When TCS = 0:
1 = Gated time accumulation is enabled
0 = Gated time accumulation is disabled

bit 5-4 **TCKPS<1:0>:** Timery Input Clock Prescale Select bits[1]
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1

bit 3-2 **Unimplemented:** Read as ‘0’

bit 1 **TCS:** Timery Clock Source Select bit[1,3]
1 = External clock is from pin, TyCK (on the rising edge)
0 = Internal clock (FP)

bit 0 **Unimplemented:** Read as ‘0’

Note 1: When 32-bit operation is enabled (TxCON<3> = 1), these bits have no effect on Timery operation; all timer functions are set through TxCON.

2: When 32-bit timer operation is enabled (T32 = 1) in the Timerx Control register (TxCON<3>), the TSIDL bit must be cleared to operate the 32-bit timer in Idle mode.

3: The TyCK pin is not available on all devices. See the **“Pin Diagrams”** section for the available pins.

14.0 INPUT CAPTURE

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Input Capture with Dedicated Timer"** (DS70000352) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The input capture module is useful in applications requiring frequency (period) and pulse measurements. The dsPIC33EPXXXGS70X/80X devices support four input capture channels.

Key features of the input capture module include:

- Hardware-Configurable for 32-Bit Operation in all modes by Cascading Two Adjacent modules
- Synchronous and Trigger modes of Output Compare Operation, with up to 21 User-Selectable Trigger/Sync Sources available
- A 4-Level FIFO Buffer for Capturing and Holding Timer Values for Several Events
- Configurable Interrupt Generation
- Up to Six Clock Sources available for each module, Driving a Separate Internal 16-Bit Counter

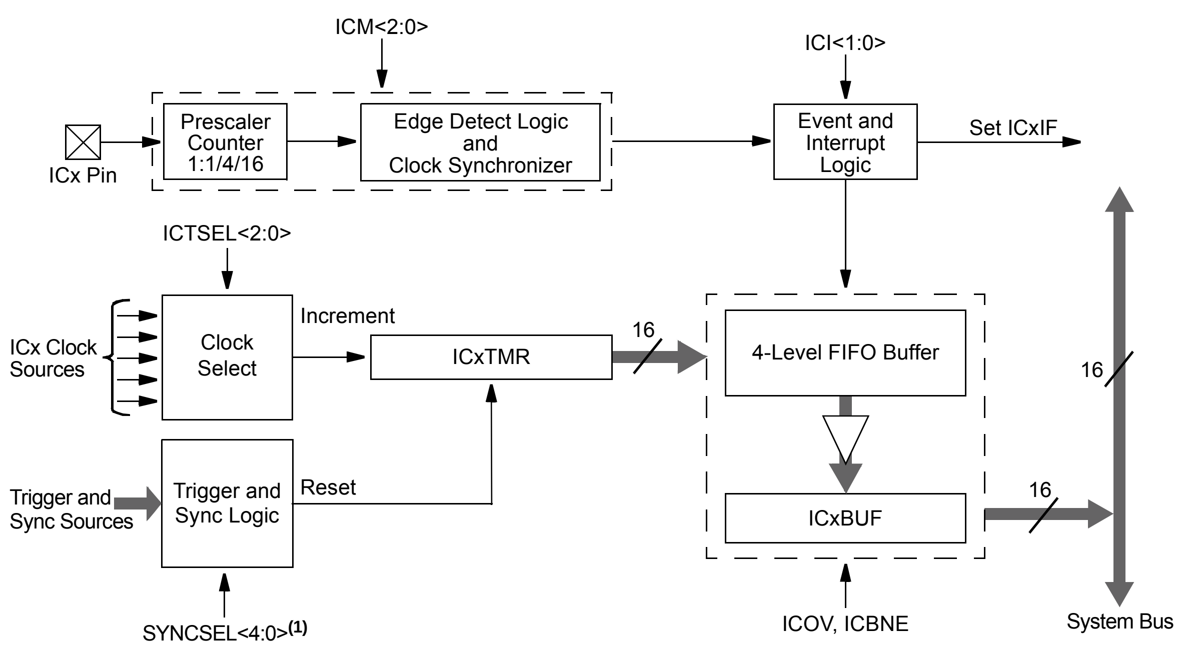
14.1 Input Capture Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

14.1.1 KEY RESOURCES

- **"Input Capture with Dedicated Timer"** (DS70000352) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

FIGURE 14-1: INPUT CAPTURE x MODULE BLOCK DIAGRAM



Note 1: The trigger/sync source is enabled by default and is set to Timer3 as a source. This timer must be enabled for proper ICx module operation or the trigger/sync source must be changed to another source option.

14.2 Input Capture Registers

REGISTER 14-1: ICxCON1: INPUT CAPTURE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R-0, HC, HS	R-0, HC, HS	R/W-0	R/W-0	R/W-0
—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	HS = Hardware Settable bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13 **ICSIDL:** Input Capture x Stop in Idle Control bit
1 = Input capture will halt in CPU Idle mode
0 = Input capture will continue to operate in CPU Idle mode

bit 12-10 **ICTSEL<2:0>:** Input Capture x Timer Select bits
111 = Peripheral clock (FP) is the clock source of the ICx
110 = Reserved
101 = Reserved
100 = T1CLK is the clock source of the ICx (only the synchronous clock is supported)
011 = T5CLK is the clock source of the ICx
010 = T4CLK is the clock source of the ICx
001 = T2CLK is the clock source of the ICx
000 = T3CLK is the clock source of the ICx

bit 9-7 **Unimplemented:** Read as ‘0’

bit 6-5 **ICI<1:0>:** Number of Captures per Interrupt Select bits (this field is not used if ICM<2:0> = 001 or 111)
11 = Interrupt on every fourth capture event
10 = Interrupt on every third capture event
01 = Interrupt on every second capture event
00 = Interrupt on every capture event

bit 4 **ICOV:** Input Capture x Overflow Status Flag bit (read-only)
1 = Input capture buffer overflow has occurred
0 = No input capture buffer overflow has occurred

bit 3 **ICBNE:** Input Capture x Buffer Not Empty Status bit (read-only)
1 = Input capture buffer is not empty, at least one more capture value can be read
0 = Input capture buffer is empty

bit 2-0 **ICM<2:0>:** Input Capture x Mode Select bits
111 = Input Capture x functions as an interrupt pin only in CPU Sleep and Idle modes (rising edge detect only, all other control bits are not applicable)
110 = Unused (module is disabled)
101 = Capture mode, every 16th rising edge (Prescaler Capture mode)
100 = Capture mode, every 4th rising edge (Prescaler Capture mode)
011 = Capture mode, every rising edge (Simple Capture mode)
010 = Capture mode, every falling edge (Simple Capture mode)
001 = Capture mode, every rising and falling edge (Edge Detect mode, ICI<1:0>, is not used in this mode)
000 = Input Capture x is turned off

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	IC32
bit 15							bit 8

R/W-0	R/W-0, HS	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
ICTRIG(2)	TRIGSTAT(3)	—	SYNCSEL4(4)	SYNCSEL3(4)	SYNCSEL2(4)	SYNCSEL1(4)	SYNCSEL0(4)
bit 7							bit 0

Legend:	HS = Hardware Settable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-9 **Unimplemented:** Read as ‘0’

bit 8 **IC32:** Input Capture x 32-Bit Timer Mode Select bit (Cascade mode)
1 = Odd ICx and even ICx form a single 32-bit input capture module(1)
0 = Cascade module operation is disabled

bit 7 **ICTRIG:** Input Capture x Trigger Operation Select bit(2)
1 = Input source is used to trigger the input capture timer (Trigger mode)
0 = Input source is used to synchronize the input capture timer to a timer of another module (Synchronization mode)

bit 6 **TRIGSTAT:** Timer Trigger Status bit(3)
1 = ICxTMR has been triggered and is running
0 = ICxTMR has not been triggered and is being held clear

bit 5 **Unimplemented:** Read as ‘0’

Note 1: The IC32 bit in both the odd and even ICx must be set to enable Cascade mode.
2: The input source is selected by the SYNCSEL<4:0> bits of the ICxCON2 register.
3: This bit is set by the selected input source (selected by SYNCSEL<4:0> bits); it can be read, set and cleared in software.
4: Do not use the ICx module as its own sync or trigger source.
5: This option should only be selected as a trigger source and not as a synchronization source.

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2 (CONTINUED)

bit 4-0 **SYNCSEL<4:0>:** Input Source Select for Synchronization and Trigger Operation bits[4]

11111 = No sync or trigger source for ICx
11110 = Reserved
11101 = Reserved
11100 = Reserved
11011 = CMP4 module synchronizes or triggers ICx[5]
11010 = CMP3 module synchronizes or triggers ICx[5]
11001 = CMP2 module synchronizes or triggers ICx[5]
11000 = CMP1 module synchronizes or triggers ICx[5]
10111 = Reserved
10110 = Reserved
10101 = Reserved
10100 = Reserved
10011 = IC4 module interrupt synchronizes or triggers ICx
10010 = IC3 module interrupt synchronizes or triggers ICx
10001 = IC2 module interrupt synchronizes or triggers ICx
10000 = IC1 module interrupt synchronizes or triggers ICx
01111 = Timer5 synchronizes or triggers ICx
01110 = Timer4 synchronizes or triggers ICx
01101 = Timer3 synchronizes or triggers ICx **(default)**
01100 = Timer2 synchronizes or triggers ICx
01011 = Timer1 synchronizes or triggers ICx
01010 = Reserved
01001 = Reserved
01000 = IC4 module synchronizes or triggers ICx
00111 = IC3 module synchronizes or triggers ICx
00110 = IC2 module synchronizes or triggers ICx
00101 = IC1 module synchronizes or triggers ICx
00100 = OC4 module synchronizes or triggers ICx
00011 = OC3 module synchronizes or triggers ICx
00010 = OC2 module synchronizes or triggers ICx
00001 = OC1 module synchronizes or triggers ICx
00000 = No sync or trigger source for ICx

Note 1: The IC32 bit in both the odd and even ICx must be set to enable Cascade mode.
2: The input source is selected by the SYNCSEL<4:0> bits of the ICxCON2 register.
3: This bit is set by the selected input source (selected by SYNCSEL<4:0> bits); it can be read, set and cleared in software.
4: Do not use the ICx module as its own sync or trigger source.
5: This option should only be selected as a trigger source and not as a synchronization source.

15.0 OUTPUT COMPARE

Note 1:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Output Compare with Dedicated Timer"** (DS70005159) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
2:	Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The output compare module can select one of six available clock sources for its time base. The module compares the value of the timer with the value of one or two Compare registers, depending on the operating mode selected. The state of the output pin changes when the timer value matches the Compare register value. The output compare module generates either a single output pulse, or a sequence of output pulses, by changing the state of the output pin on the compare match events. The output compare module can also generate interrupts on compare match events.

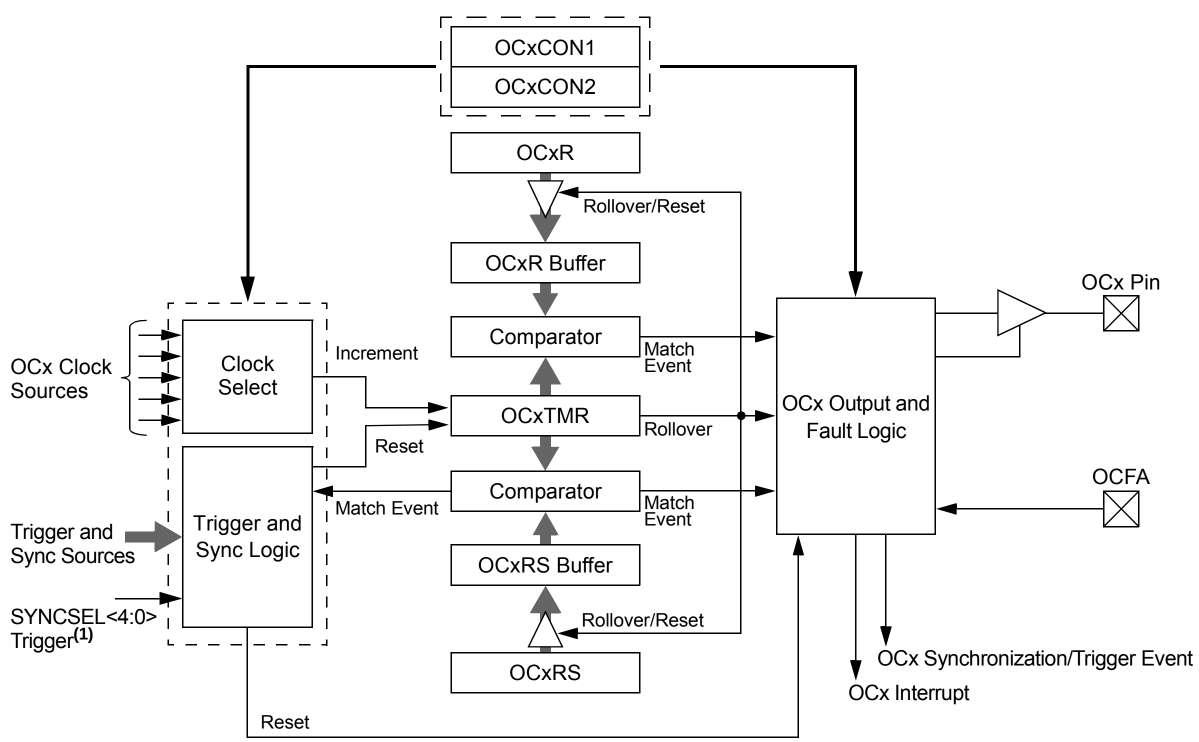
15.1 Output Compare Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

15.1.1 KEY RESOURCES

- **"Output Compare with Dedicated Timer"** (DS70005159) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

FIGURE 15-1: OUTPUT COMPARE x MODULE BLOCK DIAGRAM



Note 1: The trigger/sync source is enabled by default and is set to Timer2 as a source. This timer must be enabled for proper OCx module operation or the trigger/sync source must be changed to another source option.

15.2 Output Compare Control Registers

REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	—	—
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLTA	—	—	OCFLTA	TRIGMODE	OCM2	OCM1	OCM0
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13 **OCSIDL:** Output Compare x Stop in Idle Mode Control bit
1 = Output Compare x halts in CPU Idle mode
0 = Output Compare x continues to operate in CPU Idle mode

bit 12-10 **OCTSEL<2:0>:** Output Compare x Clock Select bits
111 = Peripheral clock (FP)
110 = Reserved
101 = Reserved
100 = T1CLK is the clock source of the OCx (only the synchronous clock is supported)
011 = T5CLK is the clock source of the OCx
010 = T4CLK is the clock source of the OCx
001 = T3CLK is the clock source of the OCx
000 = T2CLK is the clock source of the OCx

bit 9-8 **Unimplemented:** Read as ‘0’

bit 7 **ENFLTA:** Fault A Input Enable bit
1 = Output Compare Fault A input (OCFA) is enabled
0 = Output Compare Fault A input (OCFA) is disabled

bit 6-5 **Unimplemented:** Read as ‘0’

bit 4 **OCFLTA:** PWM Fault A Condition Status bit
1 = PWM Fault A condition on the OCFA pin has occurred
0 = No PWM Fault A condition on the OCFA pin has occurred

bit 3 **TRIGMODE:** Trigger Status Mode Select bit
1 = TRIGSTAT (OCxCON2<6>) is cleared when OCxRS = OCxTMR or in software
0 = TRIGSTAT is cleared only by software

Note 1: OCxR and OCxRS are double-buffered in PWM mode only.

REGISTER 15-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1 (CONTINUED)

bit 2-0 **OCM<2:0>:** Output Compare x Mode Select bits

`111` = Center-Aligned PWM mode: Output is set high when OCxTMR = OCxR and set low when OCxTMR = OCxRS[1]
`110` = Edge-Aligned PWM mode: Output is set high when OCxTMR = `0` and set low when OCxTMR = OCxR[1]
`101` = Double Compare Continuous Pulse mode: Initializes OCx pin low, toggles OCx state continuously on alternate matches of OCxR and OCxRS
`100` = Double Compare Single-Shot mode: Initializes OCx pin low, toggles OCx state on matches of OCxR and OCxRS for one cycle
`011` = Single Compare mode: Compare event with OCxR, continuously toggles OCx pin
`010` = Single Compare Single-Shot mode: Initializes OCx pin high, compare event with OCxR, forces OCx pin low
`001` = Single Compare Single-Shot mode: Initializes OCx pin low, compare event with OCxR, forces OCx pin high
`000` = Output compare channel is disabled

Note 1: OCxR and OCxRS are double-buffered in PWM mode only.

REGISTER 15-2: OCxCON2: OUTPUT COMPARE x CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
FLTMD	FLTOUT	FLTTRIEN	OCINV	—	—	—	OC32
bit 15							bit 8

R/W-0	R/W-0, HS	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
bit 7							bit 0

Legend:	HS = Hardware Settable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **FLTMD:** Fault Mode Select bit
1 = Fault mode is maintained until the Fault source is removed; the corresponding OCFLTA bit is cleared in software and a new PWMx period starts
0 = Fault mode is maintained until the Fault source is removed and a new PWMx period starts

bit 14 **FLTOUT:** Fault Out bit
1 = PWMx output is driven high on a Fault
0 = PWMx output is driven low on a Fault

bit 13 **FLTTRIEN:** Fault Output State Select bit
1 = OCx pin is tri-stated on a Fault condition
0 = OCx pin I/O state is defined by the FLTOUT bit on a Fault condition

bit 12 **OCINV:** Output Compare x Invert bit
1 = OCx output is inverted
0 = OCx output is not inverted

bit 11-9 **Unimplemented:** Read as ‘0’

bit 8 **OC32:** Cascade Two OCx Modules Enable bit (32-bit operation)
1 = Cascade module operation is enabled
0 = Cascade module operation is disabled

bit 7 **OCTRIG:** Output Compare x Trigger/Sync Select bit
1 = Triggers OCx from the source designated by the SYNCSELx bits
0 = Synchronizes OCx with the source designated by the SYNCSELx bits

bit 6 **TRIGSTAT:** Timer Trigger Status bit
1 = Timer source has been triggered and is running
0 = Timer source has not been triggered and is being held clear

bit 5 **OCTRIS:** Output Compare x Output Pin Direction Select bit
1 = OCx is tri-stated
0 = OCx module drives the OCx pin

Note 1: Do not use the OCx module as its own synchronization or trigger source.

2: When the OCy module is turned off, it sends a trigger out signal. If the OCx module uses the OCy module as a trigger source, the OCy module must be unselected as a trigger source prior to disabling it.

3: For each OCMPx instance, a different PTG trigger out is used:
OCMP1 – PTG trigger out [0]
OCMP2 – PTG trigger out [1]
OCMP3 – PTG trigger out [2]
OCMP4 – PTG trigger out [3]

REGISTER 15-2: OCxCON2: OUTPUT COMPARE x CONTROL REGISTER 2 (CONTINUED)

bit 4-0 **SYNCSEL<4:0>:** Trigger/Synchronization Source Selection bits

`11111` = OCxRS compare event is used for synchronization
`11110` = INT2 pin synchronizes or triggers OCx
`11101` = INT1 pin synchronizes or triggers OCx
`11100` = Reserved
`11011` = CMP4 module synchronizes or triggers OCx
`11010` = CMP3 module synchronizes or triggers OCx
`11001` = CMP2 module synchronizes or triggers OCx
`11000` = CMP1 module synchronizes or triggers OCx
`10111` = Reserved
`10110` = Reserved
`10101` = Reserved
`10100` = Reserved
`10011` = IC4 input capture interrupt event synchronizes or triggers OCx
`10010` = IC3 input capture interrupt event synchronizes or triggers OCx
`10001` = IC2 input capture interrupt event synchronizes or triggers OCx
`10000` = IC1 input capture interrupt event synchronizes or triggers OCx
`01111` = Timer5 synchronizes or triggers OCx
`01110` = Timer4 synchronizes or triggers OCx
`01101` = Timer3 synchronizes or triggers OCx
`01100` = Timer2 synchronizes or triggers OCx **(default)**
`01011` = Timer1 synchronizes or triggers OCx
`01010` = PTG Trigger Output x[3]
`01001` = Reserved
`01000` = IC4 input capture event synchronizes or triggers OCx
`00111` = IC3 input capture event synchronizes or triggers OCx
`00110` = IC2 input capture event synchronizes or triggers OCx
`00101` = IC1 input capture event synchronizes or triggers OCx
`00100` = OC4 module synchronizes or triggers OCx[1,2]
`00011` = OC3 module synchronizes or triggers OCx[1,2]
`00010` = OC2 module synchronizes or triggers OCx[1,2]
`00001` = OC1 module synchronizes or triggers OCx[1,2]
`00000` = No sync or trigger source for OCx

Note 1: Do not use the OCx module as its own synchronization or trigger source.

2: When the OCy module is turned off, it sends a trigger out signal. If the OCx module uses the OCy module as a trigger source, the OCy module must be unselected as a trigger source prior to disabling it.

3: For each OCMPx instance, a different PTG trigger out is used:
OCMP1 – PTG trigger out [0]
OCMP2 – PTG trigger out [1]
OCMP3 – PTG trigger out [2]
OCMP4 – PTG trigger out [3]

NOTES:

16.0 HIGH-SPEED PWM

Note:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"High-Speed PWM Module"** (DS70000323) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The high-speed PWM on dsPIC33EPXXXGS70X/80X devices supports a wide variety of PWM modes and output formats. This PWM module is ideal for power conversion applications, such as:

- AC/DC Converters
- DC/DC Converters
- Power Factor Correction
- Uninterruptible Power Supply (UPS)
- Inverters
- Battery Chargers
- Digital Lighting

16.1 Features Overview

The high-speed PWM module incorporates the following features:

- Eight PWMx Generators with Two Outputs per Generator
- Two Master Time Base modules
- Individual Time Base and Duty Cycle for each PWM Output
- Duty Cycle, Dead Time, Phase Shift and a Frequency Resolution of 1.04 ns
- Independent Fault and Current-Limit Inputs
- Redundant Output
- True Independent Output
- Center-Aligned PWM mode
- Output override control
- Chop mode (also known as Gated mode)
- Special Event Trigger
- Dual Trigger from PWMx to Analog-to-Digital Converter (ADC)
- PWMxL and PWMxH Output Pin Swapping
- Independent PWMx Frequency, Duty Cycle and Phase-Shift Changes
- Enhanced Leading-Edge Blanking (LEB) Functionality
- PWM Capture Functionality

Note:	Duty cycle, dead time, phase shift and frequency resolution is 8.32 ns in Center-Aligned PWM mode.

Figure 16-1 conceptualizes the PWM module in a simplified block diagram. Figure 16-2 illustrates how the module hardware is partitioned for each PWMx output pair for the Complementary PWM mode.

The PWM module contains eight PWM generators. The module has up to 16 PWMx output pins: PWM1H/PWM1L through PWM8H/PWM8L. For complementary outputs, these 16 I/O pins are grouped into high/low pairs. PWM1 through PWM6 can be used to trigger an ADC conversion.

16.2 Feature Description

The PWM module is designed for applications that require:

- High resolution at high PWM frequencies
- The ability to drive Standard, Edge-Aligned, Center-Aligned Complementary mode and Push-Pull mode outputs
- The ability to create multiphase PWM outputs

Two common, medium power converter topologies are push-pull and half-bridge. These designs require the PWM output signal to be switched between alternate pins, as provided by the Push-Pull PWM mode.

Phase-shifted PWM describes the situation where each PWM generator provides outputs, but the phase relationship between the generator outputs is specifiable and changeable.

Multiphase PWM is often used to improve DC/DC Converter load transient response, and reduce the size of output filter capacitors and inductors. Multiple DC/DC Converters are often operated in parallel, but phase shifted in time. A single PWM output, operating at 250 kHz, has a period of 4 µs but an array of four PWM channels, staggered by 1 µs each, yields an effective switching frequency of 1 MHz. Multiphase PWM applications typically use a fixed-phase relationship.

Variable phase PWM is useful in Zero Voltage Transition (ZVT) power converters. Here, the PWM duty cycle is always 50% and the power flow is controlled by varying the relative phase shift between the two PWM generators.

16.2.1 WRITE-PROTECTED REGISTERS

On dsPIC33EPXXXGS70X/80X family devices, write protection is implemented for the IOCONx and FCLCONx registers. The write protection feature prevents any inadvertent writes to these registers. This protection feature can be controlled by the PWMLOCK Configuration bit (FDEVOPT<0>). The default state of the write protection feature is enabled (PWMLOCK = `1`). The write protection feature can be disabled by configuring PWMLOCK = `0`.

To gain write access to these locked registers, the user application must write two consecutive values (0xABCD and 0x4321) to the PWMKEY register to perform the unlock operation. The write access to the IOCONx or FCLCONx registers must be the next SFR access following the unlock process. There can be no other SFR accesses during the unlock process and subsequent write access. To write to both the IOCONx and FCLCONx registers requires two unlock operations.

The correct unlocking sequence is described in Example 16-1.

EXAMPLE 16-1: PWM WRITE-PROTECTED REGISTER UNLOCK SEQUENCE

```
; Writing to FCLCON1 register requires unlock sequence

mov #0xabcd, w10      ; Load first unlock key to w10 register
mov #0x4321, w11      ; Load second unlock key to w11 register
mov #0x0000, w0       ; Load desired value of FCLCON1 register in w0
mov w10, PWMKEY       ; Write first unlock key to PWMKEY register
mov w11, PWMKEY       ; Write second unlock key to PWMKEY register
mov w0, FCLCON1       ; Write desired value to FCLCON1 register

; Set PWM ownership and polarity using the IOCON1 register
; Writing to IOCON1 register requires unlock sequence

mov #0xabcd, w10      ; Load first unlock key to w10 register
mov #0x4321, w11      ; Load second unlock key to w11 register
mov #0xF000, w0       ; Load desired value of IOCON1 register in w0
mov w10, PWMKEY       ; Write first unlock key to PWMKEY register
mov w11, PWMKEY       ; Write second unlock key to PWMKEY register
mov w0, IOCON1        ; Write desired value to IOCON1 register
```

16.3 PWM Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

16.3.1 KEY RESOURCES

- **"High-Speed PWM Module"** (DS70000323) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

FIGURE 16-1: HIGH-SPEED PWM MODULE ARCHITECTURAL DIAGRAM

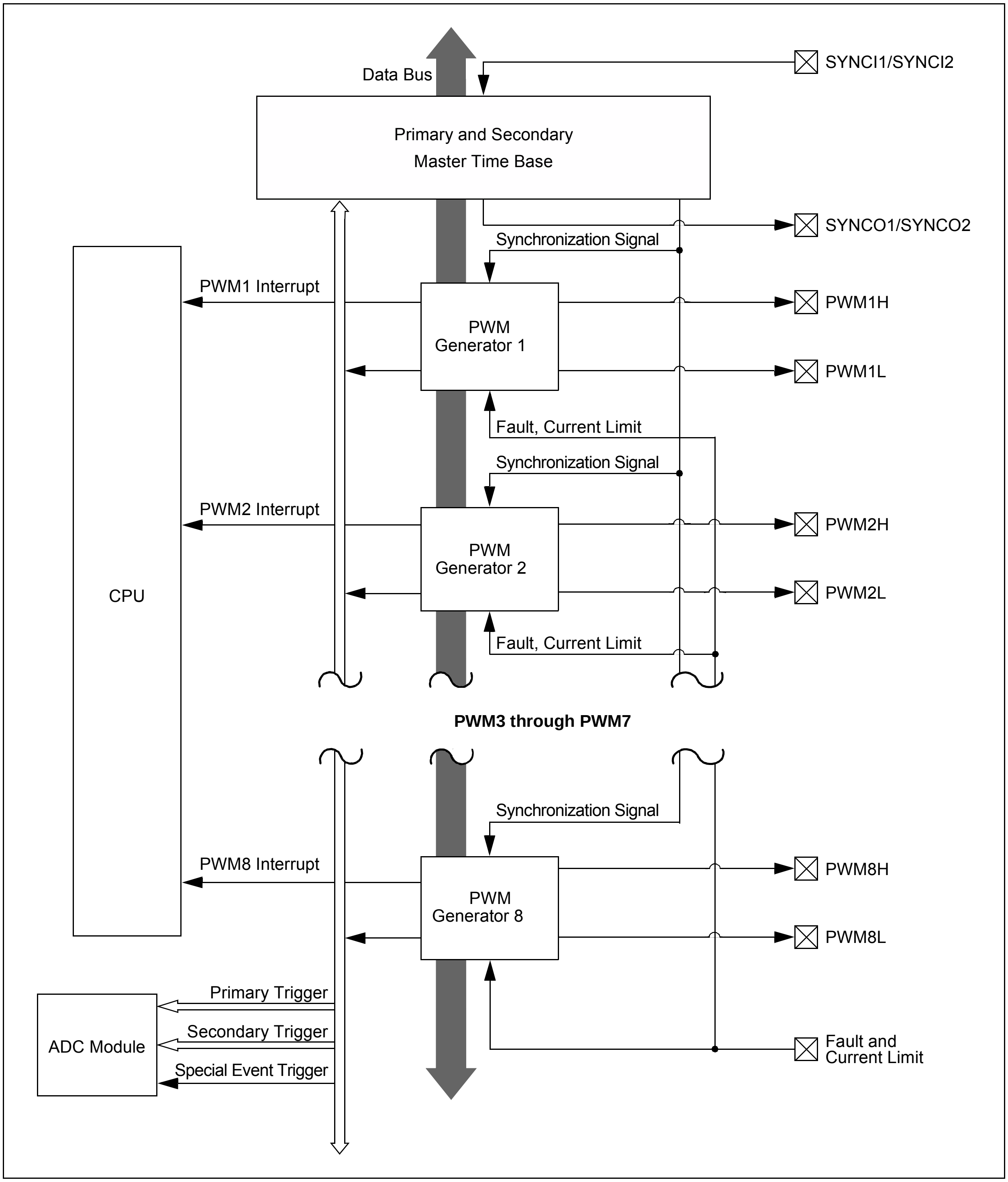
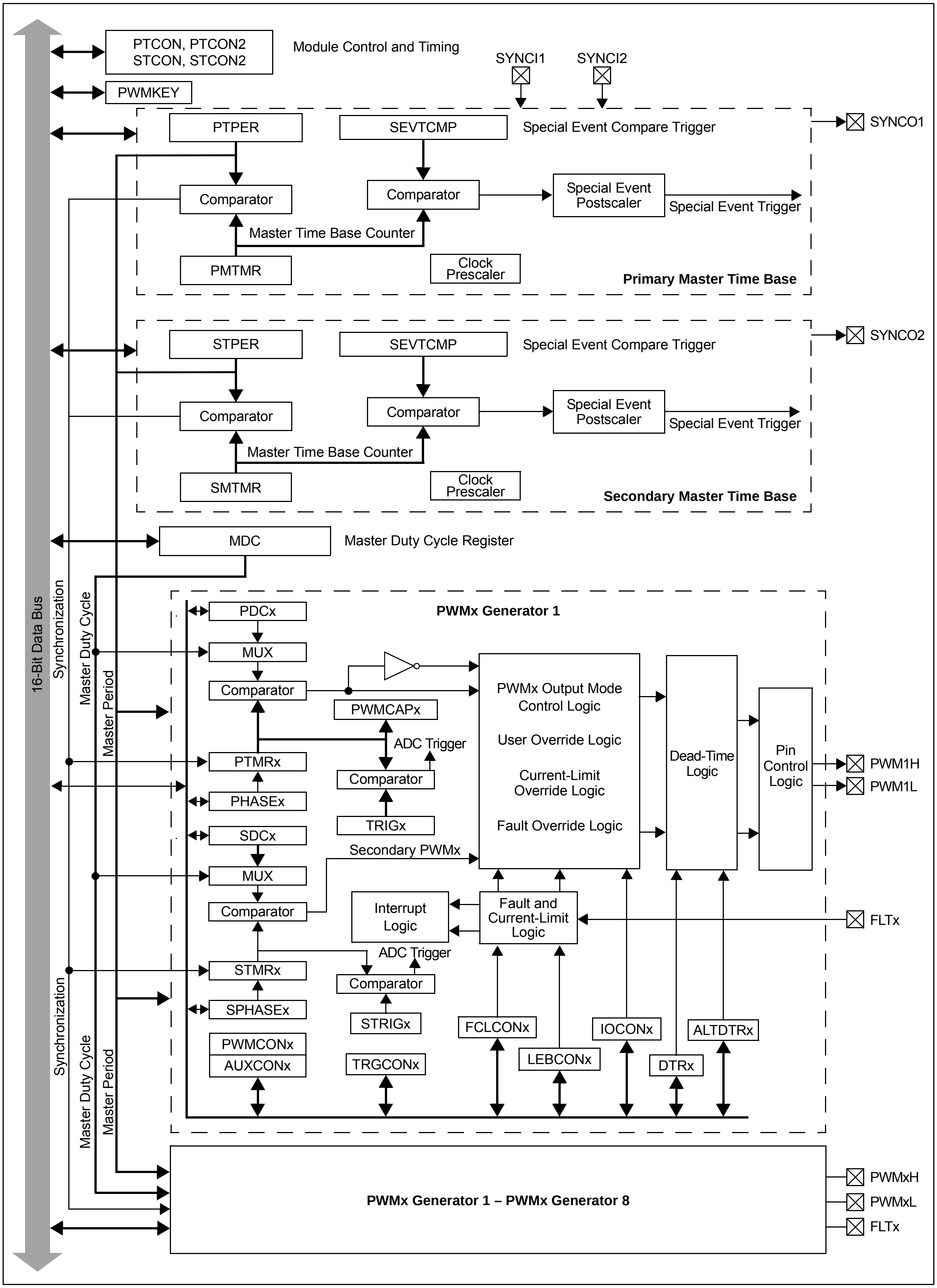


FIGURE 16-2: SIMPLIFIED CONCEPTUAL BLOCK DIAGRAM OF THE HIGH-SPEED PWM



REGISTER 16-1: PTCON: PWMx TIME BASE CONTROL REGISTER

R/W-0	U-0	R/W-0	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
PTEN	—	PTSIDL	SESTAT	SEIEN	EIPU(1)	SYNCPOL(1)	SYNCOEN(1)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCEN(1)	SYNCSRC2(1)	SYNCSRC1(1)	SYNCSRC0(1)	SEVTPS3(1)	SEVTPS2(1)	SEVTPS1(1)	SEVTPS0(1)
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **PTEN:** PWMx Module Enable bit
1 = PWMx module is enabled
0 = PWMx module is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **PTSIDL:** PWMx Time Base Stop in Idle Mode bit
1 = PWMx time base halts in CPU Idle mode
0 = PWMx time base runs in CPU Idle mode

bit 12 **SESTAT:** Special Event Interrupt Status bit
1 = Special event interrupt is pending
0 = Special event interrupt is not pending

bit 11 **SEIEN:** Special Event Interrupt Enable bit
1 = Special event interrupt is enabled
0 = Special event interrupt is disabled

bit 10 **EIPU:** Enable Immediate Period Updates bit(1)
1 = Active Period register is updated immediately
0 = Active Period register updates occur on PWMx cycle boundaries

bit 9 **SYNCPOL:** Synchronize Input and Output Polarity bit(1)
1 = SYNCIx/SYNCO1 polarity is inverted (active-low)
0 = SYNCIx/SYNCO1 is active-high

bit 8 **SYNCOEN:** Primary Time Base Synchronization Enable bit(1)
1 = SYNCO1 output is enabled
0 = SYNCO1 output is disabled

bit 7 **SYNCEN:** External Time Base Synchronization Enable bit(1)
1 = External synchronization of primary time base is enabled
0 = External synchronization of primary time base is disabled

bit 6-4 **SYNCSRC<2:0>:** Synchronous Source Selection bits(1)
111 = Reserved
101 = Reserved
100 = Reserved
011 = PTG Trigger Output 17
010 = PTG Trigger Output 16
001 = SYNCI2
000 = SYNCI1

Note 1: These bits should be changed only when PTEN = 0. In addition, when using the SYNCIx feature, the user application must program the Period register with a value that is slightly larger than the expected period of the external synchronization input signal.

REGISTER 16-1: PTCON: PWMx TIME BASE CONTROL REGISTER (CONTINUED)

bit 3-0 **SEVTPS<3:0>:** PWMx Special Event Trigger Output Postscaler Select bits[1]

1111 = 1:16 postscaler generates a Special Event Trigger on every sixteenth compare match event
•
•
•
0001 = 1:2 postscaler generates a Special Event Trigger on every second compare match event
0000 = 1:1 postscaler generates a Special Event Trigger on every compare match event

Note 1: These bits should be changed only when PTEN = 0. In addition, when using the SYNCIx feature, the user application must program the Period register with a value that is slightly larger than the expected period of the external synchronization input signal.

REGISTER 16-2: PTCON2: PWMx CLOCK DIVIDER SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0>[1]		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **Unimplemented:** Read as ‘0’

bit 2-0 **PCLKDIV<2:0>:** PWMx Input Clock Prescaler (Divider) Select bits[1]

111 = Reserved
110 = Divide-by-64, maximum PWM timing resolution
101 = Divide-by-32, maximum PWM timing resolution
100 = Divide-by-16, maximum PWM timing resolution
011 = Divide-by-8, maximum PWM timing resolution
010 = Divide-by-4, maximum PWM timing resolution
001 = Divide-by-2, maximum PWM timing resolution
000 = Divide-by-1, maximum PWM timing resolution (power-on default)

Note 1: These bits should be changed only when PTEN = 0. Changing the clock selection during operation will yield unpredictable results.

REGISTER 16-3: PTPER: PWMx PRIMARY MASTER TIME BASE PERIOD REGISTER(1,2)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PTPER<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PTPER<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PTPER<15:0>:** Primary Master Time Base (PMTMR) Period Value bits

Note 1: The PWMx time base has a minimum value of 0x0010 and a maximum value of 0xFFF8.

2: Any period value that is less than 0x0028 must have the Least Significant 3 bits set to ‘0’, thus yielding a period resolution at 8.32 ns (at fastest auxiliary clock rate).

REGISTER 16-4: SEVTCMP: PWMx SPECIAL EVENT COMPARE REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEVTCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SEVTCMP<4:0>					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **SEVTCMP<12:0>:** Special Event Compare Count Value bits

bit 2-0 **Unimplemented:** Read as ‘0’

Note 1: One LSB = 1.04 ns (at fastest auxiliary clock rate); therefore, the minimum SEVTCMP resolution is 8.32 ns.

REGISTER 16-5: STCON: PWMx SECONDARY MASTER TIME BASE CONTROL REGISTER

U-0	U-0	U-0	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	SESTAT	SEIEN	EIPU[(1)]	SYNCPOL	SYNCOEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCEN	SYNCSRC2	SYNCSRC1	SYNCSRC0	SEVTPS3	SEVTPS2	SEVTPS1	SEVTPS0
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12 **SESTAT:** Special Event Interrupt Status bit
1 = Secondary special event interrupt is pending
0 = Secondary special event interrupt is not pending

bit 11 **SEIEN:** Special Event Interrupt Enable bit
1 = Secondary special event interrupt is enabled
0 = Secondary special event interrupt is disabled

bit 10 **EIPU:** Enable Immediate Period Updates bit[(1)]
1 = Active Secondary Period register is updated immediately
0 = Active Secondary Period register updates occur on PWMx cycle boundaries

bit 9 **SYNCPOL:** Synchronize Input and Output Polarity bit
1 = SYNCIx/SYNCO2 polarity is inverted (active-low)
0 = SYNCIx/SYNCO2 polarity is active-high

bit 8 **SYNCOEN:** Secondary Master Time Base Synchronization Enable bit
1 = SYNCO2 output is enabled
0 = SYNCO2 output is disabled

bit 7 **SYNCEN:** External Secondary Master Time Base Synchronization Enable bit
1 = External synchronization of secondary time base is enabled
0 = External synchronization of secondary time base is disabled

bit 6-4 **SYNCSRC<2:0>:** Secondary Time Base Sync Source Selection bits
111 = Reserved
101 = Reserved
100 = Reserved
011 = PTG Trigger Output 17
010 = PTG Trigger Output 16
001 = SYNCI2
000 = SYNCI1

bit 3-0 **SEVTPS<3:0>:** PWMx Secondary Special Event Trigger Output Postscaler Select bits
1111 = 1:16 postscaler
0001 = 1:2 postscaler
•
•
•
0000 = 1:1 postscaler

Note 1: This bit only applies to the secondary master time base period.

REGISTER 16-6: STCON2: PWMx SECONDARY CLOCK DIVIDER SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0>[1]		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **Unimplemented:** Read as ‘0’

bit 2-0 **PCLKDIV<2:0>:** PWMx Input Clock Prescaler (Divider) Select bits[1]

111 = Reserved
110 = Divide-by-64, maximum PWM timing resolution
101 = Divide-by-32, maximum PWM timing resolution
100 = Divide-by-16, maximum PWM timing resolution
011 = Divide-by-8, maximum PWM timing resolution
010 = Divide-by-4, maximum PWM timing resolution
001 = Divide-by-2, maximum PWM timing resolution
000 = Divide-by-1, maximum PWM timing resolution (power-on default)

Note 1: These bits should be changed only when PTEN = 0. Changing the clock selection during operation will yield unpredictable results.

REGISTER 16-7: STPER: PWMx SECONDARY MASTER TIME BASE PERIOD REGISTER[1,2]

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
STPER<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
STPER<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **STPER<15:0>:** Secondary Master Time Base (SMTMR) Period Value bits

Note 1: The PWMx time base has a minimum value of 0x0010 and a maximum value of 0xFFF8.

2: Any period value that is less than 0x0028 must have the Least Significant 3 bits set to ‘0’, thus yielding a period resolution at 8.32 ns (at fastest auxiliary clock rate).

REGISTER 16-8: SSEVTCMP: PWMx SECONDARY SPECIAL EVENT COMPARE REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEVTCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SSEVTCMP<4:0>					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **SSEVTCMP<12:0>:** Special Event Compare Count Value bits

bit 2-0 **Unimplemented:** Read as ‘0’

Note 1: One LSB = 1.04 ns (at fastest auxiliary clock rate); therefore, the minimum SSEVTCMP resolution is 8.32 ns.

REGISTER 16-9: CHOP: PWMx CHOP CLOCK GENERATOR REGISTER[(1)]

R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
CHPCLKEN	—	—	—	—	—	CHOPCLK6	CHOPCLK5
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHOPCLK4	CHOPCLK3	CHOPCLK2	CHOPCLK1	CHOPCLK0	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CHPCLKEN:** Enable Chop Clock Generator bit
1 = Chop clock generator is enabled
0 = Chop clock generator is disabled

bit 14-10 **Unimplemented:** Read as ‘0’

bit 9-3 **CHOPCLK<6:0>:** Chop Clock Divider bits
Value is in 8.32 ns increments. The frequency of the chop clock signal is given by:
Chop Frequency = 1/(16.64 * (CHOP<7:3> + 1) * Primary Master PWM Input Clock Period)

bit 2-0 **Unimplemented:** Read as ‘0’

Note 1: The chop clock generator operates with the primary PWMx clock prescaler (PCLKDIV<2:0>) in the PTCON2 register (Register 16-2).

REGISTER 16-10: MDC: PWMx MASTER DUTY CYCLE REGISTER[1,2]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MDC<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MDC<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **MDC<15:0>:** PWMx Master Duty Cycle Value bits

Note 1: The smallest pulse width that can be generated on the PWMx output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period – 0x0008.

2: As the duty cycle gets closer to 0% or 100% of the PWMx period (0 to 40 ns, depending on the mode of operation), PWMx duty cycle resolution will increase from 1 to 3 LSBs.

REGISTER 16-11: PWMKEY: PWMx PROTECTION LOCK/UNLOCK KEY REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PWMKEY<15:0>:** PWMx Protection Lock/Unlock Key Value bits

REGISTER 16-12: PWMCONx: PWMx CONTROL REGISTER (x = 1 to 8)

R-0, HSC	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTSTAT[(1)]	CLSTAT[(1)]	TRGSTAT	FLTIEN	CLIEN	TRGIEN	ITB[(3)]	MDCS[(3)]
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
DTC1	DTC0	—	—	MTBS	CAM[(2,3,4)]	XPRES[(5)]	IUE
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **FLTSTAT:** Fault Interrupt Status bit[(1)]
1 = Fault interrupt is pending
0 = No Fault interrupt is pending
This bit is cleared by setting FLTIEN = 0.

bit 14 **CLSTAT:** Current-Limit Interrupt Status bit[(1)]
1 = Current-limit interrupt is pending
0 = No current-limit interrupt is pending
This bit is cleared by setting CLIEN = 0.

bit 13 **TRGSTAT:** Trigger Interrupt Status bit
1 = Trigger interrupt is pending
0 = No trigger interrupt is pending
This bit is cleared by setting TRGIEN = 0.

bit 12 **FLTIEN:** Fault Interrupt Enable bit
1 = Fault interrupt is enabled
0 = Fault interrupt is disabled and the FLTSTAT bit is cleared

bit 11 **CLIEN:** Current-Limit Interrupt Enable bit
1 = Current-limit interrupt is enabled
0 = Current-limit interrupt is disabled and the CLSTAT bit is cleared

bit 10 **TRGIEN:** Trigger Interrupt Enable bit
1 = A trigger event generates an interrupt request
0 = Trigger event interrupts are disabled and the TRGSTAT bit is cleared

bit 9 **ITB:** Independent Time Base Mode bit[(3)]
1 = PHASEx/SPHASEx registers provide the time base period for this PWMx generator
0 = PTPER register provides timing for this PWMx generator

bit 8 **MDCS:** Master Duty Cycle Register Select bit[(3)]
1 = MDC register provides duty cycle information for this PWMx generator
0 = PDCx and SDCx registers provide duty cycle information for this PWMx generator

Note 1: Software must clear the interrupt status here and in the corresponding IFSx bit in the interrupt controller.
2: The Independent Time Base mode (ITB = 1) must be enabled to use Center-Aligned mode. If ITB = 0, the CAM bit is ignored.
3: These bits should not be changed after the PWMx is enabled by setting PTEN (PTCON<15>) = 1.
4: Center-Aligned mode ignores the Least Significant 3 bits of the Duty Cycle, Phase and Dead-Time registers. The highest Center-Aligned mode resolution available is 8.32 ns with the clock prescaler set to the fastest clock.
5: Configure CLMOD (FCLCONx<8>) = 0 and ITB (PWMCONx<9>) = 1 to operate in External Period Reset mode.

REGISTER 16-12: PWMCONx: PWMx CONTROL REGISTER (x = 1 to 8) (CONTINUED)

bit 7-6 **DTC<1:0>:** Dead-Time Control bits
11 = Reserved
10 = Dead-time function is disabled
01 = Negative dead time is actively applied for Complementary Output mode
00 = Positive dead time is actively applied for all Output modes

bit 5-4 **Unimplemented:** Read as ‘0’

bit 3 **MTBS:** Master Time Base Select bit
1 = PWMx generator uses the secondary master time base for synchronization and the clock source for the PWMx generation logic (if secondary time base is available)
0 = PWMx generator uses the primary master time base for synchronization and the clock source for the PWMx generation logic

bit 2 **CAM:** Center-Aligned Mode Enable bit(2,3,4)
1 = Center-Aligned mode is enabled
0 = Edge-Aligned mode is enabled

bit 1 **XPRES:** External PWMx Reset Control bit(5)
1 = Current-limit source resets the time base for this PWMx generator if it is in Independent Time Base mode
0 = External pins do not affect the PWMx time base

bit 0 **IUE:** Immediate Update Enable bit
1 = Updates to the active Duty Cycle, Phase Offset, Dead-Time and local Time Base Period registers are immediate
0 = Updates to the active Duty Cycle, Phase Offset, Dead-Time and local Time Base Period registers are synchronized to the local PWMx time base

Note 1: Software must clear the interrupt status here and in the corresponding IFSx bit in the interrupt controller.
2: The Independent Time Base mode (ITB = 1) must be enabled to use Center-Aligned mode. If ITB = 0, the CAM bit is ignored.
3: These bits should not be changed after the PWMx is enabled by setting PTEN (PTCON<15>) = 1.
4: Center-Aligned mode ignores the Least Significant 3 bits of the Duty Cycle, Phase and Dead-Time registers. The highest Center-Aligned mode resolution available is 8.32 ns with the clock prescaler set to the fastest clock.
5: Configure CLMOD (FCLCONx<8>) = 0 and ITB (PWMCONx<9>) = 1 to operate in External Period Reset mode.

REGISTER 16-13: PDCx: PWMx GENERATOR DUTY CYCLE REGISTER (x = 1 to 8)[1,2,3]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDCx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PDCx<15:0>:** PWMx Generator Duty Cycle Value bits

Note 1: In Independent PWM mode, the PDCx register controls the PWMxH duty cycle only. In the Complementary, Redundant and Push-Pull PWM modes, the PDCx register controls the duty cycle of both the PWMxH and PWMxL.

2: The smallest pulse width that can be generated on the PWMx output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period – 0x0008.

3: As the duty cycle gets closer to 0% or 100% of the PWMx period (0 to 40 ns, depending on the mode of operation), PWMx duty cycle resolution will increase from 1 to 3 LSBs.

REGISTER 16-14: SDCx: PWMx SECONDARY DUTY CYCLE REGISTER (x = 1 to 8)[1,2,3]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDCx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **SDCx<15:0>:** PWMx Secondary Duty Cycle for PWMxL Output Pin bits

Note 1: The SDCx register is used in Independent PWM mode only. When used in Independent PWM mode, the SDCx register controls the PWMxL duty cycle.

2: The smallest pulse width that can be generated on the PWMx output corresponds to a value of 0x0008, while the maximum pulse width generated corresponds to a value of Period – 0x0008.

3: As the duty cycle gets closer to 0% or 100% of the PWMx period (0 to 40 ns, depending on the mode of operation), PWMx duty cycle resolution will increase from 1 to 3 LSBs.

REGISTER 16-15: PHASEx: PWMx PRIMARY PHASE-SHIFT REGISTER (x = 1 to 8)[1,2]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PHASEx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PHASEx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PHASEx<15:0>:** PWMx Phase-Shift Value or Independent Time Base Period for the PWMx Generator bits

Note 1: If PWMCONx<9> = 0, the following applies based on the mode of operation:
- Complementary, Redundant and Push-Pull Output mode (IOCONx<11:10> = 00, 01 or 10); PHASEx<15:0> = Phase-shift value for PWMxH and PWMxL outputs
- True Independent Output mode (IOCONx<11:10> = 11); PHASEx<15:0> = Phase-shift value for PWMxH only
- When the PHASEx/SPHASEx registers provide the phase shift with respect to the master time base; therefore, the valid range is 0x0000 through period

2: If PWMCONx<9> = 1, the following applies based on the mode of operation:
- Complementary, Redundant and Push-Pull Output mode (IOCONx<11:10> = 00, 01 or 10); PHASEx<15:0> = Independent time base period value for PWMxH and PWMxL
- True Independent Output mode (IOCONx<11:10> = 11); PHASEx<15:0> = Independent time base period value for PWMxH only
- When the PHASEx/SPHASEx registers provide the local period, the valid range is 0x0000-0xFFF8

REGISTER 16-16: SPHASEx: PWMx SECONDARY PHASE-SHIFT REGISTER (x = 1 to 8)[1,2]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPHASEx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPHASEx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **SPHASEx<15:0>:** Secondary Phase Offset for PWMxL Output Pin bits (used in Independent PWM mode only)

Note 1: If PWMCONx<9> = `0`, the following applies based on the mode of operation:
- Complementary, Redundant and Push-Pull Output mode (IOCONx<11:10> = `00`, `01` or `10`); SPHASEx<15:0> = Not used
- True Independent Output mode (IOCONx<11:10> = `11`), PHASEx<15:0> = Phase-shift value for PWMxL only

2: If PWMCONx<9> = `1`, the following applies based on the mode of operation:
- Complementary, Redundant and Push-Pull Output mode (IOCONx<11:10> = `00`, `01` or `10`); SPHASEx<15:0> = Not used
- True Independent Output mode (IOCONx<11:10> = `11`); PHASEx<15:0> = Independent time base period value for PWMxL only
- When the PHASEx/SPHASEx registers provide the local period, the valid range of values is 0x0010-0xFFF8

REGISTER 16-17: DTRx: PWMx DEAD-TIME REGISTER (x = 1 to 8)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DTRx<13:8>					
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTRx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-0 **DTRx<13:0>:** Unsigned 14-Bit Dead-Time Value for PWMx Dead-Time Unit bits

REGISTER 16-18: ALTDTRx: PWMx ALTERNATE DEAD-TIME REGISTER (x = 1 to 8)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ALTDTRx<13:8>					
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALTDTRx<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-0 **ALTDTRx<13:0>:** Unsigned 14-Bit Alternate Dead-Time Value for PWMx Dead-Time Unit bits

REGISTER 16-19: TRGCONx: PWMx TRIGGER CONTROL REGISTER (x = 1 to 8)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
TRGDIV3	TRGDIV2	TRGDIV1	TRGDIV0	—	—	—	—
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTM(1)	—	TRGSTRT5	TRGSTRT4	TRGSTRT3	TRGSTRT2	TRGSTRT1	TRGSTRT0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **TRGDIV<3:0>:** Trigger # Output Divider bits

1111 = Trigger output for every 16th trigger event
1110 = Trigger output for every 15th trigger event
1101 = Trigger output for every 14th trigger event
1100 = Trigger output for every 13th trigger event
1011 = Trigger output for every 12th trigger event
1010 = Trigger output for every 11th trigger event
1001 = Trigger output for every 10th trigger event
1000 = Trigger output for every 9th trigger event
0111 = Trigger output for every 8th trigger event
0110 = Trigger output for every 7th trigger event
0101 = Trigger output for every 6th trigger event
0100 = Trigger output for every 5th trigger event
0011 = Trigger output for every 4th trigger event
0010 = Trigger output for every 3rd trigger event
0001 = Trigger output for every 2nd trigger event
0000 = Trigger output for every trigger event

bit 11-8 **Unimplemented:** Read as ‘0’

bit 7 **DTM:** Dual Trigger Mode bit(1)

1 = Secondary trigger event is combined with the primary trigger event to create a PWM trigger
0 = Secondary trigger event is not combined with the primary trigger event to create a PWM trigger; two separate PWM triggers are generated

bit 6 **Unimplemented:** Read as ‘0’

bit 5-0 **TRGSTRT<5:0>:** Trigger Postscaler Start Enable Select bits

111111 = Wait 63 PWM cycles before generating the first trigger event after the module is enabled
•
•
•
000010 = Wait 2 PWM cycles before generating the first trigger event after the module is enabled
000001 = Wait 1 PWM cycle before generating the first trigger event after the module is enabled
000000 = Wait 0 PWM cycles before generating the first trigger event after the module is enabled

Note 1: The secondary PWMx generator cannot generate PWM trigger interrupts.

REGISTER 16-20: IOCONx: PWMx I/O CONTROL REGISTER (x = 1 to 8)

R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PENH	PENL	POLH	POLL	PMOD1[1]	PMOD0[1]	OVRENH	OVRENL
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OVRDAT1	OVRDAT0	FLTDAT1[2]	FLTDAT0[2]	CLDAT1[2]	CLDAT0[2]	SWAP	OSYNC
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **PENH:** PWMxH Output Pin Ownership bit
1 = PWMx module controls the PWMxH pin
0 = GPIO module controls the PWMxH pin

bit 14 **PENL:** PWMxL Output Pin Ownership bit
1 = PWMx module controls the PWMxL pin
0 = GPIO module controls the PWMxL pin

bit 13 **POLH:** PWMxH Output Pin Polarity bit
1 = PWMxH pin is active-low
0 = PWMxH pin is active-high

bit 12 **POLL:** PWMxL Output Pin Polarity bit
1 = PWMxL pin is active-low
0 = PWMxL pin is active-high

bit 11-10 **PMOD<1:0>:** PWMx I/O Pin Mode bits[1]
11 = PWMx I/O pin pair is in the True Independent Output mode
10 = PWMx I/O pin pair is in the Push-Pull Output mode
01 = PWMx I/O pin pair is in the Redundant Output mode
00 = PWMx I/O pin pair is in the Complementary Output mode

bit 9 **OVRENH:** Override Enable for PWMxH Pin bit
1 = OVRDAT1 provides data for output on the PWMxH pin
0 = PWMx generator provides data for the PWMxH pin

bit 8 **OVRENL:** Override Enable for PWMxL Pin bit
1 = OVRDAT0 provides data for output on the PWMxL pin
0 = PWMx generator provides data for the PWMxL pin

bit 7-6 **OVRDAT<1:0>:** Data for PWMxH, PWMxL Pins if Override is Enabled bits
If OVRENH = 1, OVRDAT1 provides data for the PWMxH pin
If OVRENL = 1, OVRDAT0 provides data for the PWMxL pin

bit 5-4 **FLTDAT<1:0>:** State for PWMxH and PWMxL Pins if FLTMOD<1:0> are Enabled bits[2]
IFLTMOD (FCLCONx<15>) = 0: Normal Fault mode:
If Fault is active, then FLTDAT1 provides the state for the PWMxH pin.
If Fault is active, then FLTDAT0 provides the state for the PWMxL pin.
IFLTMOD (FCLCONx<15>) = 1: Independent Fault mode:
If current limit is active, then FLTDAT1 provides the state for the PWMxH pin.
If Fault is active, then FLTDAT0 provides the state for the PWMxL pin.

Note 1: These bits should not be changed after the PWMx module is enabled (PTEN = 1).
2: State represents the active/inactive state of the PWMx depending on the POLH and POLL bits settings.

REGISTER 16-20: IOCONx: PWMx I/O CONTROL REGISTER (x = 1 to 8) (CONTINUED)

bit 3-2 **CLDAT<1:0>:** State for PWMxH and PWMxL Pins if CLMOD is Enabled bits[2]

IFLTMOD (FCLCONx<15>) = 0: Normal Fault mode:
If current limit is active, then CLDAT1 provides the state for the PWMxH pin.
If current limit is active, then CLDAT0 provides the state for the PWMxL pin.

IFLTMOD (FCLCONx<15>) = 1: Independent Fault mode:
CLDAT<1:0> bits are ignored.

bit 1 **SWAP:** SWAP PWMxH and PWMxL Pins bit

1 = PWMxH output signal is connected to the PWMxL pins; PWMxL output signal is connected to the PWMxH pins
0 = PWMxH and PWMxL pins are mapped to their respective pins

bit 0 **OSYNC:** Output Override Synchronization bit

1 = Output overrides via the OVRDAT<1:0> bits are synchronized to the PWMx time base
0 = Output overrides via the OVRDAT<1:0> bits occur on the next CPU clock boundary

Note 1: These bits should not be changed after the PWMx module is enabled (PTEN = 1).
2: State represents the active/inactive state of the PWMx depending on the POLH and POLL bits settings.

REGISTER 16-21: TRIGx: PWMx PRIMARY TRIGGER COMPARE VALUE REGISTER (x = 1 to 8)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TRGCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
TRGCMP<4:0>					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **TRGCMP<12:0>:** Trigger Compare Value bits

When the primary PWMx functions in the local time base, this register contains the compare values that can trigger the ADC module.

bit 2-0 **Unimplemented:** Read as ‘0’

REGISTER 16-22: FCLCONx: PWMx FAULT CURRENT-LIMIT CONTROL REGISTER (x = 1 to 8)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IFLTMOD	CLSRC4	CLSRC3	CLSRC2	CLSRC1	CLSRC0	CLPOL[1]	CLMOD
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTSRC4	FLTSRC3	FLTSRC2	FLTSRC1	FLTSRC0	FLTPOL[1]	FLTMOD1	FLTMOD0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **IFLTMOD:** Independent Fault Mode Enable bit
1 = Independent Fault mode: Current-limit input maps FLTDAT1 to the PWMxH output and the Fault input maps FLTDAT0 to the PWMxL output; the CLDAT<1:0> bits are not used for override functions
0 = Normal Fault mode: Current-Limit mode maps CLDAT<1:0> bits to the PWMxH and PWMxL outputs; the PWM Fault mode maps FLTDAT<1:0> to the PWMxH and PWMxL outputs

bit 14-10 **CLSRC<4:0>:** Current-Limit Control Signal Source Select for PWMx Generator bits
11111 = Reserved
10001 = Reserved
10000 = Analog Comparator 4
01111 = Analog Comparator 3
01110 = Analog Comparator 2
01101 = Analog Comparator 1
01100 = Fault 12
01011 = Fault 11
01010 = Fault 10
01001 = Fault 9
01000 = Fault 8
00111 = Fault 7
00110 = Fault 6
00101 = Fault 5
00100 = Fault 4
00011 = Fault 3
00010 = Fault 2
00001 = Fault 1
00000 = Reserved

bit 9 **CLPOL:** Current-Limit Polarity for PWMx Generator bit[1]
1 = The selected current-limit source is active-low
0 = The selected current-limit source is active-high

bit 8 **CLMOD:** Current-Limit Mode Enable for PWMx Generator bit
1 = Current-Limit mode is enabled
0 = Current-Limit mode is disabled

Note 1: These bits should be changed only when PTEN = 0 (PTCON<15>).

REGISTER 16-22: FCLCONx: PWMx FAULT CURRENT-LIMIT CONTROL REGISTER (x = 1 to 8) (CONTINUED)

bit 7-3	**FLTSRC<4:0>:** Fault Control Signal Source Select for PWMx Generator bits 11111 = Reserved 10001 = Reserved 10000 = Analog Comparator 4 01111 = Analog Comparator 3 01110 = Analog Comparator 2 01101 = Analog Comparator 1 01100 = Fault 12 01011 = Fault 11 01010 = Fault 10 01001 = Fault 9 01000 = Fault 8 00111 = Fault 7 00110 = Fault 6 00101 = Fault 5 00100 = Fault 4 00011 = Fault 3 00010 = Fault 2 00001 = Fault 1 00000 = Reserved
bit 2	**FLTPOL:** Fault Polarity for PWMx Generator bit[1] 1 = The selected Fault source is active-low 0 = The selected Fault source is active-high
bit 1-0	**FLTMOD<1:0>:** Fault Mode for PWMx Generator bits 11 = Fault input is disabled 10 = Reserved 01 = The selected Fault source forces the PWMxH, PWMxL pins to FLTDATx values (cycle) 00 = The selected Fault source forces the PWMxH, PWMxL pins to FLTDATx values (latched condition)

Note 1: These bits should be changed only when PTEN = 0 (PTCON<15>).

REGISTER 16-23: STRIGx: PWMx SECONDARY TRIGGER COMPARE VALUE REGISTER (x = 1 to 8)[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STRGCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
STRGCMP<4:0>					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3	**STRGCMP<12:0>:** Secondary Trigger Compare Value bits When the secondary PWMx functions in the local time base, this register contains the compare values that can trigger the ADC module.
bit 2-0	**Unimplemented:** Read as ‘0’

Note 1: STRIGx cannot generate the PWM trigger interrupts.

REGISTER 16-24: LEBCONx: PWMx LEADING-EDGE BLANKING (LEB) CONTROL REGISTER (x = 1 to 8)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	BCH[(1)]	BCL[(1)]	BPHH	BPHL	BPLH	BPLL
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **PHR:** PWMxH Rising Edge Trigger Enable bit
1 = Rising edge of PWMxH will trigger the Leading-Edge Blanking counter
0 = Leading-Edge Blanking ignores the rising edge of PWMxH

bit 14 **PHF:** PWMxH Falling Edge Trigger Enable bit
1 = Falling edge of PWMxH will trigger the Leading-Edge Blanking counter
0 = Leading-Edge Blanking ignores the falling edge of PWMxH

bit 13 **PLR:** PWMxL Rising Edge Trigger Enable bit
1 = Rising edge of PWMxL will trigger the Leading-Edge Blanking counter
0 = Leading-Edge Blanking ignores the rising edge of PWMxL

bit 12 **PLF:** PWMxL Falling Edge Trigger Enable bit
1 = Falling edge of PWMxL will trigger the Leading-Edge Blanking counter
0 = Leading-Edge Blanking ignores the falling edge of PWMxL

bit 11 **FLTLEBEN:** Fault Input Leading-Edge Blanking Enable bit
1 = Leading-Edge Blanking is applied to the selected Fault input
0 = Leading-Edge Blanking is not applied to the selected Fault input

bit 10 **CLLEBEN:** Current-Limit Leading-Edge Blanking Enable bit
1 = Leading-Edge Blanking is applied to the selected current-limit input
0 = Leading-Edge Blanking is not applied to the selected current-limit input

bit 9-6 **Unimplemented:** Read as ‘0’

bit 5 **BCH:** Blanking in Selected Blanking Signal High Enable bit[(1)]
1 = State blanking (of current-limit and/or Fault input signals) when the selected blanking signal is high
0 = No blanking when the selected blanking signal is high

bit 4 **BCL:** Blanking in Selected Blanking Signal Low Enable bit[(1)]
1 = State blanking (of current-limit and/or Fault input signals) when the selected blanking signal is low
0 = No blanking when the selected blanking signal is low

bit 3 **BPHH:** Blanking in PWMxH High Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxH output is high
0 = No blanking when the PWMxH output is high

bit 2 **BPHL:** Blanking in PWMxH Low Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxH output is low
0 = No blanking when the PWMxH output is low

Note 1: The blanking signal is selected via the BLANKSEL<3:0> bits in the AUXCONx register.

REGISTER 16-24: LEBCONx: PWMx LEADING-EDGE BLANKING (LEB) CONTROL REGISTER (x = 1 to 8) (CONTINUED)

bit 1 **BPLH:** Blanking in PWMxL High Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxL output is high
0 = No blanking when the PWMxL output is high

bit 0 **BPLL:** Blanking in PWMxL Low Enable bit
1 = State blanking (of current-limit and/or Fault input signals) when the PWMxL output is low
0 = No blanking when the PWMxL output is low

Note 1: The blanking signal is selected via the BLANKSEL<3:0> bits in the AUXCONx register.

REGISTER 16-25: LEBDLYx: PWMx LEADING-EDGE BLANKING DELAY REGISTER (x = 1 to 8)

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	LEB<8:5>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
LEB<4:0>					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-3 **LEB<8:0>:** Leading-Edge Blanking Delay for Current-Limit and Fault Inputs bits
The value is in 8.32 ns increments.

bit 2-0 **Unimplemented:** Read as ‘0’

REGISTER 16-26: AUXCONx: PWMx AUXILIARY CONTROL REGISTER (x = 1 to 8)

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
HRPDIS	HRDDIS	—	—	BLANKSEL3	BLANKSEL2	BLANKSEL1	BLANKSEL0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CHOPSEL3	CHOPSEL2	CHOPSEL1	CHOPSEL0	CHOPHEN	CHOPLEN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **HRPDIS:** High-Resolution PWMx Period Disable bit
1 = High-resolution PWMx period is disabled to reduce power consumption
0 = High-resolution PWMx period is enabled

bit 14 **HRDDIS:** High-Resolution PWMx Duty Cycle Disable bit
1 = High-resolution PWMx duty cycle is disabled to reduce power consumption
0 = High-resolution PWMx duty cycle is enabled

bit 13-12 **Unimplemented:** Read as ‘0’

bit 11-8 **BLANKSEL<3:0>:** PWMx State Blank Source Select bits
The selected state blank signal will block the current-limit and/or Fault input signals (if enabled via the BCH and BCL bits in the LEBCONx register).
1001 = Reserved
1000 = PWM8H is selected as the state blank source
0111 = PWM7H is selected as the state blank source
0110 = PWM6H is selected as the state blank source
0101 = PWM5H is selected as the state blank source
0100 = PWM4H is selected as the state blank source
0011 = PWM3H is selected as the state blank source
0010 = PWM2H is selected as the state blank source
0001 = PWM1H is selected as the state blank source
0000 = No state blanking

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-2 **CHOPSEL<3:0>:** PWMx Chop Clock Source Select bits
The selected signal will enable and disable (chop) the selected PWMx outputs.
1001 = Reserved
1000 = PWM8H is selected as the chop clock source
0111 = PWM7H is selected as the chop clock source
0110 = PWM6H is selected as the chop clock source
0101 = PWM5H is selected as the chop clock source
0100 = PWM4H is selected as the chop clock source
0011 = PWM3H is selected as the chop clock source
0010 = PWM2H is selected as the chop clock source
0001 = PWM1H is selected as the chop clock source
0000 = Chop clock generator is selected as the chop clock source

bit 1 **CHOPHEN:** PWMxH Output Chopping Enable bit
1 = PWMxH chopping function is enabled
0 = PWMxH chopping function is disabled

bit 0 **CHOPLEN:** PWMxL Output Chopping Enable bit
1 = PWMxL chopping function is enabled
0 = PWMxL chopping function is disabled

REGISTER 16-27: PWMCAPx: PWMx PRIMARY TIME BASE CAPTURE REGISTER (x = 1 to 8)

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PWMCAP<12:5>[1,2,3,4]							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	U-0	U-0	U-0
PWMCAP<4:0>[1,2,3,4]					—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-3 **PWMCAP<12:0>:** PWMx Primary Time Base Capture Value bits[1,2,3,4]
The value in this register represents the captured PWMx time base value when a leading edge is detected on the current-limit input.

bit 2-0 **Unimplemented:** Read as ‘0’

Note 1: The capture feature is only available on a primary output (PWMxH).
2: This feature is active only after LEB processing on the current-limit input signal is complete.
3: The minimum capture resolution is 8.32 ns.
4: This feature can be used when the XPRES bit (PWMCONx<1>) is set to ‘0’.

17.0 PERIPHERAL TRIGGER GENERATOR (PTG) MODULE

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Peripheral Trigger Generator (PTG)"** (DS70669) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

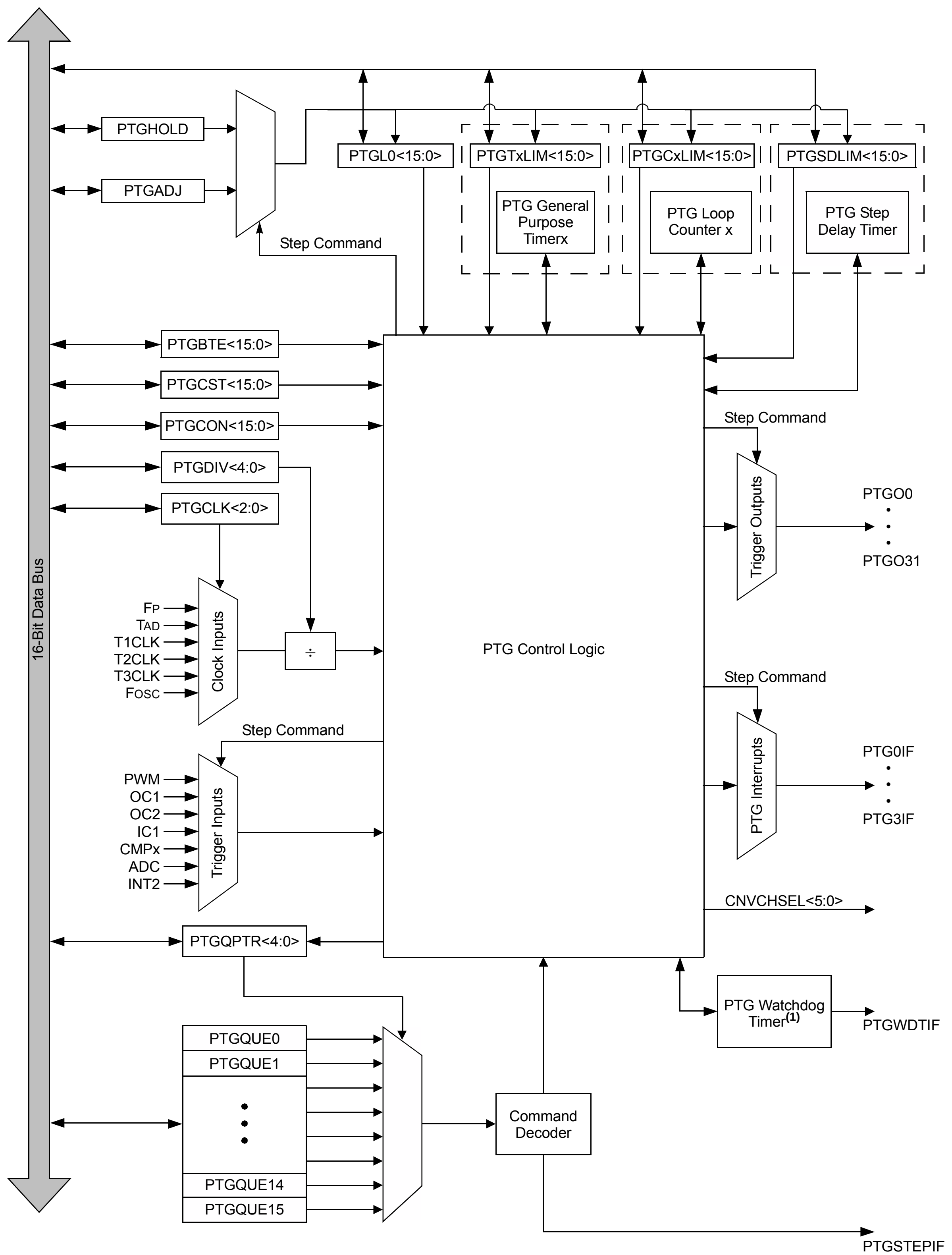
17.1 Module Introduction

The Peripheral Trigger Generator (PTG) provides a means to schedule complex, high-speed peripheral operations that would be difficult to achieve using software. The PTG module uses 8-bit commands, called "Steps", that the user writes to the PTG Queue register (PTGQUE0-PTQUE15) which performs operations, such as wait for input signal, generate output trigger and wait for timer.

The PTG module has the following major features:

- Multiple Clock Sources
- Two 16-Bit General Purpose Timers
- Two 16-Bit General Limit Counters
- Configurable for Rising or Falling Edge Triggering
- Generates Processor Interrupts to include:
  - Four configurable processor interrupts
  - Interrupt on a Step event in Single-Step mode
  - Interrupt on a PTG Watchdog Timer time-out
- Able to Receive Trigger Signals from these Peripherals:
  - ADC
  - PWM
  - Output Compare
  - Input Capture
  - Comparator
  - INT2
- Able to Trigger or Synchronize to these Peripherals:
  - Watchdog Timer
  - Output Compare
  - Input Capture
  - ADC
  - PWM
  - Comparator

FIGURE 17-1: PTG BLOCK DIAGRAM



Note 1: This is a dedicated Watchdog Timer for the PTG module and is independent of the device Watchdog Timer.

17.2 PTG Control Registers

REGISTER 17-1: PTGCST: PTG CONTROL/STATUS REGISTER

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTGEN	—	PTGSIDL	PTGTOGL	—	PTGSWT[2]	PTGSSEN	PTGIVIS
bit 15							bit 8

R/W-0	HS-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
PTGSTRT	PTGWDTO	—	—	—	—	PTGITM1[1]	PTGITM0[1]
bit 7							bit 0

Legend:	HS = Hardware Settable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **PTGEN:** PTG Module Enable bit
1 = PTG module is enabled
0 = PTG module is disabled

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **PTGSIDL:** PTG Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **PTGTOGL:** PTG TRIG Output Toggle Mode bit
1 = Toggles the state of the PTGOx for each execution of the PTGTRIG command
0 = Each execution of the PTGTRIG command will generate a single PTGOx pulse determined by the value in the PTGPWDx bits

bit 11 **Unimplemented:** Read as ‘0’

bit 10 **PTGSWT:** PTG Software Trigger bit[2]
1 = Triggers the PTG module
0 = No action (clearing this bit will have no effect)

bit 9 **PTGSSEN:** PTG Enable Single-Step bit
1 = Enables Single-Step mode
0 = Disables Single-Step mode

bit 8 **PTGIVIS:** PTG Counter/Timer Visibility Control bit
1 = Reads of the PTGSDLIM, PTGCxLIM or PTGTxLIM registers return the current values of their corresponding Counter/Timer registers (PTGSD, PTGCx, PTGTx)
0 = Reads of the PTGSDLIM, PTGCxLIM or PTGTxLIM registers return the value previously written to those PTG Limit registers

bit 7 **PTGSTRT:** Start PTG Sequencer bit
1 = Starts to sequentially execute commands (Continuous mode)
0 = Stops executing commands

bit 6 **PTGWDTO:** PTG Watchdog Timer Time-out Status bit
1 = PTG Watchdog Timer has timed out
0 = PTG Watchdog Timer has not timed out.

bit 5-2 **Unimplemented:** Read as ‘0’

Note 1: These bits apply to the PTGWHI and PTGWLO commands only.
2: This bit is only used with the PTGCTRL Step command software trigger option.

REGISTER 17-1: PTGCST: PTG CONTROL/STATUS REGISTER (CONTINUED)

bit 1-0 **PTGITM<1:0>:** PTG Input Trigger Command Operating Mode bits[1]

11 = Single level detect with Step delay is not executed on exit of command (regardless of PTGCTRL command)
10 = Single level detect with Step delay is executed on exit of command
01 = Continuous edge detect with Step delay is not executed on exit of command (regardless of PTGCTRL command)
00 = Continuous edge detect with Step delay is executed on exit of command

Note 1: These bits apply to the PTGWHI and PTGWLO commands only.

2: This bit is only used with the PTGCTRL Step command software trigger option.

REGISTER 17-2: PTGCON: PTG CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGCLK2	PTGCLK1	PTGCLK0	PTGDIV4	PTGDIV3	PTGDIV2	PTGDIV1	PTGDIV0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTGPWD3	PTGPWD2	PTGPWD1	PTGPWD0	—	PTGWDT2	PTGWDT1	PTGWDT0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **PTGCLK<2:0>:** Select PTG Module Clock Source bits
111 = CLC2
110 = CLC1
101 = PTG module clock source will be T3CLK
100 = PTG module clock source will be T2CLK
011 = PTG module clock source will be T1CLK
010 = PTG module clock source will be TAD
001 = PTG module clock source will be FOSC
000 = PTG module clock source will be FP

bit 12-8 **PTGDIV<4:0>:** PTG Module Clock Prescaler (divider) bits
11111 = Divide-by-32
11110 = Divide-by-31
•
•
•
00001 = Divide-by-2
00000 = Divide-by-1

bit 7-4 **PTGPWD<3:0>:** PTG Trigger Output Pulse-Width bits
1111 = All trigger outputs are 16 PTG clock cycles wide
1110 = All trigger outputs are 15 PTG clock cycles wide
•
•
•
0001 = All trigger outputs are 2 PTG clock cycles wide
0000 = All trigger outputs are 1 PTG clock cycle wide

bit 3 **Unimplemented:** Read as ‘0’

bit 2-0 **PTGWDT<2:0>:** Select PTG Watchdog Timer Time-out Count Value bits
111 = Watchdog Timer will time-out after 512 PTG clocks
110 = Watchdog Timer will time-out after 256 PTG clocks
101 = Watchdog Timer will time-out after 128 PTG clocks
100 = Watchdog Timer will time-out after 64 PTG clocks
011 = Watchdog Timer will time-out after 32 PTG clocks
010 = Watchdog Timer will time-out after 16 PTG clocks
001 = Watchdog Timer will time-out after 8 PTG clocks
000 = Watchdog Timer is disabled

REGISTER 17-3: PTGBTE: PTG BROADCAST TRIGGER ENABLE REGISTER[1,2]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCTS4	ADCTS3	ADCTS2	ADCTS1	IC4TSS	IC3TSS	IC2TSS	IC1TSS
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OC4CS	OC3CS	OC2CS	OC1CS	OC4TSS	OC3TSS	OC2TSS	OC1TSS
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ADCTS4:** Sample Trigger PTGO15 for ADCx bit
1 = Generates trigger when the broadcast command is executed
0 = Does not generate trigger when the broadcast command is executed

bit 14 **ADCTS3:** Sample Trigger PTGO14 for ADCx bit
1 = Generates trigger when the broadcast command is executed
0 = Does not generate trigger when the broadcast command is executed

bit 13 **ADCTS2:** Sample Trigger PTGO13 for ADCx bit
1 = Generates trigger when the broadcast command is executed
0 = Does not generate trigger when the broadcast command is executed

bit 12 **ADCTS1:** Sample Trigger PTGO12 for ADCx bit
1 = Generates trigger when the broadcast command is executed
0 = Does not generate trigger when the broadcast command is executed

bit 11 **IC4TSS:** Trigger/Synchronization Source for IC4 bit
1 = Generates trigger/synchronization when the broadcast command is executed
0 = Does not generate trigger/synchronization when the broadcast command is executed

bit 10 **IC3TSS:** Trigger/Synchronization Source for IC3 bit
1 = Generates trigger/synchronization when the broadcast command is executed
0 = Does not generate trigger/synchronization when the broadcast command is executed

bit 9 **IC2TSS:** Trigger/Synchronization Source for IC2 bit
1 = Generates trigger/synchronization when the broadcast command is executed
0 = Does not generate trigger/synchronization when the broadcast command is executed

bit 8 **IC1TSS:** Trigger/Synchronization Source for IC1 bit
1 = Generates trigger/synchronization when the broadcast command is executed
0 = Does not generate trigger/synchronization when the broadcast command is executed

bit 7 **OC4CS:** Clock Source for OC4 bit
1 = Generates clock pulse when the broadcast command is executed
0 = Does not generate clock pulse when the broadcast command is executed

bit 6 **OC3CS:** Clock Source for OC3 bit
1 = Generates clock pulse when the broadcast command is executed
0 = Does not generate clock pulse when the broadcast command is executed

bit 5 **OC2CS:** Clock Source for OC2 bit
1 = Generates clock pulse when the broadcast command is executed
0 = Does not generate clock pulse when the broadcast command is executed

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

2: This register is only used with the `PTGCTRL OPTION` = `1111` Step command.

REGISTER 17-3: PTGBTE: PTG BROADCAST TRIGGER ENABLE REGISTER[1,2] (CONTINUED)

bit 4	**OC1CS:** Clock Source for OC1 bit 1 = Generates clock pulse when the broadcast command is executed 0 = Does not generate clock pulse when the broadcast command is executed
bit 3	**OC4TSS:** Trigger/Synchronization Source for OC4 bit 1 = Generates trigger/synchronization when the broadcast command is executed 0 = Does not generate trigger/synchronization when the broadcast command is executed
bit 2	**OC3TSS:** Trigger/Synchronization Source for OC3 bit 1 = Generates trigger/synchronization when the broadcast command is executed 0 = Does not generate trigger/synchronization when the broadcast command is executed
bit 1	**OC2TSS:** Trigger/Synchronization Source for OC2 bit 1 = Generates trigger/synchronization when the broadcast command is executed 0 = Does not generate trigger/synchronization when the broadcast command is executed
bit 0	**OC1TSS:** Trigger/Synchronization Source for OC1 bit 1 = Generates trigger/synchronization when the broadcast command is executed 0 = Does not generate trigger/synchronization when the broadcast command is executed

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

2: This register is only used with the `PTGCTRL OPTION` = 1111 Step command.

REGISTER 17-4: PTGT0LIM: PTG TIMER0 LIMIT REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT0LIM<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT0LIM<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGT0LIM<15:0>:** PTG Timer0 Limit Register bits
General purpose Timer0 Limit register (effective only with a PTGT0 Step command).

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-5: PTGT1LIM: PTG TIMER1 LIMIT REGISTER[(1)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT1LIM<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT1LIM<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGT1LIM<15:0>:** PTG Timer1 Limit Register bits
General purpose Timer1 Limit register (effective only with a PTGT1 Step command).

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-6: PTGSDLIM: PTG STEP DELAY LIMIT REGISTER[1,2]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGSDLIM<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGSDLIM<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGSDLIM<15:0>:** PTG Step Delay Limit Register bits
Holds a PTG Step delay value, representing the number of additional PTG clocks, between the start of a Step command and the completion of a Step command.

Note 1: A base Step delay of one PTG clock is added to any value written to the PTGSDLIM register (Step Delay = (PTGSDLIM) + 1).
2: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-7: PTGC0LIM: PTG COUNTER 0 LIMIT REGISTER[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC0LIM<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC0LIM<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGC0LIM<15:0>:** PTG Counter 0 Limit Register bits
May be used to specify the loop count for the `PTGJMPC0` Step command or as a limit register for the General Purpose Counter 0.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-8: PTGC1LIM: PTG COUNTER 1 LIMIT REGISTER[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC1LIM<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC1LIM<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGC1LIM<15:0>:** PTG Counter 1 Limit Register bits
May be used to specify the loop count for the `PTGJMPC1` Step command or as a limit register for the General Purpose Counter 1.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-9: PTGHOLD: PTG HOLD REGISTER[1]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGHOLD<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGHOLD<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGHOLD<15:0>:** PTG General Purpose Hold Register bits
Holds user-supplied data to be copied to the PTGTxLIM, PTGCxLIM, PTGSDLIM or PTGL0 register with the `PTGCOPY` command.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-10: PTGADJ: PTG ADJUST REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGADJ<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGADJ<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGADJ<15:0>:** PTG Adjust Register bits
This register holds user-supplied data to be added to the PTGTxLIM, PTGCxLIM, PTGSDLIM or PTGL0 register with the `PTGADD` command.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-11: PTGL0: PTG LITERAL 0 REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGL0<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGL0<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PTGL0<15:0>:** PTG Literal 0 Register bits
This register holds the 6-bit value to be written to the CNVCHSEL<5:0> bits (ADCON3L<5:0>) with the `PTGCTRL` Step command.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-12: PTGQPTR: PTG STEP QUEUE POINTER REGISTER[(1)]

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	PTGQPTR<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-5 **Unimplemented:** Read as ‘0’

bit 4-0 **PTGQPTR<4:0>:** PTG Step Queue Pointer Register bits
This register points to the currently active Step command in the Step queue.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).

REGISTER 17-13: PTGQUEx: PTG STEP QUEUE REGISTER x (x = 0-15)[(1,3)]

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STEP(2x + 1)<7:0>[(2)]							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STEP(2x)<7:0>[(2)]							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **STEP(2x + 1)<7:0>:** PTG Step Queue Pointer Register bits[(2)]
A queue location for storage of the STEP(2x +1) command byte.

bit 7-0 **STEP(2x)<7:0>:** PTG Step Queue Pointer Register bits[(2)]
A queue location for storage of the STEP(2x) command byte.

Note 1: This register is read-only when the PTG module is executing Step commands (PTGEN = 1 and PTGSTRT = 1).
2: Refer to Table 17-1 for the Step command encoding.
3: The Step registers maintain their values on any type of Reset.

17.3 Step Commands and Format

TABLE 17-1: PTG STEP COMMAND FORMAT

Step Command Byte:			
STEPx<7:0>			
CMD<3:0>		OPTION<3:0>	
bit 7	bit 4	bit 3	bit 0

bit 7-4	**CMD<3:0>**	**Step Command**	**Command Description**
	`0000`	`PTGCTRL`	Execute control command as described by OPTION<3:0>
	`0001`	`PTGADD`	Add contents of PTGADJ register to target register as described by OPTION<3:0>
		`PTGCOPY`	Copy contents of PTGHOLD register to target register as described by OPTION<3:0>
	`001x`	`PTGSTRB`	Copy the value contained in CMD0:OPTION<3:0> to the CNVCHSEL<5:0> bits (ADCON3L<5:0>)
	`0100`	`PTGWHI`	Wait for a low-to-high edge input from selected PTG trigger input as described by OPTION<3:0>
	`0101`	`PTGWLO`	Wait for a high-to-low edge input from selected PTG trigger input as described by OPTION<3:0>
	`0110`	`Reserved`	Reserved
	`0111`	`PTGIRQ`	Generate individual interrupt request as described by OPTION<3:0>
	`100x`	`PTGTRIG`	Generate individual trigger output as described by <<CMD0>:OPTION<3:0>>
	`101x`	`PTGJMP`	Copy the value indicated in <<CMD0>:OPTION<3:0>> to the PTG Queue Pointer (PTGQPTR) and jump to that Step queue
	`110x`	`PTGJMPC0`	PTGC0 = PTGC0LIM: Increment the PTG Queue Pointer (PTGQPTR)
			PTGC0 ≠ PTGC0LIM: Increment PTG Counter 0 (PTGC0) and copy the value indicated in <<CMD0>:OPTION<3:0>> to the PTG Queue Pointer (PTGQPTR) and jump to that Step queue
	`111x`	`PTGJMPC1`	PTGC1 = PTGC1LIM: Increment the PTG Queue Pointer (PTGQPTR)
			PTGC1 ≠ PTGC1LIM: Increment PTG Counter 1 (PTGC1) and copy the value indicated in <<CMD0>:OPTION<3:0>> to the PTG Queue Pointer (PTGQPTR) and jump to that Step queue

Note 1: All reserved commands or options will execute but have no effect (i.e., execute as a `NOP` instruction).

2: Refer to Table 17-2 for the trigger output descriptions.

TABLE 17-1: PTG STEP COMMAND FORMAT (CONTINUED)

bit 3-0	Step Command	OPTION<3:0>	Option Description
	PTGCTRL[1]	0000	Reserved
		0001	Reserved
		0010	Disable PTG Step Delay Timer (PTGSD)
		0011	Reserved
		0100	Reserved
		0101	Reserved
		0110	Enable PTG Step Delay Timer (PTGSD)
		0111	Reserved
		1000	Start and wait for the PTG Timer0 to match the PTG Timer0 Limit register
		1001	Start and wait for the PTG Timer1 to match the PTG Timer1 Limit register
		1010	Reserved
		1011	Wait for software trigger bit transition from low-to-high before continuing (PTGSWT = 0 to 1)
		1100	Copy contents of the PTG Counter 0 register to the CNVCHSEL<5:0> bits (ADCON3L<5:0>)
		1101	Copy contents of the PTG Counter 1 register to the CNVCHSEL<5:0> bits (ADCON3L<5:0>)
		1110	Copy contents of the PTG Literal 0 register to the CNVCHSEL<5:0> bits (ADCON3L<5:0>)
		1111	Generate the triggers indicated in the PTG Broadcast Trigger Enable register (PTGBTE)
	PTGADD[1]	0000	Add contents of PTGADJ register to the PTG Counter 0 Limit register (PTGC0LIM)
		0001	Add contents of PTGADJ register to the PTG Counter 1 Limit register (PTGC1LIM)
		0010	Add contents of PTGADJ register to the PTG Timer0 Limit register (PTGT0LIM)
		0011	Add contents of PTGADJ register to the PTG Timer1 Limit register (PTGT1LIM)
		0100	Add contents of PTGADJ register to the PTG Step Delay Limit register (PTGSDLIM)
		0101	Add contents of PTGADJ register to the PTG Literal 0 register (PTGL0)
		0110	Reserved
		0111	Reserved
	PTGCOPY[1]	1000	Copy contents of PTGHOLD register to the PTG Counter 0 Limit register (PTGC0LIM)
		1001	Copy contents of PTGHOLD register to the PTG Counter 1 Limit register (PTGC1LIM)
		1010	Copy contents of PTGHOLD register to the PTG Timer0 Limit register (PTGT0LIM)
		1011	Copy contents of PTGHOLD register to the PTG Timer1 Limit register (PTGT1LIM)
		1100	Copy contents of PTGHOLD register to the PTG Step Delay Limit register (PTGSDLIM)
		1101	Copy contents of PTGHOLD register to the PTG Literal 0 register (PTGL0)
		1110	Reserved
		1111	Reserved

Note 1: All reserved commands or options will execute but have no effect (i.e., execute as a NOP instruction).

2: Refer to Table 17-2 for the trigger output descriptions.

TABLE 17-1: PTG STEP COMMAND FORMAT (CONTINUED)

bit 3-0	Step Command	OPTION<3:0>	Option Description
	PTGWHI[1] or PTGWLO[1]	0000	PWM Special Event Trigger
		0001	PWM master time base synchronization output
		0010	PWM1 interrupt
		0011	PWM2 interrupt
		0100	PWM3 interrupt
		0101	PWM4 interrupt
		0110	PWM5 interrupt
		0111	OC1 trigger event
		1000	OC2 trigger event
		1001	IC1 trigger event
		1010	CMP1 trigger event
		1011	CMP2 trigger event
		1100	CMP3 trigger event
		1101	CMP4 trigger event
		1110	ADC conversion done interrupt
		1111	INT2 external interrupt
	PTGIRQ[1]	0000	Generate PTG Interrupt 0
		0001	Generate PTG Interrupt 1
		0010	Generate PTG Interrupt 2
		0011	Generate PTG Interrupt 3
		0100	Reserved
		• • •	• • •
		1111	Reserved
	PTGTRIG[2]	00000	PTGO0
		00001	PTGO1
		• • •	• • •
		11110	PTGO30
		11111	PTGO31

Note 1: All reserved commands or options will execute but have no effect (i.e., execute as a NOP instruction).

2: Refer to Table 17-2 for the trigger output descriptions.

TABLE 17-2: PTG OUTPUT DESCRIPTIONS

PTG Output Number	PTG Output Description
PTGO0	Trigger/synchronization source for OC1
PTGO1	Trigger/synchronization source for OC2
PTGO2	Trigger/synchronization source for OC3
PTGO3	Trigger/synchronization source for OC4
PTGO4	Clock source for OC1
PTGO5	Clock source for OC2
PTGO6	Clock source for OC3
PTGO7	Clock source for OC4
PTGO8	Trigger/synchronization source for IC1
PTGO9	Trigger/synchronization source for IC2
PTGO10	Trigger/synchronization source for IC3
PTGO11	Trigger/synchronization source for IC4
PTGO12	Sample trigger for ADC
PTGO13	Reserved
PTGO14	Reserved
PTGO15	Reserved
PTGO16	PWM time base synchronous source for PWM3
PTGO17	PWM time base synchronous source for PWM4
PTGO18	PWM time base synchronous source for PWM5
PTGO19	PWM time base synchronous source for PWM6
PTGO20	Reserved
PTGO21	Reserved
PTGO22	Reserved
PTGO23	Reserved
PTGO24	Reserved
PTGO25	Reserved
PTGO26	CLC1 input
PTGO27	CLC2 input
PTGO28	CLC3 input
PTGO29	CLC4 input
PTGO30	PTG output to PPS input selection, RPI6
PTGO31	PTG output to PPS input selection, RPI7

18.0 SERIAL PERIPHERAL INTERFACE (SPI)

> **Note:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Serial Peripheral Interface (SPI) with Audio Codec Support"** (DS70005136) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The Serial Peripheral Interface (SPI) module is a synchronous serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D Converters, etc. The SPI module is compatible with the Motorola® SPI and SIOP interfaces. All devices in the dsPIC33EPXXXGS70X/80X family include three SPI modules.

The module supports operation in two buffer modes. In Standard mode, data is shifted through a single serial buffer. In Enhanced Buffer mode, data is shifted through a FIFO buffer. The FIFO level depends on the configured mode.

Variable length data can be transmitted and received, from 2 to 32 bits.

> **Note:** Do not perform Read-Modify-Write operations (such as bit-oriented instructions) on the SPIxBUF register in either Standard or Enhanced Buffer mode.

The module also supports a basic framed SPI protocol while operating in either Master or Slave mode. A total of four framed SPI configurations are supported.

SPI3 also supports Audio modes. Four different Audio modes are available.

- I^{2}S
- Left Justified
- Right Justified
- PCM/DSP

In each of these modes, the serial clock is free-running and audio data is always transferred.

If an audio protocol data transfer takes place between two devices, then usually one device is the master and the other is the slave. However, audio data can be transferred between two slaves. Because the audio protocols require free-running clocks, the master can be a third party controller. In either case, the master generates two free-running clocks: SCKx and LRC (Left, Right Channel Clock/$\overline{\text{SSx}}$/FSYNC).

The SPI serial interface consists of four pins:

- SDIx: Serial Data Input
- SDOx: Serial Data Output
- SCKx: Shift Clock Input or Output
- $\overline{\text{SSx}}$: Active-Low Slave Select or Frame Synchronization I/O Pulse

The SPI module can be configured to operate using 2, 3 or 4 pins. In the 3-pin mode, $\overline{\text{SSx}}$ is not used. In the 2-pin mode, both SDOx and $\overline{\text{SSx}}$ are not used.

The SPI module has the ability to generate three interrupts, reflecting the events that occur during the data communication. The following types of interrupts can be generated:

1. Receive interrupts are signalled by SPIxRXIF. This event occurs when:
   - RX watermark interrupt
   - SPIROV = 1
   - SPIRBF = 1
   - SPIRBE = 1

   provided the respective mask bits are enabled in SPIxIMSKL/H.
2. Transmit interrupts are signalled by SPIxTXIF. This event occurs when:
   - TX watermark interrupt
   - SPITUR = 1
   - SPITBF = 1
   - SPITBE = 1

   provided the respective mask bits are enabled in SPIxIMSKL/H.
3. General interrupts are signalled by SPIxIF. This event occurs when
   - FRMERR = 1
   - SPIBUSY = 1
   - SRMT = 1

   provided the respective mask bits are enabled in SPIxIMSKL/H.

Block diagrams of the module in Standard and Enhanced modes are shown in Figure 18-1 and Figure 18-2.

> **Note:** In this section, the SPI modules are referred to together as SPIx, or separately as SPI1, SPI2 or SPI3. Special Function Registers will follow a similar notation. For example, SPIxCON1 and SPIxCON2 refer to the control registers for any of the three SPI modules.

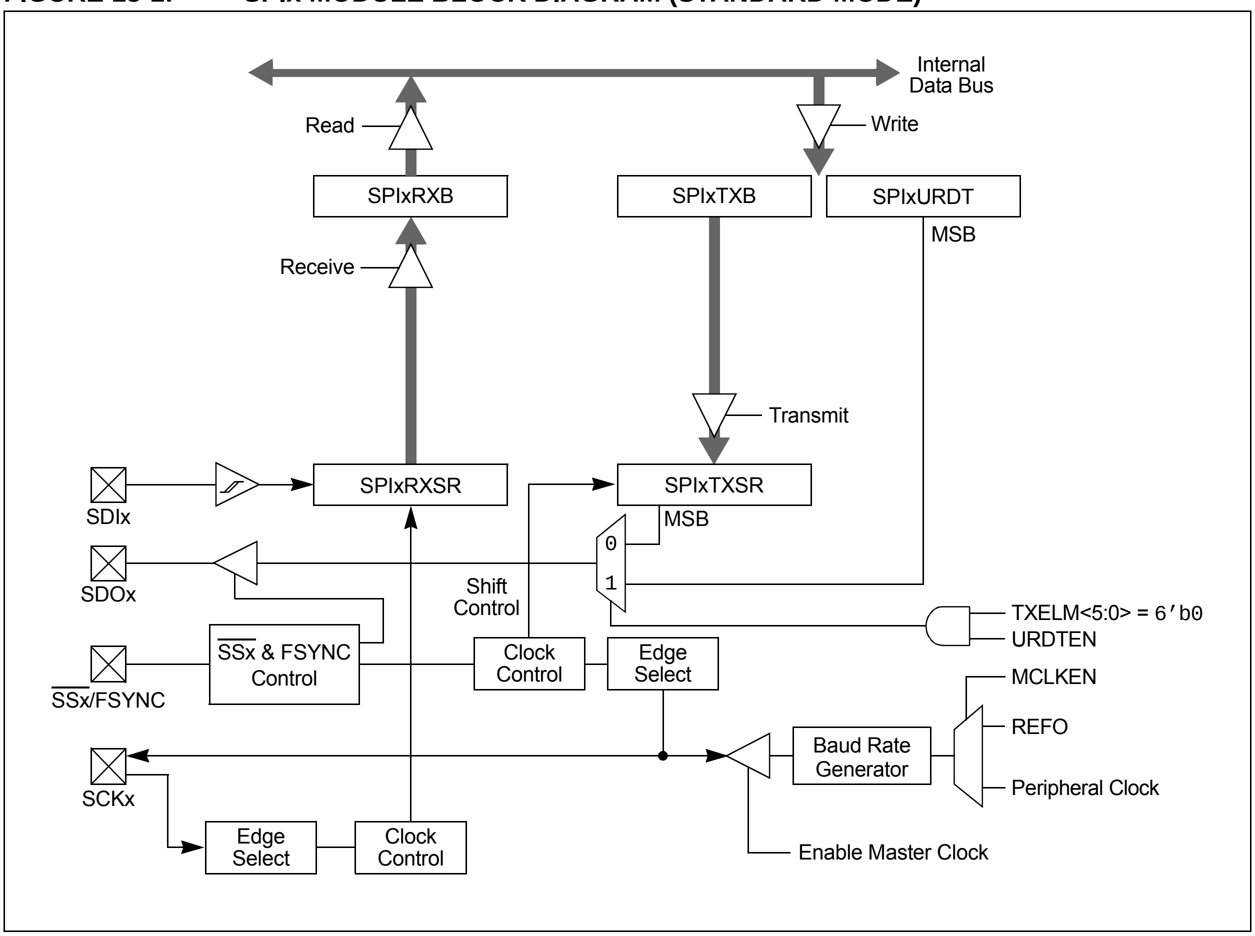
To set up the SPIx module for the Standard Master mode of operation:

1. If using interrupts:
   a) Clear the interrupt flag bits in the respective IFSx register.
   b) Set the interrupt enable bits in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
2. Write the desired settings to the SPIxCON1L and SPIxCON1H registers with the MSTEN bit (SPIxCON1L<5>) = 1.
3. Clear the SPIROV bit (SPIxSTATL<6>).
4. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
5. Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

To set up the SPIx module for the Standard Slave mode of operation:

1. Clear the SPIxBUF registers.
2. If using interrupts:
   a) Clear the SPIxBUFL and SPIxBUFH registers.
   b) Set the interrupt enable bits in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
3. Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with the MSTEN bit (SPIxCON1L<5>) = 0.
4. Clear the SMP bit.
5. If the CKE bit (SPIxCON1L<8>) is set, then the SSEN bit (SPIxCON1L<7>) must be set to enable the $\overline{\text{SSx}}$ pin.
6. Clear the SPIROV bit (SPIxSTATL<6>).
7. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).

FIGURE 18-1: SPIx MODULE BLOCK DIAGRAM (STANDARD MODE)



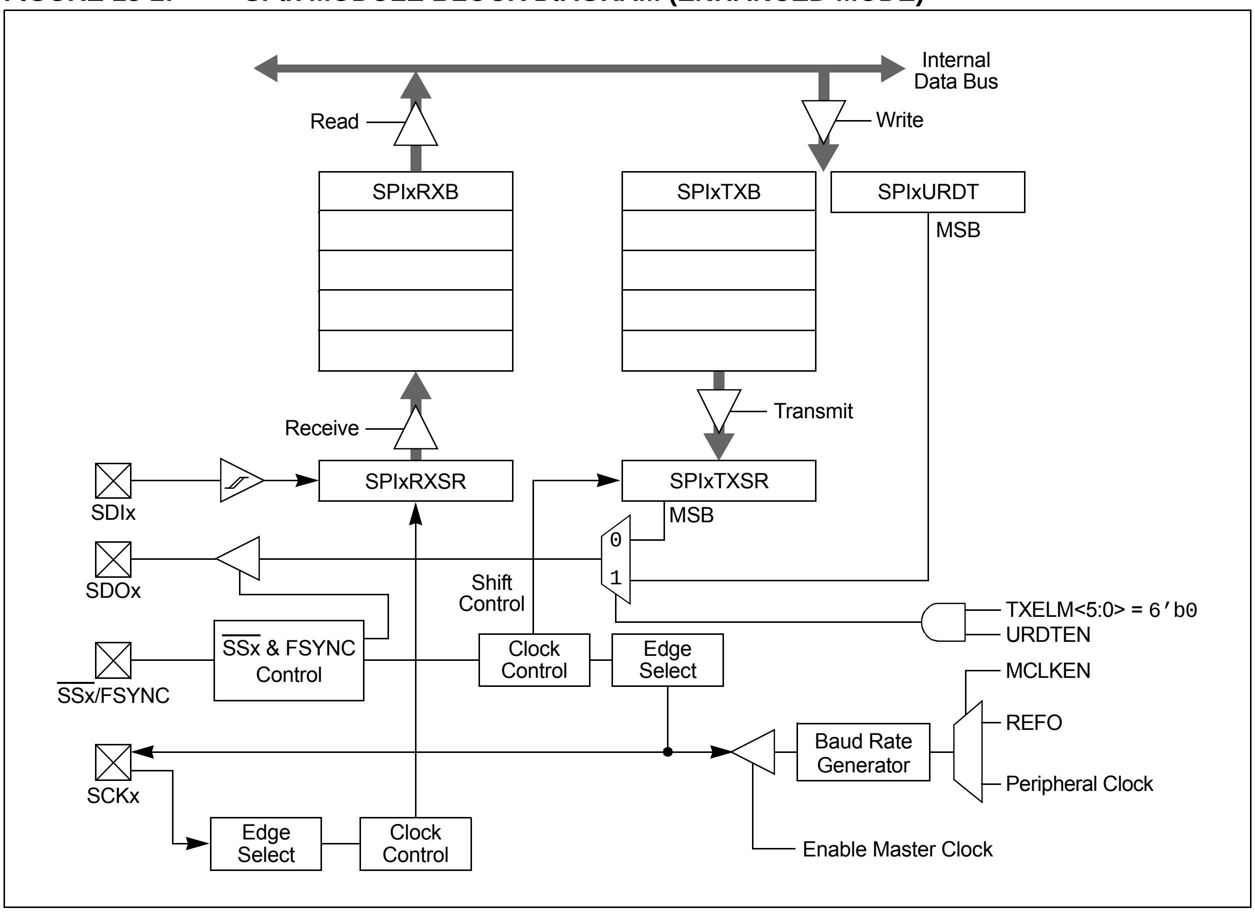
To set up the SPIx module for the Enhanced Buffer Master mode of operation:

1. If using interrupts:
   a) Clear the interrupt flag bits in the respective IFSx register.
   b) Set the interrupt enable bits in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx register.
2. Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with MSTEN (SPIxCON1L<5>) = 1.
3. Clear the SPIROV bit (SPIxSTATL<6>).
4. Select Enhanced Buffer mode by setting the ENHBUF bit (SPIxCON1L<0>).
5. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
6. Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

To set up the SPIx module for the Enhanced Buffer Slave mode of operation:

1. Clear the SPIxBUFL and SPIxBUFH registers.
2. If using interrupts:
   a) Clear the interrupt flag bits in the respective IFSx register.
   b) Set the interrupt enable bits in the respective IECx register.
   c) Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
3. Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with the MSTEN bit (SPIxCON1L<5>) = 0.
4. Clear the SMP bit.
5. If the CKE bit is set, then the SSEN bit must be set, thus enabling the $\overline{\text{SSx}}$ pin.
6. Clear the SPIROV bit (SPIxSTATL<6>).
7. Select Enhanced Buffer mode by setting the ENHBUF bit (SPIxCON1L<0>).
8. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).

FIGURE 18-2: SPIx MODULE BLOCK DIAGRAM (ENHANCED MODE)



To set up the SPIx module for Audio mode:

1. Clear the SPIxBUFL and SPIxBUFH registers.
2. If using interrupts:
   a) Clear the interrupt flag bits in the respective IFSx register.
   b) Set the interrupt enable bits in the respective IECx register.
   a) Write the SPIxIP bits in the respective IPCx register to set the interrupt priority.
3. Write the desired settings to the SPIxCON1L, SPIxCON1H and SPIxCON2L registers with AUDEN (SPIxCON1H<15>) = 1.
4. Clear the SPIROV bit (SPIxSTATL<6>).
5. Enable SPIx operation by setting the SPIEN bit (SPIxCON1L<15>).
6. Write the data to be transmitted to the SPIxBUFL and SPIxBUFH registers. Transmission (and reception) will start as soon as data is written to the SPIxBUFL and SPIxBUFH registers.

REGISTER 18-1: SPIxCON1L: SPIx CONTROL REGISTER 1 LOW

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPIEN	—	SPISIDL	DISSDO	MODE32[1,4]	MODE16[1,4]	SMP	CKE[1]
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN[2]	CKP	MSTEN	DISSDI	DISSCK	MCLKEN[3]	SPIFE	ENHBUF
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **SPIEN:** SPIx On bit
- 1 = Enables module
- 0 = Turns off and resets module, disables clocks, disables interrupt event generation, allows SFR modifications

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **SPISIDL:** SPIx Stop in Idle Mode bit
- 1 = Halts in CPU Idle mode
- 0 = Continues to operate in CPU Idle mode

bit 12 **DISSDO:** Disable SDOx Output Port bit
- 1 = SDOx pin is not used by the module; pin is controlled by port function
- 0 = SDOx pin is controlled by the module

bit 11-10 **MODE32** and **MODE16:** Serial Word Length Select bits[1,4]

MODE32	MODE16	AUDEN	Communication
1	x	0	32-Bit
0	1		16-Bit
0	0		8-Bit
1	1	1	24-Bit Data, 32-Bit FIFO, 32-Bit Channel/64-Bit Frame
1	0		32-Bit Data, 32-Bit FIFO, 32-Bit Channel/64-Bit Frame
0	1		16-Bit Data, 16-Bit FIFO, 32-Bit Channel/64-Bit Frame
0	0		16-Bit FIFO, 16-Bit Channel/32-Bit Frame

Note 1: When AUDEN (SPIxCON1H<15>) = 1, this module functions as if CKE = 0, regardless of its actual value.
2: When FRMEN = 1, SSEN is not used.
3: MCLKEN can only be written when the SPIEN bit = 0.
4: This channel is not meaningful for DSP/PCM mode as LRC follows FRMSYPW.

REGISTER 18-1: SPIxCON1L: SPIx CONTROL REGISTER 1 LOW (CONTINUED)

bit 9 **SMP:** SPIx Data Input Sample Phase bit

Master Mode:
1 = Input data is sampled at the end of data output time
0 = Input data is sampled at the middle of data output time

Slave Mode:
Input data is always sampled at the middle of data output time, regardless of the SMP setting.

bit 8 **CKE:** SPIx Clock Edge Select bit[1]

1 = Transmit happens on transition from active clock state to Idle clock state
0 = Transmit happens on transition from Idle clock state to active clock state

bit 7 **SSEN:** Slave Select Enable bit (Slave mode)[2]

1 = $\overline{\text{SSx}}$ pin is used by the macro in Slave mode; $\overline{\text{SSx}}$ pin is used as the slave select input
0 = $\overline{\text{SSx}}$ pin is not used by the macro ($\overline{\text{SSx}}$ pin will be controlled by the port I/O)

bit 6 **CKP:** Clock Polarity Select bit

1 = Idle state for clock is a high level; active state is a low level
0 = Idle state for clock is a low level; active state is a high level

bit 5 **MSTEN:** Master Mode Enable bit

1 = Master mode
0 = Slave mode

bit 4 **DISSDI:** Disable SDIx Input Port bit

1 = SDIx pin is not used by the module; pin is controlled by port function
0 = SDIx pin is controlled by the module

bit 3 **DISSCK:** Disable SCKx Output Port bit

1 = SCKx pin is not used by the module; pin is controlled by port function
0 = SCKx pin is controlled by the module

bit 2 **MCLKEN:** Master Clock Enable bit[3]

1 = REFO is used by the Baud Rate Generator (BRG)
0 = Peripheral clock is used by the BRG

bit 1 **SPIFE:** Frame Sync Pulse Edge Select bit

1 = Frame Sync pulse (Idle-to-active edge) coincides with the first bit clock
0 = Frame Sync pulse (Idle-to-active edge) precedes the first bit clock

bit 0 **ENHBUF:** Enhanced Buffer Enable bit

1 = Enhanced Buffer mode is enabled
0 = Enhanced Buffer mode is disabled

Note 1: When AUDEN (SPIxCON1H<15>) = 1, this module functions as if CKE = 0, regardless of its actual value.
2: When FRMEN = 1, SSEN is not used.
3: MCLKEN can only be written when the SPIEN bit = 0.
4: This channel is not meaningful for DSP/PCM mode as LRC follows FRMSYPW.

REGISTER 18-2: SPIxCON1H: SPIx CONTROL REGISTER 1 HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AUDEN(1)	SPISGNEXT	IGNROV	IGNTUR	AUDMONO(2)	URDTEN(3)	AUDMOD1(4)	AUDMOD0(4)
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FRMEN	FRMSYNC	FRMPOL	MSSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **AUDEN:** Audio Codec Support Enable bit(1)
1 = Audio protocol is enabled; MSTEN controls the direction of both SCKx and frame (a.k.a. LRC), and this module functions as if FRMEN = 1, FRMSYNC = MSTEN, FRMCNT<2:0> = 001 and SMP = 0, regardless of their actual values
0 = Audio protocol is disabled

bit 14 **SPISGNEXT:** SPIx Sign-Extend RX FIFO Read Data Enable bit
1 = Data from RX FIFO is sign-extended
0 = Data from RX FIFO is not sign-extended

bit 13 **IGNROV:** Ignore Receive Overflow bit
1 = A Receive Overflow (ROV) is NOT a critical error; during ROV, data in the FIFO is not overwritten by the receive data
0 = A ROV is a critical error that stops SPI operation

bit 12 **IGNTUR:** Ignore Transmit Underrun bit
1 = A Transmit Underrun (TUR) is NOT a critical error and data indicated by URDTEN is transmitted until the SPIxTXB is not empty
0 = A TUR is a critical error that stops SPI operation

bit 11 **AUDMONO:** Audio Data Format Transmit bit(2)
1 = Audio data is mono (i.e., each data word is transmitted on both left and right channels)
0 = Audio data is stereo

bit 10 **URDTEN:** Transmit Underrun Data Enable bit(3)
1 = Transmits data out of SPIxURDT register during Transmit Underrun conditions
0 = Transmits the last received data during Transmit Underrun conditions

bit 9-8 **AUDMOD<1:0>:** Audio Protocol Mode Selection bits(4)
11 = PCM/DSP mode
10 = Right Justified mode: This module functions as if SPIFE = 1, regardless of its actual value
01 = Left Justified mode: This module functions as if SPIFE = 1, regardless of its actual value
00 = I^2S mode: This module functions as if SPIFE = 0, regardless of its actual value

bit 7 **FRMEN:** Framed SPIx Support bit
1 = Framed SPIx support is enabled ($\overline{\text{SSx}}$ pin is used as the FSYNC input/output)
0 = Framed SPIx support is disabled

Note 1: AUDEN can only be written when the SPIEN bit = 0.
2: AUDMONO can only be written when the SPIEN bit = 0 and is only valid for AUDEN = 1.
3: URDTEN is only valid when IGNTUR = 1.
4: The AUDMOD<1:0> bits can only be written when the SPIEN bit = 0 and are only valid when AUDEN = 1. When NOT in PCM/DSP mode, this module functions as if FRMSYPW = 1, regardless of its actual value.

REGISTER 18-2: SPIxCON1H: SPIx CONTROL REGISTER 1 HIGH (CONTINUED)

bit 6 **FRMSYNC:** Frame Sync Pulse Direction Control bit
1 = Frame Sync pulse input (slave)
0 = Frame Sync pulse output (master)

bit 5 **FRMPOL:** Frame Sync/Slave Select Polarity bit
1 = Frame Sync pulse/slave select is active-high
0 = Frame Sync pulse/slave select is active-low

bit 4 **MSSEN:** Master Mode Slave Select Enable bit
1 = SPIx slave select support is enabled with polarity determined by FRMPOL ($\overline{\text{SSx}}$ pin is automatically driven during transmission in Master mode)
0 = Slave select SPIx support is disabled ($\overline{\text{SSx}}$ pin will be controlled by port I/O)

bit 3 **FRMSYPW:** Frame Sync Pulse-Width bit
1 = Frame Sync pulse is one serial word length wide (as defined by MODE<32,16>/WLENGTH<4:0>)
0 = Frame Sync pulse is one clock (SCK) wide

bit 2-0 **FRMCNT<2:0>:** Frame Sync Pulse Counter bits
Controls the number of serial words transmitted per Sync pulse.
111 = Reserved
110 = Reserved
101 = Generates a Frame Sync pulse on every 32 serial words
100 = Generates a Frame Sync pulse on every 16 serial words
011 = Generates a Frame Sync pulse on every 8 serial words
010 = Generates a Frame Sync pulse on every 4 serial words
001 = Generates a Frame Sync pulse on every 2 serial words (value used by audio protocols)
000 = Generates a Frame Sync pulse on each serial word

Note 1: AUDEN can only be written when the SPIEN bit = 0.
2: AUDMONO can only be written when the SPIEN bit = 0 and is only valid for AUDEN = 1.
3: URDTEN is only valid when IGNTUR = 1.
4: The AUDMOD<1:0> bits can only be written when the SPIEN bit = 0 and are only valid when AUDEN = 1. When NOT in PCM/DSP mode, this module functions as if FRMSYPW = 1, regardless of its actual value.

REGISTER 18-3: SPIxCON2L: SPIx CONTROL REGISTER 2 LOW

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	WLENGTH<4:0>(1,2)				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-5 **Unimplemented:** Read as ‘0’

bit 4-0 **WLENGTH<4:0>:** Variable Word Length bits(1,2)

11111 = 32-bit data
11110 = 31-bit data
11101 = 30-bit data
11100 = 29-bit data
11011 = 28-bit data
11010 = 27-bit data
11001 = 26-bit data
11000 = 25-bit data
10111 = 24-bit data
10110 = 23-bit data
10101 = 22-bit data
10100 = 21-bit data
10011 = 20-bit data
10010 = 19-bit data
10001 = 18-bit data
10000 = 17-bit data
01111 = 16-bit data
01110 = 15-bit data
01101 = 14-bit data
01100 = 13-bit data
01011 = 12-bit data
01010 = 11-bit data
01001 = 10-bit data
01000 = 9-bit data
00111 = 8-bit data
00110 = 7-bit data
00101 = 6-bit data
00100 = 5-bit data
00011 = 4-bit data
00010 = 3-bit data
00001 = 2-bit data
00000 = See MODE<32,16> bits in SPIxCON1L<11:10>

Note 1: These bits are effective when AUDEN = 0 only.

2: Varying the length by changing these bits does not affect the depth of the TX/RX FIFO.

REGISTER 18-4: SPIxSTATL: SPIx STATUS REGISTER LOW

U-0	U-0	U-0	R/C-0, HS	R-0, HSC	U-0	U-0	R-0, HSC
—	—	—	FRMERR	SPIBUSY	—	—	SPITUR[(1)]
bit 15							bit 8

R-0, HSC	R/C-0, HS	R-1, HSC	U-0	R-1, HSC	U-0	R-0, HSC	R-0, HSC
SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF
bit 7							bit 0

Legend:	C = Clearable bit	U = Unimplemented, read as '0'	
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	HS = Hardware Settable bit

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **FRMERR**: SPIx Frame Error Status bit
1 = Frame error is detected
0 = No frame error is detected

bit 11 **SPIBUSY:** SPIx Activity Status bit
1 = Module is currently busy with some transactions
0 = No ongoing transactions (at time of read)

bit 10-9 **Unimplemented:** Read as '0'

bit 8 **SPITUR:** SPIx Transmit Underrun Status bit[(1)]
1 = Transmit buffer has encountered a Transmit Underrun condition
0 = Transmit buffer does not have a Transmit Underrun condition

bit 7 **SRMT:** Shift Register Empty Status bit
1 = No current or pending transactions (i.e., neither SPIxTXB or SPIxTXSR contains data to transmit)
0 = Current or pending transactions

bit 6 **SPIROV:** SPIx Receive Overflow Status bit
1 = A new byte/half-word/word has been completely received when the SPIxRXB was full
0 = No overflow

bit 5 **SPIRBE:** SPIx RX Buffer Empty Status bit
1 = RX buffer is empty
0 = RX buffer is not empty
Standard Buffer mode:
Automatically set in hardware when SPIxBUF is read from, reading SPIxRXB. Automatically cleared in hardware when SPIx transfers data from SPIxRXSR to SPIxRXB.
Enhanced Buffer mode:
Indicates RXELM<5:0> = 000000.

bit 4 **Unimplemented:** Read as '0'

Note 1: SPITUR is cleared when SPIEN = 0. When IGNTUR = 1, SPITUR provides dynamic status of the Transmit Underrun condition, but does not stop RX/TX operation and does not need to be cleared by software.

REGISTER 18-4: SPIxSTATL: SPIx STATUS REGISTER LOW (CONTINUED)

bit 3 **SPITBE:** SPIx Transmit Buffer Empty Status bit
1 = SPIxTXB is empty
0 = SPIxTXB is not empty
Standard Buffer mode:
Automatically set in hardware when SPIx transfers data from SPIxTXB to SPIxTXSR. Automatically cleared in hardware when SPIxBUF is written, loading SPIxTXB.
Enhanced Buffer mode:
Indicates TXELM<5:0> = 000000.

bit 2 **Unimplemented:** Read as '0'

bit 1 **SPITBF:** SPIx Transmit Buffer Full Status bit
1 = SPIxTXB is full
0 = SPIxTXB not full
Standard Buffer mode:
Automatically set in hardware when SPIxBUF is written, loading SPIxTXB. Automatically cleared in hardware when SPIx transfers data from SPIxTXB to SPIxTXSR.
Enhanced Buffer mode:
Indicates TXELM<5:0> = 111111.

bit 0 **SPIRBF:** SPIx Receive Buffer Full Status bit
1 = SPIxRXB is full
0 = SPIxRXB is not full
Standard Buffer mode:
Automatically set in hardware when SPIx transfers data from SPIxRXSR to SPIxRXB. Automatically cleared in hardware when SPIxBUF is read from, reading SPIxRXB.
Enhanced Buffer mode:
Indicates RXELM<5:0> = 111111.

Note 1: SPITUR is cleared when SPIEN = 0. When IGNTUR = 1, SPITUR provides dynamic status of the Transmit Underrun condition, but does not stop RX/TX operation and does not need to be cleared by software.

REGISTER 18-5: SPIxSTATH: SPIx STATUS REGISTER HIGH

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	RXELM5[3]	RXELM4[2]	RXELM3[1]	RXELM2	RXELM1	RXELM0
bit 15							bit 8

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	TXELM5[3]	TXELM4[2]	TXELM3[1]	TXELM2	TXELM1	TXELM0
bit 7							bit 0

Legend:	HSC = Hardware Settable/Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as '0'

bit 13-8 **RXELM<5:0>:** Receive Buffer Element Count bits (valid in Enhanced Buffer mode)[1,2,3]

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **TXELM<5:0>:** Transmit Buffer Element Count bits (valid in Enhanced Buffer mode)[1,2,3]

Note 1: RXELM3 and TXELM3 bits are only present when FIFODEPTH = 8 or higher.

2: RXELM4 and TXELM4 bits are only present when FIFODEPTH = 16 or higher.

3: RXELM5 and TXELM5 bits are only present when FIFODEPTH = 32.

REGISTER 18-6: SPIxIMSKL: SPIx INTERRUPT MASK REGISTER LOW

U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	—	FRMERREN	BUSYEN	—	—	SPITUREN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0
SRMTEN	SPIROVEN	SPIRBEN	—	SPITBEN	—	SPITBFEN	SPIRBFEN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12 **FRMERREN:** Enable Interrupt Events via FRMERR bit
1 = Frame error generates an interrupt event
0 = Frame error does not generate an interrupt event

bit 11 **BUSYEN:** Enable Interrupt Events via SPIBUSY bit
1 = SPIBUSY generates an interrupt event
0 = SPIBUSY does not generate an interrupt event

bit 10-9 **Unimplemented:** Read as ‘0’

bit 8 **SPITUREN:** Enable Interrupt Events via SPITUR bit
1 = Transmit Underrun (TUR) generates an interrupt event
0 = Transmit Underrun does not generate an interrupt event

bit 7 **SRMTEN:** Enable Interrupt Events via SRMT bit
1 = Shift Register Empty (SRMT) generates interrupt events
0 = Shift Register Empty does not generate interrupt events

bit 6 **SPIROVEN:** Enable Interrupt Events via SPIROV bit
1 = SPIx Receive Overflow (ROV) generates an interrupt event
0 = SPIx Receive Overflow does not generate an interrupt event

bit 5 **SPIRBEN:** Enable Interrupt Events via SPIRBE bit
1 = SPIx RX buffer empty generates an interrupt event
0 = SPIx RX buffer empty does not generate an interrupt event

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **SPITBEN:** Enable Interrupt Events via SPITBE bit
1 = SPIx transmit buffer empty generates an interrupt event
0 = SPIx transmit buffer empty does not generate an interrupt event

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **SPITBFEN:** Enable Interrupt Events via SPITBF bit
1 = SPIx transmit buffer full generates an interrupt event
0 = SPIx transmit buffer full does not generate an interrupt event

bit 0 **SPIRBFEN:** Enable Interrupt Events via SPIRBF bit
1 = SPIx receive buffer full generates an interrupt event
0 = SPIx receive buffer full does not generate an interrupt event

REGISTER 18-7: SPIxIMSKH: SPIx INTERRUPT MASK REGISTER HIGH

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RXWIEN	—	RXMSK5[1]	RXMSK4[1,4]	RXMSK3[1,3]	RXMSK2[1,2]	RXMSK1[1]	RXMSK0[1]
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TXWIEN	—	TXMSK5[1]	TXMSK4[1,4]	TXMSK3[1,3]	TXMSK2[1,2]	TXMSK1[1]	TXMSK0[1]
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **RXWIEN:** Receive Watermark Interrupt Enable bit
1 = Triggers receive buffer element watermark interrupt when RXMSK<5:0> ≤ RXELM<5:0>
0 = Disables receive buffer element watermark interrupt

bit 14 **Unimplemented:** Read as ‘0’

bit 13-8 **RXMSK<5:0>:** RX Buffer Mask bits[1,2,3,4]
RX mask bits; used in conjunction with the RXWIEN bit.

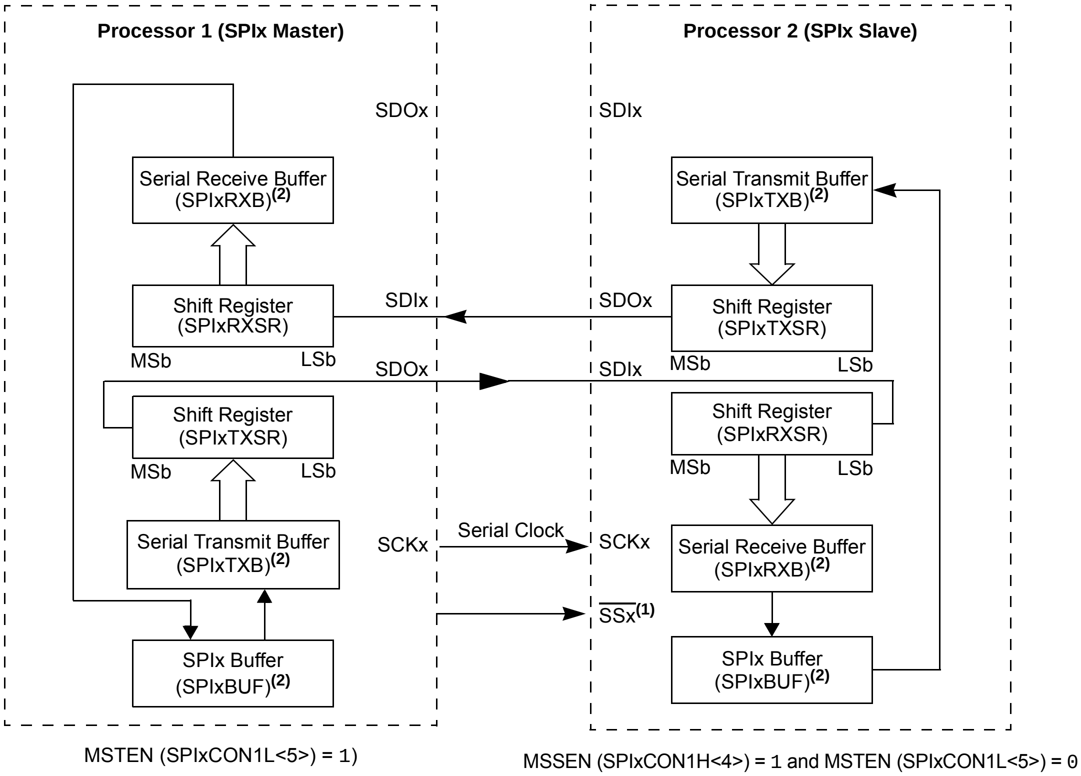
bit 7 **TXWIEN:** Transmit Watermark Interrupt Enable bit
1 = Triggers transmit buffer element watermark interrupt when TXMSK<5:0> = TXELM<5:0>
0 = Disables transmit buffer element watermark interrupt

bit 6 **Unimplemented:** Read as ‘0’

bit 5-0 **TXMSK<5:0>:** TX Buffer Mask bits[1,2,3,4]
TX mask bits; used in conjunction with the TXWIEN bit.

Note 1: Mask values higher than FIFODEPTH are not valid. The module will not trigger a match for any value in this case.
2: RXMSK2 and TXMSK2 bits are only present when FIFODEPTH = 8 or higher.
3: RXMSK3 and TXMSK3 bits are only present when FIFODEPTH = 16 or higher.
4: RXMSK4 and TXMSK4 bits are only present when FIFODEPTH = 32.

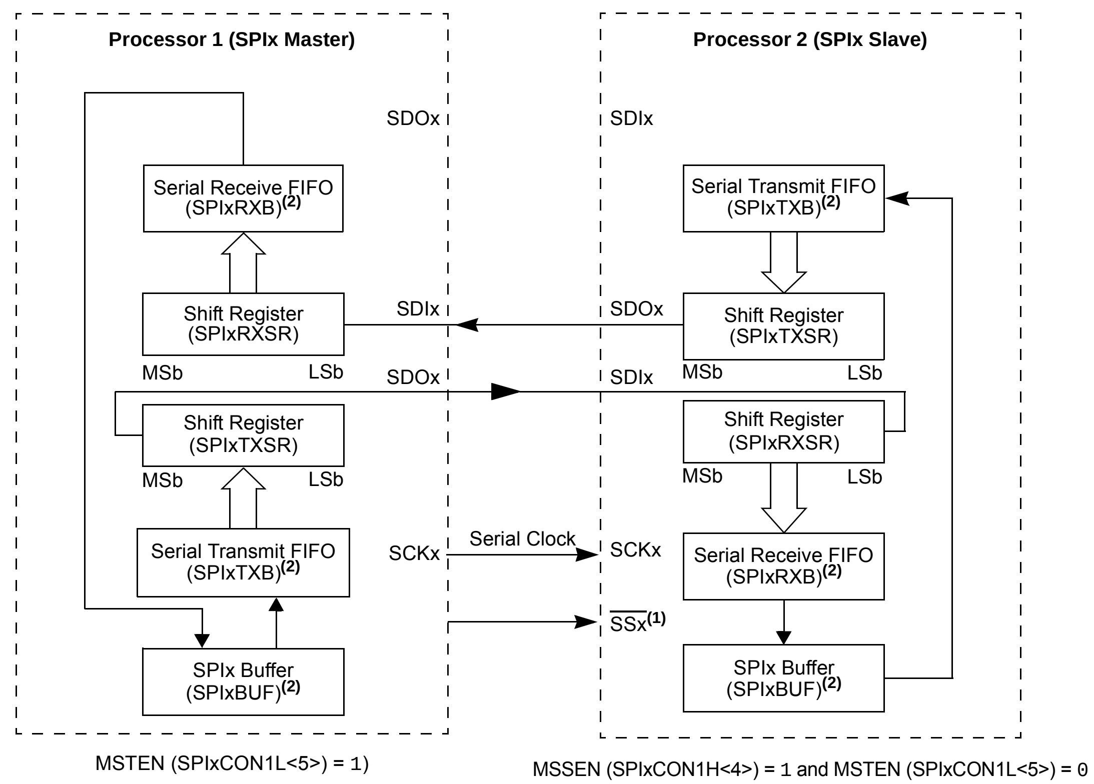
FIGURE 18-3: SPIx MASTER/SLAVE CONNECTION (STANDARD MODE)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.

2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory-mapped to SPIxBUF.

FIGURE 18-4: SPIx MASTER/SLAVE CONNECTION (ENHANCED BUFFER MODES)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.

2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory-mapped to SPIxBUF.

FIGURE 18-5: SPIx MASTER, FRAME MASTER CONNECTION DIAGRAM

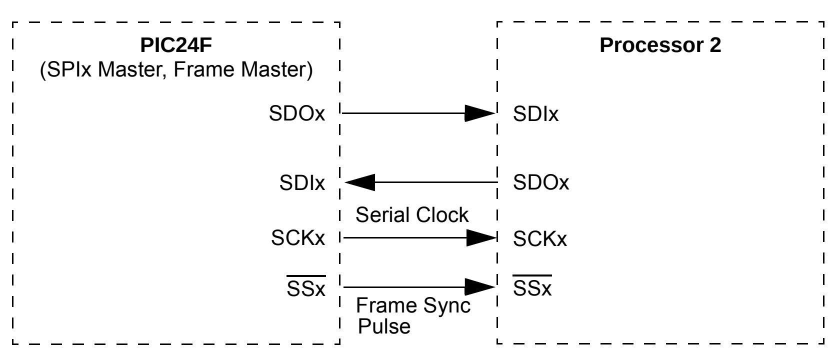


FIGURE 18-6: SPIx MASTER, FRAME SLAVE CONNECTION DIAGRAM

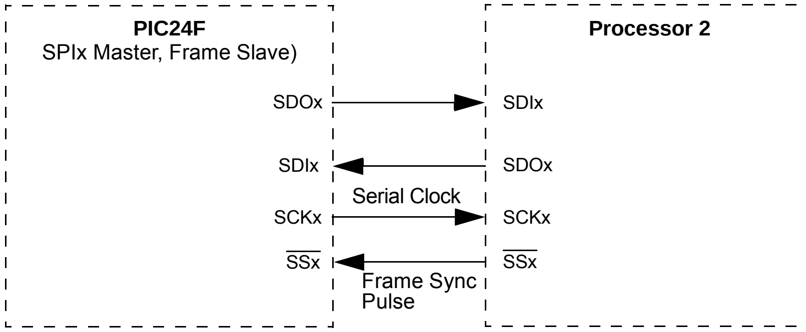


FIGURE 18-7: SPIx SLAVE, FRAME MASTER CONNECTION DIAGRAM

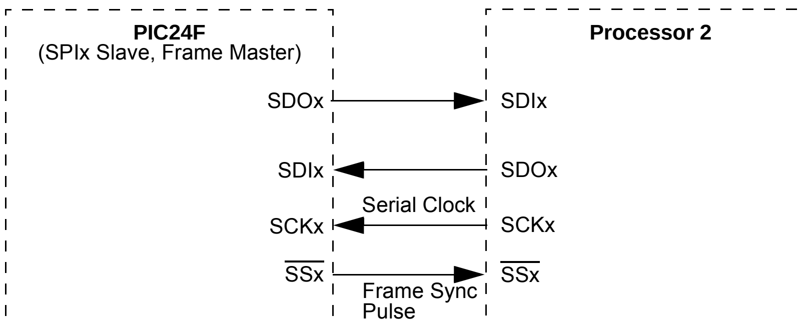
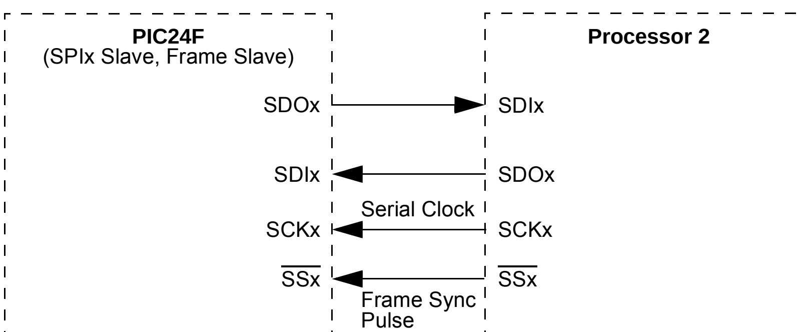


FIGURE 18-8: SPIx SLAVE, FRAME SLAVE CONNECTION DIAGRAM



EQUATION 18-1: RELATIONSHIP BETWEEN DEVICE AND SPIx CLOCK SPEED

$$\text{Baud Rate} = \frac{F_{PB}}{(2 * (\text{SPIxBRG} + 1))}$$

Where:

F_{PB} is the Peripheral Bus Clock Frequency.

19.0 INTER-INTEGRATED CIRCUIT (I^2C)

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Inter-Integrated Circuit (I^2C)"** (DS70000195) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family of devices contains two Inter-Integrated Circuit (I^2C) modules: I2C1 and I2C2.

The I^2C module provides complete hardware support for both Slave and Multi-Master modes of the I^2C serial communication standard, with a 16-bit interface.

The I^2C module has a 2-pin interface:

- The SCLx/ASCLx pin is clock
- The SDAx/ASDAx pin is data

The I^2C module offers the following key features:

- I^2C Interface supporting both Master and Slave modes of Operation
- I^2C Slave mode Supports 7 and 10-Bit Addressing
- I^2C Master mode Supports 7 and 10-Bit Addressing
- I^2C Port allows Bidirectional Transfers between Master and Slaves
- Serial Clock Synchronization for I^2C Port can be used as a Handshake Mechanism to Suspend and Resume Serial Transfer (SCLREL control)
- I^2C Supports Multi-Master Operation, Detects Bus Collision and Arbitrates Accordingly
- System Management Bus (SMBus) Support
- Alternate I^2C Pin Mapping (ASCLx/ASDAx)

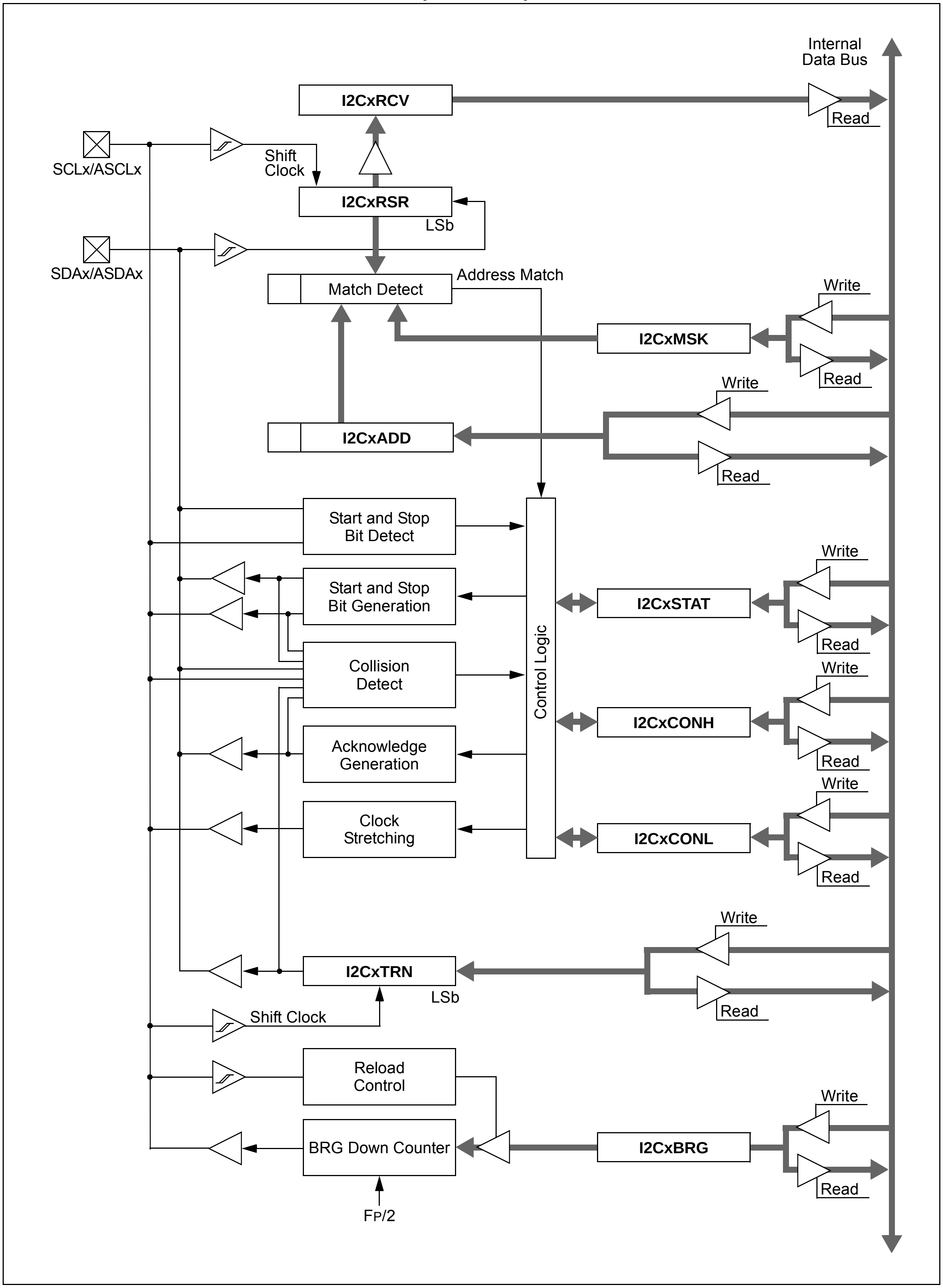
19.1 I^2C Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

19.1.1 KEY RESOURCES

- **"Inter-Integrated Circuit (I^2C)"** (DS70000195) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

FIGURE 19-1: I2Cx BLOCK DIAGRAM (x = 1 OR 2)



19.2 I²C Control Registers

REGISTER 19-1: I2CxCONL: I2Cx CONTROL REGISTER LOW

R/W-0	U-0	R/W-0	R/W-1, HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	STRICT	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

Legend:	HC = Hardware Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **I2CEN:** I2Cx Enable bit
1 = Enables the I2Cx module and configures the SDAx and SCLx pins as serial port pins
0 = Disables the I2Cx module; all I²C pins are controlled by port functions

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **I2CSIDL:** I2Cx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **SCLREL:** SCLx Release Control bit (when operating as I²C slave)
1 = Releases SCLx clock
0 = Holds SCLx clock low (clock stretch)
If STREN = 1:
Bit is R/W (i.e., software can write ‘0’ to initiate stretch and write ‘1’ to release clock). Hardware is clear at the beginning of every slave data byte transmission. Hardware is clear at the end of every slave address byte reception. Hardware is clear at the end of every slave data byte reception.
If STREN = 0:
Bit is R/S (i.e., software can only write ‘1’ to release clock). Hardware is clear at the beginning of every slave data byte transmission. Hardware is clear at the end of every slave address byte reception.

bit 11 **STRICT:** Strict I2Cx Reserved Address Enable bit
1 = Strict Reserved Addressing is Enabled:
In Slave mode, the device will NACK any reserved address. In Master mode, the device is allowed to generate addresses within the reserved address space.
0 = Reserved Addressing is Acknowledged:
In Slave mode, the device will ACK any reserved address. In Master mode, the device should not address a slave device with a reserved address.

bit 10 **A10M:** 10-Bit Slave Address bit
1 = I2CxADD is a 10-bit slave address
0 = I2CxADD is a 7-bit slave address

bit 9 **DISSLW:** Disable Slew Rate Control bit
1 = Slew rate control is disabled
0 = Slew rate control is enabled

bit 8 **SMEN:** SMBus Input Levels bit
1 = Enables I/O pin thresholds compliant with SMBus specification
0 = Disables SMBus input thresholds

bit 7 **GCEN:** General Call Enable bit (when operating as I²C slave)
1 = Enables interrupt when a general call address is received in I2CxRSR (module is enabled for reception)
0 = General call address is disabled

REGISTER 19-1: I2CxCONL: I2Cx CONTROL REGISTER LOW (CONTINUED)

bit 6 **STREN:** SCLx Clock Stretch Enable bit (when operating as I²C slave)
Used in conjunction with the SCLREL bit.
1 = Enables software or receives clock stretching
0 = Disables software or receives clock stretching

bit 5 **ACKDT:** Acknowledge Data bit (when operating as I²C master, applicable during master receive)
Value that is transmitted when the software initiates an Acknowledge sequence.
1 = Sends NACK during Acknowledge
0 = Sends ACK during Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit
(when operating as I²C master, applicable during master receive)
1 = Initiates Acknowledge sequence on SDAx and SCLx pins and transmits ACKDT data bit; hardware is clear at the end of the master Acknowledge sequence
0 = Acknowledge sequence is not in progress

bit 3 **RCEN:** Receive Enable bit (when operating as I²C master)
1 = Enables Receive mode for I²C; hardware is clear at the end of the eighth bit of the master receive data byte
0 = Receive sequence is not in progress

bit 2 **PEN:** Stop Condition Enable bit (when operating as I²C master)
1 = Initiates Stop condition on SDAx and SCLx pins; hardware is clear at the end of the master Stop sequence
0 = Stop condition is not in progress

bit 1 **RSEN:** Repeated Start Condition Enable bit (when operating as I²C master)
1 = Initiates Repeated Start condition on SDAx and SCLx pins; hardware is clear at the end of the master Repeated Start sequence
0 = Repeated Start condition is not in progress

bit 0 **SEN:** Start Condition Enable bit (when operating as I²C master)
1 = Initiates Start condition on SDAx and SCLx pins; hardware is clear at the end of the master Start sequence
0 = Start condition is not in progress

REGISTER 19-2: I2CxCONH: I2Cx CONTROL REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-7 **Unimplemented:** Read as ‘0’

bit 6 **PCIE:** Stop Condition Interrupt Enable bit (I^2C Slave mode only)
1 = Enables interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled

bit 5 **SCIE:** Start Condition Interrupt Enable bit (I^2C Slave mode only)
1 = Enables interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled

bit 4 **BOEN:** Buffer Overwrite Enable bit (I^2C Slave mode only)
1 = I2CxRCV is updated and ACK is generated for a received address/data byte, ignoring the state of the I2COV only if the RBF bit = 0
0 = I2CxRCV is only updated when I2COV is clear

bit 3 **SDAHT:** SDAx Hold Time Selection bit
1 = Minimum of 300 ns hold time on SDAx after the falling edge of SCLx
0 = Minimum of 100 ns hold time on SDAx after the falling edge of SCLx

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I^2C Slave mode only)
1 = Enables slave bus collision interrupts
0 = Slave bus collision interrupts are disabled
If the rising edge of SCLx and SDAx is sampled low when the module is in a high state, the BCL bit is set and the bus goes Idle. This Detection mode is only valid during data and ACK transmit sequences.

bit 1 **AHEN:** Address Hold Enable bit (I^2C Slave mode only)
1 = Following the 8th falling edge of SCLx for a matching received address byte, the SCLREL (I2CxCONL<12>) bit will be cleared and SCLx will be held low
0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit (I^2C Slave mode only)
1 = Following the 8th falling edge of SCLx for a received data byte, the slave hardware clears the SCLREL (I2CxCONL<12>) bit and SCLx is held low
0 = Data holding is disabled

REGISTER 19-3: I2CxSTAT: I2Cx STATUS REGISTER

R-0, HSC	R-0, HSC	R-0, HSC	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0, HS	R/C-0, HS	R-0, HSC	R/C-0, HSC	R/C-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF
bit 7							bit 0

Legend:	C = Clearable bit	'0' = Bit is cleared	HS = Hardware Settable bit
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	'1' = Bit is set	U = Unimplemented bit, read as '0'	

bit 15 **ACKSTAT:** Acknowledge Status bit (when operating as I^{2}C master, applicable to master transmit operation)
1 = NACK was received from slave
0 = ACK was received from slave
Hardware is set or clear at the end of a slave Acknowledge.

bit 14 **TRSTAT:** Transmit Status bit (when operating as I^{2}C master, applicable to master transmit operation)
1 = Master transmit is in progress (8 bits + ACK)
0 = Master transmit is not in progress
Hardware is set at the beginning of master transmission. Hardware is clear at the end of slave Acknowledge.

bit 13 **ACKTIM:** Acknowledge Time Status bit (I^{2}C Slave mode only)
1 = I^{2}C bus is an Acknowledge sequence, set on the 8th falling edge of SCLx
0 = Not an Acknowledge sequence, cleared on the 9th rising edge of SCLx

bit 12-11 **Unimplemented:** Read as '0'

bit 10 **BCL:** Master Bus Collision Detect bit
1 = A bus collision has been detected during a master operation
0 = No bus collision detected
Hardware is set at detection of a bus collision.

bit 9 **GCSTAT:** General Call Status bit
1 = General call address was received
0 = General call address was not received
Hardware is set when address matches the general call address. Hardware is clear at Stop detection.

bit 8 **ADD10:** 10-Bit Address Status bit
1 = 10-bit address was matched
0 = 10-bit address was not matched
Hardware is set at the match of the 2nd byte of the matched 10-bit address. Hardware is clear at Stop detection.

bit 7 **IWCOL:** I2Cx Write Collision Detect bit
1 = An attempt to write to the I2CxTRN register failed because the I^{2}C module is busy
0 = No collision
Hardware is set at the occurrence of a write to I2CxTRN while busy (cleared by software).

bit 6 **I2COV:** I2Cx Receive Overflow Flag bit
1 = A byte was received while the I2CxRCV register was still holding the previous byte
0 = No overflow
Hardware is set at an attempt to transfer I2CxRSR to I2CxRCV (cleared by software).

bit 5 **D_A:** Data/Address bit (I^{2}C Slave mode only)
1 = Indicates that the last byte received was data
0 = Indicates that the last byte received was a device address
Hardware is clear at a device address match. Hardware is set by reception of a slave byte.

REGISTER 19-3: I2CxSTAT: I2Cx STATUS REGISTER (CONTINUED)

bit 4	**P:** Stop bit 1 = Indicates that a Stop bit has been detected last 0 = Stop bit was not detected last Hardware is set or clear when a Start, Repeated Start or Stop is detected.
bit 3	**S:** Start bit 1 = Indicates that a Start (or Repeated Start) bit has been detected last 0 = Start bit was not detected last Hardware is set or clear when a Start, Repeated Start or Stop is detected.
bit 2	**R_W:** Read/Write Information bit (I^2C Slave mode only) 1 = Read – Indicates data transfer is output from the slave 0 = Write – Indicates data transfer is input to the slave Hardware is set or clear after reception of an I^2C device address byte.
bit 1	**RBF:** Receive Buffer Full Status bit 1 = Receive is complete, I2CxRCV is full 0 = Receive is not complete, I2CxRCV is empty Hardware is set when I2CxRCV is written with a received byte. Hardware is clear when software reads I2CxRCV.
bit 0	**TBF:** Transmit Buffer Full Status bit 1 = Transmit is in progress, I2CxTRN is full 0 = Transmit is complete, I2CxTRN is empty Hardware is set when software writes to I2CxTRN. Hardware is clear at completion of a data transmission.

REGISTER 19-4: I2CxMSK: I2Cx SLAVE MODE ADDRESS MASK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK<9:8>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as '0'

bit 9-0 **AMSK<9:0>:** Address Mask Select bits

For 10-Bit Address:
1 = Enables masking for bit Ax of incoming message address; bit match is not required in this position
0 = Disables masking for bit Ax; bit match is required in this position

For 7-Bit Address (I2CxMSK<6:0> only):
1 = Enables masking for bit Ax + 1 of incoming message address; bit match is not required in this position
0 = Disables masking for bit Ax + 1; bit match is required in this position

20.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Universal Asynchronous Receiver Transmitter (UART)"** (DS70000582) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family of devices contains two UART modules.

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the dsPIC33EPXXXGS70X/80X device family. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN/J2602, RS-232 and RS-485 interfaces. The module also supports a hardware flow control option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins, and also includes an IrDA® encoder and decoder.

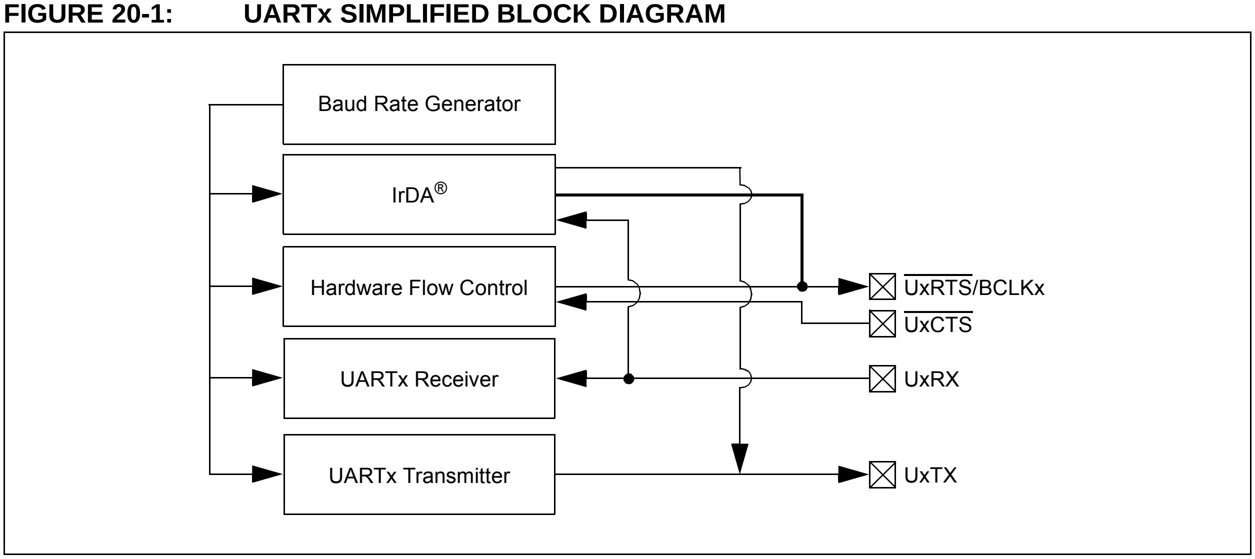
The primary features of the UARTx module are:

- Full-Duplex, 8 or 9-Bit Data Transmission through the UxTX and UxRX Pins
- Even, Odd or No Parity Options (for 8-bit data)
- One or Two Stop bits
- Hardware Flow Control Option with $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Pins
- Fully Integrated Baud Rate Generator with 16-Bit Prescaler
- Baud Rates Ranging from 4.375 Mbps to 67 bps in 16x mode at 70 MIPS
- Baud Rates Ranging from 17.5 Mbps to 267 bps in 4x mode at 70 MIPS
- 4-Deep First-In First-Out (FIFO) Transmit Data Buffer
- 4-Deep FIFO Receive Data Buffer
- Parity, Framing and Buffer Overrun Error Detection
- Support for 9-Bit mode with Address Detect (9th bit = 1)
- Transmit and Receive Interrupts
- A Separate Interrupt for all UARTx Error Conditions
- Loopback mode for Diagnostic Support
- Support for Sync and Break Characters
- Support for Automatic Baud Rate Detection
- IrDA® Encoder and Decoder Logic
- 16x Baud Clock Output for IrDA Support

A simplified block diagram of the UARTx module is shown in Figure 20-1. The UARTx module consists of these key hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

FIGURE 20-1: UARTx SIMPLIFIED BLOCK DIAGRAM



20.1 UART Helpful Tips

1. In multi-node, direct connect UART networks, UART receive inputs react to the complementary logic level defined by the URXINV bit (UxMODE<4>), which defines the Idle state, the default of which is logic high (i.e., URXINV = 0). Because remote devices do not initialize at the same time, it is likely that one of the devices, because the RX line is floating, will trigger a Start bit detection and will cause the first byte received, after the device has been initialized, to be invalid. To avoid this situation, the user should use a pull-up or pull-down resistor on the RX pin depending on the value of the URXINV bit.
   a) If URXINV = 0, use a pull-up resistor on the UxRX pin.
   b) If URXINV = 1, use a pull-down resistor on the UxRX pin.
2. The first character received on a wake-up from Sleep mode, caused by activity on the UxRX pin of the UARTx module, will be invalid. In Sleep mode, peripheral clocks are disabled. By the time the oscillator system has restarted and stabilized from Sleep mode, the baud rate bit sampling clock, relative to the incoming UxRX bit timing, is no longer synchronized, resulting in the first character being invalid; this is to be expected.

20.2 UART Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

20.2.1 KEY RESOURCES

- **"Universal Asynchronous Receiver Transmitter (UART)"** (DS70000582) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

20.3 UART Control Registers

REGISTER 20-1: UxMODE: UARTx MODE REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN[(1)]	—	USIDL	IREN[(2)]	RTSMD	—	UEN1	UEN0
bit 15							bit 8

R/W-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7							bit 0

Legend:	HC = Hardware Clearable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **UARTEN:** UARTx Enable bit[(1)]
1 = UARTx is enabled; all UARTx pins are controlled by UARTx as defined by UEN<1:0>
0 = UARTx is disabled; all UARTx pins are controlled by PORT latches; UARTx power consumption is minimal

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **USIDL:** UARTx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **IREN:** IrDA® Encoder and Decoder Enable bit[(2)]
1 = IrDA encoder and decoder are enabled
0 = IrDA encoder and decoder are disabled

bit 11 **RTSMD:** Mode Selection for $\overline{\text{UxRTS}}$ Pin bit
1 = $\overline{\text{UxRTS}}$ pin is in Simplex mode
0 = $\overline{\text{UxRTS}}$ pin is in Flow Control mode

bit 10 **Unimplemented:** Read as ‘0’

bit 9-8 **UEN<1:0>:** UARTx Pin Enable bits
11 = UxTX, UxRX and BCLKx pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by PORT latches
10 = UxTX, UxRX, $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins are enabled and used
01 = UxTX, UxRX and $\overline{\text{UxRTS}}$ pins are enabled and used; $\overline{\text{UxCTS}}$ pin is controlled by PORT latches
00 = UxTX and UxRX pins are enabled and used; $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$/BCLKx pins are controlled by PORT latches

bit 7 **WAKE:** Wake-up on Start Bit Detect During Sleep Mode Enable bit
1 = UARTx continues to sample the UxRX pin, interrupt is generated on the falling edge; bit is cleared in hardware on the following rising edge
0 = No wake-up is enabled

bit 6 **LPBACK:** UARTx Loopback Mode Select bit
1 = Enables Loopback mode
0 = Loopback mode is disabled

Note 1: Refer to **“Universal Asynchronous Receiver Transmitter (UART)”** (DS70000582) in the *“dsPIC33/PIC24 Family Reference Manual”* for information on enabling the UARTx module for receive or transmit operation.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

REGISTER 20-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 5 **ABAUD:** Auto-Baud Enable bit
1 = Enables baud rate measurement on the next character – requires reception of a Sync field (55h) before other data; cleared in hardware upon completion
0 = Baud rate measurement is disabled or completed

bit 4 **URXINV:** UARTx Receive Polarity Inversion bit
1 = UxRX Idle state is '0'
0 = UxRX Idle state is '1'

bit 3 **BRGH:** High Baud Rate Enable bit
1 = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode)
0 = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)

bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
11 = 9-bit data, no parity
10 = 8-bit data, odd parity
01 = 8-bit data, even parity
00 = 8-bit data, no parity

bit 0 **STSEL:** Stop Bit Selection bit
1 = Two Stop bits
0 = One Stop bit

Note 1: Refer to **"Universal Asynchronous Receiver Transmitter (UART)"** (DS70000582) in the *"dsPIC33/PIC24 Family Reference Manual"* for information on enabling the UARTx module for receive or transmit operation.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

REGISTER 20-2: UxSTA: UARTx STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	R/W-0, HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN[1]	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

Legend:	C = Clearable bit	HC = Hardware Clearable bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15,13 **UTXISEL<1:0>:** UARTx Transmission Interrupt Mode Selection bits
11 = Reserved; do not use
10 = Interrupt when a character is transferred to the Transmit Shift Register (TSR), and as a result, the transmit buffer becomes empty
01 = Interrupt when the last character is shifted out of the Transmit Shift Register; all transmit operations are completed
00 = Interrupt when a character is transferred to the Transmit Shift Register (this implies there is at least one character open in the transmit buffer)

bit 14 **UTXINV:** UARTx Transmit Polarity Inversion bit
If IREN = 0:
1 = UxTX Idle state is ‘0’
0 = UxTX Idle state is ‘1’
If IREN = 1:
1 = IrDA® encoded, UxTX Idle state is ‘1’
0 = IrDA encoded, UxTX Idle state is ‘0’

bit 12 **Unimplemented:** Read as ‘0’

bit 11 **UTXBRK:** UARTx Transmit Break bit
1 = Sends Sync Break on next transmission – Start bit, followed by twelve ‘0’ bits, followed by Stop bit; cleared by hardware upon completion
0 = Sync Break transmission is disabled or completed

bit 10 **UTXEN:** UARTx Transmit Enable bit[1]
1 = Transmit is enabled, UxTX pin is controlled by UARTx
0 = Transmit is disabled, any pending transmission is aborted and buffer is reset; UxTX pin is controlled by the PORT

bit 9 **UTXBF:** UARTx Transmit Buffer Full Status bit (read-only)
1 = Transmit buffer is full
0 = Transmit buffer is not full, at least one more character can be written

bit 8 **TRMT:** Transmit Shift Register Empty bit (read-only)
1 = Transmit Shift Register is empty and transmit buffer is empty (the last transmission has completed)
0 = Transmit Shift Register is not empty, a transmission is in progress or queued

bit 7-6 **URXISEL<1:0>:** UARTx Receive Interrupt Mode Selection bits
11 = Interrupt is set on UxRSR transfer, making the receive buffer full (i.e., has 4 data characters)
10 = Interrupt is set on UxRSR transfer, making the receive buffer 3/4 full (i.e., has 3 data characters)
0x = Interrupt is set when any character is received and transferred from the UxRSR to the receive buffer; receive buffer has one or more characters

Note 1: Refer to **“Universal Asynchronous Receiver Transmitter (UART)”** (DS70000582) in the *“dsPIC33/PIC24 Family Reference Manual”* for information on enabling the UARTx module for transmit operation.

REGISTER 20-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
1 = Address Detect mode is enabled; if 9-bit mode is not selected, this does not take effect
0 = Address Detect mode is disabled

bit 4 **RIDLE:** Receiver Idle bit (read-only)
1 = Receiver is Idle
0 = Receiver is active

bit 3 **PERR:** Parity Error Status bit (read-only)
1 = Parity error has been detected for the current character (character at the top of the receive FIFO)
0 = Parity error has not been detected

bit 2 **FERR:** Framing Error Status bit (read-only)
1 = Framing error has been detected for the current character (character at the top of the receive FIFO)
0 = Framing error has not been detected

bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
1 = Receive buffer has overflowed
0 = Receive buffer has not overflowed; clearing a previously set OERR bit (1 → 0 transition) resets the receiver buffer and the UxRSR to the empty state

bit 0 **URXDA:** UARTx Receive Buffer Data Available bit (read-only)
1 = Receive buffer has data, at least one more character can be read
0 = Receive buffer is empty

Note 1: Refer to **"Universal Asynchronous Receiver Transmitter (UART)"** (DS70000582) in the *"dsPIC33/PIC24 Family Reference Manual"* for information on enabling the UARTx module for transmit operation.

21.0 CONFIGURABLE LOGIC CELL (CLC)

Note:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Configurable Logic Cell (CLC)"** (DS70005298) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The Configurable Logic Cell (CLC) module allows the user to specify combinations of signals as inputs to a logic function and to use the logic output to control other peripherals or I/O pins. This provides greater flexibility and potential in embedded designs since the CLC module can operate outside the limitations of software execution and supports a vast amount of output designs.

There are four input gates to the selected logic function. These four input gates select from a pool of up to 32 signals that are selected using four data source selection multiplexers. Figure 21-1 shows an overview of the module. Figure 21-3 shows the details of the data source multiplexers and logic input gate connections.

FIGURE 21-1: CLCx MODULE

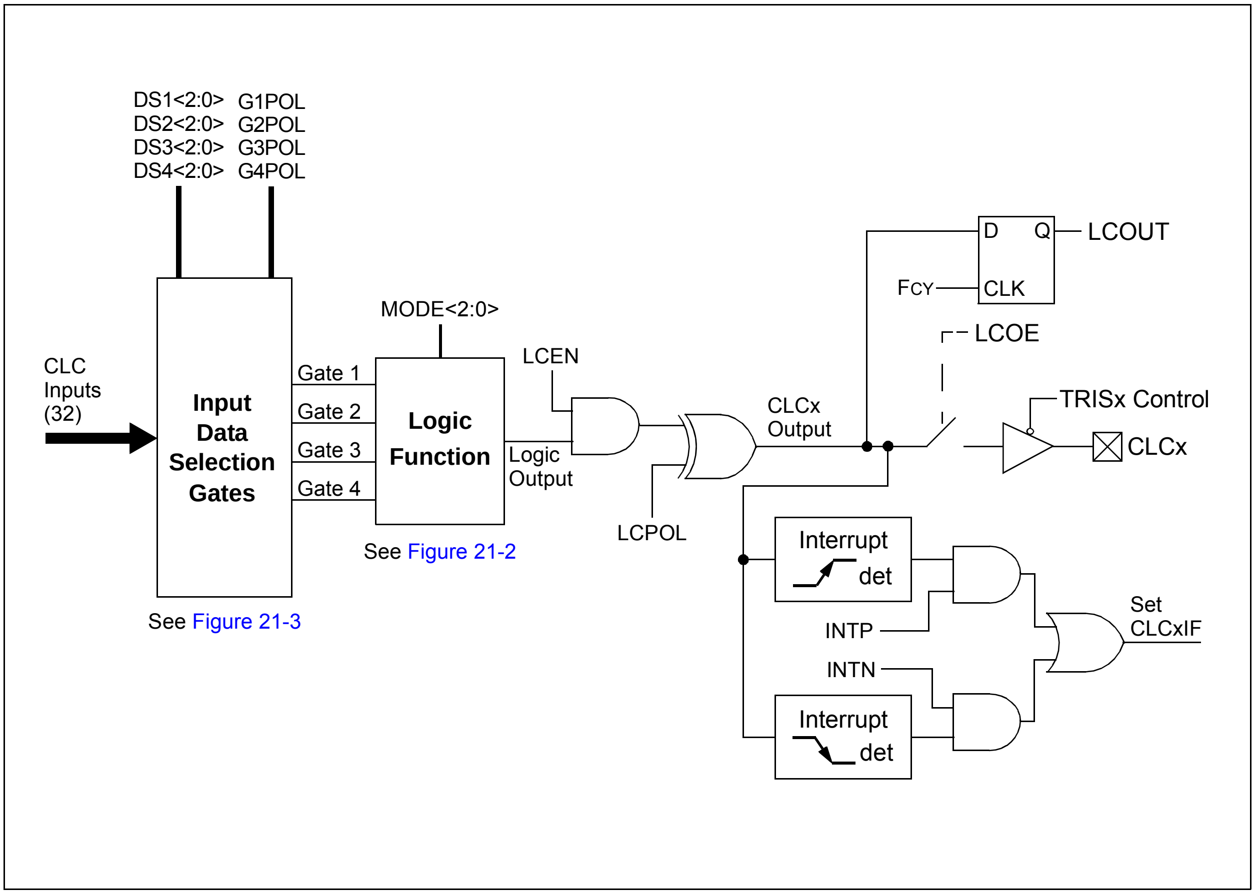


FIGURE 21-2: CLCx LOGIC FUNCTION COMBINATORIAL OPTIONS

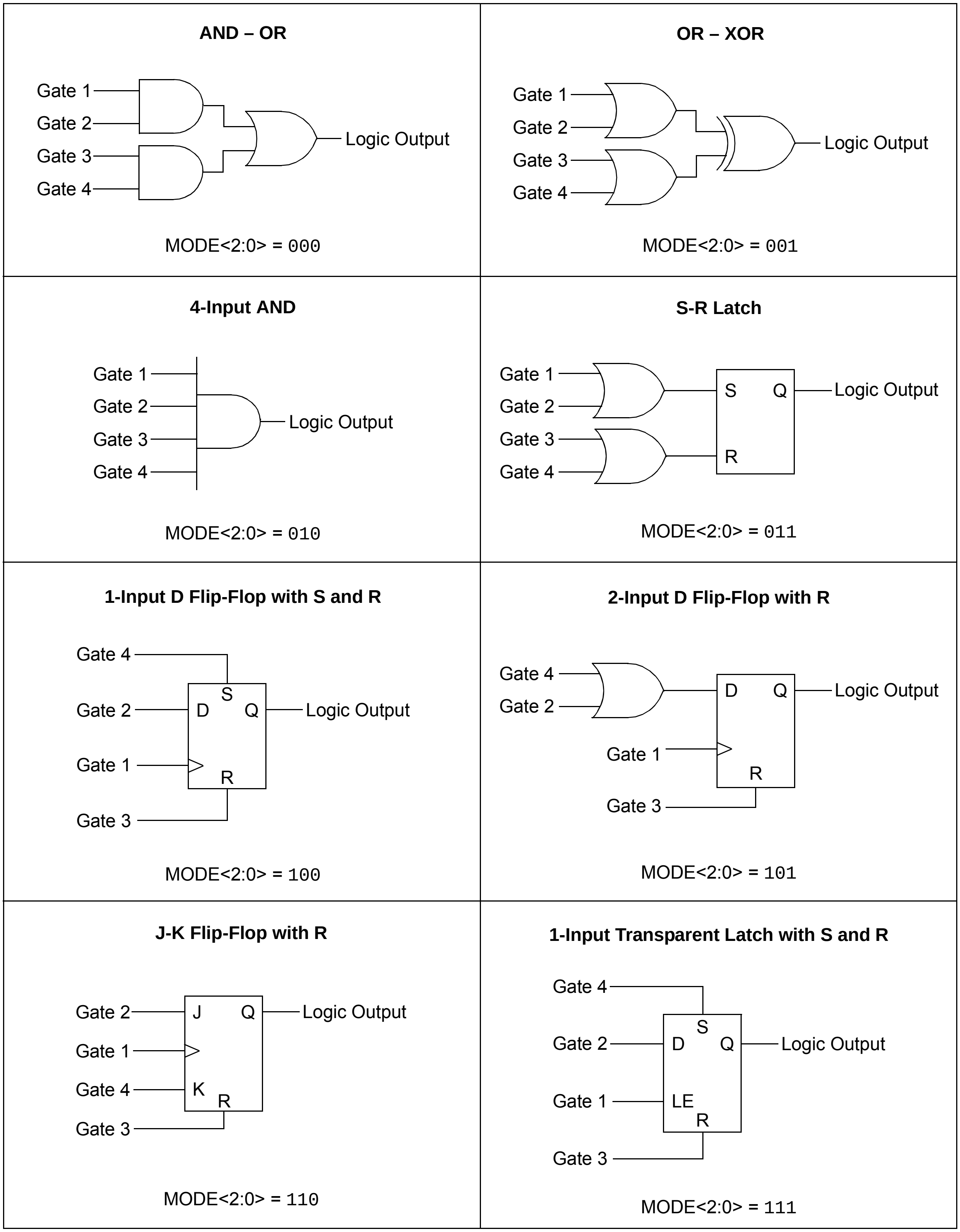
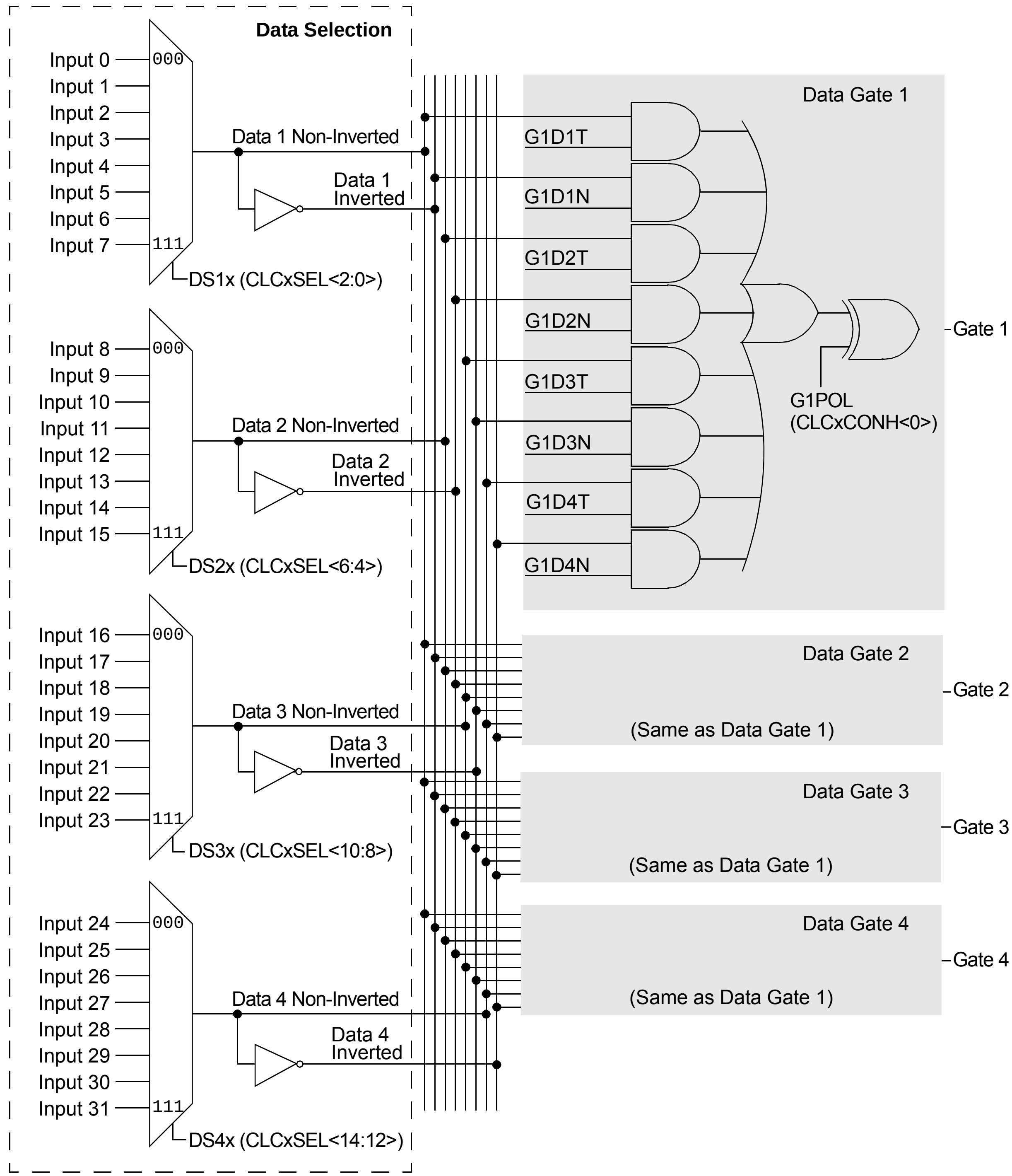


FIGURE 21-3: CLCx INPUT SOURCE SELECTION DIAGRAM



Note: All controls are undefined at power-up.

21.1 Control Registers

The CLCx module is controlled by the following registers:

- CLCxCONL
- CLCxCONH
- CLCxSEL
- CLCxGLSL
- CLCxGLSH

The CLCx Control registers (CLCxCONL and CLCxCONH) are used to enable the module and interrupts, control the output enable bit, select output polarity and select the logic function. The CLCx Control registers also allow the user to control the logic polarity of not only the cell output, but also some intermediate variables.

The CLCx Input MUX Select register (CLCxSEL) allows the user to select up to 4 data input sources using the 4 data input selection multiplexers. Each multiplexer has a list of 8 data sources available.

The CLCx Gate Logic Input Select registers (CLCxGLSL and CLCxGLSH) allow the user to select which outputs from each of the selection MUXes are used as inputs to the input gates of the logic cell. Each data source MUX outputs both a true and a negated version of its output. All of these 8 signals are enabled, ORed together by the logic cell input gates.

REGISTER 21-1: CLCxCONL: CLCx CONTROL REGISTER (LOW)

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0
LCEN	—	—	—	INTP	INTN	—	—
bit 15							bit 8

R-0	R-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
LCOE	LCOUT	LCPOL	—	—	MODE2	MODE1	MODE0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **LCEN:** CLCx Enable bit
1 = CLCx is enabled and mixing input signals
0 = CLCx is disabled and has logic zero outputs

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **INTP:** CLCx Positive Edge Interrupt Enable bit
1 = Interrupt will be generated when a rising edge occurs on LCOUT
0 = Interrupt will not be generated

bit 10 **INTN:** CLCx Negative Edge Interrupt Enable bit
1 = Interrupt will be generated when a falling edge occurs on LCOUT
0 = Interrupt will not be generated

bit 9-8 **Unimplemented:** Read as ‘0’

bit 7 **LCOE:** CLCx Port Enable bit
1 = CLCx port pin output is enabled
0 = CLCx port pin output is disabled

bit 6 **LCOUT:** CLCx Data Output Status bit
1 = CLCx output high
0 = CLCx output low

bit 5 **LCPOL:** CLCx Output Polarity Control bit
1 = The output of the module is inverted
0 = The output of the module is not inverted

bit 4-3 **Unimplemented:** Read as ‘0’

REGISTER 21-1: CLCxCONL: CLCx CONTROL REGISTER (LOW) (CONTINUED)

bit 2-0 **MODE<2:0>:** CLCx Mode bits
111 = Single Input Transparent Latch with S and R
110 = JK Flip-Flop with R
101 = Two-Input D Flip-Flop with R
100 = Single Input D Flip-Flop with S and R
011 = SR Latch
010 = Four-Input AND
001 = Four-Input OR-XOR
000 = Four-Input AND-OR

REGISTER 21-2: CLCxCONH: CLCx CONTROL REGISTER (HIGH)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	G4POL	G3POL	G2POL	G1POL
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **G4POL:** Gate 4 Polarity Control bit
1 = Channel 4 logic output is inverted when applied to the logic cell
0 = Channel 4 logic output is not inverted

bit 2 **G3POL:** Gate 3 Polarity Control bit
1 = Channel 3 logic output is inverted when applied to the logic cell
0 = Channel 3 logic output is not inverted

bit 1 **G2POL:** Gate 2 Polarity Control bit
1 = Channel 2 logic output is inverted when applied to the logic cell
0 = Channel 2 logic output is not inverted

bit 0 **G1POL:** Gate 1 Polarity Control bit
1 = Channel 1 logic output is inverted when applied to the logic cell
0 = Channel 1 logic output is not inverted

REGISTER 21-3: CLCxSEL: CLCx INPUT MUX SELECT REGISTER

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	DS4<2:0>			—	DS3<2:0>		
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	DS2<2:0>			—	DS1<2:0>		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15	**Unimplemented:** Read as ‘0’
bit 14-12	**DS4<2:0>:** Data Selection MUX 4 Signal Selection bits See Table Table 21-1 for input selections.
bit 11	**Unimplemented:** Read as ‘0’
bit 10-8	**DS3<2:0>:** Data Selection MUX 3 Signal Selection bits See Table Table 21-1 for input selections.
bit 7	**Unimplemented:** Read as ‘0’
bit 6-4	**DS2<2:0>:** Data Selection MUX 2 Signal Selection bits See Table Table 21-1 for input selections.
bit 3	**Unimplemented:** Read as ‘0’
bit 2-0	**DS1<2:0>:** Data Selection MUX 1 Signal Selection bits See Table Table 21-1 for input selections.

TABLE 21-1: CLC1 MULTIPLEXER INPUT SOURCES

DSx<2:0>		Signal Source
DS1<2:0>	000	CLCINA
	001	System Clock
	010	Timer1 Match
	011	PWM1H
	100	PWM5L
	101	High-Speed PWM Clock
	110	Timer2 Match
	111	Timer3 Match
DS2<2:0>	000	CLCINB
	001	CLC2 Out
	010	CMP1 Out
	011	UART1 TX Out
	100	ADC End-of-Conversion
	101	DMA Channel 0 Interrupt
	110	PWM1L
	111	PWM5H
DS3<2:0>	000	CLCINA
	001	CLC1 Out
	010	CMP2 Out
	011	SPI1 SDO Out
	100	UART1 RX
	101	PWM2H
	110	PWM6L
	111	OCMP2
DS4<2:0>	000	CLCINB
	001	CLC2 Out
	010	CMP3 Out
	011	SDI1
	100	PTG
	101	ECAN1
	110	PWM2L
	111	PWM6H

TABLE 21-2: CLC2 MULTIPLEXER INPUT SOURCES

DSx<2:0>		Signal Source
DS1<2:0>	000	CLCINA
	001	System Clock
	010	Timer1 Match
	011	PWM3H
	100	PWM7L
	101	High-Speed PWM Clock
	110	Timer2 Match
	111	Timer3 Match
DS2<2:0>	000	CLCINB
	001	CLC1 Out
	010	CMP1 Out
	011	UART2 TX Out
	100	ADC End-of-Conversion
	101	DMA Channel 0 Interrupt
	110	PWM3L
	111	PWM7H
DS3<2:0>	000	CLCINA
	001	CLC2 Out
	010	CMP2 Out
	011	SPI2 SDO Out
	100	UART2 RX
	101	PWM4H
	110	PWM8L
	111	OCMP2
DS4<2:0>	000	CLCINB
	001	CLC1 Out
	010	CMP3 Out
	011	SDI2
	100	PTG
	101	ECAN1
	110	PWM4L
	111	PWM8H

TABLE 21-3: CLC3 MULTIPLEXER INPUT SOURCES

DSx<2:0>		Signal Source
DS1<2:0>	000	CLCINA
	001	System Clock
	010	Timer1 Match
	011	PWM5H
	100	REFO1 Clock Output
	101	High-Speed PWM Clock
	110	Timer2 Match
	111	PWM3L
DS2<2:0>	000	CLCINB
	001	CLC4 Out
	010	CMP1 Out
	011	PWM5L
	100	ADC End-of-Conversion
	101	PWM3H
	110	ICAP1
	111	ICAP2
DS3<2:0>	000	CLCINA
	001	CLC3 Out
	010	CMP2 Out
	011	PWM6H
	100	UART1 RX
	101	DMA Channel 1 Interrupt
	110	OCMP1
	111	PWM4L
DS4<2:0>	000	CLCINB
	001	CLC4 Out
	010	CMP3 Out
	011	PWM6L
	100	PTG
	101	PWM4H
	110	PC_PWM
	111	OCMP3

TABLE 21-4: CLC4 MULTIPLEXER INPUT SOURCES

DSx<2:0>		Signal Source
DS1<2:0>	000	CLCINA
	001	PWM7H
	010	Timer1 Match
	011	INTOSC/LPRC Clock
	100	REFO1 Clock Output
	101	High-Speed PWM Clock
	110	Timer2 Match
	111	PWM1L
DS2<2:0>	000	CLCINB
	001	CLC3 Out
	010	CMP1 Out
	011	PWM7L
	100	ADC End-of-Conversion
	101	PWM1H
	110	ICAP1
	111	ICAP2
DS3<2:0>	000	CLCINA
	001	CLC4 Out
	010	CMP2 Out
	011	PWM8H
	100	UART2 RX
	101	DMA Channel 1 Interrupt
	110	OCMP1
	111	PWM2L
DS4<2:0>	000	CLCINB
	001	CLC3 Out
	010	CMP3 Out
	011	PWM8L
	100	PTG
	101	PWM2H
	110	PC_PWM
	111	OCMP3

REGISTER 21-4: CLCxGLSL: CLCx GATE LOGIC INPUT SELECT LOW REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G2D4T	G2D4N	G2D3T	G2D3N	G2D2T	G2D2N	G2D1T	G2D1N
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G1D4T	G1D4N	G1D3T	G1D3N	G1D2T	G1D2N	G1D1T	G1D1N
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **G2D4T:** Gate 2 Data Source 4 True Enable bit
1 = Data Source 4 non-inverted signal is enabled for Gate 2
0 = Data Source 4 non-inverted signal is disabled for Gate 2

bit 14 **G2D4N:** Gate 2 Data Source 4 Negated Enable bit
1 = Data Source 4 inverted signal is enabled for Gate 2
0 = Data Source 4 inverted signal is disabled for Gate 2

bit 13 **G2D3T:** Gate 2 Data Source 3 True Enable bit
1 = Data Source 3 non-inverted signal is enabled for Gate 2
0 = Data Source 3 non-inverted signal is disabled for Gate 2

bit 12 **G2D3N:** Gate 2 Data Source 3 Negated Enable bit
1 = Data Source 3 inverted signal is enabled for Gate 2
0 = Data Source 3 inverted signal is disabled for Gate 2

bit 11 **G2D2T:** Gate 2 Data Source 2 True Enable bit
1 = Data Source 2 non-inverted signal is enabled for Gate 2
0 = Data Source 2 non-inverted signal is disabled for Gate 2

bit 10 **G2D2N:** Gate 2 Data Source 2 Negated Enable bit
1 = Data Source 2 inverted signal is enabled for Gate 2
0 = Data Source 2 inverted signal is disabled for Gate 2

bit 9 **G2D1T:** Gate 2 Data Source 1 True Enable bit
1 = Data Source 1 non-inverted signal is enabled for Gate 2
0 = Data Source 1 non-inverted signal is disabled for Gate 2

bit 8 **G2D1N:** Gate 2 Data Source 1 Negated Enable bit
1 = Data Source 1 inverted signal is enabled for Gate 2
0 = Data Source 1 inverted signal is disabled for Gate 2

bit 7 **G1D4T:** Gate 1 Data Source 4 True Enable bit
1 = Data Source 4 non-inverted signal is enabled for Gate 1
0 = Data Source 4 non-inverted signal is disabled for Gate 1

bit 6 **G1D4N:** Gate 1 Data Source 4 Negated Enable bit
1 = Data Source 4 inverted signal is enabled for Gate 1
0 = Data Source 4 inverted signal is disabled for Gate 1

bit 5 **G1D3T:** Gate 1 Data Source 3 True Enable bit
1 = Data Source 3 non-inverted signal is enabled for Gate 1
0 = Data Source 3 non-inverted signal is disabled for Gate 1

bit 4 **G1D3N:** Gate 1 Data Source 3 Negated Enable bit
1 = Data Source 3 inverted signal is enabled for Gate 1
0 = Data Source 3 inverted signal is disabled for Gate 1

REGISTER 21-4: CLCxGLSL: CLCx GATE LOGIC INPUT SELECT LOW REGISTER (CONTINUED)

bit 3	**G1D2T:** Gate 1 Data Source 2 True Enable bit 1 = Data Source 2 non-inverted signal is enabled for Gate 1 0 = Data Source 2 non-inverted signal is disabled for Gate 1
bit 2	**G1D2N:** Gate 1 Data Source 2 Negated Enable bit 1 = Data Source 2 inverted signal is enabled for Gate 1 0 = Data Source 2 inverted signal is disabled for Gate 1
bit 1	**G1D1T:** Gate 1 Data Source 1 True Enable bit 1 = Data Source 1 non-inverted signal is enabled for Gate 1 0 = Data Source 1 non-inverted signal is disabled for Gate 1
bit 0	**G1D1N:** Gate 1 Data Source 1 Negated Enable bit 1 = Data Source 1 inverted signal is enabled for Gate 1 0 = Data Source 1 inverted signal is disabled for Gate 1

REGISTER 21-5: CLCxGLSH: CLCx GATE LOGIC INPUT SELECT HIGH REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G4D4T	G4D4N	G4D3T	G4D3N	G4D2T	G4D2N	G4D1T	G4D1N
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G3D4T	G3D4N	G3D3T	G3D3N	G3D2T	G3D2N	G3D1T	G3D1N
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **G4D4T:** Gate 4 Data Source 4 True Enable bit
1 = Data Source 4 non-inverted signal is enabled for Gate 4
0 = Data Source 4 non-inverted signal is disabled for Gate 4

bit 14 **G4D4N:** Gate 4 Data Source 4 Negated Enable bit
1 = Data Source 4 inverted signal is enabled for Gate 4
0 = Data Source 4 inverted signal is disabled for Gate 4

bit 13 **G4D3T:** Gate 4 Data Source 3 True Enable bit
1 = Data Source 3 non-inverted signal is enabled for Gate 4
0 = Data Source 3 non-inverted signal is disabled for Gate 4

bit 12 **G4D3N:** Gate 4 Data Source 3 Negated Enable bit
1 = Data Source 3 inverted signal is enabled for Gate 4
0 = Data Source 3 inverted signal is disabled for Gate 4

bit 11 **G4D2T:** Gate 4 Data Source 2 True Enable bit
1 = Data Source 2 non-inverted signal is enabled for Gate 4
0 = Data Source 2 non-inverted signal is disabled for Gate 4

bit 10 **G4D2N:** Gate 4 Data Source 2 Negated Enable bit
1 = Data Source 2 inverted signal is enabled for Gate 4
0 = Data Source 2 inverted signal is disabled for Gate 4

bit 9 **G4D1T:** Gate 4 Data Source 1 True Enable bit
1 = Data Source 1 non-inverted signal is enabled for Gate 4
0 = Data Source 1 non-inverted signal is disabled for Gate 4

bit 8 **G4D1N:** Gate 4 Data Source 1 Negated Enable bit
1 = Data Source 1 inverted signal is enabled for Gate 4
0 = Data Source 1 inverted signal is disabled for Gate 4

bit 7 **G3D4T:** Gate 3 Data Source 4 True Enable bit
1 = Data Source 4 non-inverted signal is enabled for Gate 3
0 = Data Source 4 non-inverted signal is disabled for Gate 3

bit 6 **G3D4N:** Gate 3 Data Source 4 Negated Enable bit
1 = Data Source 4 inverted signal is enabled for Gate 3
0 = Data Source 4 inverted signal is disabled for Gate 3

bit 5 **G3D3T:** Gate 3 Data Source 3 True Enable bit
1 = Data Source 3 non-inverted signal is enabled for Gate 3
0 = Data Source 3 non-inverted signal is disabled for Gate 3

bit 4 **G3D3N:** Gate 3 Data Source 3 Negated Enable bit
1 = Data Source 3 inverted signal is enabled for Gate 3
0 = Data Source 3 inverted signal is disabled for Gate 3

REGISTER 21-5: CLCxGLSH: CLCx GATE LOGIC INPUT SELECT HIGH REGISTER (CONTINUED)

bit 3	**G3D2T:** Gate 3 Data Source 2 True Enable bit 1 = Data Source 2 non-inverted signal is enabled for Gate 3 0 = Data Source 2 non-inverted signal is disabled for Gate 3
bit 2	**G3D2N:** Gate 3 Data Source 2 Negated Enable bit 1 = Data Source 2 inverted signal is enabled for Gate 3 0 = Data Source 2 inverted signal is disabled for Gate 3
bit 1	**G3D1T:** Gate 3 Data Source 1 True Enable bit 1 = Data Source 1 non-inverted signal is enabled for Gate 3 0 = Data Source 1 non-inverted signal is disabled for Gate 3
bit 0	**G3D1N:** Gate 3 Data Source 1 Negated Enable bit 1 = Data Source 1 inverted signal is enabled for Gate 3 0 = Data Source 1 inverted signal is disabled for Gate 3

22.0 HIGH-SPEED, 12-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"12-Bit High-Speed, Multiple SARs A/D Converter (ADC)"** (DS70005213) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

dsPIC33EPXXXGS70X/80X devices have a high-speed, 12-bit Analog-to-Digital Converter (ADC) that features a low conversion latency, high resolution and oversampling capabilities to improve performance in AC/DC, DC/DC power converters.

22.1 Features Overview

The high-speed, 12-bit multiple SARs Analog-to-Digital Converter (ADC) includes the following features:

- Five ADC Cores: Four Dedicated Cores and One Shared (common) Core
- User-Configurable Resolution of up to 12 Bits for each Core
- Up to 3.25 Msps Conversion Rate per Channel at 12-Bit Resolution
- Low Latency Conversion
- Up to 22 Analog Input Channels, with a Separate 16-Bit Conversion Result Register for each Input
- Conversion Result can be Formatted as Unsigned or Signed Data, on a per Channel Basis, for All Channels
- Single-Ended and Pseudodifferential Conversions are available on All ADC Cores
- Simultaneous Sampling of up to 5 Analog Inputs
- Channel Scan Capability
- Multiple Conversion Trigger Options for each Core, including:
  - PWM1 through PWM6 (primary and secondary triggers, and current-limit event trigger)
  - PWM Special Event Trigger
  - Timer1/Timer2 period match
  - Output Compare 1 and event trigger
  - External pin trigger event (ADTRG31)
  - Software trigger
- Two Integrated Digital Comparators with Dedicated Interrupts:
  - Multiple comparison options
  - Assignable to specific analog inputs
- Two Oversampling Filters with Dedicated Interrupts:
  - Provide increased resolution
  - Assignable to a specific analog input

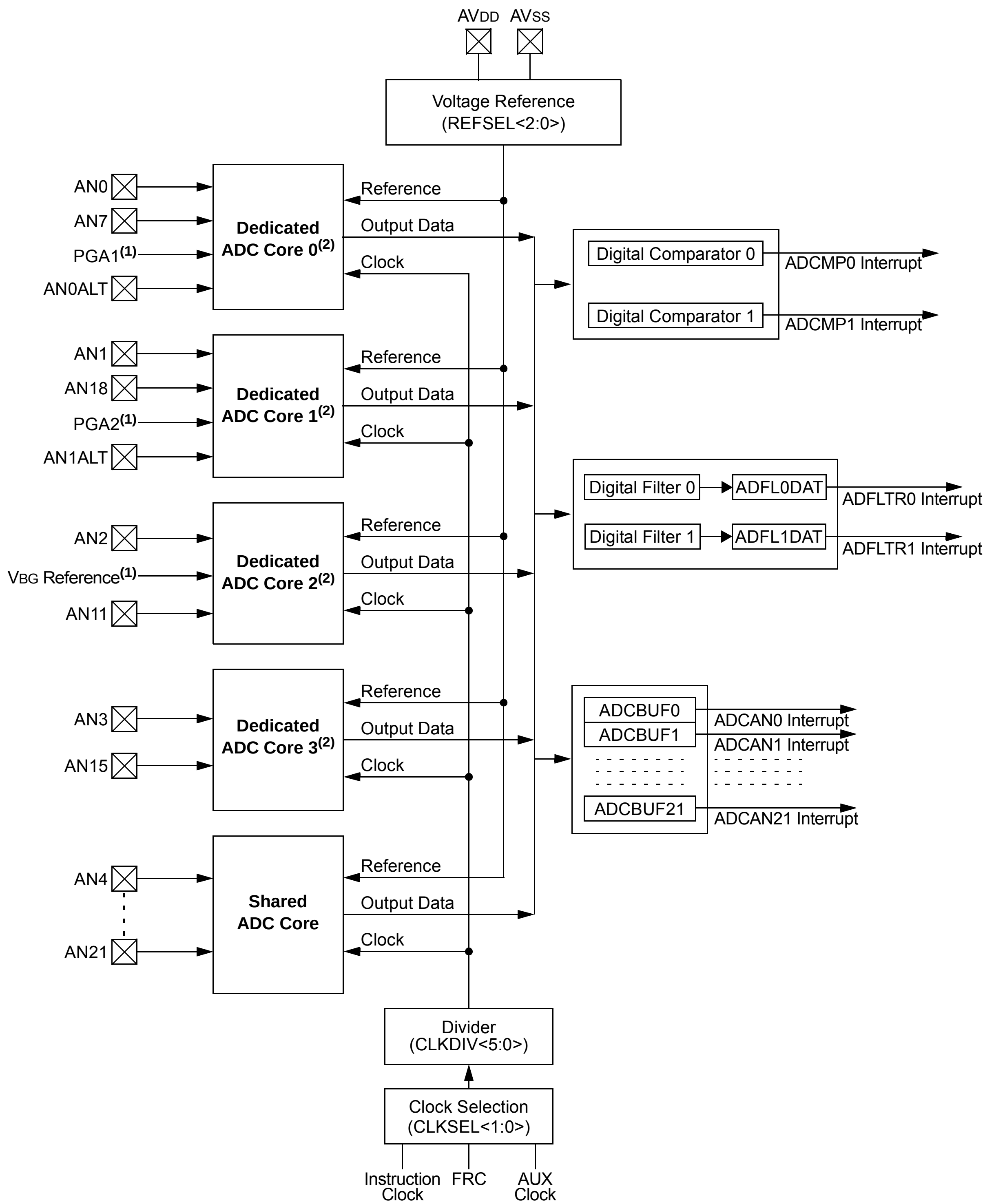
The module consists of five independent SAR ADC cores. Simplified block diagrams of the multiple SARs 12-bit ADC are shown in Figure 22-1, Figure 22-2 and Figure 22-3.

The analog inputs (channels) are connected through multiplexers and switches to the Sample-and-Hold (S&H) circuit of each ADC core. The core uses the channel information (the output format, the Measurement mode and the input number) to process the analog sample. When conversion is complete, the result is stored in the result buffer for the specific analog input, and passed to the digital filter and digital comparator if they were configured to use data from this particular channel.

The ADC module can sample up to five inputs at a time (four inputs from the dedicated SAR cores and one from the shared SAR core). If multiple ADC inputs request conversion on the shared core, the module will convert them in a sequential manner, starting with the lowest order input.

The ADC provides each analog input the ability to specify its own trigger source. This capability allows the ADC to sample and convert analog inputs that are associated with PWM generators operating on independent time bases.

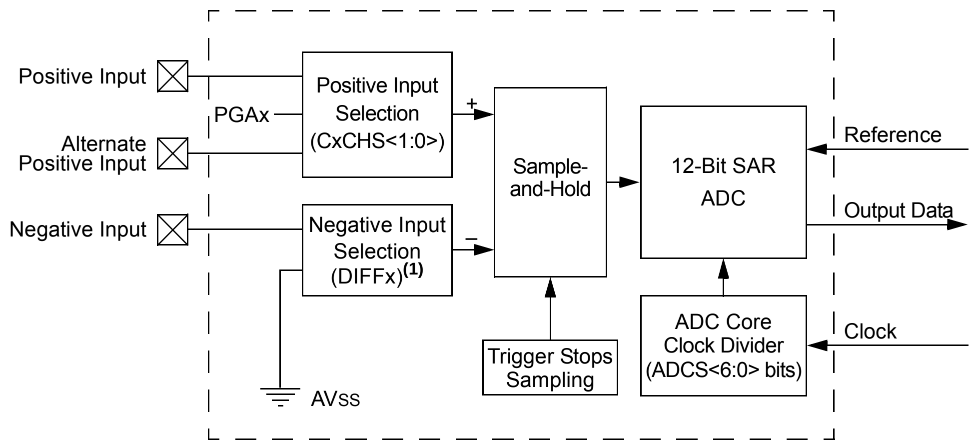
FIGURE 22-1: ADC MODULE BLOCK DIAGRAM



Note 1: PGA1, PGA2 and the Band Gap Reference (VBG) are internal analog inputs and are not available on device pins.

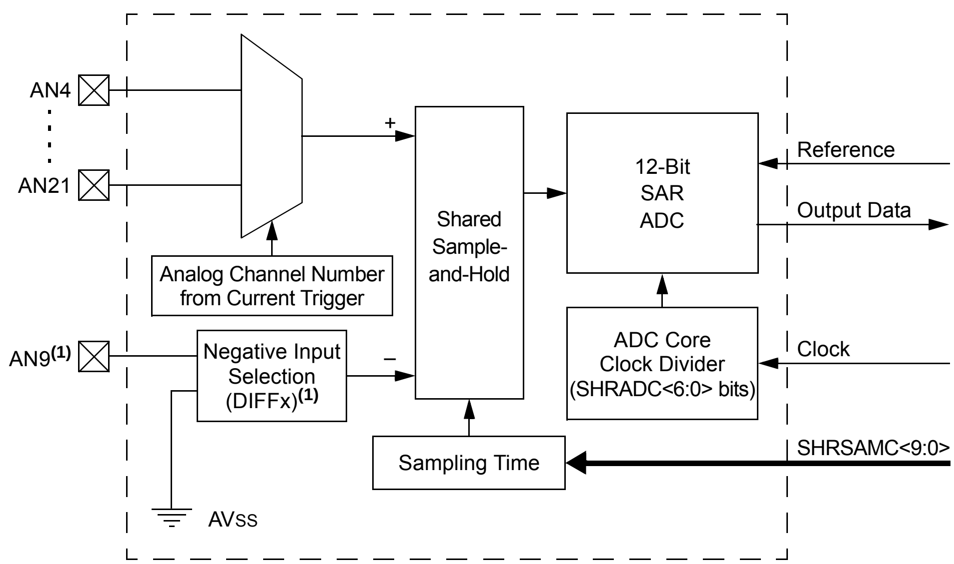
2: If the dedicated core uses an alternate channel, then shared core function cannot be used.

FIGURE 22-2: DEDICATED ADC CORES 0 TO 3 BLOCK DIAGRAM



Note 1: The DIFFx bit for the corresponding positive input channel must be set in order to use the negative differential input.

FIGURE 22-3: SHARED ADC CORE BLOCK DIAGRAM



Note 1: Differential-mode conversion is not available for the shared ADC core in dsPIC33EPXXGS70X/80X devices. For all other devices, the DIFFx bit for the corresponding positive input channel must be set to use AN9 as the negative differential input.

22.2 Analog-to-Digital Converter Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

22.2.1 KEY RESOURCES

- **"12-Bit High-Speed, Multiple SARs A/D Converter (ADC)"** (DS70005213) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

REGISTER 22-1: ADCON1L: ADC CONTROL REGISTER 1 LOW

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
ADON[1]	—	ADSIDL	—	—	—	—	—
bit 15							bit 8

U-0	r-0	r-0	r-0	r-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:	r = Reserved bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **ADON:** ADC Enable bit[1]
1 = ADC module is enabled
0 = ADC module is off

bit 14 **Unimplemented:** Read as '0'

bit 13 **ADSIDL:** ADC Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-7 **Unimplemented:** Read as '0'

bit 6-3 **Reserved:** Maintain as '0'

bit 2-0 **Unimplemented:** Read as '0'

Note 1: Set the ADON bit only after the ADC module has been configured. Changing ADC Configuration bits when ADON = 1 will result in unpredictable behavior.

REGISTER 22-2: ADCON1H: ADC CONTROL REGISTER 1 HIGH

r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-1	R/W-1	r-0	r-0	r-0	r-0	r-0
FORM	SHRRES1	SHRRES0	—	—	—	—	—
bit 7							bit 0

Legend:	r = Reserved bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Reserved:** Maintain as ‘0’

bit 7 **FORM:** Fractional Data Output Format bit
1 = Fractional
0 = Integer

bit 6-5 **SHRRES<1:0>:** Shared ADC Core Resolution Selection bits
11 = 12-bit resolution
10 = 10-bit resolution
01 = 8-bit resolution
00 = 6-bit resolution

bit 4-0 **Reserved:** Maintain as ‘0’

REGISTER 22-3: ADCON2L: ADC CONTROL REGISTER 2 LOW

R/W-0	R/W-0	r-0	R/W-0	r-0	R/W-0	R/W-0	R/W-0
REFCIE	REFERCIE	—	EIEN	—	SHREISEL2[(1)]	SHREISEL1[(1)]	SHREISEL0[(1)]
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SHRADCS6	SHRADCS5	SHRADCS4	SHRADCS3	SHRADCS2	SHRADCS1	SHRADCS0
bit 7							bit 0

Legend:	r = Reserved bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **REFCIE:** Band Gap and Reference Voltage Ready Common Interrupt Enable bit
1 = Common interrupt will be generated when the band gap will become ready
0 = Common interrupt is disabled for the band gap ready event

bit 14 **REFERCIE:** Band Gap or Reference Voltage Error Common Interrupt Enable bit
1 = Common interrupt will be generated when a band gap or reference voltage error is detected
0 = Common interrupt is disabled for the band gap and reference voltage error event

bit 13 **Reserved:** Maintain as ‘0’

bit 12 **EIEN:** Early Interrupts Enable bit
1 = The early interrupt feature is enabled for the input channel interrupts (when the EISTATx flag is set)
0 = The individual interrupts are generated when conversion is done (when the ANxRDY flag is set)

bit 11 **Reserved:** Maintain as ‘0’

bit 10-8 **SHREISEL<2:0>:** Shared Core Early Interrupt Time Selection bits[(1)]
111 = Early interrupt is set and interrupt is generated 8 TADCORE clocks prior to when the data is ready
110 = Early interrupt is set and interrupt is generated 7 TADCORE clocks prior to when the data is ready
101 = Early interrupt is set and interrupt is generated 6 TADCORE clocks prior to when the data is ready
100 = Early interrupt is set and interrupt is generated 5 TADCORE clocks prior to when the data is ready
011 = Early interrupt is set and interrupt is generated 4 TADCORE clocks prior to when the data is ready
010 = Early interrupt is set and interrupt is generated 3 TADCORE clocks prior to when the data is ready
001 = Early interrupt is set and interrupt is generated 2 TADCORE clocks prior to when the data is ready
000 = Early interrupt is set and interrupt is generated 1 TADCORE clock prior to when the data is ready

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **SHRADCS<6:0>:** Shared ADC Core Input Clock Divider bits
These bits determine the number of TCORESRC (Source Clock Periods) for one shared TADCORE (Core Clock Period).
1111111 = 254 Source Clock Periods
•
•
•
0000011 = 6 Source Clock Periods
0000010 = 4 Source Clock Periods
0000001 = 2 Source Clock Periods
0000000 = 2 Source Clock Periods

Note 1: For the 6-bit shared ADC core resolution (SHRRES<1:0> = 00), the SHREISEL<2:0> settings, from ‘100’ to ‘111’, are not valid and should not be used. For the 8-bit shared ADC core resolution (SHRRES<1:0> = 01), the SHREISEL<2:0> settings, ‘110’ and ‘111’, are not valid and should not be used.

REGISTER 22-4: ADCON2H: ADC CONTROL REGISTER 2 HIGH

R-0, HSC	R-0, HSC	r-0	r-0	r-0	r-0	R/W-0	R/W-0
REFRDY	REFERR	—	—	—	—	SHRSAMC9	SHRSAMC8
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRSAMC7	SHRSAMC6	SHRSAMC5	SHRSAMC4	SHRSAMC3	SHRSAMC2	SHRSAMC1	SHRSAMC0
bit 7							bit 0

Legend:	r = Reserved bit	U = Unimplemented bit, read as ‘0’	
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **REFRDY:** Band Gap and Reference Voltage Ready Flag bit
1 = Band gap is ready
0 = Band gap is not ready

bit 14 **REFERR:** Band Gap or Reference Voltage Error Flag bit
1 = Band gap was removed after the ADC module was enabled (ADON = 1)
0 = No band gap error was detected

bit 13-10 **Reserved:** Maintain as ‘0’

bit 9-0 **SHRSAMC<9:0>:** Shared ADC Core Sample Time Selection bits
These bits specify the number of shared ADC Core Clock Periods (TADCORE) for the shared ADC core sample time.
1111111111 = 1025 TADCORE
•
•
•
0000000001 = 3 TADCORE
0000000000 = 2 TADCORE

REGISTER 22-5: ADCON3L: ADC CONTROL REGISTER 3 LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HSC	R/W-0	R-0, HSC
REFSEL2	REFSEL1	REFSEL0	SUSPEND	SUSPCIE	SUSPRDY	SHRSAMP	CNVRTCH
bit 15							bit 8

R/W-0	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SWLCTRG	SWCTRG	CNVCHSEL5	CNVCHSEL4	CNVCHSEL3	CNVCHSEL2	CNVCHSEL1	CNVCHSEL0
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **REFSEL<2:0>:** ADC Reference Voltage Selection bits

Value	VREFH	VREFL
000	AVDD	AVSS

001-111 = **Unimplemented:** Do not use

bit 12 **SUSPEND:** All ADC Cores Triggers Disable bit

1 = All new trigger events for all ADC cores are disabled
0 = All ADC cores can be triggered

bit 11 **SUSPCIE:** Suspend All ADC Cores Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC core triggers are suspended (SUSPEND bit = 1) and all previous conversions are finished (SUSPRDY bit becomes set)
0 = Common interrupt is not generated for suspend ADC cores event

bit 10 **SUSPRDY:** All ADC Cores Suspended Flag bit

1 = All ADC cores are suspended (SUSPEND bit = 1) and have no conversions in progress
0 = ADC cores have previous conversions in progress

bit 9 **SHRSAMP:** Shared ADC Core Sampling Direct Control bit

This bit should be used with the individual channel conversion trigger controlled by the CNVRTCH bit. It connects an analog input, specified by the CNVCHSEL<5:0> bits, to the shared ADC core and allows extending the sampling time. This bit is not controlled by hardware and must be cleared before the conversion starts (setting CNVRTCH to ‘1’).
1 = Shared ADC core samples an analog input specified by the CNVCHSEL<5:0> bits
0 = Sampling is controlled by the shared ADC core hardware

bit 8 **CNVRTCH:** Software Individual Channel Conversion Trigger bit

1 = Single trigger is generated for an analog input specified by the CNVCHSEL<5:0> bits; when the bit is set, it is automatically cleared by hardware on the next instruction cycle
0 = Next individual channel conversion trigger can be generated

bit 7 **SWLCTRG:** Software Level-Sensitive Common Trigger bit

1 = Triggers are continuously generated for all channels with the software, level-sensitive common trigger selected as a source in the ADTRIGxL and ADTRIGxH registers
0 = No software, level-sensitive common triggers are generated

bit 6 **SWCTRG:** Software Common Trigger bit

1 = Single trigger is generated for all channels with the software, common trigger selected as a source in the ADTRIGxL and ADTRIGxH registers; when the bit is set, it is automatically cleared by hardware on the next instruction cycle
0 = Ready to generate the next software common trigger

bit 5-0 **CNVCHSEL <5:0>:** Channel Number Selection for Software Individual Channel Conversion Trigger bits

These bits define a channel to be converted when the CNVRTCH bit is set.

REGISTER 22-6: ADCON3H: ADC CONTROL REGISTER 3 HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLKSEL1	CLKSEL0	CLKDIV5	CLKDIV4	CLKDIV3	CLKDIV2	CLKDIV1	CLKDIV0
bit 15							bit 8

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHREN	—	—	—	C3EN	C2EN	C1EN	C0EN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **CLKSEL<1:0>:** ADC Module Clock Source Selection bits

11 = APLL
10 = FRC
01 = FOSC (System Clock x 2)
00 = FSYS (System Clock)

bit 13-8 **CLKDIV<5:0>:** ADC Module Clock Source Divider bits

The divider forms a TCORESRC clock used by all ADC cores (shared and dedicated) from the TSRC ADC module clock source selected by the CLKSEL<1:0> bits. Then, each ADC core individually divides the TCORESRC clock to get a core-specific TADCORE clock using the ADCS<6:0> bits in the ADCORExH register or the SHRADCS<6:0> bits in the ADCON2L register.

111111 = 64 Source Clock Periods
•
•
•
000011 = 4 Source Clock Periods
000010 = 3 Source Clock Periods
000001 = 2 Source Clock Periods
000000 = 1 Source Clock Period

bit 7 **SHREN:** Shared ADC Core Enable bit

1 = Shared ADC core is enabled
0 = Shared ADC core is disabled

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **C3EN:** Dedicated ADC Core 3 Enable bits

1 = Dedicated ADC Core 3 is enabled
0 = Dedicated ADC Core 3 is disabled

bit 2 **C2EN:** Dedicated ADC Core 2 Enable bits

1 = Dedicated ADC Core 2 is enabled
0 = Dedicated ADC Core 2 is disabled

bit 1 **C1EN:** Dedicated ADC Core 1 Enable bits

1 = Dedicated ADC Core 1 is enabled
0 = Dedicated ADC Core 1 is disabled

bit 0 **C0EN:** Dedicated ADC Core 0 Enable bits

1 = Dedicated ADC Core 0 is enabled
0 = Dedicated ADC Core 0 is disabled

REGISTER 22-7: ADCON4L: ADC CONTROL REGISTER 4 LOW

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SAMC3EN	SAMC2EN	SAMC1EN	SAMC0EN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-4 **Unimplemented:** Read as ‘0’

bit 3 **SAMC3EN:** Dedicated ADC Core 3 Conversion Delay Enable bit
1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE3L register
0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

bit 2 **SAMC2EN:** Dedicated ADC Core 2 Conversion Delay Enable bit
1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE2L register
0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

bit 1 **SAMC1EN:** Dedicated ADC Core 1 Conversion Delay Enable bit
1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE1L register
0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

bit 0 **SAMC0EN:** Dedicated ADC Core 0 Conversion Delay Enable bit
1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE0L register
0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

REGISTER 22-8: ADCON4H: ADC CONTROL REGISTER 4 HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C3CHS1	C3CHS0	C2CHS1	C2CHS0	C1CHS1	C1CHS0	C0CHS1	C0CHS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-6 **C3CHS<1:0>:** Dedicated ADC Core 3 Input Channel Selection bits
1x = Reserved
01 = AN15 (differential negative input when DIFF3 (ADMOD0L<7>) = 1)
00 = AN3

bit 5-4 **C2CHS<1:0>:** Dedicated ADC Core 2 Input Channel Selection bits
11 = Reserved
10 = VREF band gap
01 = AN11 (differential negative input when DIFF2 (ADMOD0L<5>) = 1)
00 = AN2

bit 3-2 **C1CHS<1:0>:** Dedicated ADC Core 1 Input Channel Selection bits
11 = AN1ALT
10 = PGA2
01 = AN18 (differential negative input when DIFF1 (ADMOD0L<3>) = 1)
00 = AN1

bit 1-0 **C0CHS<1:0>:** Dedicated ADC Core 0 Input Channel Selection bits
11 = AN0ALT
10 = PGA1
01 = AN7 (differential negative input when DIFF0 (ADMOD0L<1>) = 1)
00 = AN0

REGISTER 22-9: ADCON5L: ADC CONTROL REGISTER 5 LOW

R-0, HSC	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
SHRRDY	—	—	—	C3RDY	C2RDY	C1RDY	C0RDY
bit 15							bit 8

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRPWR	—	—	—	C3PWR	C2PWR	C1PWR	C0PWR
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **SHRRDY:** Shared ADC Core Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **C3RDY:** Dedicated ADC Core 3 Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 10 **C2RDY:** Dedicated ADC Core 2 Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 9 **C1RDY:** Dedicated ADC Core 1 Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 8 **C0RDY:** Dedicated ADC Core 0 Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 7 **SHRPWR:** Shared ADC Core x Power Enable bit
1 = ADC Core x is powered
0 = ADC Core x is off

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **C3PWR:** Dedicated ADC Core 3 Power Enable bit
1 = ADC core is powered
0 = ADC core is off

bit 2 **C2PWR:** Dedicated ADC Core 2 Power Enable bit
1 = ADC core is powered
0 = ADC core is off

bit 1 **C1PWR:** Dedicated ADC Core 1 Power Enable bit
1 = ADC core is powered
0 = ADC core is off

bit 0 **C0PWR:** Dedicated ADC Core 0 Power Enable bit
1 = ADC core is powered
0 = ADC core is off

REGISTER 22-10: ADCON5H: ADC CONTROL REGISTER 5 HIGH

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	WARMTIME3	WARMTIME2	WARMTIME1	WARMTIME0
bit 15							bit 8

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRCIE	—	—	—	C3CIE	C2CIE	C1CIE	C0CIE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-8 **WARMTIME<3:0>:** ADC Dedicated Core x Power-up Delay bits

These bits determine the power-up delay in the number of the Core Source Clock Periods ($T_{CORESRC}$) for all ADC cores.
1111 = 32768 Source Clock Periods
1110 = 16384 Source Clock Periods
1101 = 8192 Source Clock Periods
1100 = 4096 Source Clock Periods
1011 = 2048 Source Clock Periods
1010 = 1024 Source Clock Periods
1001 = 512 Source Clock Periods
1000 = 256 Source Clock Periods
0111 = 128 Source Clock Periods
0110 = 64 Source Clock Periods
0101 = 32 Source Clock Periods
0100 = 16 Source Clock Periods
00xx = 16 Source Clock Periods

bit 7 **SHRCIE:** Shared ADC Core Ready Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC core is powered and ready for operation
0 = Common interrupt is disabled for an ADC core ready event

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **C3CIE:** Dedicated ADC Core 3 Ready Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC Core 3 is powered and ready for operation
0 = Common interrupt is disabled for an ADC Core 3 ready event

bit 2 **C2CIE:** Dedicated ADC Core 2 Ready Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC Core 2 is powered and ready for operation
0 = Common interrupt is disabled for an ADC Core 2 ready event

bit 1 **C1CIE:** Dedicated ADC Core 1 Ready Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC Core 1 is powered and ready for operation
0 = Common interrupt is disabled for an ADC Core 1 ready event

bit 0 **C0CIE:** Dedicated ADC Core 0 Ready Common Interrupt Enable bit

1 = Common interrupt will be generated when ADC Core 0 is powered and ready for operation
0 = Common interrupt is disabled for an ADC Core 0 ready event

REGISTER 22-11: ADCORExL: DEDICATED ADC CORE x CONTROL REGISTER LOW (x = 0 to 3)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SAMC<9:8>	
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SAMC<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as ‘0’

bit 9-0 **SAMC<9:0>:** Dedicated ADC Core x Conversion Delay Selection bits

These bits determine the time between the trigger event and the start of conversion in the number of the Core Clock Periods (TADCORE). During this time, the ADC Core x still continues sampling. This feature is enabled by the SAMCxEN bits in the ADCON4L register.

1111111111 = 1025 TADCORE
•
•
•
0000000001 = 3 TADCORE
0000000000 = 2 TADCORE

REGISTER 22-12: ADCORExH: DEDICATED ADC CORE x CONTROL REGISTER HIGH (x = 0 to 3)[1]

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
—	—	—	EISEL2	EISEL1	EISEL0	RES1	RES0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-10 **EISEL<2:0>:** ADC Core x Early Interrupt Time Selection bits
111 = Early interrupt is set and an interrupt is generated 8 TADCORE clocks prior to when the data is ready
110 = Early interrupt is set and an interrupt is generated 7 TADCORE clocks prior to when the data is ready
101 = Early interrupt is set and an interrupt is generated 6 TADCORE clocks prior to when the data is ready
100 = Early interrupt is set and an interrupt is generated 5 TADCORE clocks prior to when the data is ready
011 = Early interrupt is set and an interrupt is generated 4 TADCORE clocks prior to when the data is ready
010 = Early interrupt is set and an interrupt is generated 3 TADCORE clocks prior to when the data is ready
001 = Early interrupt is set and an interrupt is generated 2 TADCORE clocks prior to when the data is ready
000 = Early interrupt is set and an interrupt is generated 1 TADCORE clock prior to when the data is ready

bit 9-8 **RES<1:0>:** ADC Core x Resolution Selection bits
11 = 12-bit resolution
10 = 10-bit resolution
01 = 8-bit resolution
00 = 6-bit resolution

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **ADCS<6:0>:** ADC Core x Input Clock Divider bits
These bits determine the number of Source Clock Periods (TCORESRC) for one Core Clock Period (TADCORE).
1111111 = 254 Source Clock Periods
•
•
•
0000011 = 6 Source Clock Periods
0000010 = 4 Source Clock Periods
0000001 = 2 Source Clock Periods
0000000 = 2 Source Clock Periods

Note 1: For the 6-bit ADC core resolution (RES<1:0> = 00), the EISEL<2:0> bits settings, from ‘100’ to ‘111’, are not valid and should not be used. For the 8-bit ADC core resolution (RES<1:0> = 01), the EISEL<2:0> bits settings, ‘110’ and ‘111’, are not valid and should not be used.

REGISTER 22-13: ADLVLTRGL: ADC LEVEL-SENSITIVE TRIGGER CONTROL REGISTER LOW

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
LVLEN<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LVLEN<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **LVLEN<15:0>:** Level Trigger for Corresponding Analog Input Enable bits
1 = Input trigger is level-sensitive
0 = Input trigger is edge-sensitive

REGISTER 22-14: ADLVLTRGH: ADC LEVEL-SENSITIVE TRIGGER CONTROL REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	LVLEN<21:16>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **LVLEN<21:16>:** Level Trigger for Corresponding Analog Input Enable bits
1 = Input trigger is level-sensitive
0 = Input trigger is edge-sensitive

REGISTER 22-15: ADEIEL: ADC EARLY INTERRUPT ENABLE REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EIEN<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EIEN<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **EIEN<15:0>:** Early Interrupt Enable for Corresponding Analog Inputs bits
1 = Early interrupt is enabled for the channel
0 = Early interrupt is disabled for the channel

REGISTER 22-16: ADEIEH: ADC EARLY INTERRUPT ENABLE REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	EIEN<21:16>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **EIEN<21:16>:** Early Interrupt Enable for Corresponding Analog Inputs bits
1 = Early interrupt is enabled for the channel
0 = Early interrupt is disabled for the channel

REGISTER 22-17: ADEISTATL: ADC EARLY INTERRUPT STATUS REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EISTAT<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EISTAT<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **EISTAT<15:0>:** Early Interrupt Status for Corresponding Analog Inputs bits
1 = Early interrupt was generated
0 = Early interrupt was not generated since the last ADCBUFx read

REGISTER 22-18: ADEISTATH: ADC EARLY INTERRUPT STATUS REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	EISTAT<21:16>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **EISTAT<21:16>:** Early Interrupt Status for Corresponding Analog Inputs bits
1 = Early interrupt was generated
0 = Early interrupt was not generated since the last ADCBUFx read

REGISTER 22-19: ADMOD0L: ADC INPUT MODE CONTROL REGISTER 0 LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-1(odd) **DIFF<7:0>:** Differential-Mode for Corresponding Analog Inputs bits
1 = Channel is differential
0 = Channel is single-ended

bit 14-0 (even) **SIGN<7:0>:** Output Data Sign for Corresponding Analog Inputs bits
1 = Channel output data is signed
0 = Channel output data is unsigned

REGISTER 22-20: ADMOD0H: ADC INPUT MODE CONTROL REGISTER 0 HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF15	SIGN15	DIFF14	SIGN14	DIFF13	SIGN13	DIFF12	SIGN12
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-1(odd) **DIFF<15:8>:** Differential-Mode for Corresponding Analog Inputs bits
1 = Channel is differential
0 = Channel is single-ended

bit 14-0 (even) **SIGN<15:8>:** Output Data Sign for Corresponding Analog Inputs bits
1 = Channel output data is signed
0 = Channel output data is unsigned

REGISTER 22-21: ADMOD1L: ADC INPUT MODE CONTROL REGISTER 1 LOW

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	DIFF21	SIGN21	DIFF20	SIGN20
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF19	SIGN19	DIFF18	SIGN18	DIFF17	SIGN17	DIFF16	SIGN16
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-1(odd) **DIFF<21:16>:** Differential-Mode for Corresponding Analog Inputs bits
1 = Channel is differential
0 = Channel is single-ended

bit 10-0 (even) **SIGN<21:16>:** Output Data Sign for Corresponding Analog Inputs bits
1 = Channel output data is signed
0 = Channel output data is unsigned

REGISTER 22-22: ADIEL: ADC INTERRUPT ENABLE REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IE<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IE<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **IE<15:0>:** Common Interrupt Enable bits
1 = Common and individual interrupts are enabled for the corresponding channel
0 = Common and individual interrupts are disabled for the corresponding channel

REGISTER 22-23: ADIEH: ADC INTERRUPT ENABLE REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	IE<21:16>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **IE<21:16>:** Common Interrupt Enable bits
1 = Common and individual interrupts are enabled for the corresponding channel
0 = Common and individual interrupts are disabled for the corresponding channel

REGISTER 22-24: ADSTATL: ADC DATA READY STATUS REGISTER LOW

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
AN<15:8>RDY							
bit 15							bit 8

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
AN<7:0>RDY							
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **AN<15:0>RDY:** Common Interrupt Enable for Corresponding Analog Inputs bits
1 = Channel conversion result is ready in the corresponding ADCBUFx register
0 = Channel conversion result is not ready

REGISTER 22-25: ADSTATH: ADC DATA READY STATUS REGISTER HIGH

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	AN<21:16>RDY					
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **AN<21:16>RDY:** Common Interrupt Enable for Corresponding Analog Inputs bits
1 = Channel conversion result is ready in the corresponding ADCBUFx register
0 = Channel conversion result is not ready

REGISTER 22-26: ADTRIGxL: ADC CHANNEL TRIGGER x SELECTION REGISTER LOW (x = 0 to 5)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+1)<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x)<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **TRGSRC(4x+1)<4:0>:** Trigger Source Selection for Corresponding Analog Inputs bits

11111 = ADTRG31
11110 = PTG Trigger Output 12
11101 = PWM Generator 6 current-limit trigger
11100 = PWM Generator 5 current-limit trigger
11011 = PWM Generator 4 current-limit trigger
11010 = PWM Generator 3 current-limit trigger
11001 = PWM Generator 2 current-limit trigger
11000 = PWM Generator 1 current-limit trigger
10111 = Output Compare 2 trigger
10110 = Output Compare 1 trigger
10101 = CLC2 output
10100 = PWM Generator 6 secondary trigger
10011 = PWM Generator 5 secondary trigger
10010 = PWM Generator 4 secondary trigger
10001 = PWM Generator 3 secondary trigger
10000 = PWM Generator 2 secondary trigger
01111 = PWM Generator 1 secondary trigger
01110 = PWM secondary Special Event Trigger
01101 = Timer2 period match
01100 = Timer1 period match
01011 = CLC1 output
01010 = PWM Generator 6 primary trigger
01001 = PWM Generator 5 primary trigger
01000 = PWM Generator 4 primary trigger
00111 = PWM Generator 3 primary trigger
00110 = PWM Generator 2 primary trigger
00101 = PWM Generator 1 primary trigger
00100 = PWM Special Event Trigger
00011 = Reserved
00010 = Level software trigger
00001 = Common software trigger
00000 = No trigger is enabled

bit 7-5 **Unimplemented:** Read as ‘0’

REGISTER 22-26: ADTRIGxL: ADC CHANNEL TRIGGER x SELECTION REGISTER LOW (x = 0 to 5) (CONTINUED)

bit 4-0 **TRGSRC(4x)<4:0>:** Trigger Source Selection for Corresponding Analog Inputs bits

`11111` = ADTRG31
`11110` = PTG Trigger Output 30
`11101` = PWM Generator 6 current-limit trigger
`11100` = PWM Generator 5 current-limit trigger
`11011` = PWM Generator 4 current-limit trigger
`11010` = PWM Generator 3 current-limit trigger
`11001` = PWM Generator 2 current-limit trigger
`11000` = PWM Generator 1 current-limit trigger
`10111` = Output Compare 2 trigger
`10110` = Output Compare 1 trigger
`10101` = CLC2 output
`10100` = PWM Generator 6 secondary trigger
`10011` = PWM Generator 5 secondary trigger
`10010` = PWM Generator 4 secondary trigger
`10001` = PWM Generator 3 secondary trigger
`10000` = PWM Generator 2 secondary trigger
`01111` = PWM Generator 1 secondary trigger
`01110` = PWM secondary Special Event Trigger
`01101` = Timer2 period match
`01100` = Timer1 period match
`01011` = CLC1 output
`01010` = PWM Generator 6 primary trigger
`01001` = PWM Generator 5 primary trigger
`01000` = PWM Generator 4 primary trigger
`00111` = PWM Generator 3 primary trigger
`00110` = PWM Generator 2 primary trigger
`00101` = PWM Generator 1 primary trigger
`00100` = PWM Special Event Trigger
`00011` = Reserved
`00010` = Level software trigger
`00001` = Common software trigger
`00000` = No trigger is enabled

REGISTER 22-27: ADTRIGxH: ADC CHANNEL TRIGGER x SELECTION REGISTER HIGH (x = 0 to 5)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+3)<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+2)<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12-8 **TRGSRC(4x+3)<4:0>:** Trigger Source Selection for Corresponding Analog Inputs bits

`11111` = ADTRG31
`11110` = PTG Trigger Output 30
`11101` = PWM Generator 6 current-limit trigger
`11100` = PWM Generator 5 current-limit trigger
`11011` = PWM Generator 4 current-limit trigger
`11010` = PWM Generator 3 current-limit trigger
`11001` = PWM Generator 2 current-limit trigger
`11000` = PWM Generator 1 current-limit trigger
`10111` = Output Compare 2 trigger
`10110` = Output Compare 1 trigger
`10101` = CLC2 output
`10100` = PWM Generator 6 secondary trigger
`10011` = PWM Generator 5 secondary trigger
`10010` = PWM Generator 4 secondary trigger
`10001` = PWM Generator 3 secondary trigger
`10000` = PWM Generator 2 secondary trigger
`01111` = PWM Generator 1 secondary trigger
`01110` = PWM secondary Special Event Trigger
`01101` = Timer2 period match
`01100` = Timer1 period match
`01011` = CLC1 output
`01010` = PWM Generator 6 primary trigger
`01001` = PWM Generator 5 primary trigger
`01000` = PWM Generator 4 primary trigger
`00111` = PWM Generator 3 primary trigger
`00110` = PWM Generator 2 primary trigger
`00101` = PWM Generator 1 primary trigger
`00100` = PWM Special Event Trigger
`00011` = Reserved
`00010` = Level software trigger
`00001` = Common software trigger
`00000` = No trigger is enabled

bit 7-5 **Unimplemented:** Read as '0'

REGISTER 22-27: ADTRIGxH: ADC CHANNEL TRIGGER x SELECTION REGISTER HIGH (x = 0 to 5) (CONTINUED)

bit 4-0 **TRGSRC(4x+2)<4:0>:** Trigger Source Selection for Corresponding Analog Inputs bits

11111 = ADTRG31
11110 = PTG Trigger Output 30
11101 = PWM Generator 6 current-limit trigger
11100 = PWM Generator 5 current-limit trigger
11011 = PWM Generator 4 current-limit trigger
11010 = PWM Generator 3 current-limit trigger
11001 = PWM Generator 2 current-limit trigger
11000 = PWM Generator 1 current-limit trigger
10111 = Output Compare 2 trigger
10110 = Output Compare 1 trigger
10101 = CLC2 output
10100 = PWM Generator 6 secondary trigger
10011 = PWM Generator 5 secondary trigger
10010 = PWM Generator 4 secondary trigger
10001 = PWM Generator 3 secondary trigger
10000 = PWM Generator 2 secondary trigger
01111 = PWM Generator 1 secondary trigger
01110 = PWM secondary Special Event Trigger
01101 = Timer2 period match
01100 = Timer1 period match
01011 = CLC1 output
01010 = PWM Generator 6 primary trigger
01001 = PWM Generator 5 primary trigger
01000 = PWM Generator 4 primary trigger
00111 = PWM Generator 3 primary trigger
00110 = PWM Generator 2 primary trigger
00101 = PWM Generator 1 primary trigger
00100 = PWM Special Event Trigger
00011 = Reserved
00010 = Level software trigger
00001 = Common software trigger
00000 = No trigger is enabled

REGISTER 22-28: ADCAL0L: ADC CALIBRATION REGISTER 0 LOW

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL1RDY	—	—	—	CAL1SKIP	CAL1DIFF	CAL1EN	CAL1RUN
bit 15							bit 8

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL0RDY	—	—	—	CAL0SKIP	CAL0DIFF	CAL0EN	CAL0RUN
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CAL1RDY:** Dedicated ADC Core 1 Calibration Status Flag bit
1 = Dedicated ADC Core 1 calibration is finished
0 = Dedicated ADC Core 1 calibration is in progress

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **CAL1SKIP:** Dedicated ADC Core 1 Calibration Bypass bit
1 = After power-up, the dedicated ADC Core 1 will not be calibrated
0 = After power-up, the dedicated ADC Core 1 will be calibrated

bit 10 **CAL1DIFF:** Dedicated ADC Core 1 Differential-Mode Calibration bit
1 = Dedicated ADC Core 1 will be calibrated in Differential Input mode
0 = Dedicated ADC Core 1 will be calibrated in Single-Ended Input mode

bit 9 **CAL1EN:** Dedicated ADC Core 1 Calibration Enable bit
1 = Dedicated ADC Core 1 calibration bits (CALxRDY, CALxSKIP, CALxDIFF and CALxRUN) can be accessed by software
0 = Dedicated ADC Core 1 calibration bits are disabled

bit 8 **CAL1RUN:** Dedicated ADC Core 1 Calibration Start bit
1 = If this bit is set by software, the dedicated ADC Core 1 calibration cycle is started; this bit is automatically cleared by hardware
0 = Software can start the next calibration cycle

bit 7 **CAL0RDY:** Dedicated ADC Core 0 Calibration Status Flag bit
1 = Dedicated ADC Core 0 calibration is finished
0 = Dedicated ADC Core 0 calibration is in progress

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **CAL0SKIP:** Dedicated ADC Core 0 Calibration Bypass bit
1 = After power-up, the dedicated ADC Core 0 will not be calibrated
0 = After power-up, the dedicated ADC Core 0 will be calibrated

bit 2 **CAL0DIFF:** Dedicated ADC Core 0 Differential-Mode Calibration bit
1 = Dedicated ADC Core 0 will be calibrated in Differential Input mode
0 = Dedicated ADC Core 0 will be calibrated in Single-Ended Input mode

bit 1 **CAL0EN:** Dedicated ADC Core 0 Calibration Enable bit
1 = Dedicated ADC Core 0 calibration bits (CALxRDY, CALxSKIP, CALxDIFF and CALxRUN) can be accessed by software
0 = Dedicated ADC Core 0 calibration bits are disabled

bit 0 **CAL0RUN:** Dedicated ADC Core 0 Calibration Start bit
1 = If this bit is set by software, the dedicated ADC Core 0 calibration cycle is started; this bit is automatically cleared by hardware
0 = Software can start the next calibration cycle

REGISTER 22-29: ADCAL0H: ADC CALIBRATION REGISTER 0 HIGH

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL3RDY	—	—	—	CAL3SKIP	CAL3DIFF	CAL3EN	CAL3RUN
bit 15							bit 8

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL2RDY	—	—	—	CAL2SKIP	CAL2DIFF	CAL2EN	CAL2RUN
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CAL3RDY:** Dedicated ADC Core 3 Calibration Status Flag bit
1 = Dedicated ADC Core 3 calibration is finished
0 = Dedicated ADC Core 3 calibration is in progress

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **CAL3SKIP:** Dedicated ADC Core 3 Calibration Bypass bit
1 = After power-up, the dedicated ADC Core 3 will not be calibrated
0 = After power-up, the dedicated ADC Core 3 will be calibrated

bit 10 **CAL3DIFF:** Dedicated ADC Core 3 Differential-Mode Calibration bit
1 = Dedicated ADC Core 3 will be calibrated in Differential Input mode
0 = Dedicated ADC Core 3 will be calibrated in Single-Ended Input mode

bit 9 **CAL3EN:** Dedicated ADC Core 3 Calibration Enable bit
1 = Dedicated ADC Core 3 calibration bits (CALxRDY, CALxSKIP, CALxDIFF and CALxRUN) can be accessed by software
0 = Dedicated ADC Core 3 calibration bits are disabled

bit 8 **CAL3RUN:** Dedicated ADC Core 3 Calibration Start bit
1 = If this bit is set by software, the dedicated ADC Core 3 calibration cycle is started; this bit is automatically cleared by hardware
0 = Software can start the next calibration cycle

bit 7 **CAL2RDY:** Dedicated ADC Core 2 Calibration Status Flag bit
1 = Dedicated ADC Core 2 calibration is finished
0 = Dedicated ADC Core 2 calibration is in progress

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **CAL2SKIP:** Dedicated ADC Core 2 Calibration Bypass bit
1 = After power-up, the dedicated ADC Core 2 will not be calibrated
0 = After power-up, the dedicated ADC Core 2 will be calibrated

bit 2 **CAL2DIFF:** Dedicated ADC Core 2 Differential-Mode Calibration bit
1 = Dedicated ADC Core 2 will be calibrated in Differential Input mode
0 = Dedicated ADC Core 2 will be calibrated in Single-Ended Input mode

bit 1 **CAL2EN:** Dedicated ADC Core 2 Calibration Enable bit
1 = Dedicated ADC Core 2 calibration bits (CALxRDY, CALxSKIP, CALxDIFF and CALxRUN) can be accessed by software
0 = Dedicated ADC Core 2 calibration bits are disabled

bit 0 **CAL2RUN:** Dedicated ADC Core 2 Calibration Start bit
1 = If this bit is set by software, the dedicated ADC Core 2 calibration cycle is started; this bit is automatically cleared by hardware
0 = Software can start the next calibration cycle

REGISTER 22-30: ADCAL1H: ADC CALIBRATION REGISTER 1 HIGH

R/W-0, HS	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CSHRRDY	—	—	—	CSHRSKIP	CSHRDIFF	CSHREN	CSHRRUN
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:	HS = Hardware Settable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CSHRRDY:** Shared ADC Core Calibration Status Flag bit
1 = Shared ADC core calibration is finished
0 = Shared ADC core calibration is in progress

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **CSHRSKIP:** Shared ADC Core Calibration Bypass bit
1 = After power-up, the shared ADC core will not be calibrated
0 = After power-up, the shared ADC core will be calibrated

bit 10 **CSHRDIFF:** Shared ADC Core Differential-Mode Calibration bit
1 = Shared ADC core will be calibrated in Differential Input mode
0 = Shared ADC core will be calibrated in Single-Ended Input mode

bit 9 **CSHREN:** Shared ADC Core Calibration Enable bit
1 = Shared ADC core calibration bits (CSHRRDY, CSHRSKIP, CSHRDIFF and CSHRRUN) can be accessed by software
0 = Shared ADC core calibration bits are disabled

bit 8 **CSHRRUN:** Shared ADC Core Calibration Start bit
1 = If this bit is set by software, the shared ADC core calibration cycle is started; this bit is cleared automatically by hardware
0 = Software can start the next calibration cycle

bit 7-0 **Unimplemented:** Read as ‘0’

REGISTER 22-31: ADCMPxCON: ADC DIGITAL COMPARATOR x CONTROL REGISTER (x = 0 or 1)

U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	—	CHNL4	CHNL3	CHNL2	CHNL1	CHNL0
bit 15							bit 8

R/W-0	R/W-0	R-0, HC, HS	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN	IE	STAT	BTWN	HIHI	HILO	LOHI	LOLO
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	U = Unimplemented bit, read as ‘0’	
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	HS = Hardware Settable bit

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **CHNL<4:0>:** Input Channel Number bits
If the comparator has detected an event for a channel, this channel number is written to these bits.
11111 = Reserved
•
•
10110 = Reserved
10101 = AN21
10100 = AN20
•
•
00001 = AN1
00000 = AN0

bit 7 **CMPEN:** Comparator Enable bit
1 = Comparator is enabled
0 = Comparator is disabled and the STAT status bit is cleared

bit 6 **IE:** Comparator Common ADC Interrupt Enable bit
1 = Common ADC interrupt will be generated if the comparator detects a comparison event
0 = Common ADC interrupt will not be generated for the comparator

bit 5 **STAT:** Comparator Event Status bit
This bit is cleared by hardware when the channel number is read from the CHNL<4:0> bits.
1 = A comparison event has been detected since the last read of the CHNL<4:0> bits
0 = A comparison event has not been detected since the last read of the CHNL<4:0> bits

bit 4 **BTWN:** Between Low/High Comparator Event bit
1 = Generates a comparator event when ADCMPxLO ≤ ADCBUFx < ADCMPxHI
0 = Does not generate a digital comparator event when ADCMPxLO ≤ ADCBUFx < ADCMPxHI

bit 3 **HIHI:** High/High Comparator Event bit
1 = Generates a digital comparator event when ADCBUFx ≥ ADCMPxHI
0 = Does not generate a digital comparator event when ADCBUFx ≥ ADCMPxHI

bit 2 **HILO:** High/Low Comparator Event bit
1 = Generates a digital comparator event when ADCBUFx < ADCMPxHI
0 = Does not generate a digital comparator event when ADCBUFx < ADCMPxHI

bit 1 **LOHI:** Low/High Comparator Event bit
1 = Generates a digital comparator event when ADCBUFx ≥ ADCMPxLO
0 = Does not generate a digital comparator event when ADCBUFx ≥ ADCMPxLO

bit 0 **LOLO:** Low/Low Comparator Event bit
1 = Generates a digital comparator event when ADCBUFx < ADCMPxLO
0 = Does not generate a digital comparator event when ADCBUFx < ADCMPxLO

REGISTER 22-32: ADCMPxENL: ADC DIGITAL COMPARATOR x CHANNEL ENABLE REGISTER LOW (x = 0 or 1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN<15:8>							
bit 15							bit 8

R/W/0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **CMPEN<15:0>:** Comparator Enable for Corresponding Input Channels bits
1 = Conversion result for corresponding channel is used by the comparator
0 = Conversion result for corresponding channel is not used by the comparator

REGISTER 22-33: ADCMPxENH: ADC DIGITAL COMPARATOR x CHANNEL ENABLE REGISTER HIGH (x = 0 or 1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CMPEN<21:16>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **CMPEN<21:16>:** Comparator Enable for Corresponding Input Channels bits
1 = Conversion result for corresponding channel is used by the comparator
0 = Conversion result for corresponding channel is not used by the comparator

REGISTER 22-34: ADFLxCON: ADC DIGITAL FILTER x CONTROL REGISTER (x = 0 or 1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HSC
FLEN	MODE1	MODE0	OVRSAM2	OVRSAM1	OVRSAM0	IE	RDY
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FLCHSEL4	FLCHSEL3	FLCHSEL2	FLCHSEL1	FLCHSEL0
bit 7							bit 0

Legend:	U = Unimplemented bit, read as ‘0’		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **FLEN:** Filter Enable bit
1 = Filter is enabled
0 = Filter is disabled and the RDY bit is cleared

bit 14-13 **MODE<1:0>:** Filter Mode bits
11 = Averaging mode
10 = Reserved
01 = Reserved
00 = Oversampling mode

bit 12-10 **OVRSAM<2:0>:** Filter Averaging/Oversampling Ratio bits
If MODE<1:0> = 00:
111 = 128x (16-bit result in the ADFLxDAT register is in 12.4 format)
110 = 32x (15-bit result in the ADFLxDAT register is in 12.3 format)
101 = 8x (14-bit result in the ADFLxDAT register is in 12.2 format)
100 = 2x (13-bit result in the ADFLxDAT register is in 12.1 format)
011 = 256x (16-bit result in the ADFLxDAT register is in 12.4 format)
010 = 64x (15-bit result in the ADFLxDAT register is in 12.3 format)
001 = 16x (14-bit result in the ADFLxDAT register is in 12.2 format)
000 = 4x (13-bit result in the ADFLxDAT register is in 12.1 format)
If MODE<1:0> = 11 (12-bit result in the ADFLxDAT register in all instances):
111 = 256x
110 = 128x
101 = 64x
100 = 32x
011 = 16x
010 = 8x
001 = 4x
000 = 2x

bit 9 **IE:** Filter Common ADC Interrupt Enable bit
1 = Common ADC interrupt will be generated when the filter result will be ready
0 = Common ADC interrupt will not be generated for the filter

bit 8 **RDY:** Oversampling Filter Data Ready Flag bit
This bit is cleared by hardware when the result is read from the ADFLxDAT register.
1 = Data in the ADFLxDAT register is ready
0 = The ADFLxDAT register has been read and new data in the ADFLxDAT register is not ready

bit 7-5 **Unimplemented:** Read as ‘0’

REGISTER 22-34: ADFLxCON: ADC DIGITAL FILTER x CONTROL REGISTER (x = 0 or 1) (CONTINUED)

bit 4-0 **FLCHSEL<4:0>:** Oversampling Filter Input Channel Selection bits

11111 = Reserved
•
•
•
10110 = Reserved
10101 = AN21
10100 = AN20
•
•
•
00001 = AN1
00000 = AN0

NOTES:

23.0 CONTROLLER AREA NETWORK (CAN) MODULE (dsPIC33EPXXXGS80X DEVICES ONLY)

> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Enhanced Controller Area Network (ECAN™)"** (DS70353) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

23.1 Overview

The Controller Area Network (CAN) module is a serial interface, useful for communicating with other CAN modules or microcontroller devices. This interface/protocol was designed to allow communications within noisy environments. The dsPIC33EPXXXGS80X devices contain two CAN modules.

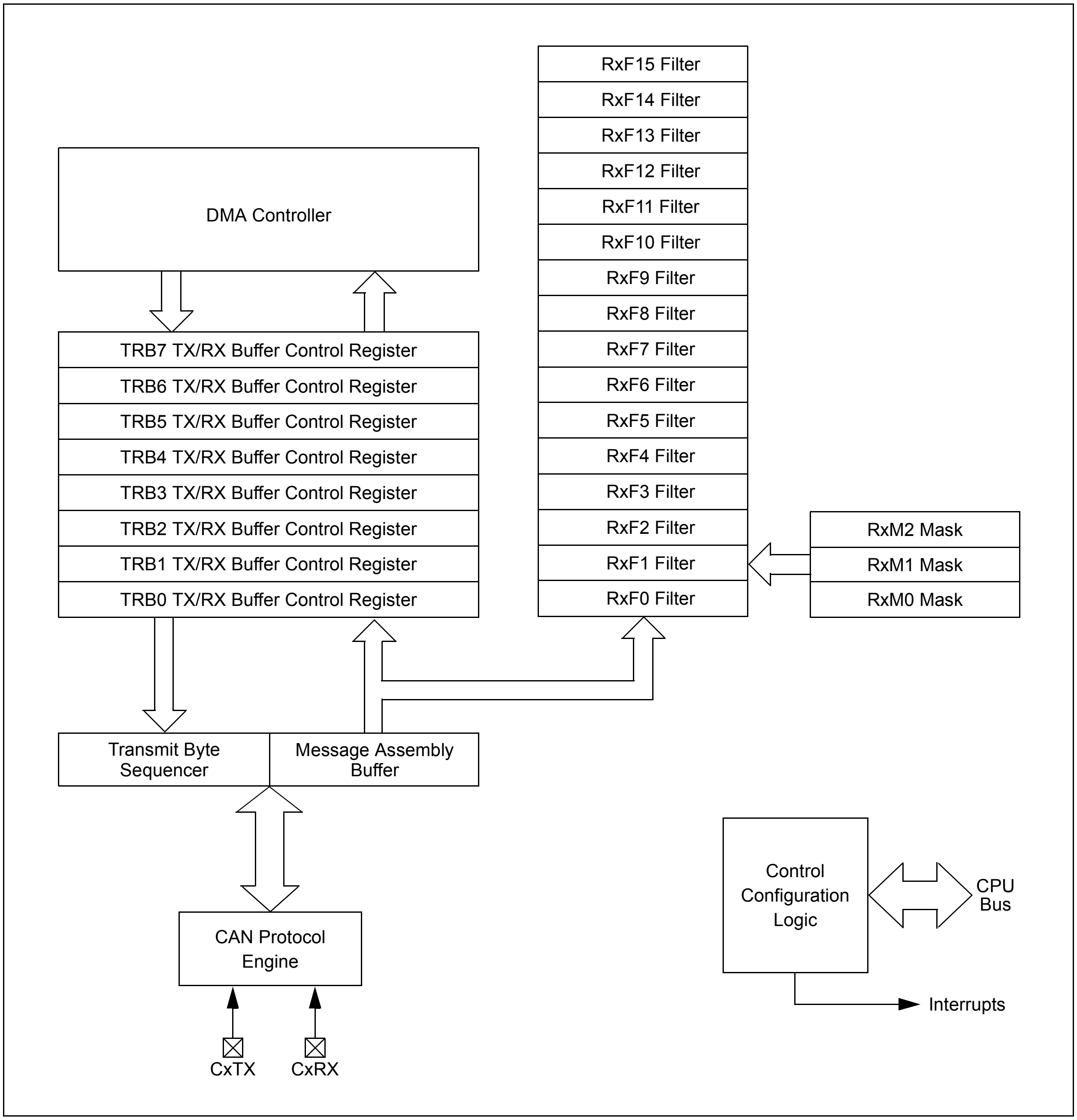
The CAN module is a communication controller, implementing the CAN 2.0 A/B protocol, as defined in the BOSCH CAN specification. The module supports CAN 1.2, CAN 2.0A, CAN 2.0B Passive and CAN 2.0B Active versions of the protocol. The module implementation is a full CAN system. The CAN specification is not covered within this data sheet. The reader can refer to the BOSCH CAN specification for further details.

The CAN module features are as follows:

- Implementation of the CAN Protocol, CAN 1.2, CAN 2.0A and CAN 2.0B
- Standard and Extended Data Frames
- 0-8 Bytes of Data Length
- Programmable Bit Rate, up to 1 Mbit/sec
- Automatic Response to Remote Transmission Requests
- Up to 8 Transmit Buffers with Application Specified Prioritization and Abort Capability (each buffer can contain up to 8 bytes of data)
- Up to 32 Receive Buffers (each buffer can contain up to 8 bytes of data)
- Up to 16 Full (Standard/Extended Identifier) Acceptance Filters
- Three Full Acceptance Filter Masks
- DeviceNet™ Addressing Support
- Programmable Wake-up Functionality with Integrated Low-Pass Filter
- Programmable Loopback mode supports Self-Test Operation
- Signaling via Interrupt Capabilities for All CAN Receiver and Transmitter Error States
- Programmable Clock Source
- Programmable Link to Input Capture 2 (IC2) module for Timestamping and Network Synchronization
- Low-Power Sleep and Idle modes

The CAN bus module consists of a protocol engine and message buffering/control. The CAN protocol engine handles all functions for receiving and transmitting messages on the CAN bus. Messages are transmitted by first loading the appropriate data registers. Status and errors can be checked by reading the appropriate registers. Any message detected on the CAN bus is checked for errors and then matched against filters to see if it should be received and stored in one of the receive registers.

FIGURE 23-1: CANx MODULE BLOCK DIAGRAM



23.2 Modes of Operation

The CANx module can operate in one of several operation modes selected by the user. These modes include:

- Initialization mode
- Disable mode
- Normal Operation mode
- Listen Only mode
- Listen All Messages mode
- Loopback mode

Modes are requested by setting the REQOP<2:0> bits (CxCTRL1<10:8>). Entry into a mode is Acknowledged by monitoring the OPMODE<2:0> bits (CxCTRL1<7:5>). The module does not change the mode and the OPMODEx bits until a change in mode is acceptable, generally during bus Idle time, which is defined as at least 11 consecutive recessive bits.

23.3 CAN Control Registers

REGISTER 23-1: CxCTRL1: CANx CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
—	—	CSIDL	ABAT	CANCKS	REQOP2	REQOP1	REQOP0
bit 15							bit 8

R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
OPMODE2	OPMODE1	OPMODE0	—	CANCAP	—	—	WIN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13 **CSIDL:** CANx Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **ABAT:** Abort All Pending Transmissions bit
1 = Signals all transmit buffers to abort transmission
0 = Module will clear this bit when all transmissions are aborted

bit 11 **CANCKS:** CANx Module Clock (FCAN) Source Select bit
1 = FCAN is equal to 2 * FP
0 = FCAN is equal to FP

bit 10-8 **REQOP<2:0>:** Request Operation Mode bits
111 = Set Listen All Messages mode
110 = Reserved
101 = Reserved
100 = Set Configuration mode
011 = Set Listen Only mode
010 = Set Loopback mode
001 = Set Disable mode
000 = Set Normal Operation mode

bit 7-5 **OPMODE<2:0>**: Operation Mode bits
111 = Module is in Listen All Messages mode
110 = Reserved
101 = Reserved
100 = Module is in Configuration mode
011 = Module is in Listen Only mode
010 = Module is in Loopback mode
001 = Module is in Disable mode
000 = Module is in Normal Operation mode

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **CANCAP:** CANx Message Receive Timer Capture Event Enable bit
1 = Enables input capture based on CAN message receive
0 = Disables CAN capture

bit 2-1 **Unimplemented:** Read as ‘0’

bit 0 **WIN:** SFR Map Window Select bit
1 = Uses filter window
0 = Uses buffer window

REGISTER 23-2: CxCTRL2: CANx CONTROL REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	DNCNT<4:0>				
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-5 **Unimplemented:** Read as ‘0’

bit 4-0 **DNCNT<4:0>:** DeviceNet™ Filter Bit Number bits

10010-11111 = Invalid selection
10001 = Compare up to Data Byte 3, bit 6 with EID<17>
•
•
•
00001 = Compare up to Data Byte 1, bit 7 with EID<0>
00000 = Do not compare data bytes

REGISTER 23-3: CxVEC: CANx INTERRUPT CODE REGISTER

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
—	ICODE6	ICODE5	ICODE4	ICODE3	ICODE2	ICODE1	ICODE0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **FILHIT<4:0>:** Filter Hit Number bits
10000-11111 = Reserved
01111 = Filter 15
•
•
•
00001 = Filter 1
00000 = Filter 0

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **ICODE<6:0>:** Interrupt Flag Code bits
1000101-1111111 = Reserved
1000100 = FIFO almost full interrupt
1000011 = Receiver overflow interrupt
1000010 = Wake-up interrupt
1000001 = Error interrupt
1000000 = No interrupt
•
•
•
0010000-0111111 = Reserved
0001111 = RB15 buffer interrupt
•
•
•
0001001 = RB9 buffer interrupt
0001000 = RB8 buffer interrupt
0000111 = TRB7 buffer interrupt
0000110 = TRB6 buffer interrupt
0000101 = TRB5 buffer interrupt
0000100 = TRB4 buffer interrupt
0000011 = TRB3 buffer interrupt
0000010 = TRB2 buffer interrupt
0000001 = TRB1 buffer interrupt
0000000 = TRB0 buffer interrupt

REGISTER 23-4: CxFCTRL: CANx FIFO CONTROL REGISTER

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
DMABS2	DMABS1	DMABS0	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FSA4	FSA3	FSA2	FSA1	FSA0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **DMABS<2:0>:** DMA Buffer Size bits
111 = Reserved
110 = 32 buffers in RAM
101 = 24 buffers in RAM
100 = 16 buffers in RAM
011 = 12 buffers in RAM
010 = 8 buffers in RAM
001 = 6 buffers in RAM
000 = 4 buffers in RAM

bit 12-5 **Unimplemented:** Read as ‘0’

bit 4-0 **FSA<4:0>:** FIFO Area Starts with Buffer bits
11111 = Receive Buffer RB31
11110 = Receive Buffer RB30
•
•
•
00001 = Transmit/Receive Buffer TRB1
00000 = Transmit/Receive Buffer TRB0

REGISTER 23-5: CxFIFO: CANx FIFO STATUS REGISTER

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FBP5	FBP4	FBP3	FBP2	FBP1	FBP0
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FNRB5	FNRB4	FNRB3	FNRB2	FNRB1	FNRB0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **FBP<5:0>:** FIFO Buffer Pointer bits

011111 = RB31 buffer
011110 = RB30 buffer
•
•
•
000001 = TRB1 buffer
000000 = TRB0 buffer

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **FNRB<5:0>:** FIFO Next Read Buffer Pointer bits

011111 = RB31 buffer
011110 = RB30 buffer
•
•
•
000001 = TRB1 buffer
000000 = TRB0 buffer

REGISTER 23-6: CxINTF: CANx INTERRUPT FLAG REGISTER

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN
bit 15							bit 8

R/C-0	R/C-0	R/C-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF
bit 7							bit 0

Legend:	C = Writable bit, but only ‘0’ can be Written to Clear bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **Unimplemented:** Read as ‘0’

bit 13 **TXBO:** Transmitter in Error State Bus Off bit
1 = Transmitter is in Bus Off state
0 = Transmitter is not in Bus Off state

bit 12 **TXBP:** Transmitter in Error State Bus Passive bit
1 = Transmitter is in Bus Passive state
0 = Transmitter is not in Bus Passive state

bit 11 **RXBP:** Receiver in Error State Bus Passive bit
1 = Receiver is in Bus Passive state
0 = Receiver is not in Bus Passive state

bit 10 **TXWAR:** Transmitter in Error State Warning bit
1 = Transmitter is in Error Warning state
0 = Transmitter is not in Error Warning state

bit 9 **RXWAR:** Receiver in Error State Warning bit
1 = Receiver is in Error Warning state
0 = Receiver is not in Error Warning state

bit 8 **EWARN:** Transmitter or Receiver in Error State Warning bit
1 = Transmitter or receiver is in Error Warning state
0 = Transmitter or receiver is not in Error Warning state

bit 7 **IVRIF:** Invalid Message Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **WAKIF:** Bus Wake-up Activity Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **ERRIF:** Error Interrupt Flag bit (multiple sources in CxINTF<13:8> register)
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **FIFOIF:** FIFO Almost Full Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **RBOVIF:** RX Buffer Overflow Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 23-6: CxINTF: CANx INTERRUPT FLAG REGISTER (CONTINUED)

bit 1 **RBIF:** RX Buffer Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **TBIF:** TX Buffer Interrupt Flag bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 23-7: CxINTE: CANx INTERRUPT ENABLE REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7 **IVRIE:** Invalid Message Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 6 **WAKIE:** Bus Wake-up Activity Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 5 **ERRIE:** Error Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **FIFOIE:** FIFO Almost Full Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 2 **RBOVIE:** RX Buffer Overflow Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **RBIE:** RX Buffer Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **TBIE:** TX Buffer Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

REGISTER 23-8: CxEC: CANx TRANSMIT/RECEIVE ERROR COUNT REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TERRCNT7	TERRCNT6	TERRCNT5	TERRCNT4	TERRCNT3	TERRCNT2	TERRCNT1	TERRCNT0
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RERRCNT7	RERRCNT6	RERRCNT5	RERRCNT4	RERRCNT3	RERRCNT2	RERRCNT1	RERRCNT0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **TERRCNT<7:0>:** Transmit Error Count bits

bit 7-0 **RERRCNT<7:0>:** Receive Error Count bits

REGISTER 23-9: CxCFG1: CANx BAUD RATE CONFIGURATION REGISTER 1

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-6 **SJW<1:0>:** Synchronization Jump Width bits
11 = Length is 4 x TQ
10 = Length is 3 x TQ
01 = Length is 2 x TQ
00 = Length is 1 x TQ

bit 5-0 **BRP<5:0>:** Baud Rate Prescaler bits
11 1111 = TQ = 2 x 64 x 1/FCAN
•
•
•
00 0010 = TQ = 2 x 3 x 1/FCAN
00 0001 = TQ = 2 x 2 x 1/FCAN
00 0000 = TQ = 2 x 1 x 1/FCAN

REGISTER 23-10: CxCFG2: CANx BAUD RATE CONFIGURATION REGISTER 2

U-0	R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	WAKFIL	—	—	—	SEG2PH2	SEG2PH1	SEG2PH0
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEG2PHTS	SAM	SEG1PH2	SEG1PH1	SEG1PH0	PRSEG2	PRSEG1	PRSEG0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **Unimplemented:** Read as ‘0’

bit 14 **WAKFIL:** Select CAN Bus Line Filter for Wake-up bit
1 = Uses CAN bus line filter for wake-up
0 = CAN bus line filter is not used for wake-up

bit 13-11 **Unimplemented:** Read as ‘0’

bit 10-8 **SEG2PH<2:0>:** Phase Segment 2 bits
111 = Length is 8 x TQ
•
•
•
000 = Length is 1 x TQ

bit 7 **SEG2PHTS:** Phase Segment 2 Time Select bit
1 = Freely programmable
0 = Maximum of SEG1PHx bits or Information Processing Time (IPT), whichever is greater

bit 6 **SAM:** Sample of the CAN Bus Line bit
1 = Bus line is sampled three times at the sample point
0 = Bus line is sampled once at the sample point

bit 5-3 **SEG1PH<2:0>:** Phase Segment 1 bits
111 = Length is 8 x TQ
•
•
•
000 = Length is 1 x TQ

bit 2-0 **PRSEG<2:0>:** Propagation Time Segment bits
111 = Length is 8 x TQ
•
•
•
000 = Length is 1 x TQ

REGISTER 23-11: CxFEN1: CANx ACCEPTANCE FILTER ENABLE REGISTER 1

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **FLTEN<15:0>:** Enable Filter n to Accept Messages bits
`1` = Enables Filter n
`0` = Disables Filter n

REGISTER 23-12: CxBUFPNT1: CANx FILTERS 0-3 BUFFER POINTER REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3BP3	F3BP2	F3BP1	F3BP0	F2BP3	F2BP2	F2BP1	F2BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F1BP3	F1BP2	F1BP1	F1BP0	F0BP3	F0BP2	F0BP1	F0BP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **F3BP<3:0>:** RX Buffer Mask for Filter 3 bits
`1111` = Filter hits received in RX FIFO buffer
`1110` = Filter hits received in RX Buffer 14
•
•
•
`0001` = Filter hits received in RX Buffer 1
`0000` = Filter hits received in RX Buffer 0

bit 11-8 **F2BP<3:0>:** RX Buffer Mask for Filter 2 bits (same values as bits 15-12)

bit 7-4 **F1BP<3:0>:** RX Buffer Mask for Filter 1 bits (same values as bits 15-12)

bit 3-0 **F0BP<3:0>:** RX Buffer Mask for Filter 0 bits (same values as bits 15-12)

REGISTER 23-13: CxBUFPNT2: CANx FILTERS 4-7 BUFFER POINTER REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7BP3	F7BP2	F7BP1	F7BP0	F6BP3	F6BP2	F6BP1	F6BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F5BP3	F5BP2	F5BP1	F5BP0	F4BP3	F4BP2	F4BP1	F4BP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **F7BP<3:0>:** RX Buffer Mask for Filter 7 bits
`1111` = Filter hits received in RX FIFO buffer
`1110` = Filter hits received in RX Buffer 14
•
•
•
`0001` = Filter hits received in RX Buffer 1
`0000` = Filter hits received in RX Buffer 0

bit 11-8 **F6BP<3:0>:** RX Buffer Mask for Filter 6 bits (same values as bits 15-12)

bit 7-4 **F5BP<3:0>:** RX Buffer Mask for Filter 5 bits (same values as bits 15-12)

bit 3-0 **F4BP<3:0>:** RX Buffer Mask for Filter 4 bits (same values as bits 15-12)

REGISTER 23-14: CxBUFPNT3: CANx FILTERS 8-11 BUFFER POINTER REGISTER 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11BP3	F11BP2	F11BP1	F11BP0	F10BP3	F10BP2	F10BP1	F10BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F9BP3	F9BP2	F9BP1	F9BP0	F8BP3	F8BP2	F8BP1	F8BP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **F11BP<3:0>:** RX Buffer Mask for Filter 11 bits
`1111` = Filter hits received in RX FIFO buffer
`1110` = Filter hits received in RX Buffer 14
•
•
•
`0001` = Filter hits received in RX Buffer 1
`0000` = Filter hits received in RX Buffer 0

bit 11-8 **F10BP<3:0>:** RX Buffer Mask for Filter 10 bits (same values as bits 15-12)

bit 7-4 **F9BP<3:0>:** RX Buffer Mask for Filter 9 bits (same values as bits 15-12)

bit 3-0 **F8BP<3:0>:** RX Buffer Mask for Filter 8 bits (same values as bits 15-12)

REGISTER 23-15: CxBUFPNT4: CANx FILTERS 12-15 BUFFER POINTER REGISTER 4

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15BP3	F15BP2	F15BP1	F15BP0	F14BP3	F14BP2	F14BP1	F14BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F13BP3	F13BP2	F13BP1	F13BP0	F12BP3	F12BP2	F12BP1	F12BP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **F15BP<3:0>:** RX Buffer Mask for Filter 15 bits
1111 = Filter hits received in RX FIFO buffer
1110 = Filter hits received in RX Buffer 14
•
•
•
0001 = Filter hits received in RX Buffer 1
0000 = Filter hits received in RX Buffer 0

bit 11-8 **F14BP<3:0>:** RX Buffer Mask for Filter 14 bits (same values as bits 15-12)

bit 7-4 **F13BP<3:0>:** RX Buffer Mask for Filter 13 bits (same values as bits 15-12)

bit 3-0 **F12BP<3:0>:** RX Buffer Mask for Filter 12 bits (same values as bits 15-12)

REGISTER 23-16: CxRXFnSID: CANx ACCEPTANCE FILTER n STANDARD IDENTIFIER REGISTER (n = 0-15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-5 **SID<10:0>:** Standard Identifier bits
1 = Message address bit, SIDx, must be ‘1’ to match filter
0 = Message address bit, SIDx, must be ‘0’ to match filter

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **EXIDE:** Extended Identifier Enable bit
If MIDE = 1:
1 = Matches only messages with Extended Identifier addresses
0 = Matches only messages with Standard Identifier addresses
If MIDE = 0:
Ignores EXIDE bit.

bit 2 **Unimplemented:** Read as ‘0’

bit 1-0 **EID<17:16>:** Extended Identifier bits
1 = Message address bit, EIDx, must be ‘1’ to match filter
0 = Message address bit, EIDx, must be ‘0’ to match filter

REGISTER 23-17: CxRXFnEID: CANx ACCEPTANCE FILTER n EXTENDED IDENTIFIER REGISTER (n = 0-15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **EID<15:0>:** Extended Identifier bits
1 = Message address bit, EIDx, must be ‘1’ to match filter
0 = Message address bit, EIDx, must be ‘0’ to match filter

REGISTER 23-18: CxFMSKSEL1: CANx FILTERS 7-0 MASK SELECTION REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7MSK1	F7MSK0	F6MSK1	F6MSK0	F5MSK1	F5MSK0	F4MSK1	F4MSK0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3MSK1	F3MSK0	F2MSK1	F2MSK0	F1MSK1	F1MSK0	F0MSK1	F0MSK0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14 **F7MSK<1:0>:** Mask Source for Filter 7 bits
11 = Reserved
10 = Acceptance Mask 2 registers contain mask
01 = Acceptance Mask 1 registers contain mask
00 = Acceptance Mask 0 registers contain mask

bit 13-12 **F6MSK<1:0>:** Mask Source for Filter 6 bits (same values as bits 15-14)

bit 11-10 **F5MSK<1:0>:** Mask Source for Filter 5 bits (same values as bits 15-14)

bit 9-8 **F4MSK<1:0>:** Mask Source for Filter 4 bits (same values as bits 15-14)

bit 7-6 **F3MSK<1:0>:** Mask Source for Filter 3 bits (same values as bits 15-14)

bit 5-4 **F2MSK<1:0>:** Mask Source for Filter 2 bits (same values as bits 15-14)

bit 3-2 **F1MSK<1:0>:** Mask Source for Filter 1 bits (same values as bits 15-14)

bit 1-0 **F0MSK<1:0>:** Mask Source for Filter 0 bits (same values as bits 15-14)

REGISTER 23-19: CxFMSKSEL2: CANx FILTERS 15-8 MASK SELECTION REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15MSK1	F15MSK0	F14MSK1	F14MSK0	F13MSK1	F13MSK0	F12MSK1	F12MSK0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11MSK1	F11MSK0	F10MSK1	F10MSK0	F9MSK1	F9MSK0	F8MSK1	F8MSK0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-14	**F15MSK<1:0>:** Mask Source for Filter 15 bits 11 = Reserved 10 = Acceptance Mask 2 registers contain mask 01 = Acceptance Mask 1 registers contain mask 00 = Acceptance Mask 0 registers contain mask
bit 13-12	**F14MSK<1:0>:** Mask Source for Filter 14 bits (same values as bits 15-14)
bit 11-10	**F13MSK<1:0>:** Mask Source for Filter 13 bits (same values as bits 15-14)
bit 9-8	**F12MSK<1:0>:** Mask Source for Filter 12 bits (same values as bits 15-14)
bit 7-6	**F11MSK<1:0>:** Mask Source for Filter 11 bits (same values as bits 15-14)
bit 5-4	**F10MSK<1:0>:** Mask Source for Filter 10 bits (same values as bits 15-14)
bit 3-2	**F9MSK<1:0>:** Mask Source for Filter 9 bits (same values as bits 15-14)
bit 1-0	**F8MSK<1:0>:** Mask Source for Filter 8 bits (same values as bits 15-14)

REGISTER 23-20: CxRXMnSID: CANx ACCEPTANCE FILTER MASK n STANDARD IDENTIFIER REGISTER (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	MIDE	—	EID17	EID16
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-5 **SID<10:0>:** Standard Identifier bits
1 = Includes bit, SIDx, in filter comparison
0 = Bit, SIDx, is a don’t care in filter comparison

bit 4 **Unimplemented:** Read as ‘0’

bit 3 **MIDE:** Identifier Receive Mode bit
1 = Matches only message types (standard or extended address) that correspond to the EXIDE bit in the filter
0 = Matches either standard or extended address message if filters match (i.e., if (Filter SIDx) = (Message SIDx) or if (Filter SIDx/EIDx) = (Message SIDx/EIDx))

bit 2 **Unimplemented:** Read as ‘0’

bit 1-0 **EID<17:16>:** Extended Identifier bits
1 = Includes bit, EIDx, in filter comparison
0 = Bit, EIDx, is a don’t care in filter comparison

REGISTER 23-21: CxRXMnEID: CANx ACCEPTANCE FILTER MASK n EXTENDED IDENTIFIER REGISTER (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **EID<15:0>:** Extended Identifier bits
1 = Includes bit, EIDx, in filter comparison
0 = Bit, EIDx, is a don’t care in filter comparison

REGISTER 23-22: CxRXFUL1: CANx RECEIVE BUFFER FULL REGISTER 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<15:8>							
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<7:0>							
bit 7							bit 0

Legend:	C = Writable bit, but only ‘0’ can be Written to Clear bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **RXFUL<15:0>:** Receive Buffer n Full bits
1 = Buffer is full (set by module)
0 = Buffer is empty (cleared by user software)

REGISTER 23-23: CxRXFUL2: CANx RECEIVE BUFFER FULL REGISTER 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<31:24>							
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<23:16>							
bit 7							bit 0

Legend:	C = Writable bit, but only ‘0’ can be Written to Clear bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **RXFUL<31:16>:** Receive Buffer n Full bits
1 = Buffer is full (set by module)
0 = Buffer is empty (cleared by user software)

REGISTER 23-24: CxRXOVF1: CANx RECEIVE BUFFER OVERFLOW REGISTER 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<15:8>							
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<7:0>							
bit 7							bit 0

Legend:	C = Writable bit, but only ‘0’ can be Written to Clear bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **RXOVF<15:0>:** Receive Buffer n Overflow bits
`1` = Module attempted to write to a full buffer (set by module)
`0` = No overflow condition (cleared by user software)

REGISTER 23-25: CxRXOVF2: CANx RECEIVE BUFFER OVERFLOW REGISTER 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<31:24>							
bit 15							bit 8

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<23:16>							
bit 7							bit 0

Legend:	C = Writable bit, but only ‘0’ can be Written to Clear bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **RXOVF<31:16>:** Receive Buffer n Overflow bits
`1` = Module attempted to write to a full buffer (set by module)
`0` = No overflow condition (cleared by user software)

REGISTER 23-26: CxTRmnCON: CANx TX/RX BUFFER mn CONTROL REGISTER (m = 0,2,4,6; n = 1,3,5,7)

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENn	TXABTn	TXLARBn	TXERRn	TXREQn	RTRENn	TXnPRI1	TXnPRI0
bit 15							bit 8

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENm	TXABTm(1)	TXLARBm(1)	TXERRm(1)	TXREQm	RTRENm	TXmPRI1	TXmPRI0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 See Definition for bits 7-0, controls Buffer n.

bit 7 **TXENm:** TX/RX Buffer m Selection bit
1 = Buffer, TRBm, is a transmit buffer
0 = Buffer, TRBm, is a receive buffer

bit 6 **TXABTm:** Message Aborted bit(1)
1 = Message was aborted
0 = Message completed transmission successfully

bit 5 **TXLARBm:** Message Lost Arbitration bit(1)
1 = Message lost arbitration while being sent
0 = Message did not lose arbitration while being sent

bit 4 **TXERRm:** Error Detected During Transmission bit(1)
1 = A bus error occurred while the message was being sent
0 = A bus error did not occur while the message was being sent

bit 3 **TXREQm:** Message Send Request bit
1 = Requests that a message be sent; the bit automatically clears when the message is successfully sent
0 = Clearing the bit to ‘0’ while set requests a message abort

bit 2 **RTRENm:** Auto-Remote Transmit Enable bit
1 = When a remote transmit is received, TXREQx will be set
0 = When a remote transmit is received, TXREQx will be unaffected

bit 1-0 **TXmPRI<1:0>:** Message Transmission Priority bits
11 = Highest message priority
10 = High intermediate message priority
01 = Low intermediate message priority
00 = Lowest message priority

Note 1: This bit is cleared when TXREQmn is set.

Note: The buffers, SIDx, EIDx, DLCx, Data Field and Receive Status registers, are located in DMA RAM.

23.4 CAN Message Buffers

CAN Message Buffers are part of RAM memory. They are not CAN Special Function Registers. The user application must directly write into the RAM area that is configured for CAN Message Buffers. The location and size of the buffer area is defined by the user application.

BUFFER 21-1: CANx MESSAGE BUFFER WORD 0

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	SID10	SID9	SID8	SID7	SID6
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID5	SID4	SID3	SID2	SID1	SID0	SRR	IDE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12-2 **SID<10:0>:** Standard Identifier bits

bit 1 **SRR:** Substitute Remote Request bit

When IDE = 0:
1 = Message will request remote transmission
0 = Normal message

When IDE = 1:
The SRR bit must be set to '1'.

bit 0 **IDE:** Extended Identifier bit

1 = Message will transmit an Extended Identifier
0 = Message will transmit a Standard Identifier

BUFFER 21-2: CANx MESSAGE BUFFER WORD 1

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	EID<17:14>			
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<13:6>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as '0'

bit 11-0 **EID<17:6>:** Extended Identifier bits

BUFFER 21-3: CANx MESSAGE BUFFER WORD 2

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID5	EID4	EID3	EID2	EID1	EID0	RTR	RB1
bit 15							bit 8

U-x	U-x	U-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	RB0	DLC3	DLC2	DLC1	DLC0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-10 **EID<5:0>:** Extended Identifier bits

bit 9 **RTR:** Remote Transmission Request bit
When IDE = 1:
1 = Message will request remote transmission
0 = Normal message
When IDE = 0:
The RTR bit is ignored.

bit 8 **RB1:** Reserved Bit 1
User must set this bit to ‘0’ per CAN protocol.

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4 **RB0:** Reserved Bit 0
User must set this bit to ‘0’ per CAN protocol.

bit 3-0 **DLC<3:0>:** Data Length Code bits

BUFFER 21-4: CANx MESSAGE BUFFER WORD 3

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 1<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 0<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Byte 1<15:8>:** CANx Message Byte 1 bits

bit 7-0 **Byte 0<7:0>:** CANx Message Byte 0 bits

BUFFER 21-5: CANx MESSAGE BUFFER WORD 4

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 3<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 2<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Byte 3<15:8>:** CANx Message Byte 3 bits

bit 7-0 **Byte 2<7:0>:** CANx Message Byte 2 bits

BUFFER 21-6: CANx MESSAGE BUFFER WORD 5

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 5<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 4<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Byte 5<15:8>:** CANx Message Byte 5 bits

bit 7-0 **Byte 4<7:0>:** CANx Message Byte 4 bits

BUFFER 21-7: CANx MESSAGE BUFFER WORD 6

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 7<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Byte 6<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Byte 7<15:8>:** CANx Message Byte 7 bits

bit 7-0 **Byte 6<7:0>:** CANx Message Byte 6 bits

BUFFER 21-8: CANx MESSAGE BUFFER WORD 7

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	FILHIT<4:0>[1]				
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as ‘0’

bit 12-8 **FILHIT<4:0>:** Filter Hit Code bits[1]
Encodes number of filter that resulted in writing this buffer.

bit 7-0 **Unimplemented:** Read as ‘0’

Note 1: Only written by module for receive buffers, unused for transmit buffers.

NOTES:

24.0 HIGH-SPEED ANALOG COMPARATOR

Note 1: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"High-Speed Analog Comparator Module"** (DS70005128) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The high-speed analog comparator module monitors current and/or voltage transients that may be too fast for the CPU and ADC to capture.

24.1 Features Overview

The Switch Mode Power Supply (SMPS) comparator module offers the following major features:

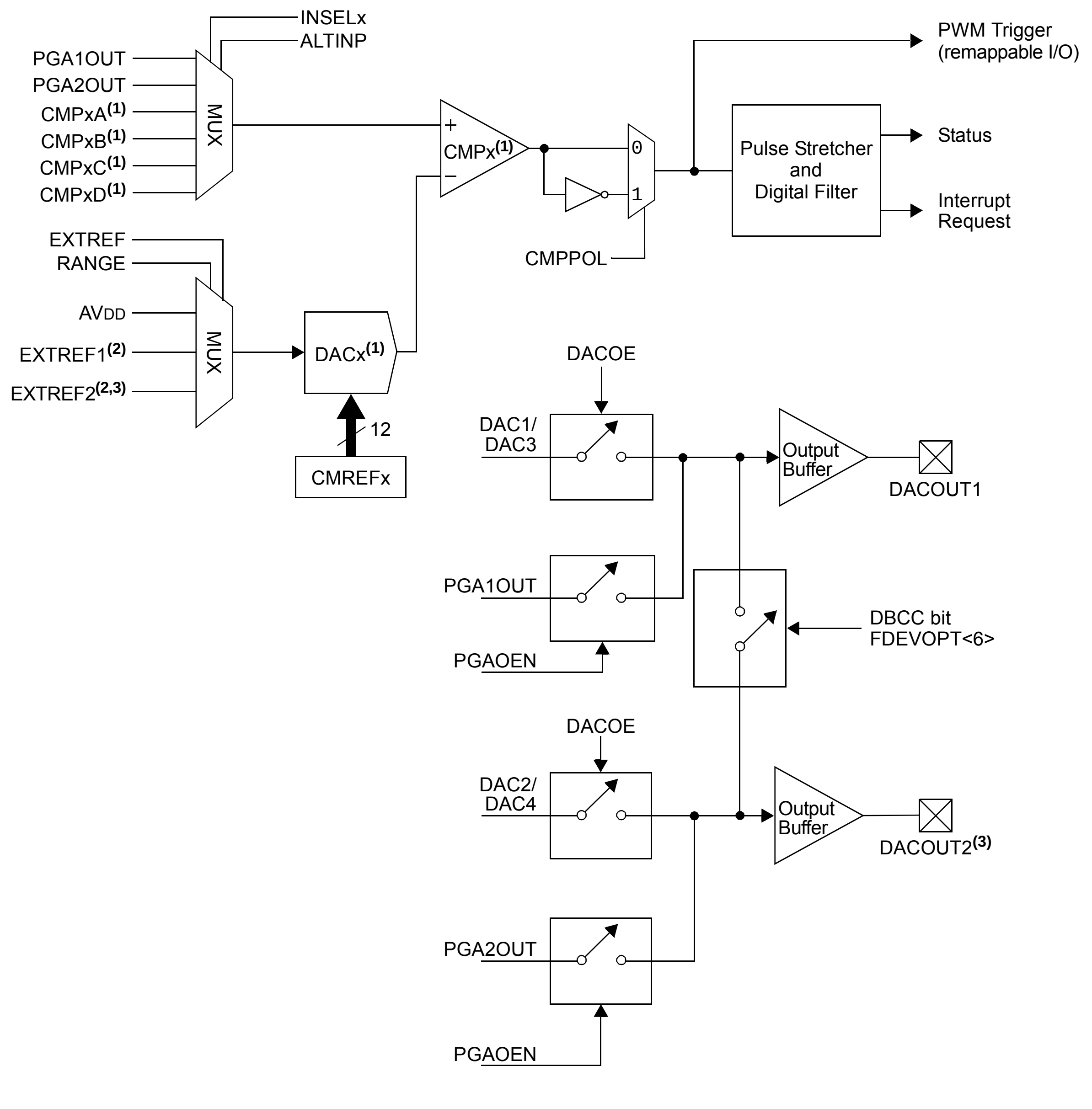
- Four Rail-to-Rail Analog Comparators
- Dedicated 12-Bit DAC for each Analog Comparator
- Up to Six Selectable Input Sources per Comparator:
  - Four external inputs
  - Two internal inputs from the PGAx module
- Programmable Comparator Hysteresis
- Programmable Output Polarity
- Up to Two DAC Outputs to Device Pins
- Multiple Voltage References for the DAC:
  - External References (EXTREF1 or EXTREF2)
  - AVDD
- Interrupt Generation Capability
- Functional Support for PWMx:
  - PWMx duty cycle control
  - PWMx period control
  - PWMx Fault detected

24.2 Module Description

Figure 24-1 shows a functional block diagram of one analog comparator from the high-speed analog comparator module. The analog comparator provides high-speed operation with a typical delay of 15 ns. The negative input of the comparator is always connected to the DACx circuit. The positive input of the comparator is connected to an analog multiplexer that selects the desired source pin.

The analog comparator input pins are typically shared with pins used by the Analog-to-Digital Converter (ADC) module. Both the comparator and the ADC can use the same pins at the same time. This capability enables a user to measure an input voltage with the ADC and detect voltage transients with the comparator.

FIGURE 24-1: HIGH-SPEED ANALOG COMPARATOR x MODULE BLOCK DIAGRAM



Note 1: x = 1-4

2: EXTREF1 is connected to DAC1/DAC3. EXTREF2 is connected to DAC2/DAC4.

3: Not available on all devices.

24.3 Module Applications

This module provides a means for the SMPS dsPIC® DSC devices to monitor voltage and currents in a power conversion application. The ability to detect transient conditions and stimulate the dsPIC DSC processor and/or peripherals, without requiring the processor and ADC to constantly monitor voltages or currents, frees the dsPIC DSC to perform other tasks.

The comparator module has a high-speed comparator and an associated 12-bit DAC that provides a programmable reference voltage to the inverting input of the comparator. The polarity of the comparator output is user-programmable. The output of the module can be used in the following modes:

- Generate an Interrupt
- Trigger an ADC Sample and Convert Process
- Truncate the PWMx Signal (current limit)
- Truncate the PWMx Period (current minimum)
- Disable the PWMx Outputs (Fault latch)

The output of the comparator module may be used in multiple modes at the same time, such as: 1) generate an interrupt, 2) have the ADC take a sample and convert it, and 3) truncate the PWMx output in response to a voltage being detected beyond its expected value.

The comparator module can also be used to wake-up the system from Sleep or Idle mode when the analog input voltage exceeds the programmed threshold voltage.

24.4 Digital-to-Analog Comparator (DAC)

Each analog comparator has a dedicated 12-bit DAC that is used to program the comparator threshold voltage via the CMPxDAC register. The DAC voltage reference source is selected using the EXTREF and RANGE bits in the CMPxCON register.

The EXTREF bit selects either the external voltage reference, EXTREFx, or an internal source as the voltage reference source. The EXTREFx input enables users to connect to a voltage reference that better suits their application. The RANGE bit enables AVDD as the voltage reference source for the DAC when an internal voltage reference is selected.

Note:	EXTREF2 is not available on all devices.

Each DACx has an output enable bit, DACOE, in the CMPxCON register that enables the DACx reference voltage to be routed to an external output pin (DACOUTx). Refer to Figure 24-1 for connecting the DACx output voltage to the DACOUTx pins.

Note 1:	Ensure that multiple DACOE bits are not set in software. The output on the DACOUTx pin will be indeterminate if multiple comparators enable the DACx output.
2:	DACOUT2 is not available on all devices.

24.5 Pulse Stretcher and Digital Logic

The analog comparator can respond to very fast transient signals. After the comparator output is given the desired polarity, the signal is passed to a pulse stretching circuit. The pulse stretching circuit has an asynchronous set function and a delay circuit that ensures the minimum pulse width is three system clock cycles wide to allow the attached circuitry to properly respond to a narrow pulse event.

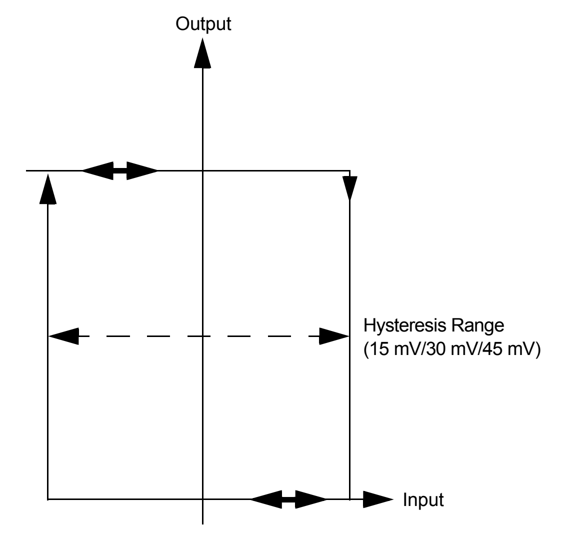
The pulse stretcher circuit is followed by a digital filter. The digital filter is enabled via the FLTREN bit in the CMPxCON register. The digital filter operates with the clock specified via the FCLKSEL bit in the CMPxCON register. The comparator signal must be stable in a high or low state, for at least three of the selected clock cycles, for it to pass through the digital filter.

24.6 Hysteresis

An additional feature of the module is hysteresis control. Hysteresis can be enabled or disabled and its amplitude can be controlled by the HYSSEL<1:0> bits in the CMPxCON register. Three different values are available: 15 mV, 30 mV and 45 mV. It is also possible to select the edge (rising or falling) to which hysteresis is to be applied.

Hysteresis control prevents the comparator output from continuously changing state because of small perturbations (noise) at the input (see Figure 24-2).

FIGURE 24-2: HYSTERESIS CONTROL



24.7 Analog Comparator Resources

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

24.7.1 KEY RESOURCES

- **"High-Speed Analog Comparator Module"** (DS70005128) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

REGISTER 24-1: CMPxCON: COMPARATOR x CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPON	—	CMPSIDL	HYSSEL1	HYSSEL0	FLTREN	FCLKSEL	DACOE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	HC-0, HS	R/W-0	R/W-0	R/W-0
INSEL1	INSEL0	EXTREF	HYSPOL	CMPSTAT	ALTINP	CMPPOL	RANGE
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	HS = Hardware Settable bit	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **CMPON:** Comparator Operating Mode bit
1 = Comparator module is enabled
0 = Comparator module is disabled (reduces power consumption)

bit 14 **Unimplemented:** Read as ‘0’

bit 13 **CMPSIDL:** Comparator Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode.
0 = Continues module operation in Idle mode
If a device has multiple comparators, any CMPSIDL bit set to ‘1’ disables all comparators while in Idle mode.

bit 12-11 **HYSSEL<1:0>:** Comparator Hysteresis Select bits
11 = 45 mV hysteresis
10 = 30 mV hysteresis
01 = 15 mV hysteresis
00 = No hysteresis is selected

bit 10 **FLTREN:** Digital Filter Enable bit
1 = Digital filter is enabled
0 = Digital filter is disabled

bit 9 **FCLKSEL:** Digital Filter and Pulse Stretcher Clock Select bit
1 = Digital filter and pulse stretcher operate with the PWM clock
0 = Digital filter and pulse stretcher operate with the system clock

bit 8 **DACOE:** DACx Output Enable bit
1 = DACx analog voltage is connected to the DACOUTx pin(1)
0 = DACx analog voltage is not connected to the DACOUTx pin

bit 7-6 **INSEL<1:0>:** Input Source Select for Comparator bits
If ALTINP = 0, Select from Comparator Inputs:
11 = Selects CMPxD input pin
10 = Selects CMPxC input pin
01 = Selects CMPxB input pin
00 = Selects CMPxA input pin
If ALTINP = 1, Select from Alternate Inputs:
11 = Reserved
10 = Reserved
01 = Selects PGA2 output
00 = Selects PGA1 output

Note 1: DACOUTx can be associated only with a single comparator at any given time. The software must ensure that multiple comparators do not enable the DACx output by setting their respective DACOE bit.

REGISTER 24-1: CMPxCON: COMPARATOR x CONTROL REGISTER (CONTINUED)

bit 5	**EXTREF:** Enable External Reference bit 1 = External source provides reference to DACx (maximum DAC voltage is determined by the external voltage source) 0 = AVDD provides reference to DACx (maximum DAC voltage is AVDD)
bit 4	**HYSPOL:** Comparator Hysteresis Polarity Select bit 1 = Hysteresis is applied to the falling edge of the comparator output 0 = Hysteresis is applied to the rising edge of the comparator output
bit 3	**CMPSTAT:** Comparator Current State bit Reflects the current output state of Comparator x, including the setting of the CMPPOL bit.
bit 2	**ALTINP:** Alternate Input Select bit 1 = INSEL<1:0> bits select alternate inputs 0 = INSEL<1:0> bits select comparator inputs
bit 1	**CMPPOL:** Comparator Output Polarity Control bit 1 = Output is inverted 0 = Output is non-inverted
bit 0	**RANGE:** DACx Output Voltage Range Select bit 1 = AVDD is the maximum DACx output voltage 0 = Unimplemented, do not use

Note 1: DACOUTx can be associated only with a single comparator at any given time. The software must ensure that multiple comparators do not enable the DACx output by setting their respective DACOE bit.

REGISTER 24-2: CMPxDAC: COMPARATOR x DAC CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CMREF<11:8>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMREF<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-12 **Unimplemented:** Read as ‘0’

bit 11-0 **CMREF<11:0>:** Comparator Reference Voltage Select bits

111111111111
•
•
• = ([CMREF<11:0>] * (AV_{DD})/4096) volts (EXTREF = 0)
• or ([CMREF<11:0>] * (EXTREF)/4096) volts (EXTREF = 1)
•
•
000000000000

NOTES:

25.0 PROGRAMMABLE GAIN AMPLIFIER (PGA)

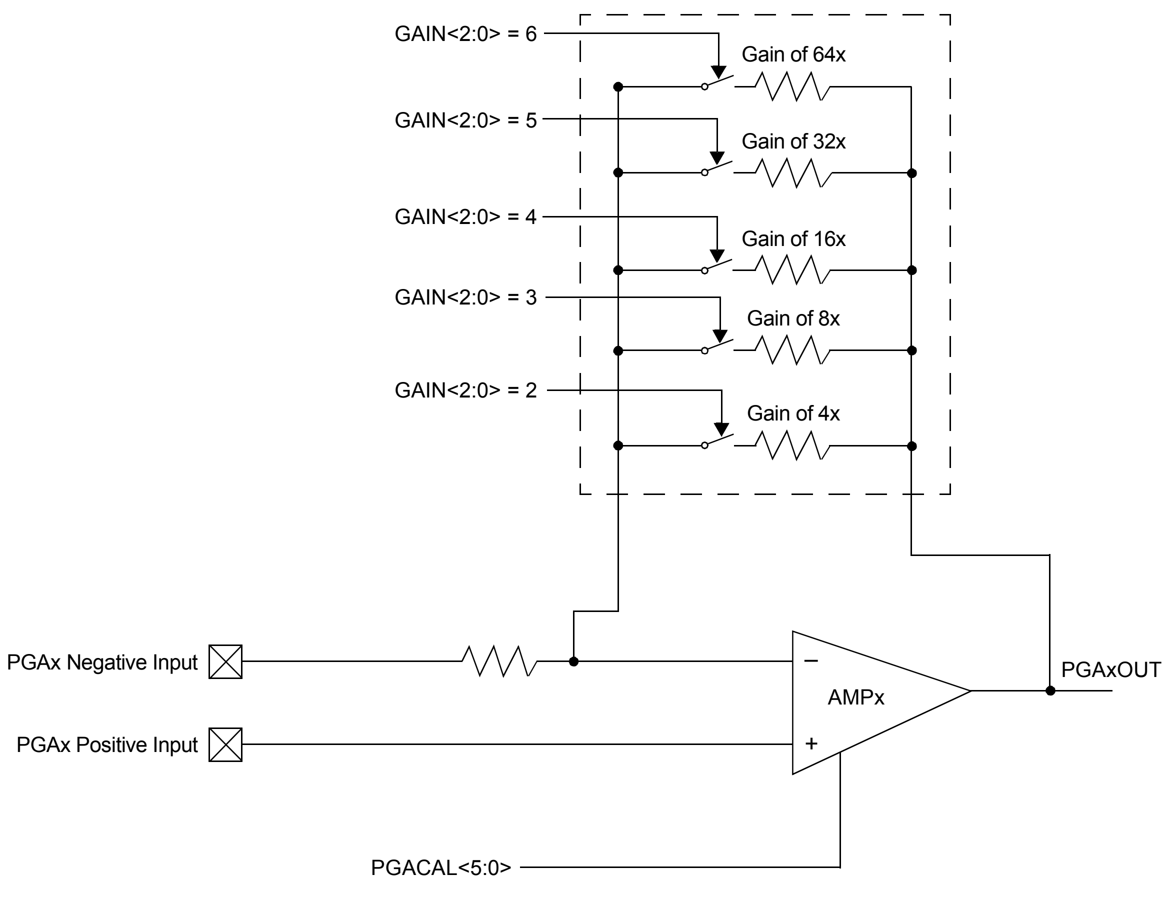
> **Note 1:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Programmable Gain Amplifier (PGA)"** (DS70005146) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
>
> **2:** Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The dsPIC33EPXXXGS70X/80X family devices have two Programmable Gain Amplifiers (PGA1, PGA2). The PGA is an op amp-based, non-inverting amplifier with user-programmable gains. The output of the PGA can be connected to a number of dedicated Sample-and-Hold inputs of the Analog-to-Digital Converter and/or to the high-speed analog comparator module. The PGA has five selectable gains and may be used as a ground referenced amplifier (single-ended) or used with an independent ground reference point.

Key features of the PGA module include:

- Single-Ended or Independent Ground Reference
- Selectable Gains: 4x, 8x, 16x, 32x and 64x
- High Gain Bandwidth
- Rail-to-Rail Output Voltage
- Wide Input Voltage Range

FIGURE 25-1: PGAx MODULE BLOCK DIAGRAM



Note 1: x = 1 and 2.

25.1 Module Description

The Programmable Gain Amplifiers are used to amplify small voltages (i.e., voltages across burden/shunt resistors) to improve the signal-to-noise ratio of the measured signal. The PGAx output voltage can be read by any of the four dedicated Sample-and-Hold circuits on the ADC module. The output voltage can also be fed to the comparator module for overcurrent/voltage protection. Figure 25-2 shows a functional block diagram of the PGAx module. Refer to **Section 22.0 "High-Speed, 12-Bit Analog-to-Digital Converter (ADC)"** and **Section 24.0 "High-Speed Analog Comparator"** for more interconnection details.

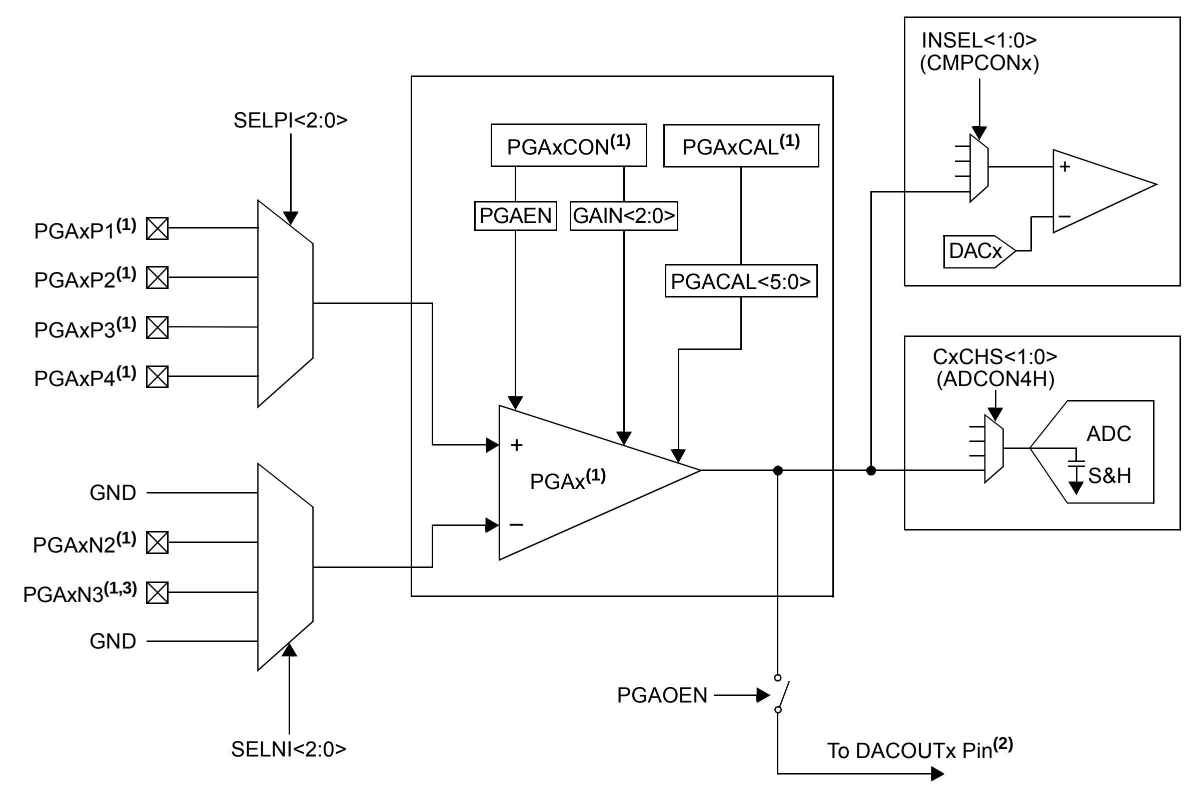
The gain of the PGAx module is selectable via the GAIN<2:0> bits in the PGAxCON register. There are five selectable gains, ranging from 4x to 64x. The SELPI<2:0> and SELNI<2:0> bits in the PGAxCON register select one of four positive/negative inputs to the PGAx module. For single-ended applications, the SELNI<2:0> bits will select the ground as the negative input source. To provide an independent ground reference, the PGAxN2 and PGAxN3 pins are available as the negative input source to the PGAx module.

> **Note 1:** Not all PGA positive/negative inputs are available on all devices. Refer to the specific device pinout for available input source pins.

The output voltage of the PGAx module can be connected to the DACOUTx pin by setting the PGAOEN bit in the PGAxCON register. When the PGAOEN bit is enabled, the output voltage of PGA1 is connected to DACOUT1 and PGA2 is connected to DACOUT2. For devices with a single DACOUTx pin, the output voltage of PGA2 can be connected to DACOUT1 by configuring the DBCC Configuration bit in the FDEVOPT register (FDEVOPT<6>).

If both the DACx output voltage and PGAx output voltage are connected to the DACOUTx pin, the resulting output voltage would be a combination of signals. There is no assigned priority between the PGAx module and the DACx module.

FIGURE 25-2: PGAx FUNCTIONAL BLOCK DIAGRAM



Note 1: x = 1 and 2.
2: The DACOUT2 device pin is only available on 64-pin devices.
3: The PGAxN3 input is not available on 28-pin devices.

25.2 PGA Resources

Many useful resources are provided on the main product page of the Microchip website for the devices listed in this data sheet. This product page contains the latest updates and additional information.

25.2.1 KEY RESOURCES

- **"Programmable Gain Amplifier (PGA)"** (DS70005146) in the *"dsPIC33/PIC24 Family Reference Manual"*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All Related *"dsPIC33/PIC24 Family Reference Manual"* Sections
- Development Tools

REGISTER 25-1: PGAxCON: PGAx CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PGAEN	PGAOEN	SELPI2	SELPI1	SELPI0	SELNI2	SELNI1	SELNI0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	GAIN2	GAIN1	GAIN0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **PGAEN:** PGAx Enable bit
1 = PGAx module is enabled
0 = PGAx module is disabled (reduces power consumption)

bit 14 **PGAOEN:** PGAx Output Enable bit
1 = PGAx output is connected to the DACOUTx pin
0 = PGAx output is not connected to the DACOUTx pin

bit 13-11 **SELPI<2:0>:** PGAx Positive Input Selection bits
111 = Reserved
110 = Reserved
101 = Reserved
100 = Reserved
011 = PGAxP4
010 = PGAxP3
001 = PGAxP2
000 = PGAxP1

bit 10-8 **SELNI<2:0>:** PGAx Negative Input Selection bits
111 = Reserved
110 = Reserved
101 = Reserved
100 = Reserved
011 = Ground (Single-Ended mode)
010 = PGAxN3
001 = PGAxN2
000 = Ground (Single-Ended mode)

bit 7-3 **Unimplemented:** Read as '0'

REGISTER 25-1: PGAxCON: PGAx CONTROL REGISTER (CONTINUED)

bit 2-0 **GAIN<2:0>:** PGAx Gain Selection bits
111 = Reserved
110 = Gain of 64x
101 = Gain of 32x
100 = Gain of 16x
011 = Gain of 8x
010 = Gain of 4x
001 = Reserved
000 = Reserved

REGISTER 25-2: PGAxCAL: PGAx CALIBRATION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PGACAL<5:0>					
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-6 **Unimplemented:** Read as ‘0’

bit 5-0 **PGACAL<5:0>:** PGAx Offset Calibration bits
The calibration values for PGA1 and PGA2 must be copied from Flash addresses, 0x800E48 and 0x800E4C, respectively, into these bits before the module is enabled. Refer to the calibration data address table (Table 27-3) in **Section 27.0 “Special Features”** for more information.

26.0 CONSTANT-CURRENT SOURCE

Note 1:	This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the related section of the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).
2:	Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The constant-current source module is a precision current generator and is used in conjunction with the ADC module to measure the resistance of external resistors connected to device pins.

26.1 Features Overview

The constant-current source module offers the following major features:

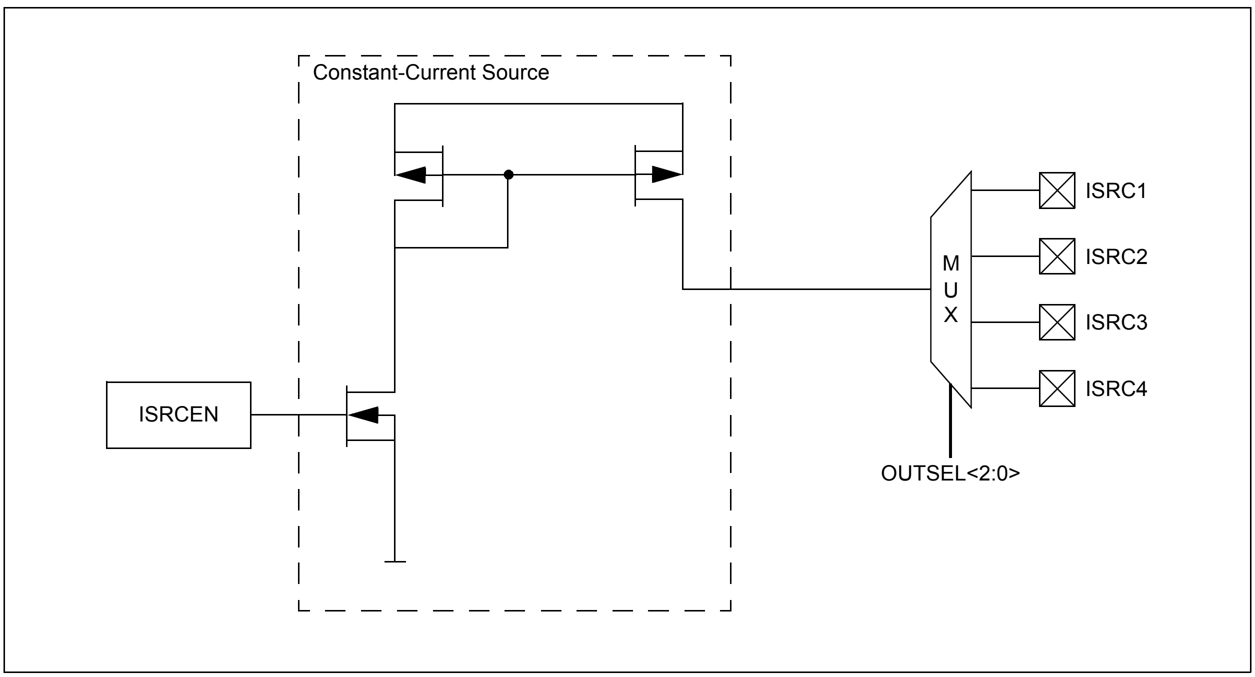
- Constant-Current Generator (10 μA nominal)
- Internal Selectable Connection to One of Four Pins
- Enable/Disable bit

26.2 Module Description

Figure 26-1 shows a functional block diagram of the constant-current source module. It consists of a precision current generator with a nominal value of 10 μA. The module can be enabled and disabled using the ISRCEN bit in the ISRCCON register. The output of the current generator is internally connected to a device pin. The dsPIC33EPXXXGS70X/80X family can have up to 4 selectable current source pins. The OUTSEL<2:0> bits in the ISRCCON register allow selection of the target pin.

The current source is calibrated during testing.

FIGURE 26-1: CONSTANT-CURRENT SOURCE MODULE BLOCK DIAGRAM



26.3 Current Source Control Register

REGISTER 26-1: ISRCCON: CONSTANT-CURRENT SOURCE CONTROL REGISTER

R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
ISRCEN	—	—	—	—	OUTSEL2	OUTSEL1	OUTSEL0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ISRCCAL5	ISRCCAL4	ISRCCAL3	ISRCCAL2	ISRCCAL1	ISRCCAL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15 **ISRCEN:** Constant-Current Source Enable bit
1 = Current source is enabled
0 = Current source is disabled

bit 14-11 **Unimplemented:** Read as ‘0’

bit 10-8 **OUTSEL<2:0>:** Output Constant-Current Select bits
111 = Reserved
110 = Reserved
101 = Reserved
100 = Input pin, ISRC4 (AN4)
011 = Input pin, ISRC3 (AN5)
010 = Input pin, ISRC2 (AN6)
001 = Input pin, ISRC1 (AN12)
000 = No output is selected

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **ISRCCAL<5:0>:** Constant-Current Source Calibration bits
The calibration value must be copied from Flash address, 0x800E78, into these bits before the module is enabled. Refer to the calibration data address table (Table 27-3) in **Section 27.0 “Special Features”** for more information.

27.0 SPECIAL FEATURES

Note: This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Device Configuration"** (DS70000618), **"Watchdog Timer and Power-Saving Modes"** (DS70615) and **"CodeGuard™ Intermediate Security"** (DS70005182) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The dsPIC33EPXXXGS70X/80X family devices include several features intended to maximize application flexibility and reliability, and minimize cost through elimination of external components. These are:

- Flexible Configuration
- Watchdog Timer (WDT)
- Code Protection and CodeGuard™ Security
- JTAG Boundary Scan Interface
- In-Circuit Serial Programming™ (ICSP™)
- In-Circuit Emulation
- Brown-out Reset (BOR)

27.1 Configuration Bits

In dsPIC33EPXXXGS70X/80X family devices, the Configuration Words are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored at the end of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 27-1 with detailed descriptions in Table 27-2. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration Shadow registers during device Resets.

For devices operating in Dual Partition Flash modes, the BSEQx bits (FBTSEQ<11:0>) determine which panel is the Active Partition at start-up and the Configuration Words from that panel are loaded into the Configuration Shadow registers.

Note: Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Words for configuration data in their code for the compiler. This is to make certain that program code is not stored in this address when the code is compiled. Program code executing out of configuration space will cause a device Reset.

Note: Performing a page erase operation on the last page of program memory clears the Flash Configuration Words.

TABLE 27-1: CONFIGURATION REGISTER MAP[3]

Name	Address	Device Memory Size (Kbytes)	Bits 23-16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3
FSEC	00AF80	64	—	AIVTDIS	—	—	—	CSS<2:0>			CWRP	GSS<1:0>		GWRP	—	BSEN
	015780	128														
FBSLIM	00AF90	64	—	—	—	—	BSLIM<12:0>									
	015790	128														
FSIGN	00AF90	64	—	Reserved[2]	—	—	—	—	—	—	—	—	—	—	—	—
	015794	128														
FOSCSEL	00AF98	64	—	—	—	—	—	—	—	—	—	IESO	—	—	—	—
	015798	128														
FOSC	00AF9C	64	—	—	—	—	—	—	—	—	PLLKEN	FCKSM<1:0>		IOL1WAY	—	—
	01579C	128														
FWDT	00AFA0	64	—	—	—	—	—	—	—	WDTWIN<1:0>		WINDIS	WDTEN<1:0>		WDTPRE	
	0157A0	128														
FPOR	00AFA4	64	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	0157A4	128														
FICD	00AFA8	64	—	BTSWP	—	—	—	—	—	—	—	Reserved[1]	—	JTAGEN	—	—
	0157A8	128														
FDEVOPT	00AFAC	64	—	—	—	—	—	—	—	—	—	—	DBCC	—	ALTI2C2	ALTI2C1
	0157AC	128														
FALTREG	00AFB0	64	—	—	CTXT4<2:0>			—	CTXT3<3:0>			—	CTXT2 <2:0>			—
	0157B0	128														
FBTSEQ	00AFFC	64	IBSEQ<11:0>					BSEQ<11:0>								
	0157FC	128														
FBOOT[4]	801000	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Note 1: These bits are reserved and must be programmed as '1'.

2: This bit is reserved and must be programmed as '0'.

3: When operating in Dual Partition Flash mode, each partition will have dedicated Configuration registers. On a device Reset, the configuration values of the Active Partition ar swap condition, the configuration settings of the newly Active Partition are ignored.

4: FBOOT resides in configuration memory space.

TABLE 27-2: CONFIGURATION BITS DESCRIPTION

Bit Field	Description
BSS<1:0>	Boot Segment Code-Protect Level bits 11 = Boot Segment is not code-protected other than BWRP 10 = Standard security 0x = High security
BSEN	Boot Segment Control bit 1 = No Boot Segment is enabled 0 = Boot Segment size is determined by the BSLIM<12:0> bits
BWRP	Boot Segment Write-Protect bit 1 = Boot Segment can be written 0 = Boot Segment is write-protected
BSLIM<12:0>	Boot Segment Flash Page Address Limit bits Contains the last active Boot Segment page. The value to be programmed is the inverted page address, such that programming additional '0's can only increase the Boot Segment size (i.e., 0x1FFD = 2 Pages or 1024 IW).
GSS<1:0>	General Segment Code-Protect Level bits 11 = User program memory is not code-protected 10 = Standard security 0x = High security
GWRP	General Segment Write-Protect bit 1 = User program memory is not write-protected 0 = User program memory is write-protected
CWRP	Configuration Segment Write-Protect bit 1 = Configuration data is not write-protected 0 = Configuration data is write-protected
CSS<2:0>	Configuration Segment Code-Protect Level bits 111 = Configuration data is not code-protected 110 = Standard security 10x = Enhanced security 0xx = High security
BTSWP	BOOTSWP Instruction Enable/Disable bit 1 = BOOTSWP instruction is disabled 0 = BOOTSWP instruction is enabled
BSEQ<11:0>	Boot Sequence Number bits (Dual Partition modes only) Relative value defining which partition will be active after device Reset; the partition containing a lower boot number will be active.
IBSEQ<11:0>	Inverse Boot Sequence Number bits (Dual Partition modes only) The one's complement of BSEQ<11:0>; must be calculated by the user and written for device programming. If BSEQx and IBSEQx are not complements of each other, the Boot Sequence Number is considered to be invalid.
AIVTDIS[1]	Alternate Interrupt Vector Table bit 1 = Alternate Interrupt Vector Table is disabled 0 = Alternate Interrupt Vector Table is enabled if INTCON2<8> = 1
IESO	Two-Speed Oscillator Start-up Enable bit 1 = Starts up device with FRC, then automatically switches to the user-selected oscillator source when ready 0 = Starts up device with the user-selected oscillator source
PWMLOCK	PWMx Lock Enable bit 1 = Certain PWMx registers may only be written after a key sequence 0 = PWMx registers may be written without a key sequence

Note 1: The Boot Segment must be present to use the Alternate Interrupt Vector Table.

TABLE 27-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Description
FNOSC<2:0>	Oscillator Selection bits 111 = Fast RC Oscillator with Divide-by-N (FRCDIVN) 110 = Fast RC Oscillator with Divide-by-16 101 = Low-Power RC Oscillator (LPRC) 100 = Reserved; do not use 011 = Primary Oscillator with PLL module (XT+PLL, HS+PLL, EC+PLL) 010 = Primary Oscillator (XT, HS, EC) 001 = Fast RC Oscillator with Divide-by-N with PLL module (FRCPLL) 000 = Fast RC Oscillator (FRC)
FCKSM<1:0>	Clock Switching Mode bits 1x = Clock switching is disabled, Fail-Safe Clock Monitor is disabled 01 = Clock switching is enabled, Fail-Safe Clock Monitor is disabled 00 = Clock switching is enabled, Fail-Safe Clock Monitor is enabled
IOL1WAY	Peripheral Pin Select Configuration bit 1 = Allows only one reconfiguration 0 = Allows multiple reconfigurations
OSCIOFNC	OSC2 Pin Function bit (except in XT and HS modes) 1 = OSC2 is the clock output 0 = OSC2 is a general purpose digital I/O pin
POSCMD<1:0>	Primary Oscillator Mode Select bits 11 = Primary Oscillator is disabled 10 = HS Crystal Oscillator mode 01 = XT Crystal Oscillator mode 00 = EC (External Clock) mode
WDTEN<1:0>	Watchdog Timer Enable bits 11 = Watchdog Timer is always enabled (LPRC oscillator cannot be disabled; clearing the SWDTEN bit in the RCON register will have no effect) 10 = Watchdog Timer is enabled/disabled by user software (LPRC can be disabled by clearing the SWDTEN bit in the RCON register) 01 = Watchdog Timer is enabled only while device is active and is disabled while in Sleep mode; software control is disabled in this mode 00 = Watchdog Timer and SWDTEN bit are disabled
WINDIS	Watchdog Timer Window Enable bit 1 = Watchdog Timer is in Non-Window mode 0 = Watchdog Timer is in Window mode
PLLKEN	PLL Lock Enable bit 1 = PLL lock is enabled 0 = PLL lock is disabled
WDTPRE	Watchdog Timer Prescaler bit 1 = 1:128 0 = 1:32
WDTPOST<3:0>	Watchdog Timer Postscaler bits 1111 = 1:32,768 1110 = 1:16,384 • • • 0001 = 1:2 0000 = 1:1

Note 1: The Boot Segment must be present to use the Alternate Interrupt Vector Table.

TABLE 27-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Description
WDTWIN<1:0>	Watchdog Timer Window Select bits 11 = WDT window is 25% of the WDT period 10 = WDT window is 37.5% of the WDT period 01 = WDT window is 50% of the WDT period 00 = WDT window is 75% of the WDT period
ALTI2C1	Alternate I2C1 Pin bit 1 = I2C1 is mapped to the SDA1/SCL1 pins 0 = I2C1 is mapped to the ASDA1/ASCL1 pins
ALTI2C2	Alternate I2C2 Pin bit 1 = I2C2 is mapped to the SDA2/SCL2 pins 0 = I2C2 is mapped to the ASDA2/ASCL2 pins
JTAGEN	JTAG Enable bit 1 = JTAG is enabled 0 = JTAG is disabled
ICS<1:0>	ICD Communication Channel Select bits 11 = Communicates on PGEC1 and PGED1 10 = Communicates on PGEC2 and PGED2 01 = Communicates on PGEC3 and PGED3 00 = Reserved, do not use
DBCC	DACx Output Cross Connection Select bit 1 = No cross connection between DAC outputs 0 = Interconnects DACOUT1 and DACOUT2
CTXT1<2:0>	Alternate Working Register Set 1 Interrupt Priority Level (IPL) Select bits 111 = Reserved 110 = Assigned to IPL of 7 101 = Assigned to IPL of 6 100 = Assigned to IPL of 5 011 = Assigned to IPL of 4 010 = Assigned to IPL of 3 001 = Assigned to IPL of 2 000 = Assigned to IPL of 1
CTXT2<2:0>	Alternate Working Register Set 2 Interrupt Priority Level (IPL) Select bits 111 = Reserved 110 = Assigned to IPL of 7 101 = Assigned to IPL of 6 100 = Assigned to IPL of 5 011 = Assigned to IPL of 4 010 = Assigned to IPL of 3 001 = Assigned to IPL of 2 000 = Assigned to IPL of 1
CTXT3<2:0>	Alternate Working Register Set 3 Interrupt Priority Level (IPL) Select bits 111 = Reserved 110 = Assigned to IPL of 7 101 = Assigned to IPL of 6 100 = Assigned to IPL of 5 011 = Assigned to IPL of 4 010 = Assigned to IPL of 3 001 = Assigned to IPL of 2 000 = Assigned to IPL of 1

Note 1: The Boot Segment must be present to use the Alternate Interrupt Vector Table.

TABLE 27-2: CONFIGURATION BITS DESCRIPTION (CONTINUED)

Bit Field	Description
CTXT4<2:0>	Alternate Working Register Set 4 Interrupt Priority Level (IPL) Select bits 111 = Reserved 110 = Assigned to IPL of 7 101 = Assigned to IPL of 6 100 = Assigned to IPL of 5 011 = Assigned to IPL of 4 010 = Assigned to IPL of 3 001 = Assigned to IPL of 2 000 = Assigned to IPL of 1
BTMODE<1:0>	Boot Mode Configuration bits 11 = Single Partition mode 10 = Dual Partition mode 01 = Protected Dual Partition mode 00 = Privileged Dual Partition mode

Note 1: The Boot Segment must be present to use the Alternate Interrupt Vector Table.

27.2 Device Calibration and Identification

The PGAx and current source modules on the dsPIC33EPXXXGS70X/80X family devices require Calibration Data registers to improve performance of the module over a wide operating range. These Calibration registers are read-only and are stored in configuration memory space. Prior to enabling the module, the calibration data must be read (TBLPAG and Table Read instruction) and loaded into its respective SFR registers. The device calibration addresses are shown in Table 27-3.

The dsPIC33EPXXXGS70X/80X devices have two Identification registers near the end of configuration memory space that store the Device ID (DEVID) and Device Revision (DEVREV). These registers are used to determine the mask, variant and manufacturing information about the device. These registers are read-only and are shown in Register 27-1 and Register 27-2.

TABLE 27-3: DEVICE CALIBRATION ADDRESSES[(1)]

Calibration Name	Address	Bits 23-16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PGA1CAL	800E48	—	—	—	—	—	—	—	—	—	—	—	PGA1 Calibration Data					
PGA2CAL	800E4C	—	—	—	—	—	—	—	—	—	—	—	PGA2 Calibration Data					
ISRCCAL	800E78	—	—	—	—	—	—	—	—	—	—	—	Current Source Calibration Data					

Note 1: The calibration data must be copied into its respective SFR registers prior to enabling the module.

REGISTER 27-1: DEVID: DEVICE ID REGISTER

R	R	R	R	R	R	R	R
DEVID<23:16>							
bit 23							bit 16

R	R	R	R	R	R	R	R
DEVID<15:8>							
bit 15							bit 8

R	R	R	R	R	R	R	R
DEVID<7:0>							
bit 7							bit 0

Legend:	R = Read-Only bit	U = Unimplemented bit

bit 23-0 **DEVID<23:0>:** Device Identifier bits

REGISTER 27-2: DEVREV: DEVICE REVISION REGISTER

R	R	R	R	R	R	R	R
DEVREV<23:16>							
bit 23							bit 16

R	R	R	R	R	R	R	R
DEVREV<15:8>							
bit 15							bit 8

R	R	R	R	R	R	R	R
DEVREV<7:0>							
bit 7							bit 0

Legend:	R = Read-only bit	U = Unimplemented bit

bit 23-0 **DEVREV<23:0>:** Device Revision bits

27.3 User OTP Memory

The dsPIC33EPXXXGS70X/80X family devices contain 64 words of user One-Time-Programmable (OTP) memory, located at addresses, 0x800F80 through 0x800FFC. The user OTP Words can be used for storing checksum, code revisions, product information, such as serial numbers, system manufacturing dates, manufacturing lot numbers and other application-specific information. These words can only be written once at program time and not at run time; they can be read at run time.

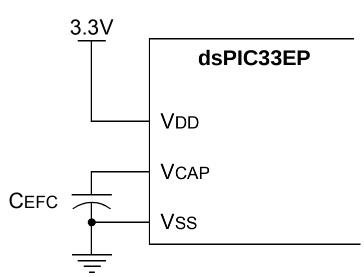
27.4 On-Chip Voltage Regulator

All the dsPIC33EPXXXGS70X/80X family devices power their core digital logic at a nominal 1.8V. This can create a conflict for designs that are required to operate at a higher typical voltage, such as 3.3V. To simplify system design, all devices in the dsPIC33EPXXXGS70X/80X family incorporate an on-chip regulator that allows the device to run its core logic from VDD.

The regulator provides power to the core from the other VDD pins. A low-ESR (less than 1 Ohm) capacitor (such as tantalum or ceramic) must be connected to the VCAP pin (Figure 27-1). This helps to maintain the stability of the regulator. The recommended value for the filter capacitor is provided in Table 30-5, located in **Section 30.0 "Electrical Characteristics"**.

Note:	It is important for the low-ESR capacitor to be placed as close as possible to the VCAP pin.

FIGURE 27-1: CONNECTIONS FOR THE ON-CHIP VOLTAGE REGULATOR[1,2,3]



Note 1: These are typical operating voltages. Refer to Table 30-5 located in **Section 30.0 "Electrical Characteristics"** for the full operating ranges of VDD and VCAP.

2: It is important for the low-ESR capacitor to be placed as close as possible to the VCAP pin.

3: Typical VCAP pin voltage = 1.8V when VDD ≥ VDDMIN.

27.5 Brown-out Reset (BOR)

The Brown-out Reset (BOR) module is based on an internal voltage reference circuit that monitors the regulated supply voltage, VCAP. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (for example, missing portions of the AC cycle waveform due to bad power transmission lines or voltage sags due to excessive current draw when a large inductive load is turned on).

A BOR generates a Reset pulse which resets the device. The BOR selects the clock source based on the device Configuration bit values (FNOSC<2:0> and POSCMD<1:0>).

If an Oscillator mode is selected, the BOR activates the Oscillator Start-up Timer (OST). The system clock is held until OST expires. If the PLL is used, the clock is held until the LOCK bit (OSCCON<5>) is '1'.

Concurrently, the Power-up Timer (PWRT) Time-out (TPWRT) is applied before the internal Reset is released. If TPWRT = 0 and a crystal oscillator is being used, then a nominal delay of TFSCM is applied. The total delay in this case is TFSCM. Refer to Parameter SY35 in Table 30-23 of **Section 30.0 "Electrical Characteristics"** for specific TFSCM values.

The BOR status bit (RCON<1>) is set to indicate that a BOR has occurred. The BOR circuit continues to operate while in Sleep or Idle modes and resets the device should VDD fall below the BOR threshold voltage.

27.6 Watchdog Timer (WDT)

For dsPIC33EPXXXGS70X/80X family devices, the WDT is driven by the LPRC oscillator. When the WDT is enabled, the clock source is also enabled.

27.6.1 PRESCALER/POSTSCALER

The nominal WDT clock source from LPRC is 32 kHz. This feeds a prescaler that can be configured for either 5-bit (divide-by-32) or 7-bit (divide-by-128) operation. The prescaler is set by the WDTPRE Configuration bit. With a 32 kHz input, the prescaler yields a WDT Time-out Period (T_{WDT}), as shown in Parameter SY12 in Table 30-23.

A variable postscaler divides down the WDT prescaler output and allows for a wide range of time-out periods. The postscaler is controlled by the WDTPOST<3:0> Configuration bits (FWDT<3:0>), which allow the selection of 16 settings, from 1:1 to 1:32,768. Using the prescaler and postscaler time-out periods, ranges from 1 ms to 131 seconds can be achieved.

The WDT, prescaler and postscaler are reset:

- On any device Reset
- On the completion of a clock switch, whether invoked by software (i.e., setting the OSWEN bit after changing the NOSCx bits) or by hardware (i.e., Fail-Safe Clock Monitor)
- When a PWRSAV instruction is executed (i.e., Sleep or Idle mode is entered)
- When the device exits Sleep or Idle mode to resume normal operation
- By a CLRWDT instruction during normal execution

Note: The CLRWDT and PWRSAV instructions clear the prescaler and postscaler counts when executed.

27.6.2 SLEEP AND IDLE MODES

If the WDT is enabled, it continues to run during Sleep or Idle modes. When the WDT time-out occurs, the device wakes and code execution continues from where the PWRSAV instruction was executed. The corresponding SLEEP or IDLE bit (RCON<3:2>) needs to be cleared in software after the device wakes up.

27.6.3 ENABLING WDT

The WDT is enabled or disabled by the WDTEN<1:0> Configuration bits in the FWDT Configuration register. When the WDTEN<1:0> Configuration bits have been programmed to '0b11', the WDT is always enabled.

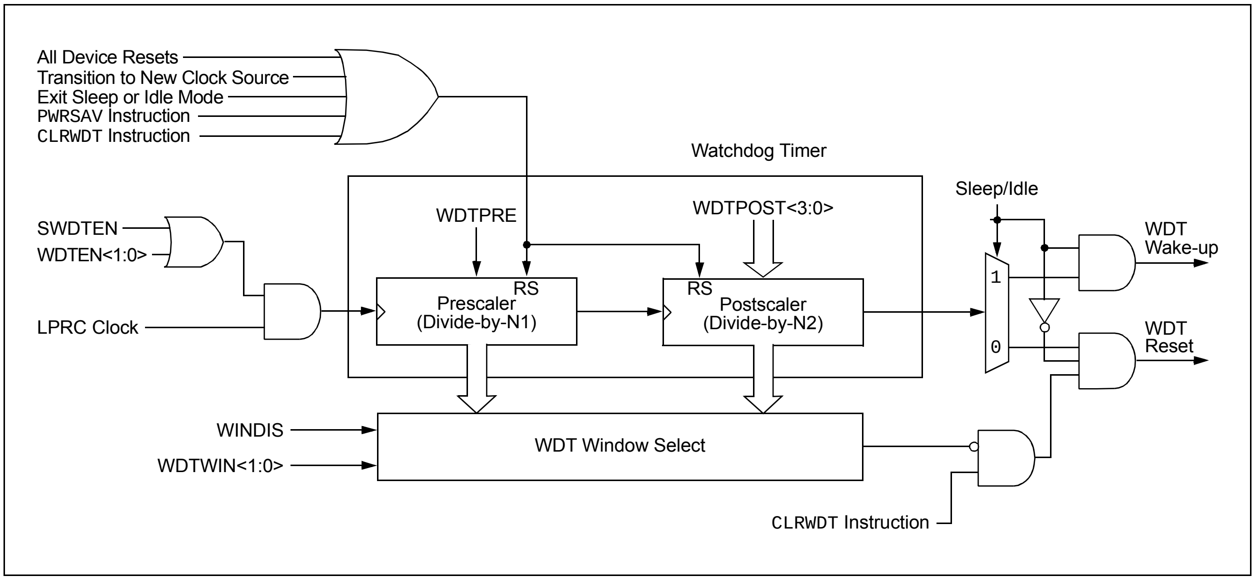
The WDT can be optionally controlled in software when the WDTEN<1:0> Configuration bits have been programmed to '0b10'. The WDT is enabled in software by setting the SWDTEN control bit (RCON<5>). The SWDTEN control bit is cleared on any device Reset. The software WDT option allows the user application to enable the WDT for critical code segments and disables the WDT during non-critical segments for maximum power savings.

The WDT Time-out flag bit, WDTO (RCON<4>), is not automatically cleared following a WDT time-out. To detect subsequent WDT events, the flag must be cleared in software.

27.6.4 WDT WINDOW

The Watchdog Timer has an optional Windowed mode, enabled by programming the WINDIS bit in the WDT Configuration register (FWDT<7>). In the Windowed mode (WINDIS = 0), the WDT should be cleared based on the settings in the programmable Watchdog Timer Window select bits (WDTWIN<1:0>).

FIGURE 27-2: WDT BLOCK DIAGRAM



27.7 JTAG Interface

The dsPIC33EPXXXGS70X/80X family devices implement a JTAG interface, which supports boundary scan device testing. Detailed information on this interface is provided in future revisions of the document.

Note:	Refer to **"Programming and Diagnostics"** (DS70608) in the *"dsPIC33/PIC24 Family Reference Manual"* for further information on usage, configuration and operation of the JTAG interface.

27.8 In-Circuit Serial Programming™ (ICSP™)

The dsPIC33EPXXXGS70X/80X family devices can be serially programmed while in the end application circuit. This is done with two lines for clock and data, and three other lines for power, ground and the programming sequence. Serial programming allows customers to manufacture boards with unprogrammed devices and then program the device just before shipping the product. Serial programming also allows the most recent firmware or a custom firmware to be programmed. Refer to the *"dsPIC33E/PIC24E Flash Programming Specification for Devices with Volatile Configuration Bits"* (DS70663) for details about In-Circuit Serial Programming™ (ICSP™).

Any of the three pairs of programming clock/data pins can be used:

- PGEC1 and PGED1
- PGEC2 and PGED2
- PGEC3 and PGED3

27.9 In-Circuit Debugger

When MPLAB® ICD 3 or REAL ICE™ emulator is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. Debugging functionality is controlled through the PGECx (Emulation/Debug Clock) and PGEDx (Emulation/Debug Data) pin functions.

Any of the three pairs of debugging clock/data pins can be used:

- PGEC1 and PGED1
- PGEC2 and PGED2
- PGEC3 and PGED3

To use the in-circuit debugger function of the device, the design must implement ICSP connections to $\overline{MCLR}$, V_{DD}, V_{SS} and the PGECx/PGEDx pin pair. In addition, when the feature is enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins (PGECx and PGEDx).

27.10 Code Protection and CodeGuard™ Security

dsPIC33EPXXXGS70X/80X devices offer multiple levels of security for protecting individual intellectual property. The program Flash protection can be broken up into three segments: Boot Segment (BS), General Segment (GS) and Configuration Segment (CS). Boot Segment has the highest security privilege and can be thought to have limited restrictions when accessing other segments. General Segment has the least security and is intended for the end user system code. Configuration Segment contains only the device user configuration data which is located at the end of the program memory space.

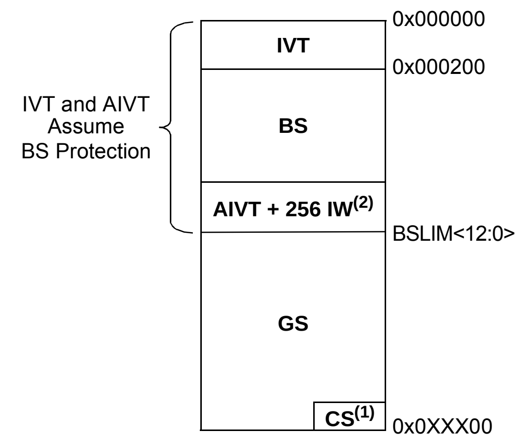
The code protection features are controlled by the Configuration registers, FSEC and FBSLIM. The FSEC register controls the code-protect level for each segment and if that segment is write-protected. The size of BS and GS will depend on the BSLIM<12:0> bits setting and if the Alternate Interrupt Vector Table (AIVT) is enabled. The BSLIM<12:0> bits define the number of pages for BS with each page containing 512 IW. The smallest BS size is one page, which will consist of the Interrupt Vector Table (IVT) and 256 IW of code protection.

If the AIVT is enabled, the last page of BS will contain the AIVT and will not contain any BS code. With AIVT enabled, the smallest BS size is now two pages (1024 IW), with one page for the IVT and BS code, and the other page for the AIVT. Write protection of the BS does not cover the AIVT. The last page of BS can always be programmed or erased by BS code. The General Segment will start at the next page and will consume the rest of program Flash except for the Flash Configuration Words. The IVT will assume GS security only if BS is not enabled. The IVT is protected from being programmed or page erased when either security segment has enabled write protection.

Note:	Refer to **"CodeGuard™ Intermediate Security"** (DS70005182) in the *"dsPIC33/PIC24 Family Reference Manual"* for further information on usage, configuration and operation of CodeGuard Security.

The different device security segments are shown in Figure 27-3. Here, all three segments are shown but are not required. If only basic code protection is required, then GS can be enabled independently or combined with CS, if desired.

FIGURE 27-3: SECURITY SEGMENTS EXAMPLE FOR dsPIC33EPXXXGS70X/80X DEVICES



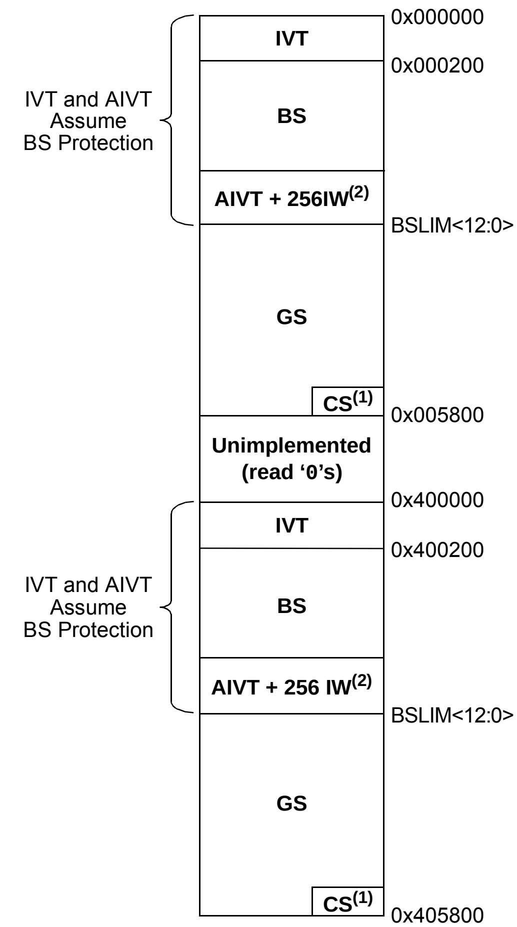
Note 1: If CS is write-protected, the last page (GS + CS) of program memory will be protected from an erase condition.
2: The last half (256 IW) of the last page of BS is unusable program memory.

dsPIC33EPXXXGS70X/80X family devices can be operated in Dual Partition mode, where security is required for each partition. When operating in Dual Partition mode, the Active and Inactive Partitions both contain unique copies of the Reset vector, Interrupt Vector Tables (IVT and AIVT, if enabled) and the Flash Configuration Words. Both partitions have the three security segments described previously. Code may not be executed from the Inactive Partition, but it may be programmed by, and read from, the Active Partition, subject to defined code protection. Figure 27-4 and Figure 27-5 show the different security segments for devices operating in Dual Partition mode.

The device may also operate in a Protected Dual Partition mode or in Privileged Dual Partition mode. In Protected Dual Partition mode, Partition 1 is permanently erase/write-protected. This implementation allows for a “Factory Default” mode, which provides a fail-safe backup image to be stored in Partition 1. For example, a fail-safe bootloader can be placed in Partition 1, along with a fail-safe backup code image, which can be used or rewritten into Partition 2 in the event of a failed Flash update to Partition 2.

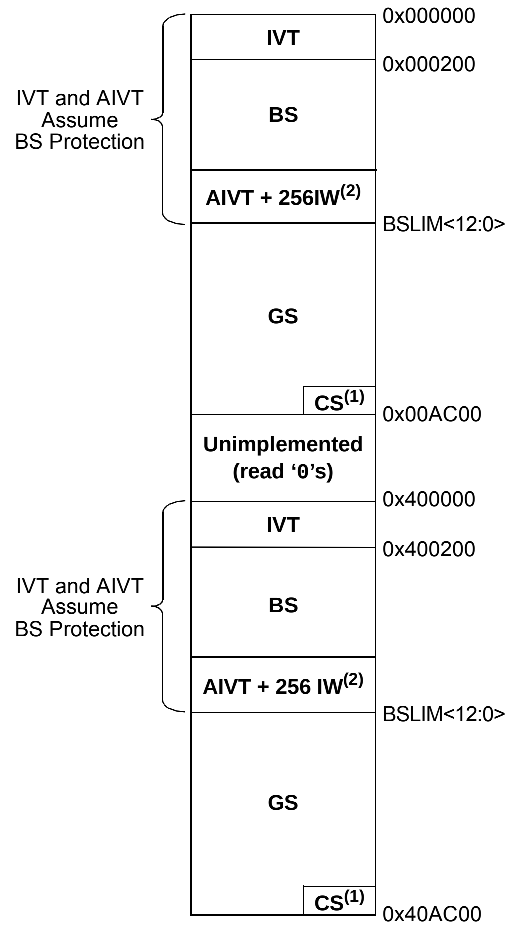
Privileged Dual Partition mode performs the same function as Protected Dual Partition mode, except additional constraints are applied in an effort to prevent code in the Boot Segment and General Segment from being used against each other.

FIGURE 27-4: SECURITY SEGMENTS EXAMPLE FOR dsPIC33EP64GS70X/80X DEVICES (DUAL PARTITION MODES)



Note 1: If CS is write-protected, the last page (GS + CS) of program memory will be protected from an erase condition.
2: The last half (256 IW) of the last page of BS is unusable program memory.

FIGURE 27-5: SECURITY SEGMENTS EXAMPLE FOR dsPIC33EP128GS70X/80X DEVICES (DUAL PARTITION MODES)



Note 1: If CS is write-protected, the last page (GS + CS) of program memory will be protected from an erase condition.

2: The last half (256 IW) of the last page of BS is unusable program memory.

NOTES:

28.0 INSTRUCTION SET SUMMARY

> **Note:** This data sheet summarizes the features of the dsPIC33EPXXXGS70X/80X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to the related section of the "*dsPIC33/PIC24 Family Reference Manual*", which is available from the Microchip web site (www.microchip.com).

The dsPIC33EP instruction set is almost identical to that of the dsPIC30F and dsPIC33F.

Most instructions are a single program memory word (24 bits). Only three instructions require two program memory locations.

Each single-word instruction is a 24-bit word, divided into an 8-bit opcode, which specifies the instruction type and one or more operands, which further specify the operation of the instruction.

The instruction set is highly orthogonal and is grouped into five basic categories:

- Word or byte-oriented operations
- Bit-oriented operations
- Literal operations
- DSP operations
- Control operations

Table 28-1 lists the general symbols used in describing the instructions.

The dsPIC33E instruction set summary in Table 28-2 lists all the instructions, along with the status flags affected by each instruction.

Most word or byte-oriented W register instructions (including barrel shift instructions) have three operands:

- The first source operand, which is typically a register 'Wb' without any address modifier
- The second source operand, which is typically a register 'Ws' with or without an address modifier
- The destination of the result, which is typically a register 'Wd' with or without an address modifier

However, word or byte-oriented file register instructions have two operands:

- The file register specified by the value 'f'
- The destination, which could be either the file register 'f' or the W0 register, which is denoted as 'WREG'

Most bit-oriented instructions (including simple rotate/shift instructions) have two operands:

- The W register (with or without an address modifier) or file register (specified by the value of 'Ws' or 'f')
- The bit in the W register or file register (specified by a literal value or indirectly by the contents of register 'Wb')

The literal instructions that involve data movement can use some of the following operands:

- A literal value to be loaded into a W register or file register (specified by 'k')
- The W register or file register where the literal value is to be loaded (specified by 'Wb' or 'f')

However, literal instructions that involve arithmetic or logical operations use some of the following operands:

- The first source operand, which is a register 'Wb' without any address modifier
- The second source operand, which is a literal value
- The destination of the result (only if not the same as the first source operand), which is typically a register 'Wd' with or without an address modifier

The `MAC` class of DSP instructions can use some of the following operands:

- The accumulator (A or B) to be used (required operand)
- The W registers to be used as the two operands
- The X and Y address space prefetch operations
- The X and Y address space prefetch destinations
- The accumulator write back destination

The other DSP instructions do not involve any multiplication and can include:

- The accumulator to be used (required)
- The source or destination operand (designated as Wso or Wdo, respectively) with or without an address modifier
- The amount of shift specified by a W register 'Wn' or a literal value

The control instructions can use some of the following operands:

- A program memory address
- The mode of the Table Read and Table Write instructions

Most instructions are a single word. Certain double-word instructions are designed to provide all the required information in these 48 bits. In the second word, the 8 MSbs are '0's. If this second word is executed as an instruction (by itself), it executes as a NOP.

The double-word instructions execute in two instruction cycles.

Most single-word instructions are executed in a single instruction cycle, unless a conditional test is true or the Program Counter is changed as a result of the instruction, or a PSV or Table Read is performed. In these cases, the execution takes multiple instruction cycles, with the additional instruction cycle(s) executed as a NOP. Certain instructions that involve skipping over the subsequent instruction require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double-word moves require two cycles.

> **Note:** For more details on the instruction set, refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157).

TABLE 28-1: SYMBOLS USED IN OPCODE DESCRIPTIONS

Field	Description
#text	Means literal defined by "text"
(text)	Means "content of text"
[text]	Means "the location addressed by text"
{ }	Optional field or operation
a ∈ {b, c, d}	a is selected from the set of values b, c, d
<n:m>	Register bit field
.b	Byte mode selection
.d	Double-Word mode selection
.S	Shadow register select
.w	Word mode selection (default)
Acc	One of two accumulators {A, B}
AWB	Accumulator Write-Back Destination Address register ∈ {W13, [W13]+ = 2}
bit4	4-bit bit selection field (used in word addressed instructions) ∈ {0...15}
C, DC, N, OV, Z	MCU Status bits: Carry, Digit Carry, Negative, Overflow, Sticky Zero
Expr	Absolute address, label or expression (resolved by the linker)
f	File register address ∈ {0x0000...0x1FFF}
lit1	1-bit unsigned literal ∈ {0,1}
lit4	4-bit unsigned literal ∈ {0...15}
lit5	5-bit unsigned literal ∈ {0...31}
lit8	8-bit unsigned literal ∈ {0...255}
lit10	10-bit unsigned literal ∈ {0...255} for Byte mode, {0:1023} for Word mode
lit14	14-bit unsigned literal ∈ {0...16384}
lit16	16-bit unsigned literal ∈ {0...65535}
lit23	23-bit unsigned literal ∈ {0...8388608}; LSb must be '0'
None	Field does not require an entry, can be blank
OA, OB, SA, SB	DSP Status bits: ACCA Overflow, ACCB Overflow, ACCA Saturate, ACCB Saturate
PC	Program Counter
Slit10	10-bit signed literal ∈ {-512...511}
Slit16	16-bit signed literal ∈ {-32768...32767}
Slit6	6-bit signed literal ∈ {-16...16}
Wb	Base W register ∈ {W0...W15}
Wd	Destination W register ∈ { Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd] }
Wdo	Destination W register ∈ { Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] }
Wm,Wn	Dividend, Divisor Working register pair (Direct Addressing)

TABLE 28-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

Field	Description
Wm*Wm	Multiplicand and Multiplier Working register pair for Square instructions ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	Multiplicand and Multiplier Working register pair for DSP instructions ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
Wn	One of 16 Working registers ∈ {W0...W15}
Wnd	One of 16 Destination Working registers ∈ {W0...W15}
Wns	One of 16 Source Working registers ∈ {W0...W15}
WREG	W0 (Working register used in file register instructions)
Ws	Source W register ∈ { Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws] }
Wso	Source W register ∈ { Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb] }
Wx	X Data Space Prefetch Address register for DSP instructions ∈ {[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12], none}
Wxd	X Data Space Prefetch Destination register for DSP instructions ∈ {W4...W7}
Wy	Y Data Space Prefetch Address register for DSP instructions ∈ {[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], none}
Wyd	Y Data Space Prefetch Destination register for DSP instructions ∈ {W4...W7}

TABLE 28-2: INSTRUCTION SET OVERVIEW

Base Instr #	Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles[1]	Status Flags Affected
1	ADD	ADD	Acc	Add Accumulators	1	1	OA,OB,SA,SB
		ADD	f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD	f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD	Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD	Wso,#Slit4,Acc	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	ADDC	ADDC	f	f = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	f,WREG	WREG = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND	f	f = f .AND. WREG	1	1	N,Z
		AND	f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
		AND	Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N,Z
		AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR	f	f = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	f,WREG	WREG = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C,N,OV,Z
		ASR	Wb,Wns,Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N,Z
		ASR	Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N,Z
5	BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
		BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
6	BOOTSWP	BOOTSWP		Swap the active and inactive program Flash Space	1	2	None
7	BRA	BRA	C,Expr	Branch if Carry	1	1 (4)	None
		BRA	GE,Expr	Branch if greater than or equal	1	1 (4)	None
		BRA	GEU,Expr	Branch if unsigned greater than or equal	1	1 (4)	None
		BRA	GT,Expr	Branch if greater than	1	1 (4)	None
		BRA	GTU,Expr	Branch if unsigned greater than	1	1 (4)	None
		BRA	LE,Expr	Branch if less than or equal	1	1 (4)	None
		BRA	LEU,Expr	Branch if unsigned less than or equal	1	1 (4)	None
		BRA	LT,Expr	Branch if less than	1	1 (4)	None
		BRA	LTU,Expr	Branch if unsigned less than	1	1 (4)	None
		BRA	N,Expr	Branch if Negative	1	1 (4)	None
		BRA	NC,Expr	Branch if Not Carry	1	1 (4)	None
		BRA	NN,Expr	Branch if Not Negative	1	1 (4)	None
		BRA	NOV,Expr	Branch if Not Overflow	1	1 (4)	None
		BRA	NZ,Expr	Branch if Not Zero	1	1 (4)	None
		BRA	OA,Expr	Branch if Accumulator A overflow	1	1 (4)	None
		BRA	OB,Expr	Branch if Accumulator B overflow	1	1 (4)	None
		BRA	OV,Expr	Branch if Overflow	1	1 (4)	None
		BRA	SA,Expr	Branch if Accumulator A saturated	1	1 (4)	None
		BRA	SB,Expr	Branch if Accumulator B saturated	1	1 (4)	None
		BRA	Expr	Branch Unconditionally	1	4	None
		BRA	Z,Expr	Branch if Zero	1	1 (4)	None
		BRA	Wn	Computed Branch	1	4	None
8	BSET	BSET	f,#bit4	Bit Set f	1	1	None
		BSET	Ws,#bit4	Bit Set Ws	1	1	None

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles[(1)]	Status Flags Affected
9	BSW	BSW.C Ws,Wb	Write C bit to Ws<Wb>	1	1	None
		BSW.Z Ws,Wb	Write Z bit to Ws<Wb>	1	1	None
10	BTG	BTG f,#bit4	Bit Toggle f	1	1	None
		BTG Ws,#bit4	Bit Toggle Ws	1	1	None
11	BTSC	BTSC f,#bit4	Bit Test f, Skip if Clear	1	1 (2 or 3)	None
		BTSC Ws,#bit4	Bit Test Ws, Skip if Clear	1	1 (2 or 3)	None
12	BTSS	BTSS f,#bit4	Bit Test f, Skip if Set	1	1 (2 or 3)	None
		BTSS Ws,#bit4	Bit Test Ws, Skip if Set	1	1 (2 or 3)	None
13	BTST	BTST f,#bit4	Bit Test f	1	1	Z
		BTST.C Ws,#bit4	Bit Test Ws to C	1	1	C
		BTST.Z Ws,#bit4	Bit Test Ws to Z	1	1	Z
		BTST.C Ws,Wb	Bit Test Ws<Wb> to C	1	1	C
		BTST.Z Ws,Wb	Bit Test Ws<Wb> to Z	1	1	Z
14	BTSTS	BTSTS f,#bit4	Bit Test then Set f	1	1	Z
		BTSTS.C Ws,#bit4	Bit Test Ws to C, then Set	1	1	C
		BTSTS.Z Ws,#bit4	Bit Test Ws to Z, then Set	1	1	Z
15	CALL	CALL lit23	Call subroutine	2	4	SFA
		CALL Wn	Call indirect subroutine	1	4	SFA
		CALL.L Wn	Call indirect subroutine (long address)	1	4	SFA
16	CLR	CLR f	f = 0x0000	1	1	None
		CLR WREG	WREG = 0x0000	1	1	None
		CLR Ws	Ws = 0x0000	1	1	None
		CLR Acc,Wx,Wxd,Wy,Wyd,AWB	Clear Accumulator	1	1	OA,OB,SA,SB
17	CLRWDT	CLRWDT	Clear Watchdog Timer	1	1	WDTO,Sleep
18	COM	COM f	f = $\overline{f}$	1	1	N,Z
		COM f,WREG	WREG = $\overline{f}$	1	1	N,Z
		COM Ws,Wd	Wd = $\overline{Ws}$	1	1	N,Z
19	CP	CP f	Compare f with WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit8	Compare Wb with lit8	1	1	C,DC,N,OV,Z
		CP Wb,Ws	Compare Wb with Ws (Wb – Ws)	1	1	C,DC,N,OV,Z
20	CP0	CP0 f	Compare f with 0x0000	1	1	C,DC,N,OV,Z
		CP0 Ws	Compare Ws with 0x0000	1	1	C,DC,N,OV,Z
21	CPB	CPB f	Compare f with WREG, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,#lit8	Compare Wb with lit8, with Borrow	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	Compare Wb with Ws, with Borrow (Wb – Ws – $\overline{C}$)	1	1	C,DC,N,OV,Z
22	CPSEQ	CPSEQ Wb,Wn	Compare Wb with Wn, skip if =	1	1 (2 or 3)	None
	CPBEQ	CPBEQ Wb,Wn,Expr	Compare Wb with Wn, branch if =	1	1 (5)	None
23	CPSGT	CPSGT Wb,Wn	Compare Wb with Wn, skip if >	1	1 (2 or 3)	None
	CPBGT	CPBGT Wb,Wn,Expr	Compare Wb with Wn, branch if >	1	1 (5)	None
24	CPSLT	CPSLT Wb,Wn	Compare Wb with Wn, skip if <	1	1 (2 or 3)	None
	CPBLT	CPBLT Wb,Wn,Expr	Compare Wb with Wn, branch if <	1	1 (5)	None
25	CPSNE	CPSNE Wb,Wn	Compare Wb with Wn, skip if ≠	1	1 (2 or 3)	None
	CPBNE	CPBNE Wb,Wn,Expr	Compare Wb with Wn, branch if ≠	1	1 (5)	None

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles(1)	Status Flags Affected
26	CTXTSWP	CTXTSWP #lit3	Switch CPU register context to context defined by lit3	1	2	None
		CTXTSWP Wn	Switch CPU register context to context defined by Wn	1	2	None
27	DAW	DAW Wn	Wn = decimal adjust Wn	1	1	C
28	DEC	DEC f	f = f – 1	1	1	C,DC,N,OV,Z
		DEC f,WREG	WREG = f – 1	1	1	C,DC,N,OV,Z
		DEC Ws,Wd	Wd = Ws – 1	1	1	C,DC,N,OV,Z
29	DEC2	DEC2 f	f = f – 2	1	1	C,DC,N,OV,Z
		DEC2 f,WREG	WREG = f – 2	1	1	C,DC,N,OV,Z
		DEC2 Ws,Wd	Wd = Ws – 2	1	1	C,DC,N,OV,Z
30	DISI	DISI #lit14	Disable Interrupts for k instruction cycles	1	1	None
31	DIV	DIV.S Wm,Wn	Signed 16/16-bit Integer Divide	1	18	N,Z,C,OV
		DIV.SD Wm,Wn	Signed 32/16-bit Integer Divide	1	18	N,Z,C,OV
		DIV.U Wm,Wn	Unsigned 16/16-bit Integer Divide	1	18	N,Z,C,OV
		DIV.UD Wm,Wn	Unsigned 32/16-bit Integer Divide	1	18	N,Z,C,OV
32	DIVF	DIVF Wm,Wn	Signed 16/16-bit Fractional Divide	1	18	N,Z,C,OV
33	DO	DO #lit15,Expr	Do code to PC + Expr, lit15 + 1 times	2	2	None
		DO Wn,Expr	Do code to PC + Expr, (Wn) + 1 times	2	2	None
34	ED	ED Wm*Wm,Acc,Wx,Wy,Wxd	Euclidean Distance (no accumulate)	1	1	OA,OB,OAB, SA,SB,SAB
35	EDAC	EDAC Wm*Wm,Acc,Wx,Wy,Wxd	Euclidean Distance	1	1	OA,OB,OAB, SA,SB,SAB
36	EXCH	EXCH Wns,Wnd	Swap Wns with Wnd	1	1	None
37	FBCL	FBCL Ws,Wnd	Find Bit Change from Left (MSb) Side	1	1	C
38	FF1L	FF1L Ws,Wnd	Find First One from Left (MSb) Side	1	1	C
39	FF1R	FF1R Ws,Wnd	Find First One from Right (LSb) Side	1	1	C
40	GOTO	GOTO Expr	Go to address	2	4	None
		GOTO Wn	Go to indirect	1	4	None
		GOTO.L Wn	Go to indirect (long address)	1	4	None
41	INC	INC f	f = f + 1	1	1	C,DC,N,OV,Z
		INC f,WREG	WREG = f + 1	1	1	C,DC,N,OV,Z
		INC Ws,Wd	Wd = Ws + 1	1	1	C,DC,N,OV,Z
42	INC2	INC2 f	f = f + 2	1	1	C,DC,N,OV,Z
		INC2 f,WREG	WREG = f + 2	1	1	C,DC,N,OV,Z
		INC2 Ws,Wd	Wd = Ws + 2	1	1	C,DC,N,OV,Z
43	IOR	IOR f	f = f .IOR. WREG	1	1	N,Z
		IOR f,WREG	WREG = f .IOR. WREG	1	1	N,Z
		IOR #lit10,Wn	Wd = lit10 .IOR. Wd	1	1	N,Z
		IOR Wb,Ws,Wd	Wd = Wb .IOR. Ws	1	1	N,Z
		IOR Wb,#lit5,Wd	Wd = Wb .IOR. lit5	1	1	N,Z
44	LAC	LAC Wso,#Slit4,Acc	Load Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
45	LNK	LNK #lit14	Link Frame Pointer	1	1	SFA
46	LSR	LSR f	f = Logical Right Shift f	1	1	C,N,OV,Z
		LSR f,WREG	WREG = Logical Right Shift f	1	1	C,N,OV,Z
		LSR Ws,Wd	Wd = Logical Right Shift Ws	1	1	C,N,OV,Z
		LSR Wb,Wns,Wnd	Wnd = Logical Right Shift Wb by Wns	1	1	N,Z
		LSR Wb,#lit5,Wnd	Wnd = Logical Right Shift Wb by lit5	1	1	N,Z
47	MAC	MAC Wm*Wn,Acc,Wx,Wxd,Wy,Wyd,AWB	Multiply and Accumulate	1	1	OA,OB,OAB, SA,SB,SAB
		MAC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Square and Accumulate	1	1	OA,OB,OAB, SA,SB,SAB

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles[(1)]	Status Flags Affected
48	MOV	MOV f,Wn	Move f to Wn	1	1	None
		MOV f	Move f to f	1	1	None
		MOV f,WREG	Move f to WREG	1	1	None
		MOV #lit16,Wn	Move 16-bit literal to Wn	1	1	None
		MOV.b #lit8,Wn	Move 8-bit literal to Wn	1	1	None
		MOV Wn,f	Move Wn to f	1	1	None
		MOV Wso,Wdo	Move Ws to Wd	1	1	None
		MOV WREG,f	Move WREG to f	1	1	None
		MOV.D Wns,Wd	Move Double from W(ns):W(ns + 1) to Wd	1	2	None
		MOV.D Ws,Wnd	Move Double from Ws to W(nd + 1):W(nd)	1	2	None
49	MOVPAG	MOVPAG #lit10,DSRPAG	Move 10-bit literal to DSRPAG	1	1	None
		MOVPAG #lit8,TBLPAG	Move 8-bit literal to TBLPAG	1	1	None
		MOVPAGW Ws, DSRPAG	Move Ws<9:0> to DSRPAG	1	1	None
		MOVPAGW Ws, TBLPAG	Move Ws<7:0> to TBLPAG	1	1	None
50	MOVSAC	MOVSAC Acc,Wx,Wxd,Wy,Wyd,AWB	Prefetch and store accumulator	1	1	None
51	MPY	MPY Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	Multiply Wm by Wn to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Square Wm to Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
52	MPY.N	MPY.N Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	-(Multiply Wm by Wn) to Accumulator	1	1	None
53	MSC	MSC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd,AWB	Multiply and Subtract from Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
54	MUL	MUL.SS Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * signed(Ws)	1	1	None
		MUL.SS Wb,Ws,Acc	Accumulator = signed(Wb) * signed(Ws)	1	1	None
		MUL.SU Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(Ws)	1	1	None
		MUL.SU Wb,Ws,Acc	Accumulator = signed(Wb) * unsigned(Ws)	1	1	None
		MUL.SU Wb,#lit5,Acc	Accumulator = signed(Wb) * unsigned(lit5)	1	1	None
		MUL.US Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * signed(Ws)	1	1	None
		MUL.US Wb,Ws,Acc	Accumulator = unsigned(Wb) * signed(Ws)	1	1	None
		MUL.UU Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(Ws)	1	1	None
		MUL.UU Wb,#lit5,Acc	Accumulator = unsigned(Wb) * unsigned(lit5)	1	1	None
		MUL.UU Wb,Ws,Acc	Accumulator = unsigned(Wb) * unsigned(Ws)	1	1	None
		MULW.SS Wb,Ws,Wnd	Wnd = signed(Wb) * signed(Ws)	1	1	None
		MULW.SU Wb,Ws,Wnd	Wnd = signed(Wb) * unsigned(Ws)	1	1	None
		MULW.US Wb,Ws,Wnd	Wnd = unsigned(Wb) * signed(Ws)	1	1	None
		MULW.UU Wb,Ws,Wnd	Wnd = unsigned(Wb) * unsigned(Ws)	1	1	None
		MUL.SU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(lit5)	1	1	None
		MUL.SU Wb,#lit5,Wnd	Wnd = signed(Wb) * unsigned(lit5)	1	1	None
		MUL.UU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(lit5)	1	1	None
		MUL.UU Wb,#lit5,Wnd	Wnd = unsigned(Wb) * unsigned(lit5)	1	1	None
		MUL f	W3:W2 = f * WREG	1	1	None

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax		Description	# of Words	# of Cycles[1]	Status Flags Affected
55	NEG	NEG	Acc	Negate Accumulator	1	1	OA,OB,OAB, SA,SB,SAB
		NEG	f	$f = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG	f,WREG	$WREG = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG	Ws,Wd	$Wd = \overline{Ws} + 1$	1	1	C,DC,N,OV,Z
56	NOP	NOP		No Operation	1	1	None
		NOPR		No Operation	1	1	None
57	POP	POP	f	Pop f from Top-of-Stack (TOS)	1	1	None
		POP	Wdo	Pop from Top-of-Stack (TOS) to Wdo	1	1	None
		POP.D	Wnd	Pop from Top-of-Stack (TOS) to W(nd):W(nd + 1)	1	2	None
		POP.S		Pop Shadow Registers	1	1	All
58	PUSH	PUSH	f	Push f to Top-of-Stack (TOS)	1	1	None
		PUSH	Wso	Push Wso to Top-of-Stack (TOS)	1	1	None
		PUSH.D	Wns	Push W(ns):W(ns + 1) to Top-of-Stack (TOS)	1	2	None
		PUSH.S		Push Shadow Registers	1	1	None
59	PWRSAV	PWRSAV	#lit1	Go into Sleep or Idle mode	1	1	WDTO,Sleep
60	RCALL	RCALL	Expr	Relative Call	1	4	SFA
		RCALL	Wn	Computed Call	1	4	SFA
61	REPEAT	REPEAT	#lit15	Repeat Next Instruction lit15 + 1 times	1	1	None
		REPEAT	Wn	Repeat Next Instruction (Wn) + 1 times	1	1	None
62	RESET	RESET		Software device Reset	1	1	None
63	RETFIE	RETFIE		Return from interrupt	1	6 (5)	SFA
64	RETLW	RETLW	#lit10,Wn	Return with literal in Wn	1	6 (5)	SFA
65	RETURN	RETURN		Return from Subroutine	1	6 (5)	SFA
66	RLC	RLC	f	f = Rotate Left through Carry f	1	1	C,N,Z
		RLC	f,WREG	WREG = Rotate Left through Carry f	1	1	C,N,Z
		RLC	Ws,Wd	Wd = Rotate Left through Carry Ws	1	1	C,N,Z
67	RLNC	RLNC	f	f = Rotate Left (No Carry) f	1	1	N,Z
		RLNC	f,WREG	WREG = Rotate Left (No Carry) f	1	1	N,Z
		RLNC	Ws,Wd	Wd = Rotate Left (No Carry) Ws	1	1	N,Z
68	RRC	RRC	f	f = Rotate Right through Carry f	1	1	C,N,Z
		RRC	f,WREG	WREG = Rotate Right through Carry f	1	1	C,N,Z
		RRC	Ws,Wd	Wd = Rotate Right through Carry Ws	1	1	C,N,Z
69	RRNC	RRNC	f	f = Rotate Right (No Carry) f	1	1	N,Z
		RRNC	f,WREG	WREG = Rotate Right (No Carry) f	1	1	N,Z
		RRNC	Ws,Wd	Wd = Rotate Right (No Carry) Ws	1	1	N,Z
70	SAC	SAC	Acc,#Slit4,Wdo	Store Accumulator	1	1	None
		SAC.R	Acc,#Slit4,Wdo	Store Rounded Accumulator	1	1	None
71	SE	SE	Ws,Wnd	Wnd = sign-extended Ws	1	1	C,N,Z
72	SETM	SETM	f	f = 0xFFFF	1	1	None
		SETM	WREG	WREG = 0xFFFF	1	1	None
		SETM	Ws	Ws = 0xFFFF	1	1	None
73	SFTAC	SFTAC	Acc,Wn	Arithmetic Shift Accumulator by (Wn)	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC	Acc,#Slit6	Arithmetic Shift Accumulator by Slit6	1	1	OA,OB,OAB, SA,SB,SAB

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

TABLE 28-2: INSTRUCTION SET OVERVIEW (CONTINUED)

Base Instr #	Assembly Mnemonic	Assembly Syntax	Description	# of Words	# of Cycles[1]	Status Flags Affected
74	SL	SL f	f = Left Shift f	1	1	C,N,OV,Z
		SL f,WREG	WREG = Left Shift f	1	1	C,N,OV,Z
		SL Ws,Wd	Wd = Left Shift Ws	1	1	C,N,OV,Z
		SL Wb,Wns,Wnd	Wnd = Left Shift Wb by Wns	1	1	N,Z
		SL Wb,#lit5,Wnd	Wnd = Left Shift Wb by lit5	1	1	N,Z
75	SUB	SUB Acc	Subtract Accumulators	1	1	OA,OB,OAB, SA,SB,SAB
		SUB f	f = f – WREG	1	1	C,DC,N,OV,Z
		SUB f,WREG	WREG = f – WREG	1	1	C,DC,N,OV,Z
		SUB #lit10,Wn	Wn = Wn – lit10	1	1	C,DC,N,OV,Z
		SUB Wb,Ws,Wd	Wd = Wb – Ws	1	1	C,DC,N,OV,Z
		SUB Wb,#lit5,Wd	Wd = Wb – lit5	1	1	C,DC,N,OV,Z
76	SUBB	SUBB f	f = f – WREG – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB f,WREG	WREG = f – WREG – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB #lit10,Wn	Wn = Wn – lit10 – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,Ws,Wd	Wd = Wb – Ws – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,#lit5,Wd	Wd = Wb – lit5 – ($\overline{C}$)	1	1	C,DC,N,OV,Z
77	SUBR	SUBR f	f = WREG – f	1	1	C,DC,N,OV,Z
		SUBR f,WREG	WREG = WREG – f	1	1	C,DC,N,OV,Z
		SUBR Wb,Ws,Wd	Wd = Ws – Wb	1	1	C,DC,N,OV,Z
		SUBR Wb,#lit5,Wd	Wd = lit5 – Wb	1	1	C,DC,N,OV,Z
78	SUBBR	SUBBR f	f = WREG – f – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR f,WREG	WREG = WREG – f – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,Ws,Wd	Wd = Ws – Wb – ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,#lit5,Wd	Wd = lit5 – Wb – ($\overline{C}$)	1	1	C,DC,N,OV,Z
79	SWAP	SWAP.b Wn	Wn = nibble swap Wn	1	1	None
		SWAP Wn	Wn = byte swap Wn	1	1	None
80	TBLRDH	TBLRDH Ws,Wd	Read Prog<23:16> to Wd<7:0>	1	5	None
81	TBLRDL	TBLRDL Ws,Wd	Read Prog<15:0> to Wd	1	5	None
82	TBLWTH	TBLWTH Ws,Wd	Write Ws<7:0> to Prog<23:16>	1	2	None
83	TBLWTL	TBLWTL Ws,Wd	Write Ws to Prog<15:0>	1	2	None
84	ULNK	ULNK	Unlink Frame Pointer	1	1	SFA
85	XOR	XOR f	f = f .XOR. WREG	1	1	N,Z
		XOR f,WREG	WREG = f .XOR. WREG	1	1	N,Z
		XOR #lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N,Z
		XOR Wb,Ws,Wd	Wd = Wb .XOR. Ws	1	1	N,Z
		XOR Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N,Z
86	ZE	ZE Ws,Wnd	Wnd = Zero-extend Ws	1	1	C,Z,N

Note 1: Read and Read-Modify-Write (e.g., bit operations and logical operations) on non-CPU SFRs incur an additional instruction cycle.

NOTES:

29.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

29.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

29.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16 and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

29.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

29.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

29.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

29.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

29.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

29.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

29.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

29.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

29.11 Demonstration/Development Boards, Evaluation Kits and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

29.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

30.0 ELECTRICAL CHARACTERISTICS

This section provides an overview of the dsPIC33EPXXXGS70X/80X family electrical characteristics. Additional information will be provided in future revisions of this document as it becomes available.

Absolute maximum ratings for the dsPIC33EPXXXGS70X/80X family are listed below. Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these, or any other conditions above the parameters indicated in the operation listings of this specification, is not implied.

Absolute Maximum Ratings[1]

Ambient temperature under bias .... -40°C to +125°C
Storage temperature .... -65°C to +150°C
Voltage on VDD with respect to VSS .... -0.3V to +4.0V
Voltage on any pin that is not 5V tolerant with respect to VSS[3] .... -0.3V to (VDD + 0.3V)
Voltage on any 5V tolerant pin with respect to VSS when VDD ≥ 3.0V[3] .... -0.3V to +5.5V
Voltage on any 5V tolerant pin with respect to Vss when VDD < 3.0V[3] .... -0.3V to +3.6V
Maximum current out of VSS pin .... 300 mA
Maximum current into VDD pin[2] .... 300 mA
Maximum current sunk/sourced by any 4x I/O pin .... 15 mA
Maximum current sunk/sourced by any 8x I/O pin .... 25 mA
Maximum current sunk by all ports[2] .... 200 mA

Note 1: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those, or any other conditions above those indicated in the operation listings of this specification, is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

2: Maximum allowable current is a function of device maximum power dissipation (see Table 30-2).

3: See the **“Pin Diagrams”** section for the 5V tolerant pins.

30.1 DC Characteristics

TABLE 30-1: OPERATING MIPS vs. VOLTAGE

Characteristic	VDD Range (in Volts)	Temperature Range (in °C)	Maximum MIPS
			dsPIC33EPXXXGS70X/80X Family
—	3.0V to 3.6V[1]	-40°C to +85°C	70
—	3.0V to 3.6V[1]	-40°C to +125°C	60

Note 1: Device is functional at VBORMIN < VDD < VDDMIN. Analog modules (ADC, PGAs and comparators) may have degraded performance. Device functionality is tested but not characterized. Refer to Parameter BO10 in Table 30-13 for the minimum and maximum BOR values.

TABLE 30-2: THERMAL OPERATING CONDITIONS

Rating	Symbol	Min.	Typ.	Max.	Unit
Industrial Temperature Devices					
Operating Junction Temperature Range	T_J	-40	—	+125	°C
Operating Ambient Temperature Range	T_A	-40	—	+85	°C
Extended Temperature Devices					
Operating Junction Temperature Range	T_J	-40	—	+140	°C
Operating Ambient Temperature Range	T_A	-40	—	+125	°C
Power Dissipation: Internal Chip Power Dissipation: $P_{INT} = V_{DD} \times (I_{DD} - \Sigma I_{OH})$ I/O Pin Power Dissipation: $I/O = \Sigma (\{V_{DD} - V_{OH}\} \times I_{OH}) + \Sigma (V_{OL} \times I_{OL})$	P_D	$P_{INT} + P_{I/O}$			W
Maximum Allowed Power Dissipation	P_{DMAX}	$(T_J - T_A)/\theta_{JA}$			W

TABLE 30-3: THERMAL PACKAGING CHARACTERISTICS

Characteristic	Symbol	Typ.	Max.	Unit	Notes
Package Thermal Resistance, 80-Pin TQFP 12x12x1 mm	θ_{JA}	53.0	—	°C/W	1
Package Thermal Resistance, 64-Pin TQFP 10x10x1 mm	θ_{JA}	49.0	—	°C/W	1
Package Thermal Resistance, 48-Pin TQFP 7x7x1 mm	θ_{JA}	63.0	—	°C/W	1
Package Thermal Resistance, 44-Pin QFN 8x8 mm	θ_{JA}	29.0	—	°C/W	1
Package Thermal Resistance, 44-Pin TQFP 10x10x1 mm	θ_{JA}	50.0	—	°C/W	1
Package Thermal Resistance, 28-Pin QFN-S 6x6x0.9 mm	θ_{JA}	30.0	—	°C/W	1
Package Thermal Resistance, 28-Pin UQFN 6x6x0.55 mm	θ_{JA}	26.0	—	°C/W	1
Package Thermal Resistance, 28-Pin SOIC 7.50 mm	θ_{JA}	70.0	—	°C/W	1

Note 1: Junction to ambient thermal resistance, Theta-JA (θ_{JA}) numbers are achieved by package simulations.

TABLE 30-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[1] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
Operating Voltage							
DC10	VDD	**Supply Voltage**	3.0	—	3.6	V	
DC12	VDR	**RAM Retention Voltage**[2]	—	—	1.95	V	+25°C, +85°C, +125°C
			—	—	2.0	V	-40°C
DC16	VPOR	**VDD Start Voltage** to Ensure Internal Power-on Reset Signal	—	—	VSS	V	
DC17	SVDD	**VDD Rise Rate** to Ensure Internal Power-on Reset Signal	1.0	—	—	V/ms	0V-3V in 3 ms

Note 1: Device is functional at VBORMIN < VDD < VDDMIN. Analog modules (ADC, PGAs and comparators) may have degraded performance. Device functionality is tested but not characterized. Refer to Parameter BO10 in Table 30-13 for the minimum and maximum BOR values.

2: This is the limit to which VDD may be lowered and the RAM contents will always be retained.

TABLE 30-5: FILTER CAPACITOR (CEFC) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated): Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended							
Param No.	Symbol	Characteristics	Min.	Typ.	Max.	Units	Comments
	CEFC	External Filter Capacitor Value[1]	4.7	—	10	μF	Capacitor must have a low series resistance (<1 Ohm)

Note 1: Typical VCAP Voltage = 1.8 volts when VDD ≥ VDDMIN.

TABLE 30-6: DC CHARACTERISTICS: OPERATING CURRENT (IDD)

DC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended		
Parameter No.	**Typ.**	**Max.**	**Units**	**Conditions**		
Operating Current (IDD)[1]						
DC20d	8	13	mA	-40°C	3.3V	10 MIPS
DC20a	8	13	mA	+25°C		
DC20b	8	13	mA	+85°C		
DC20c	8	13	mA	+125°C		
DC22d	12	20	mA	-40°C	3.3V	20 MIPS
DC22a	12	20	mA	+25°C		
DC22b	12	20	mA	+85°C		
DC22c	12	20	mA	+125°C		
DC24d	19	30	mA	-40°C	3.3V	40 MIPS
DC24a	19	30	mA	+25°C		
DC24b	19	30	mA	+85°C		
DC24c	19	30	mA	+125°C		
DC25d	27	42	mA	-40°C	3.3V	60 MIPS
DC25a	27	42	mA	+25°C		
DC25b	27	42	mA	+85°C		
DC25c	27	42	mA	+125°C		
DC26d	30	46	mA	-40°C	3.3V	70 MIPS
DC26a	30	46	mA	+25°C		
DC26b	30	46	mA	+85°C		
DC27d	57	75	mA	-40°C	3.3V	70 MIPS (**Note 2**)
DC27a	57	75	mA	+25°C		
DC27b	57	75	mA	+85°C		

Note 1: IDD is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDD measurements are as follows:

- Oscillator is configured in EC mode with PLL, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}}$ = VDD, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating or being clocked (all defined PMDx bits are set)
- CPU is executing `while(1)` statement
- JTAG is disabled

2: For this specification, the following test conditions apply:

- APLL clock is enabled
- All 8 PWMs enabled and operating at maximum speed (PTCON2<2:0> = `000`), PTPER = 1000h, 50% duty cycle
- All other peripherals are disabled (corresponding PMDx bits are set)

TABLE 30-7: DC CHARACTERISTICS: IDLE CURRENT (I_{IDLE})

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.	**Typ.**	**Max.**	**Units**	**Conditions**		
Idle Current (I_{IDLE})[1]						
DC40d	2	4	mA	-40°C	3.3V	10 MIPS
DC40a	2	4	mA	+25°C		
DC40b	2	4	mA	+85°C		
DC40c	2	4	mA	+125°C		
DC42d	3	6	mA	-40°C	3.3V	20 MIPS
DC42a	3	6	mA	+25°C		
DC42b	4	7	mA	+85°C		
DC42c	4	7	mA	+125°C		
DC44d	6	12	mA	-40°C	3.3V	40 MIPS
DC44a	6	12	mA	+25°C		
DC44b	6	12	mA	+85°C		
DC44c	6	12	mA	+125°C		
DC45d	9	17	mA	-40°C	3.3V	60 MIPS
DC45a	9	17	mA	+25°C		
DC45b	9	17	mA	+85°C		
DC45c	9	17	mA	+125°C		
DC46d	10	20	mA	-40°C	3.3V	70 MIPS
DC46a	10	20	mA	+25°C		
DC46b	10	20	mA	+85°C		

Note 1: Base Idle current (I_{IDLE}) is measured as follows:

- CPU core is off, oscillator is configured in EC mode and external clock is active; OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as inputs and pulled to V_{SS}
- $\overline{MCLR}$ = V_{DD}, WDT and FSCM are disabled
- No peripheral modules are operating or being clocked (all defined PMDx bits are set)
- The NVMSIDL bit (NVMCON<12>) = 1 (i.e., Flash regulator is set to standby while the device is in Idle mode)
- The VREGSF bit (RCON<11>) = 0 (i.e., Flash regulator is set to standby while the device is in Sleep mode)
- JTAG is disabled

TABLE 30-8: DC CHARACTERISTICS: POWER-DOWN CURRENT (I_{PD})

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended		
Parameter No.	**Typ.**	**Max.**	**Units**	**Conditions**	
Power-Down Current (I_{PD})[1]					
DC60d	15	110	μA	-40°C	3.3V
DC60a	20	150	μA	+25°C	
DC60b	150	500	μA	+85°C	
DC60c	500	1200	μA	+125°C	

Note 1: IPD (Sleep) current is measured as follows:
- CPU core is off, oscillator is configured in EC mode and external clock is active; OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}}$ = VDD, WDT and FSCM are disabled
- All peripheral modules are disabled (PMDx bits are all set)
- The VREGS bit (RCON<8>) = 0 (i.e., core regulator is set to standby while the device is in Sleep mode)
- The VREGSF bit (RCON<11>) = 0 (i.e., Flash regulator is set to standby while the device is in Sleep mode)
- JTAG is disabled

TABLE 30-9: DC CHARACTERISTICS: WATCHDOG TIMER DELTA CURRENT (ΔI_{WDT})[1]

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended		
Parameter No.	**Typ.**	**Max.**	**Units**	**Conditions**	
DC61d	1	10	μA	-40°C	3.3V
DC61a	1	10	μA	+25°C	
DC61b	2	17	μA	+85°C	
DC61c	2	20	μA	+125°C	

Note 1: The ΔIWDT current is the additional current consumed when the module is enabled. This current should be added to the base IPD current. All parameters are characterized but not tested during manufacturing.

TABLE 30-10: DC CHARACTERISTICS: DOZE CURRENT (I_{DOZE})

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Parameter No.	**Typ.**	**Max.**	**Doze Ratio**	**Units**	**Conditions**		
Doze Current (I_{DOZE})(1)							
DC73a(2)	20	40	1:2	mA	-40°C	3.3V	FOSC = 140 MHz
DC73g	10	22	1:128	mA			
DC70a(2)	20	40	1:2	mA	+25°C	3.3V	FOSC = 140 MHz
DC70g	10	22	1:128	mA			
DC71a(2)	20	40	1:2	mA	+85°C	3.3V	FOSC = 140 MHz
DC71g	10	22	1:128	mA			
DC72a(2)	20	40	1:2	mA	+125°C	3.3V	FOSC = 120 MHz
DC72g	10	22	1:128	mA			

Note 1: IDOZE is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all IDOZE measurements are as follows:

- Oscillator is configured in EC mode and external clock is active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as inputs and pulled to VSS
- $\overline{\text{MCLR}}$ = VDD, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating or being clocked (all defined PMDx bits are set)
- CPU is executing `while(1)` statement
- JTAG is disabled

2: These parameter are characterized but not tested in manufacturing.

TABLE 30-11: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ.[1]	Max.	Units	Conditions
	VIL	**Input Low Voltage**					
DI10		Any I/O Pin and $\overline{\text{MCLR}}$	VSS	—	0.2 VDD	V	
DI18		I/O Pins with SDAx, SCLx	VSS	—	0.3 VDD	V	SMBus disabled
DI19		I/O Pins with SDAx, SCLx	VSS	—	0.8	V	SMBus enabled
	VIH	**Input High Voltage**					
DI20		I/O Pins Not 5V Tolerant[4]	0.8 VDD	—	VDD	V	
		I/O Pins 5V Tolerant and $\overline{\text{MCLR}}$[4]	0.8 VDD	—	5.5	V	
		5V Tolerant I/O Pins with SDAx, SCLx[4]	0.8 VDD	—	5.5	V	SMBus disabled
		5V Tolerant I/O Pins with SDAx, SCLx[4]	2.1	—	5.5	V	SMBus enabled
		I/O Pins with SDAx, SCLx Not 5V Tolerant[4]	0.8 VDD	—	VDD	V	SMBus disabled
		I/O Pins with SDAx, SCLx Not 5V Tolerant[4]	2.1	—	VDD	V	SMBus enabled
DI30	ICNPU	**Input Change Notification Pull-up Current**	100	230	550	μA	VDD = 3.3V, VPIN = VSS
DI31	ICNPD	**Input Change Notification Pull-Down Current**[5]	100	230	400	μA	VDD = 3.3V, VPIN = VDD

Note 1: Data in “Typ.” column is at 3.3V, +25°C unless otherwise stated.

2: The leakage current on the $\overline{\text{MCLR}}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current can be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: See the **“Pin Diagrams”** section for the 5V tolerant I/O pins.

5: VIL Source < (VSS – 0.3). Characterized but not tested.

6: VIH Source > (VDD + 0.3) for pins that are not 5V tolerant only.

7: Digital 5V tolerant pins do not have internal high-side diodes to VDD and cannot tolerate any “positive” input injection current.

8: Injection Currents > | 0 | can affect the ADC results by approximately 4-6 counts.

9: Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical “absolute instantaneous” sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 30-11: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.[1]**	**Max.**	**Units**	**Conditions**
	IIL	**Input Leakage Current[2,3]**					
DI50		I/O Pins 5V Tolerant[4]	-1	—	+1	μA	VSS ≤ VPIN ≤ VDD, pin at high-impedance
DI51		I/O Pins Not 5V Tolerant[4]	-1	—	+1	μA	VSS ≤ VPIN ≤ VDD, pin at high-impedance, -40°C ≤ TA ≤ +85°C
DI51a		I/O Pins Not 5V Tolerant[4]	-1	—	+1	μA	Analog pins shared with external reference pins, -40°C ≤ TA ≤ +85°C
DI51b		I/O Pins Not 5V Tolerant[4]	-1	—	+1	μA	VSS ≤ VPIN ≤ VDD, pin at high-impedance, -40°C ≤ TA ≤ +125°C
DI51c		I/O Pins Not 5V Tolerant[4]	-1	—	+1	μA	Analog pins shared with external reference pins, -40°C ≤ TA ≤ +125°C
DI55		$\overline{\text{MCLR}}$	-5	—	+5	μA	VSS ≤ VPIN ≤ VDD
DI56		OSC1	-5	—	+5	μA	VSS ≤ VPIN ≤ VDD, XT and HS modes

Note 1: Data in "Typ." column is at 3.3V, +25°C unless otherwise stated.

2: The leakage current on the $\overline{\text{MCLR}}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current can be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: See the **"Pin Diagrams"** section for the 5V tolerant I/O pins.

5: VIL Source < (VSS – 0.3). Characterized but not tested.

6: VIH Source > (VDD + 0.3) for pins that are not 5V tolerant only.

7: Digital 5V tolerant pins do not have internal high-side diodes to VDD and cannot tolerate any "positive" input injection current.

8: Injection Currents > | 0 | can affect the ADC results by approximately 4-6 counts.

9: Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 30-11: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS (CONTINUED)

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DI60a	IICL	**Input Low Injection Current**	0	—	-5(5,8)	mA	All pins except VDD, VSS, AVDD, AVSS, MCLR, VCAP and RB7
DI60b	IICH	**Input High Injection Current**	0	—	+5(6,7,8)	mA	All pins except VDD, VSS, AVDD, AVSS, MCLR, VCAP, RB7 and all 5V tolerant pins(7)
DI60c	∑IICT	**Total Input Injection Current (sum of all I/O and control pins)**	-20(9)	—	+20(9)	mA	Absolute instantaneous sum of all ± input injection currents from all I/O pins (| IICL | + | IICH |) ≤ ∑IICT

Note 1: Data in "Typ." column is at 3.3V, +25°C unless otherwise stated.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current can be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: See the **"Pin Diagrams"** section for the 5V tolerant I/O pins.

5: VIL Source < (VSS – 0.3). Characterized but not tested.

6: VIH Source > (VDD + 0.3) for pins that are not 5V tolerant only.

7: Digital 5V tolerant pins do not have internal high-side diodes to VDD and cannot tolerate any "positive" input injection current.

8: Injection Currents > | 0 | can affect the ADC results by approximately 4-6 counts.

9: Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the mathematical "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

TABLE 30-12: DC CHARACTERISTICS: I/O PIN OUTPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
DO10	V_{OL}	**Output Low Voltage** 4x Sink Driver Pins(2)	—	—	0.4	V	V_{DD} = 3.3V, I_{OL} ≤ 6 mA, -40°C ≤ T_A ≤ +85°C, I_{OL} ≤ 5 mA, +85°C < T_A ≤ +125°C
		Output Low Voltage 8x Sink Driver Pins(3)	—	—	0.4	V	V_{DD} = 3.3V, I_{OL} ≤ 12 mA, -40°C ≤ T_A ≤ +85°C, I_{OL} ≤ 8 mA, +85°C < T_A ≤ +125°C
DO20	V_{OH}	**Output High Voltage** 4x Source Driver Pins(2)	2.4	—	—	V	I_{OH} ≥ -10 mA, V_{DD} = 3.3V
		Output High Voltage 8x Source Driver Pins(3)	2.4	—	—	V	I_{OH} ≥ -15 mA, V_{DD} = 3.3V
DO20A	V_{OH1}	**Output High Voltage** 4x Source Driver Pins(2)	1.5(1)	—	—	V	I_{OH} ≥ -14 mA, V_{DD} = 3.3V
			2.0(1)	—	—		I_{OH} ≥ -12 mA, V_{DD} = 3.3V
			3.0(1)	—	—		I_{OH} ≥ -7 mA, V_{DD} = 3.3V
		Output High Voltage 8x Source Driver Pins(3)	1.5(1)	—	—	V	I_{OH} ≥ -22 mA, V_{DD} = 3.3V
			2.0(1)	—	—		I_{OH} ≥ -18 mA, V_{DD} = 3.3V
			3.0(1)	—	—		I_{OH} ≥ -10 mA, V_{DD} = 3.3V

Note 1: Parameters are characterized but not tested.

2: Includes RA0-RA2, RB0-RB1, RB9, RC1-RC2, RC9-RC10, RC12, RD7, RD8, RE4-RE5, RE8-RE9 and RE12-RE13 pins.

3: Includes all I/O pins that are not 4x driver pins (see **Note 2**).

TABLE 30-13: ELECTRICAL CHARACTERISTICS: BOR

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)(1) Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.(2)**	**Typ.**	**Max.**	**Units**	**Conditions**
BO10	V_{BOR}	BOR Event on V_{DD} Transition High-to-Low	2.65	—	2.95	V	V_{DD} (**Notes 2** and **3**)

Note 1: Device is functional at V_{BORMIN} < V_{DD} < V_{DDMIN}, but will have degraded performance. Device functionality is tested, but not characterized. Analog modules (ADC, PGAs and comparators) may have degraded performance.

2: Parameters are for design guidance only and are not tested in manufacturing.

3: The V_{BOR} specification is relative to V_{DD}.

TABLE 30-14: DC CHARACTERISTICS: PROGRAM MEMORY

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.[1]**	**Max.**	**Units**	**Conditions**
		Program Flash Memory					
D130	EP	Cell Endurance	10,000	—	—	E/W	-40°C to +125°C
D131	VPR	VDD for Read	3.0	—	3.6	V	
D132b	VPEW	VDD for Self-Timed Write	3.0	—	3.6	V	
D134	TRETD	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated, -40°C to +125°C
D135	IDDP	Supply Current during Programming[2]	—	10	—	mA	
D136	IPEAK	Instantaneous Peak Current During Start-up	—	—	150	mA	
D137a	TPE	Page Erase Time	19.7	—	20.1	ms	TPE = 146893 FRC cycles, TA = +85°C **(Note 3)**
D137b	TPE	Page Erase Time	19.5	—	20.3	ms	TPE = 146893 FRC cycles, TA = +125°C **(Note 3)**
D138a	TWW	Word Write Cycle Time	46.5	—	47.3	µs	TWW = 346 FRC cycles, TA = +85°C **(Note 3)**
D138b	TWW	Word Write Cycle Time	46.0	—	47.9	µs	TWW = 346 FRC cycles, TA = +125°C **(Note 3)**
D139a	TRW	Row Write Time	667	—	679	µs	TRW = 4965 FRC cycles, TA = +85°C **(Note 3)**
D139b	TRW	Row Write Time	660	—	687	µs	TRW = 4965 FRC cycles, TA = +125°C **(Note 3)**

Note 1: Data in “Typ.” column is at 3.3V, +25°C unless otherwise stated.

2: Parameter characterized but not tested in manufacturing.

3: Other conditions: FRC = 7.37 MHz, TUN<5:0> = `011111` (for Minimum), TUN<5:0> = `100000` (for Maximum). This parameter depends on the FRC accuracy (see Table 30-20) and the value of the FRC Oscillator Tuning register (see Register 9-4). For complete details on calculating the Minimum and Maximum time, see **Section 5.3 “Programming Operations”**.

30.2 AC Characteristics and Timing Parameters

This section defines the dsPIC33EPXXXGS70X/80X family AC characteristics and timing parameters.

TABLE 30-15: TEMPERATURE AND VOLTAGE SPECIFICATIONS – AC

AC CHARACTERISTICS	**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended Operating voltage VDD range as described in **Section 30.1 "DC Characteristics"**.

FIGURE 30-1: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS

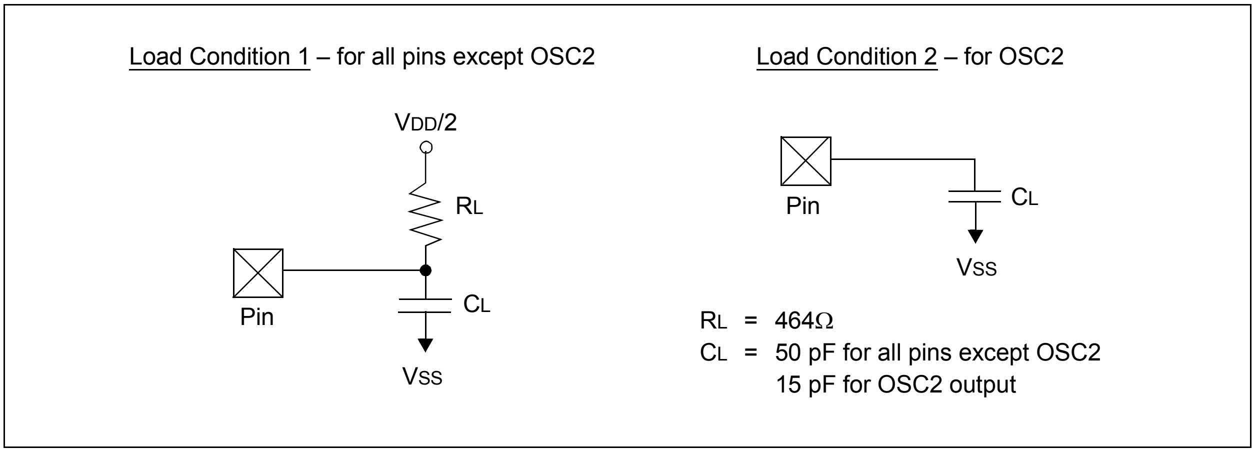


TABLE 30-16: CAPACITIVE LOADING REQUIREMENTS ON OUTPUT PINS

Param No.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
DO50	COSCO	OSC2 Pin	—	—	15	pF	In XT and HS modes, when external clock is used to drive OSC1
DO56	CIO	All I/O Pins and OSC2	—	—	50	pF	EC mode
DO58	CB	SCLx, SDAx	—	—	400	pF	In I^2C mode

FIGURE 30-2: EXTERNAL CLOCK TIMING

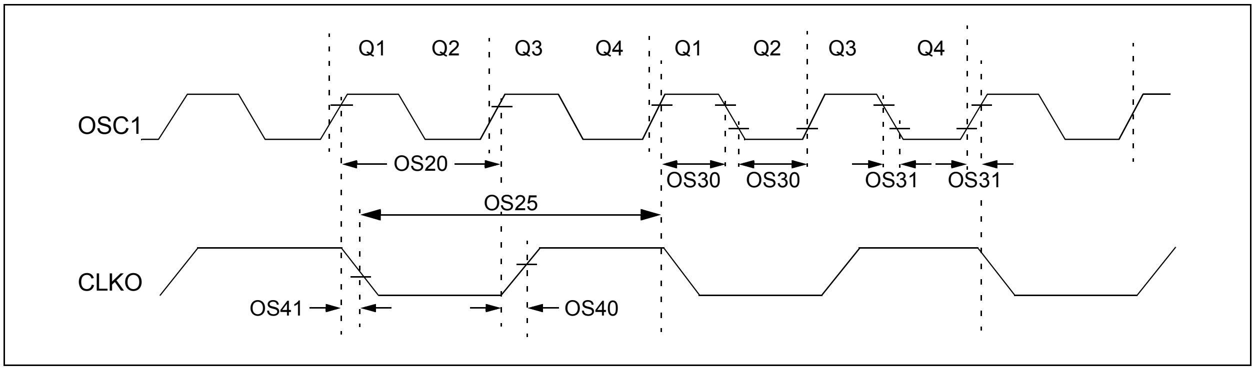


TABLE 30-17: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Sym**	**Characteristic**	**Min.**	**Typ.[1]**	**Max.**	**Units**	**Conditions**
OS10	FIN	External CLKI Frequency (External clocks allowed only in EC and ECPLL modes)	DC	—	60	MHz	EC
		Oscillator Crystal Frequency	3.5 10	— —	10 40	MHz MHz	XT HS
OS20	TOSC	TOSC = 1/FOSC	8.33	—	DC	ns	+125°C
		TOSC = 1/FOSC	7.14	—	DC	ns	+85°C
OS25	TCY	Instruction Cycle Time[2]	16.67	—	DC	ns	+125°C
		Instruction Cycle Time[2]	14.28	—	DC	ns	+85°C
OS30	TosL, TosH	External Clock in (OSC1) High or Low Time	0.45 x TOSC	—	0.55 x TOSC	ns	EC
OS31	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time[3,4]	—	5.2	—	ns	
OS41	TckF	CLKO Fall Time[3,4]	—	5.2	—	ns	
OS42	GM	External Oscillator Transconductance[4]	—	12	—	mA/V	HS, VDD = 3.3V, TA = +25°C
			—	6	—	mA/V	XT, VDD = 3.3V, TA = +25°C

Note 1: Data in "Typ." column is at 3.3V, +25°C unless otherwise stated.

2: Instruction cycle period (TCY) equals two times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type, under standard operating conditions, with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "Minimum" values with an external clock applied to the OSC1 pin. When an external clock input is used, the "Maximum" cycle time limit is "DC" (no clock) for all devices.

3: Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.

4: This parameter is characterized but not tested in manufacturing.

TABLE 30-18: PLL CLOCK TIMING SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ.[(1)]	Max.	Units	Conditions
OS50	FPLLI	PLL Voltage Controlled Oscillator (VCO) Input Frequency Range	0.8	—	8.0	MHz	ECPLL, XTPLL modes
OS51	FVCO	On-Chip VCO System Frequency	120	—	340	MHz	
OS52	TLOCK	PLL Start-up Time (Lock Time)	0.9	1.5	3.1	ms	
OS53	DCLK	CLKO Stability (Jitter)[(2)]	-3	0.5	3	%	

Note 1: Data in "Typ." column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: This jitter specification is based on clock cycle-by-clock cycle measurements. To get the effective jitter for individual time bases, or communication clocks used by the application, use the following formula:

$$Effective\ Jitter = \frac{D_{CLK}}{\sqrt{\frac{F_{OSC}}{Time\ Base\ or\ Communication\ Clock}}}$$

For example, if FOSC = 120 MHz and the SPIx bit rate = 10 MHz, the effective jitter is as follows:

$$Effective\ Jitter = \frac{D_{CLK}}{\sqrt{\frac{120}{10}}} = \frac{D_{CLK}}{\sqrt{12}} = \frac{D_{CLK}}{3.464}$$

TABLE 30-19: AUXILIARY PLL CLOCK TIMING SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ[(1)]	Max	Units	Conditions
OS56	FHPOUT	On-Chip 16x PLL CCO Frequency	112	118	120	MHz	
OS57	FHPIN	On-Chip 16x PLL Phase Detector Input Frequency	7.0	7.37	7.5	MHz	
OS58	TSU	Frequency Generator Lock Time	—	—	10	µs	

Note 1: Data in "Typ" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested in manufacturing.

TABLE 30-20: INTERNAL FRC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**	
Internal FRC Accuracy @ FRC Frequency = 7.37 MHz[1]							
F20a	FRC	-2	0.5	+2	%	-40°C ≤ TA ≤ -10°C	VDD = 3.0-3.6V
		-0.9	0.5	+0.9	%	-10°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F20b	FRC	-2	1	+2	%	+85°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

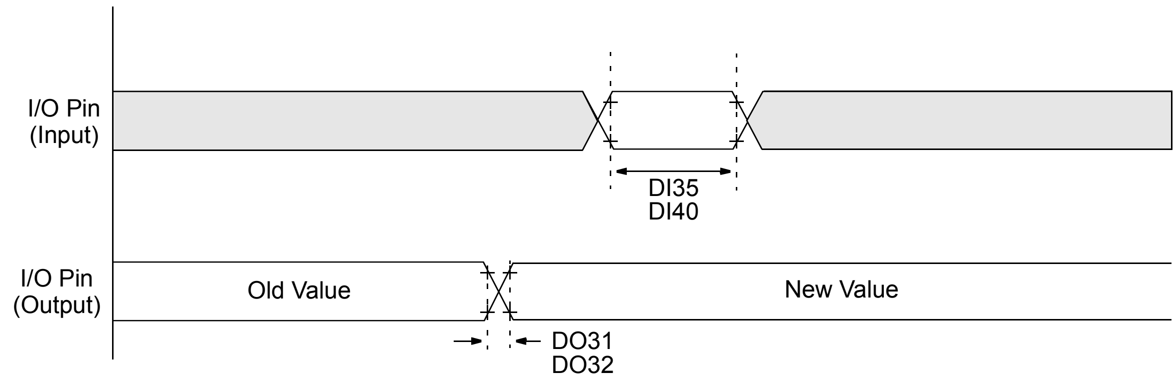
Note 1: Frequency is calibrated at +25°C and 3.3V. TUNx bits can be used to compensate for temperature drift.

TABLE 30-21: INTERNAL LPRC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**	
LPRC @ 32.768 kHz[1]							
F21a	LPRC	-30	—	+30	%	-40°C ≤ TA ≤ -10°C	VDD = 3.0-3.6V
		-20	—	+20	%	-10°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F21b	LPRC	-30	—	+30	%	+85°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

Note 1: This is the change of the LPRC frequency as VDD changes.

FIGURE 30-3: I/O TIMING CHARACTERISTICS



Note: Refer to Figure 30-1 for load conditions.

TABLE 30-22: I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DO31	TIOR	Port Output Rise Time	—	5	10	ns	
DO32	TIOF	Port Output Fall Time	—	5	10	ns	
DI35	TINP	INTx Pin High or Low Time (input)	20	—	—	ns	
DI40	TRBP	CNx High or Low Time (input)	2	—	—	TCY	

Note 1: Data in "Typ." column is at 3.3V, +25°C unless otherwise stated.

FIGURE 30-4: BOR AND MASTER CLEAR RESET TIMING CHARACTERISTICS

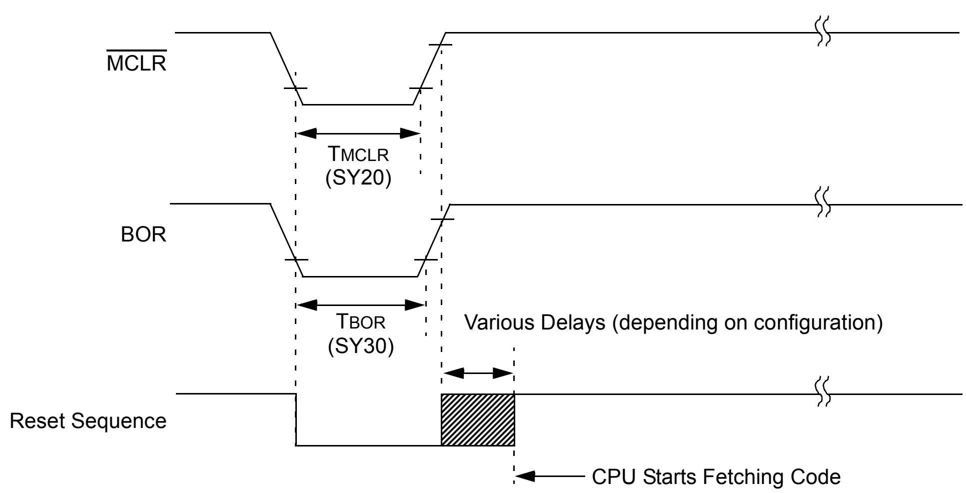
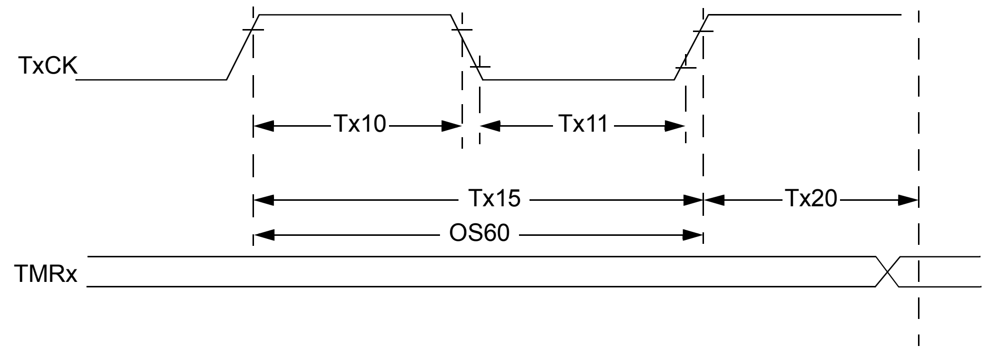


TABLE 30-23: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SY00	TPU	Power-up Period	—	400	600	µs	
SY10	TOST	Oscillator Start-up Time	—	1024 TOSC	—	—	TOSC = OSC1 period
SY12	TWDT	Watchdog Timer Time-out Period	0.81	—	1.22	ms	WDTPRE = 0, WDTPOST<3:0> = 0000, using LPRC tolerances indicated in F21 (see Table 30-21) at +85°C
			3.25	—	4.88	ms	WDTPRE = 1, WDTPOST<3:0> = 0000, using LPRC tolerances indicated in F21 (see Table 30-21) at +85°C
SY13	TIOZ	I/O High-Impedance from MCLR Low or Watchdog Timer Reset	0.68	0.72	1.2	µs	
SY20	TMCLR	MCLR Pulse Width (low)	2	—	—	µs	
SY30	TBOR	BOR Pulse Width (low)	1	—	—	µs	
SY35	TFSCM	Fail-Safe Clock Monitor Delay	—	500	900	µs	-40°C to +85°C
SY36	TVREG	Voltage Regulator Standby-to-Active mode Transition Time	—	—	30	µs	
SY37	TOSCDFRC	FRC Oscillator Start-up Delay	—	48	—	µs	
SY38	TOSCDLPRC	LPRC Oscillator Start-up Delay	—	—	70	µs	

Note 1: These parameters are characterized but not tested in manufacturing.
2: Data in “Typ.” column is at 3.3V, +25°C unless otherwise stated.

FIGURE 30-5: TIMER1-TIMER5 EXTERNAL CLOCK TIMING CHARACTERISTICS



Note: Refer to Figure 30-1 for load conditions.

TABLE 30-24: TIMER1 EXTERNAL CLOCK TIMING REQUIREMENTS[1]

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[2]**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
TA10	TTXH	T1CK High Time	Synchronous mode	Greater of: 20 or (TCY + 20)/N	—	—	ns	Must also meet Parameter TA15, N = Prescale Value (1, 8, 64, 256)
			Asynchronous mode	35	—	—	ns	
TA11	TTXL	T1CK Low Time	Synchronous mode	Greater of: 20 or (TCY + 20)/N	—	—	ns	Must also meet Parameter TA15, N = Prescale Value (1, 8, 64, 256)
			Asynchronous mode	10	—	—	ns	
TA15	TTXP	T1CK Input Period	Synchronous mode	Greater of: 40 or (2 TCY + 40)/N	—	—	ns	N = Prescale Value (1, 8, 64, 256)
OS60	Ft1	T1CK Oscillator Input Frequency Range (oscillator enabled by setting bit, TCS (T1CON<1>))		DC	—	50	kHz	
TA20	TCKEXTMRL	Delay from External T1CK Clock Edge to Timer Increment		0.75 TCY + 40	—	1.75 TCY + 40	ns	

Note 1: Timer1 is a Type A timer.

2: These parameters are characterized but not tested in manufacturing.

TABLE 30-25: TIMER2 AND TIMER4 (TYPE B TIMER) EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
TB10	TtxH	TxCK High Time	Synchronous mode	Greater of: 20 or (TCY + 20)/N	—	—	ns	Must also meet Parameter TB15, N = Prescale Value (1, 8, 64, 256)
TB11	TtxL	TxCK Low Time	Synchronous mode	Greater of: 20 or (TCY + 20)/N	—	—	ns	Must also meet Parameter TB15, N = Prescale Value (1, 8, 64, 256)
TB15	TtxP	TxCK Input Period	Synchronous mode	Greater of: 40 or (2 TCY + 40)/N	—	—	ns	N = Prescale Value (1, 8, 64, 256)
TB20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 TCY + 40	—	1.75 TCY + 40	ns	

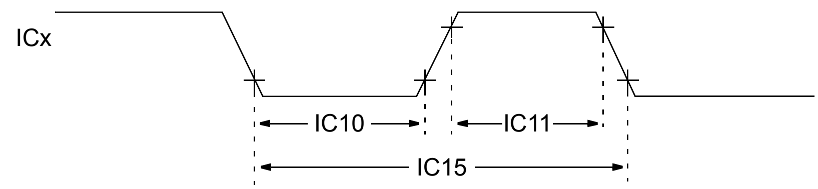
Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 30-26: TIMER3 AND TIMER5 (TYPE C TIMER) EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic[1]**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
TC10	TtxH	TxCK High Time	Synchronous	TCY + 20	—	—	ns	Must also meet Parameter TC15
TC11	TtxL	TxCK Low Time	Synchronous	TCY + 20	—	—	ns	Must also meet Parameter TC15
TC15	TtxP	TxCK Input Period	Synchronous with Prescaler	2 TCY + 40	—	—	ns	N = Prescale Value (1, 8, 64, 256)
TC20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.75 TCY + 40	—	1.75 TCY + 40	ns	

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 30-6: INPUT CAPTURE x (ICx) TIMING CHARACTERISTICS



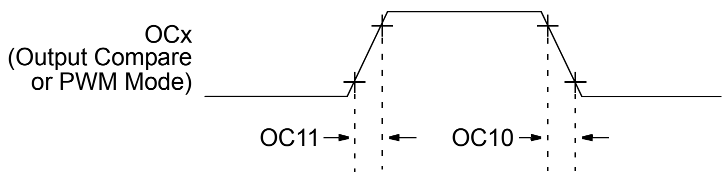
Note: Refer to Figure 30-1 for load conditions.

TABLE 30-27: INPUT CAPTURE x MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param. No.	**Symbol**	**Characteristics[1]**	**Min.**	**Max.**	**Units**	**Conditions**	
IC10	TCCL	ICx Input Low Time	Greater of: 12.5 + 25 or (0.5 TCY/N) + 25	—	ns	Must also meet Parameter IC15	
IC11	TCCH	ICx Input High Time	Greater of: 12.5 + 25 or (0.5 TCY/N) + 25	—	ns	Must also meet Parameter IC15	N = Prescale Value (1, 4, 16)
IC15	TCCP	ICx Input Period	Greater of: 25 + 50 or (1 TCY/N) + 50	—	ns		

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 30-7: OUTPUT COMPARE x MODULE (OCx) TIMING CHARACTERISTICS



Note: Refer to Figure 30-1 for load conditions.

TABLE 30-28: OUTPUT COMPARE x MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic[1]	Min.	Typ.	Max.	Units	Conditions
OC10	TccF	OCx Output Fall Time	—	—	—	ns	See Parameter DO32
OC11	TccR	OCx Output Rise Time	—	—	—	ns	See Parameter DO31

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 30-8: OCx/PWMx MODULE TIMING CHARACTERISTICS

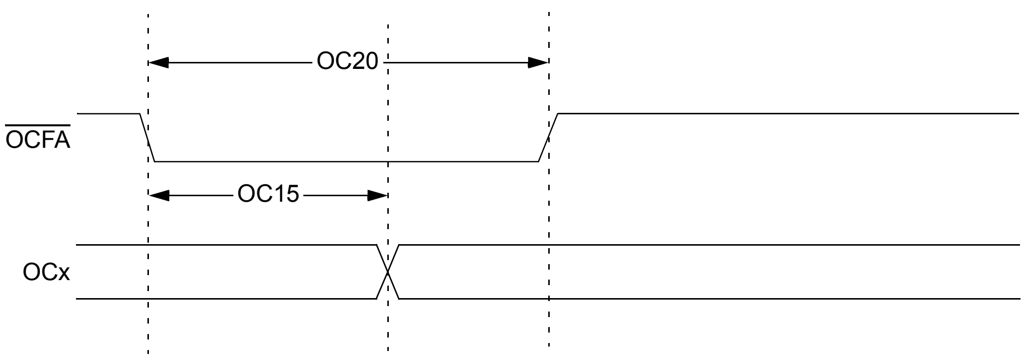


TABLE 30-29: OCx/PWMx MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic[1]	Min.	Typ.	Max.	Units	Conditions
OC15	TFD	Fault Input to PWMx I/O Change	—	—	TCY + 20	ns	
OC20	TFLT	Fault Input Pulse Width	TCY + 20	—	—	ns	

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 30-9: HIGH-SPEED PWMx MODULE FAULT TIMING CHARACTERISTICS

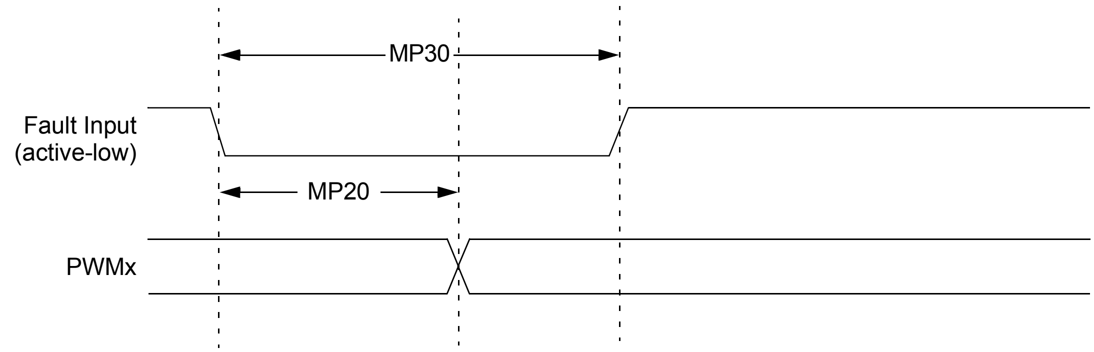
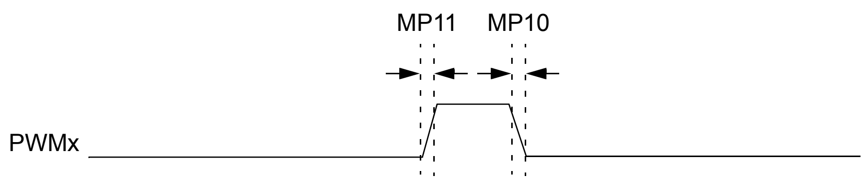


FIGURE 30-10: HIGH-SPEED PWMx MODULE TIMING CHARACTERISTICS



Note: Refer to Figure 30-1 for load conditions.

TABLE 30-30: HIGH-SPEED PWMx MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			**Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic(1)**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
MP10	TFPWM	PWMx Output Fall Time	—	—	—	ns	See Parameter DO32
MP11	TRPWM	PWMx Output Rise Time	—	—	—	ns	See Parameter DO31
MP20	TFD	Fault Input ↓ to PWMx I/O Change	—	—	15	ns	
MP30	TFH	Fault Input Pulse Width	15	—	—	ns	

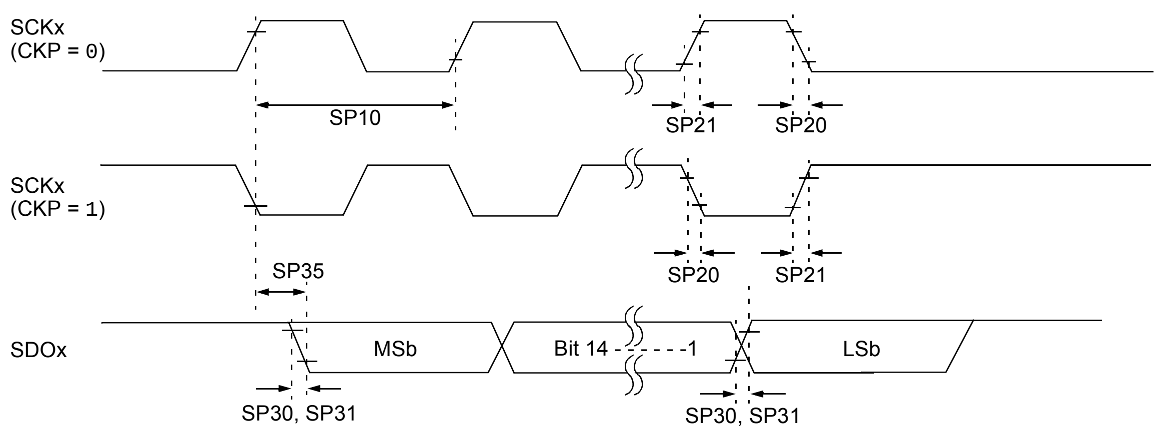
Note 1: These parameters are characterized but not tested in manufacturing.

TABLE 30-31: SPI1, SPI2 AND SPI3 MAXIMUM DATA/CLOCK RATE SUMMARY[1]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Maximum Data Rate	**Master Transmit Only (Half-Duplex)**	**Master Transmit/Receive (Full-Duplex)**	**Slave Transmit/Receive (Full-Duplex)**	**CKE**	**CKP**	**SMP**
15 MHz	Table 30-32	—	—	0,1	0,1	0,1
9 MHz	—	Table 30-33	—	1	0,1	1
9 MHz	—	Table 30-34	—	0	0,1	1
15 MHz	—	—	Table 30-35	1	0	0
11 MHz	—	—	Table 30-36	1	1	0
15 MHz	—	—	Table 30-37	0	1	0
11 MHz	—	—	Table 30-38	0	0	0

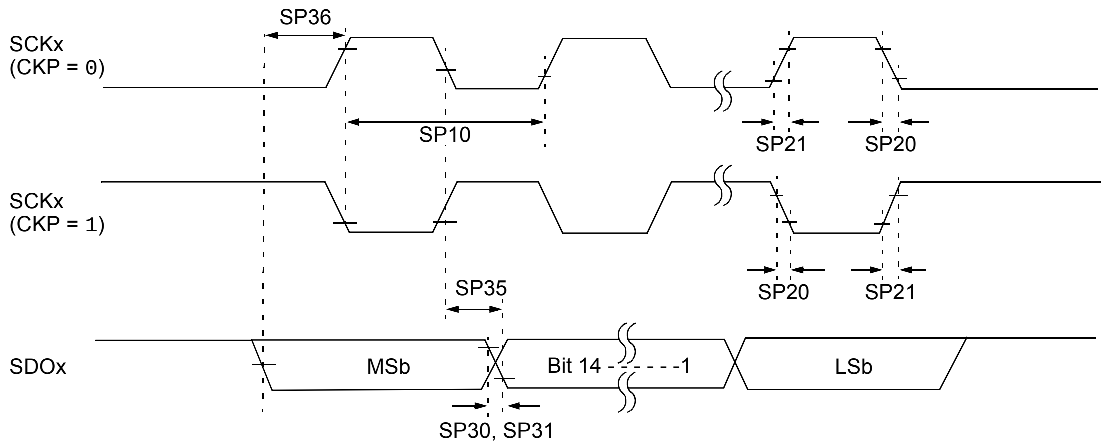
Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-11: SPI1, SPI2 AND SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY, CKE = 0) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

FIGURE 30-12: SPI1, SPI2 AND SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY, CKE = 1) TIMING CHARACTERISTICS(1,2)



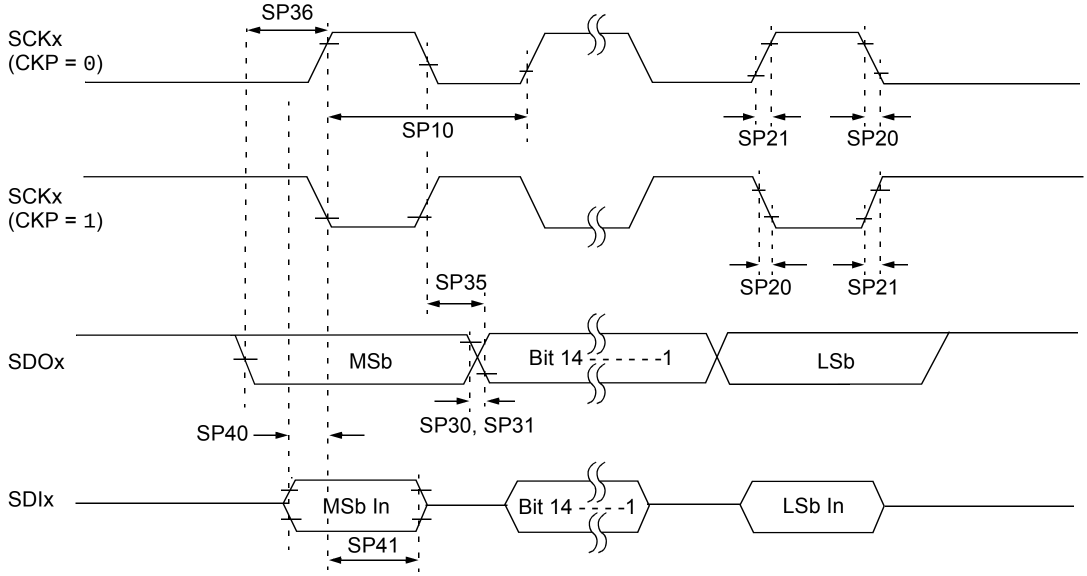
Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

TABLE 30-32: SPI1, SPI2 AND SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING REQUIREMENTS(5)

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic(1)**	**Min.**	**Typ.(2)**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCKx Frequency	—	—	15	MHz	(Note 3)
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdiV2scH, TdiV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCKx is 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.
4: Assumes 50 pF load on all SPIx pins.
5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-13: SPI1, SPI2 AND SPI3 MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-33: SPI1, SPI2 AND SPI3 MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCKx Frequency	—	—	9	MHz	(Note 3)
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2sc, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.

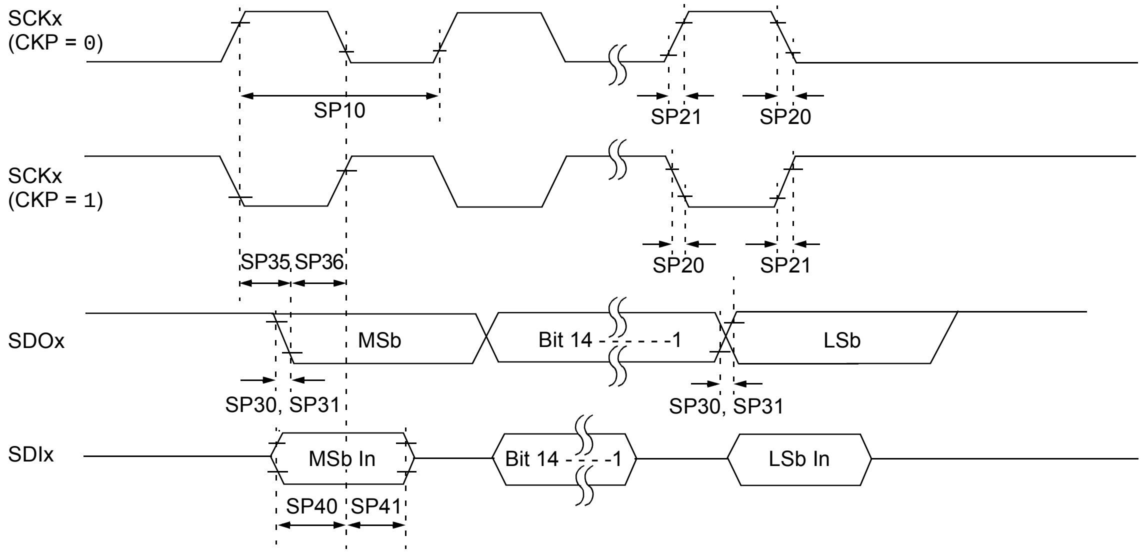
2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCKx is 111 ns. The clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-14: SPI1, SPI2 AND SPI3 MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-34: SPI1, SPI2 AND SPI3 MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCKx Frequency	—	—	9	MHz	-40ºC to +125ºC (**Note 3**)
SP20	TscF	SCKx Output Fall Time	—	—	—	ns	See Parameter DO32 (**Note 4**)
SP21	TscR	SCKx Output Rise Time	—	—	—	ns	See Parameter DO31 (**Note 4**)
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (**Note 4**)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (**Note 4**)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.

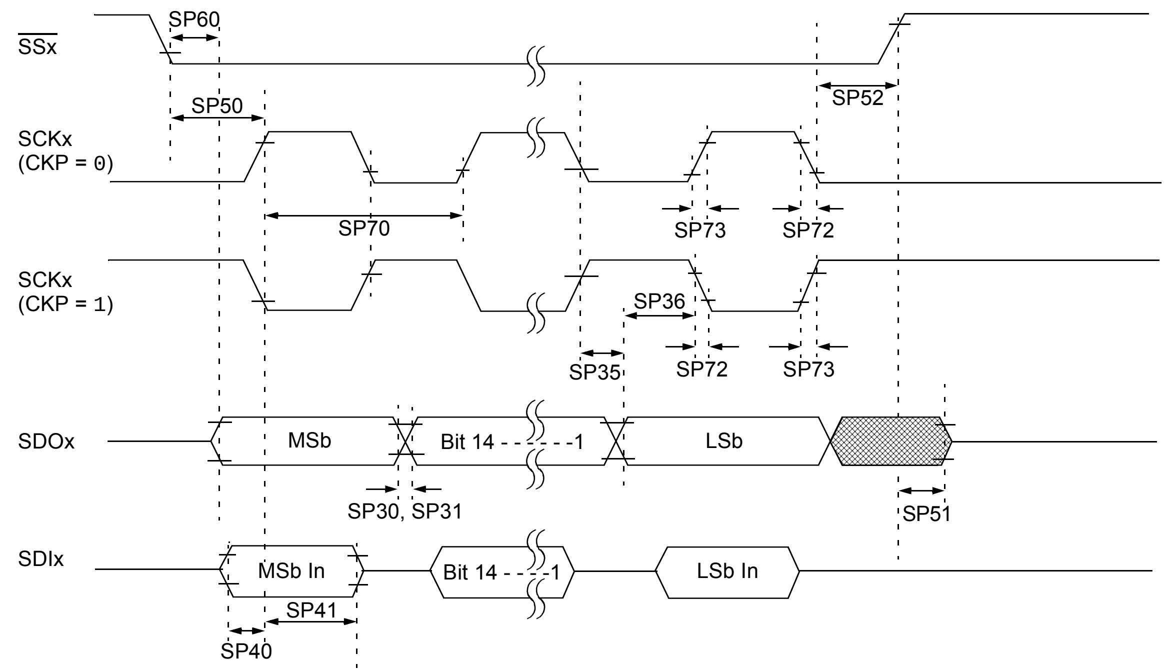
2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCKx is 111 ns. The clock generated in Master mode must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-15: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

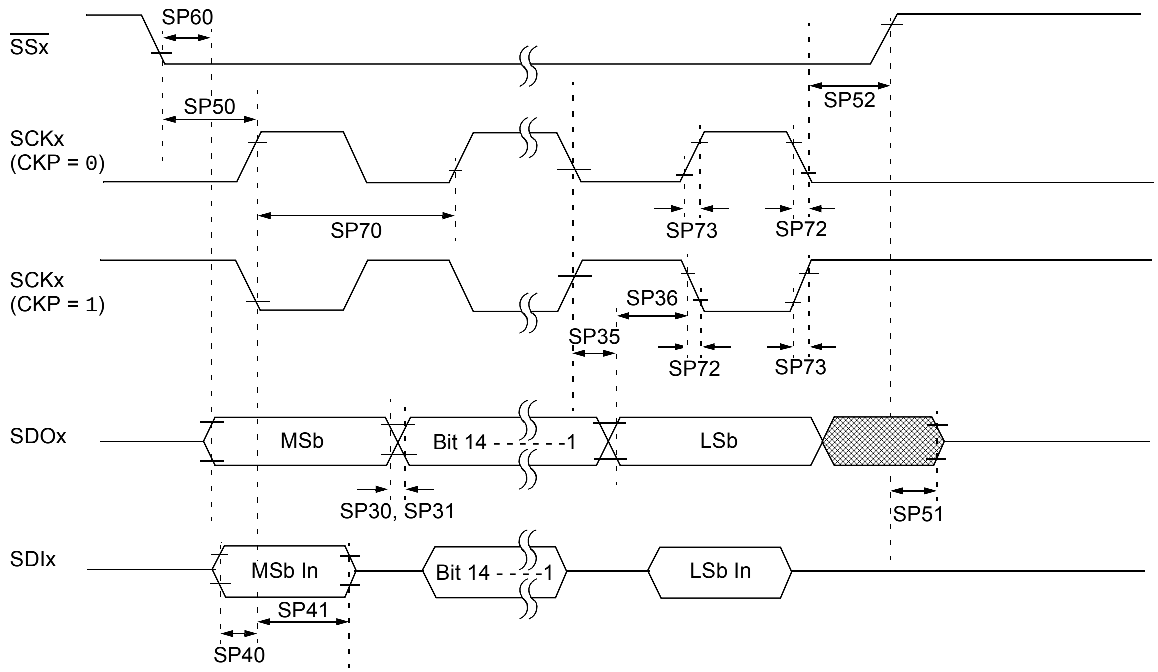
2: Refer to Figure 30-1 for load conditions.

TABLE 30-35: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCKx Input Frequency	—	—	15	MHz	(Note 3)
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ ↑ after SCKx Edge	1.5 TCY + 40	—	—	ns	(Note 4)
SP60	TssL2doV	SDOx Data Output Valid after $\overline{\text{SSx}}$ Edge	—	—	50	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCKx is 66.7 ns. Therefore, the SCKx clock generated by the master must not violate this specification.
4: Assumes 50 pF load on all SPIx pins.
5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-16: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-36: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCKx Input Frequency	—	—	11	MHz	(Note 3)
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↑ or SCKx ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH TscL2ssH	$\overline{SSx}$ ↑ after SCKx Edge	1.5 TCY + 40	—	—	ns	(Note 4)
SP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	50	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.

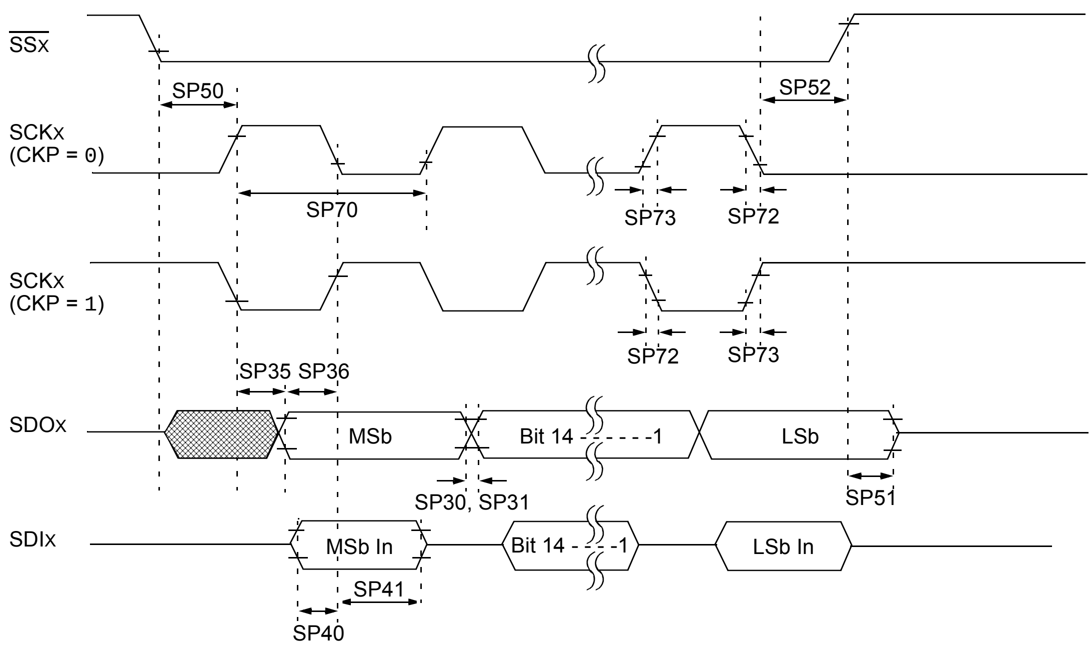
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCKx is 91 ns. Therefore, the SCKx clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-17: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-37: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS(5)

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic(1)**	**Min.**	**Typ.(2)**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCKx Input Frequency	—	—	15	MHz	(Note 3)
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↑ or SCKx ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH TscL2ssH	$\overline{SSx}$ ↑ after SCKx Edge	1.5 TCY + 40	—	—	ns	(Note 4)

Note 1: These parameters are characterized, but are not tested in manufacturing.

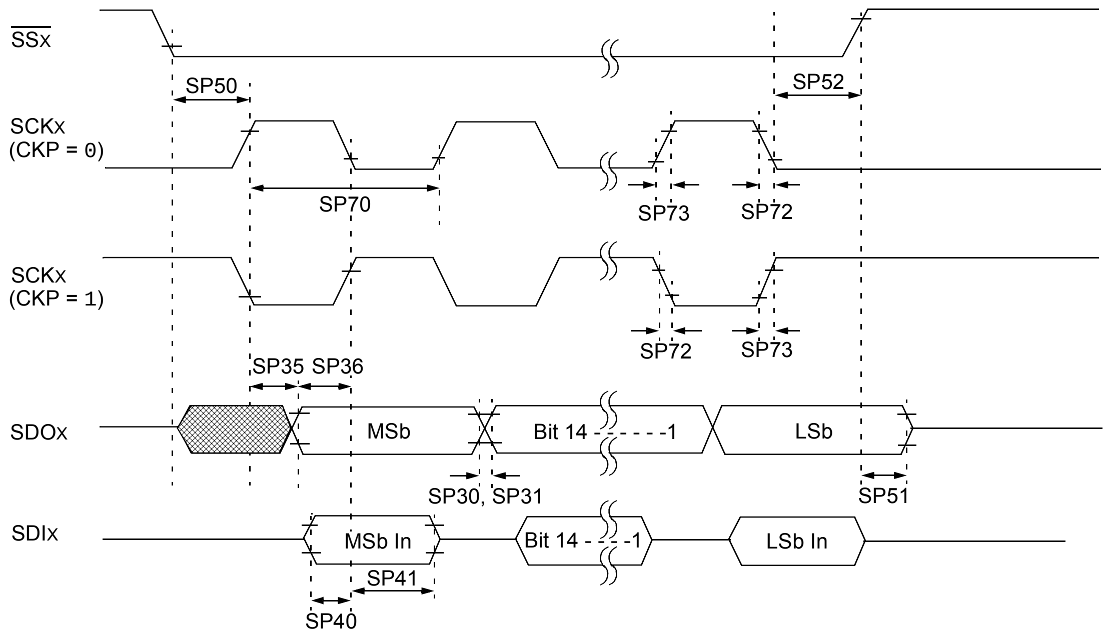
2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCKx is 66.7 ns. Therefore, the SCKx clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

FIGURE 30-18: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-38: SPI1, SPI2 AND SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCKx Input Frequency	—	—	11	MHz	(Note 3)
SP72	TscF	SCKx Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCKx Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDOx Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDOx Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx Data Output Setup to First SCKx Edge	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	30	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{\text{SSx}}$ ↓ to SCKx ↑ or SCKx ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{\text{SSx}}$ ↑ to SDOx Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH TscL2ssH	$\overline{\text{SSx}}$ ↑ after SCKx Edge	1.5 TCY + 40	—	—	ns	(Note 4)

Note 1: These parameters are characterized, but are not tested in manufacturing.

2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCKx is 91 ns. Therefore, the SCKx clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPIx pins.

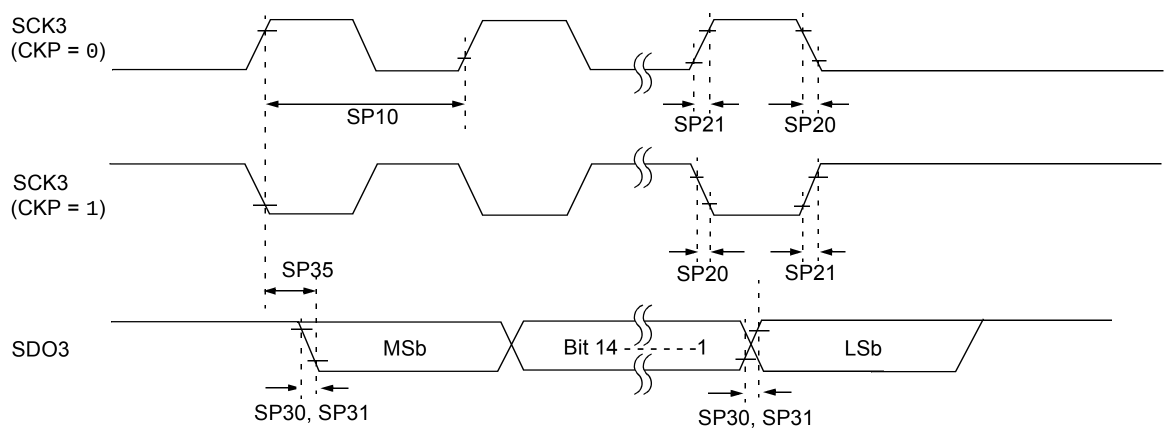
5: Pertaining to SPI3: dsPIC33EPXXXGS702, dsPIC33EPXXXGSX04 and dsPIC33EPXXXGSX05 devices with a remappable SCK3 pin.

TABLE 30-39: SPI3 MAXIMUM DATA/CLOCK RATE SUMMARY[1]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Maximum Data Rate	**Master Transmit Only (Half-Duplex)**	**Master Transmit/Receive (Full-Duplex)**	**Slave Transmit/Receive (Full-Duplex)**	**CKE**	**CKP**	**SMP**
25 MHz	Table 30-40	—	—	0,1	0,1	0,1
25 MHz	—	Table 30-41	—	1	0,1	1
25 MHz	—	Table 30-42	—	0	0,1	1
25 MHz	—	—	Table 30-43	1	0	0
25 MHz	—	—	Table 30-44	1	1	0
25 MHz	—	—	Table 30-45	0	1	0
25 MHz	—	—	Table 30-46	0	0	0

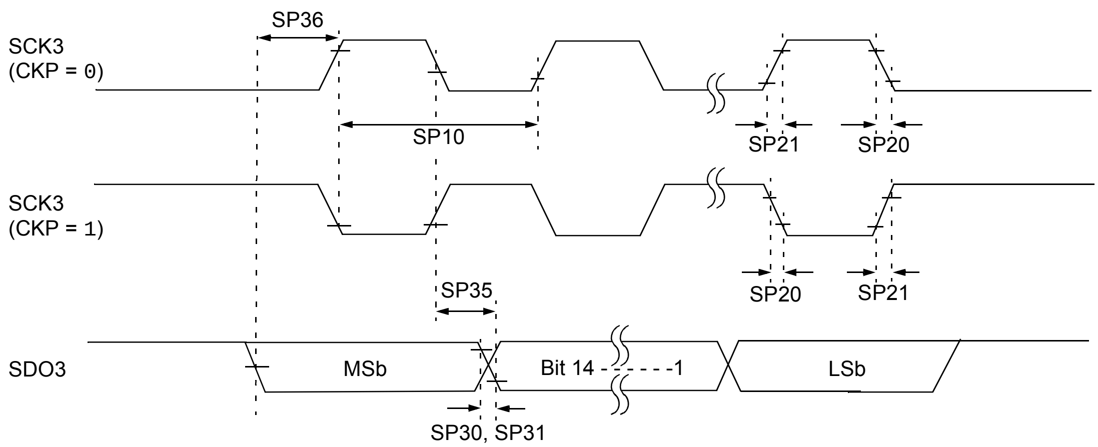
Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-19: SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY, CKE = 0) TIMING CHARACTERISTICS[1,2]



Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

FIGURE 30-20: SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY, CKE = 1) TIMING CHARACTERISTICS[1,2]



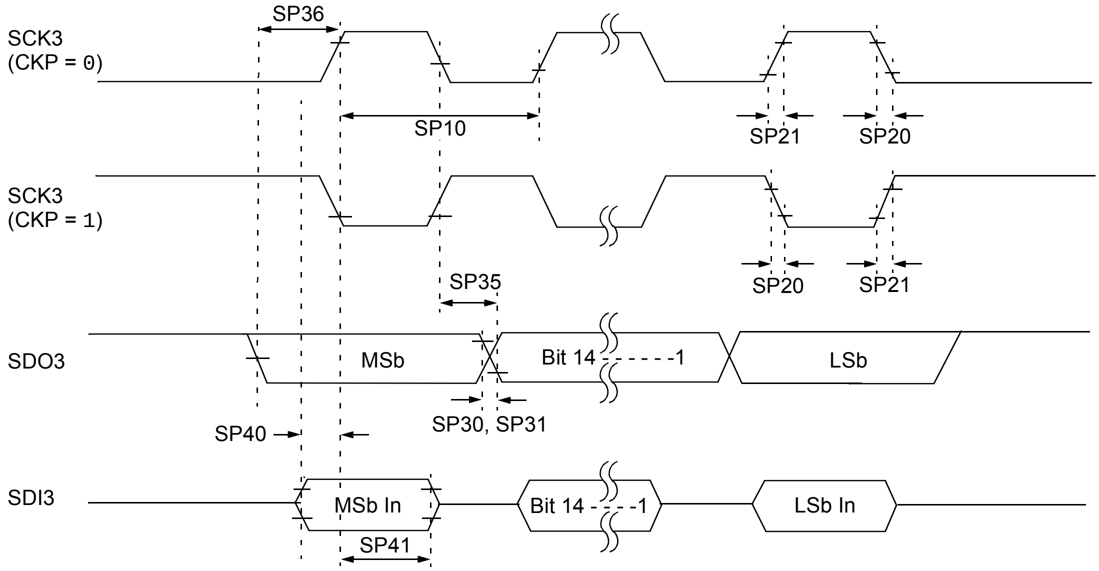
Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

TABLE 30-40: SPI3 MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCK3 Frequency	—	—	25	MHz	(Note 3)
SP20	TscF	SCK3 Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP21	TscR	SCK3 Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdiV2scH, TdiV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCK3 is 66.7 ns. Therefore, the clock generated in Master mode must not violate this specification.
4: Assumes 50 pF load on all SPI3 pins.
5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-21: SPI3 MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING CHARACTERISTICS[1,2]



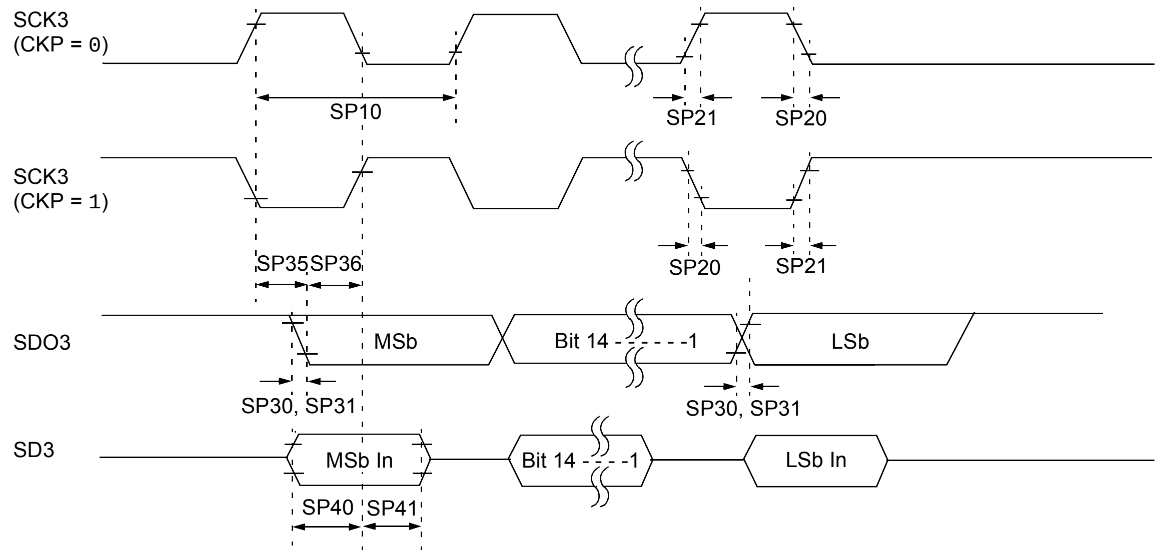
Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

TABLE 30-41: SPI3 MASTER MODE (FULL-DUPLEX, CKE = 1, CKP = x, SMP = 1) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCK3 Frequency	—	—	25	MHz	(Note 3)
SP20	TscF	SCK3 Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP21	TscR	SCK3 Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2sc, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3 Edge	15	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCK3 is 100 ns. The clock generated in Master mode must not violate this specification.
4: Assumes 50 pF load on all SPI3 pins.
5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-22: SPI3 MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING CHARACTERISTICS[1,2]



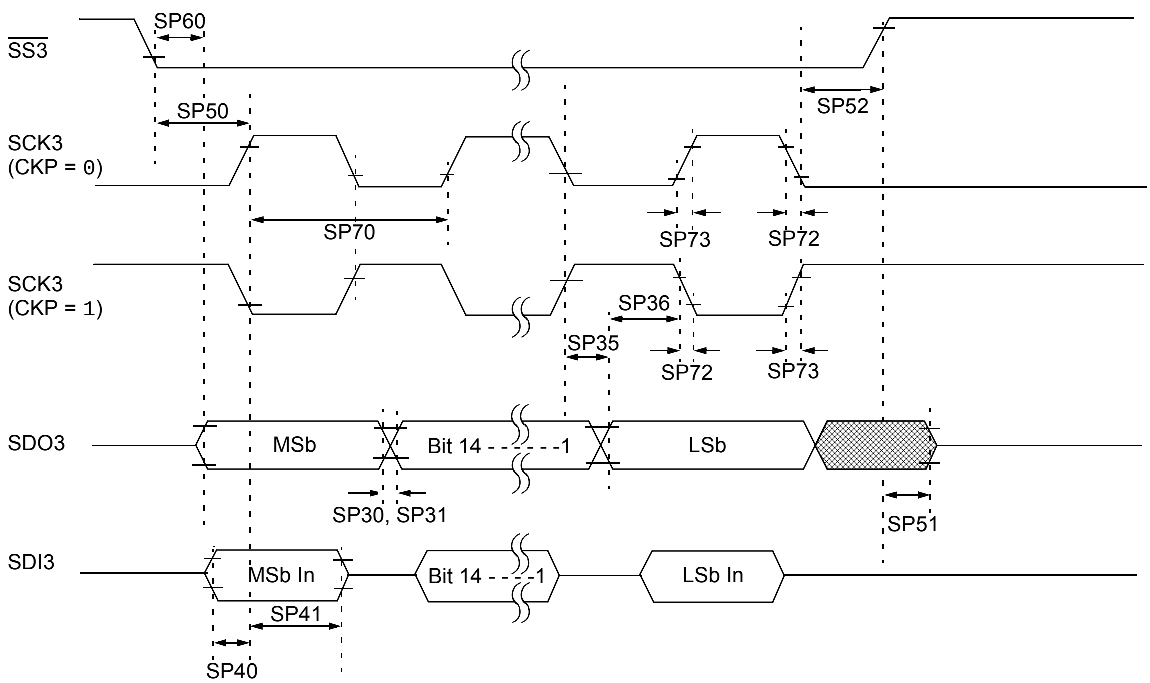
Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.
2: Refer to Figure 30-1 for load conditions.

TABLE 30-42: SPI3 MASTER MODE (FULL-DUPLEX, CKE = 0, CKP = x, SMP = 1) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP10	FscP	Maximum SCK3 Frequency	—	—	25	MHz	-40ºC to +125ºC (**Note 3**)
SP20	TscF	SCK3 Output Fall Time	—	—	—	ns	See Parameter DO32 (**Note 4**)
SP21	TscR	SCK3 Output Rise Time	—	—	—	ns	See Parameter DO31 (**Note 4**)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (**Note 4**)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (**Note 4**)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3Edge	20	—	—	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCK3 is 100 ns. The clock generated in Master mode must not violate this specification.
4: Assumes 50 pF load on all SPI3 pins.
5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-23: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-43: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 0, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCK3 Input Frequency	—	—	25	MHz	(Note 3)
SP72	TscF	SCK3 Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCK3 Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3 Edge	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ to SCK3 ↑ or SCK3 ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ to SDO3 Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH TscL2ssH	$\overline{SS3}$ ↑ after SCK3 Edge	1.5 TCY + 40	—	—	ns	(Note 4)
SP60	TssL2doV	SDO3 Data Output Valid after $\overline{SS3}$ Edge	—	—	50	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.

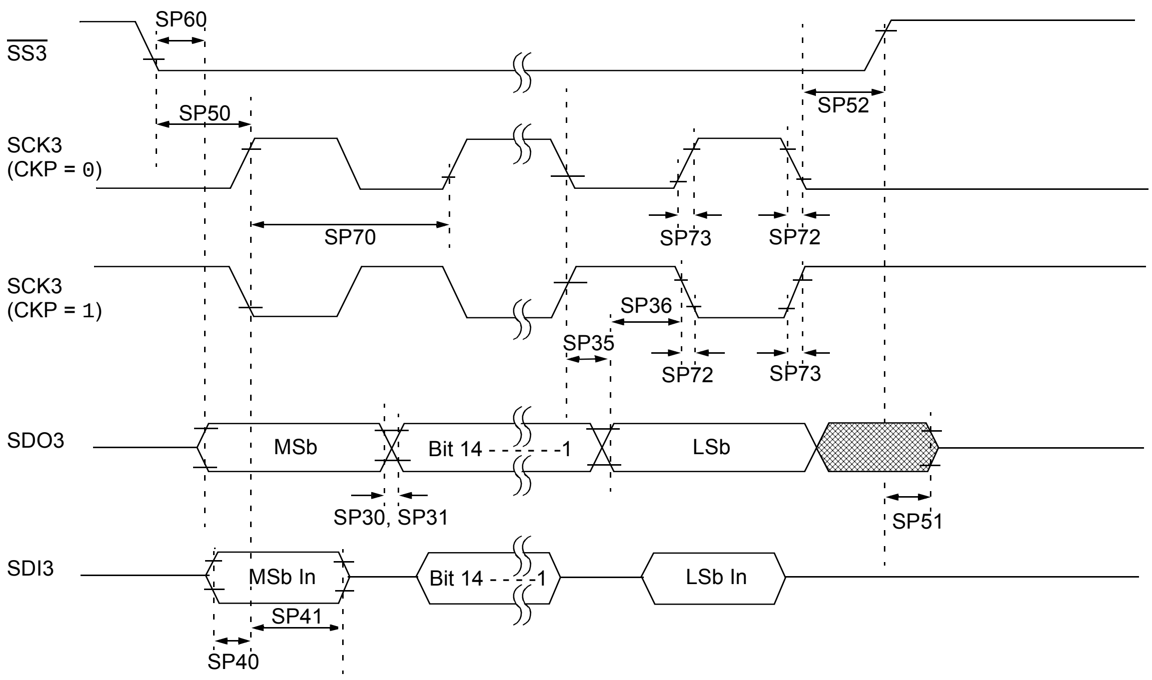
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCK3 is 66.7 ns. Therefore, the SCK3 clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPI3 pins.

5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-24: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

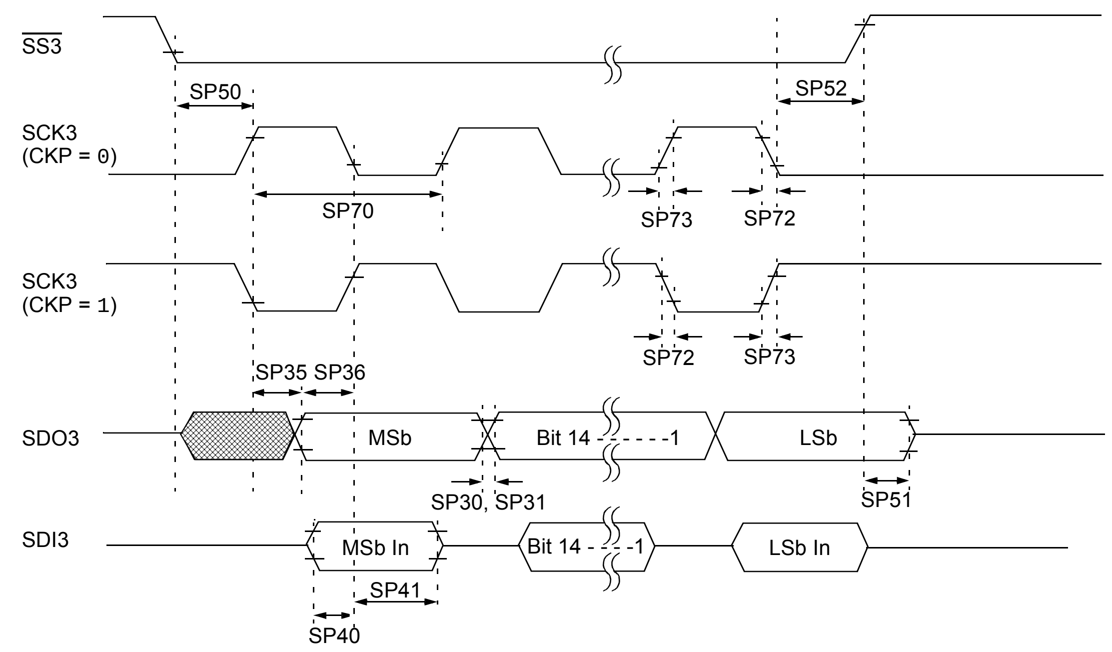
2: Refer to Figure 30-1 for load conditions.

TABLE 30-44: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	Symbol	Characteristic[1]	Min.	Typ.[2]	Max.	Units	Conditions
SP70	FscP	Maximum SCK3 Input Frequency	—	—	25	MHz	(Note 3)
SP72	TscF	SCK3 Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCK3 Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3 Edge	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{\text{SS3}}$ ↓ to SCK3 ↑ or SCK3 ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{\text{SS3}}$ ↑ to SDO3 Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH, TscL2ssH	$\overline{\text{SS3}}$ ↑ after SCK3 Edge	1.5 TCY + 40	—	—	ns	(Note 4)
SP60	TssL2doV	SDO3 Data Output Valid after $\overline{\text{SS3}}$ Edge	—	—	50	ns	

Note 1: These parameters are characterized, but are not tested in manufacturing.
2: Data in "Typical" column is at 3.3V, +25°C unless otherwise stated.
3: The minimum clock period for SCK3 is 91 ns. Therefore, the SCK3 clock generated by the master must not violate this specification.
4: Assumes 50 pF load on all SPI3 pins.
5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-25: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-45: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 1, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCK3 Input Frequency	—	—	25	MHz	(Note 3)
SP72	TscF	SCK3 Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCK3 Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3 Edge	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ to SCK3 ↑ or SCK3 ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ to SDO3 Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH, TscL2ssH	$\overline{SS3}$ ↑ after SCK3 Edge	1.5 TCY + 40	—	—	ns	(Note 4)

Note 1: These parameters are characterized, but are not tested in manufacturing.

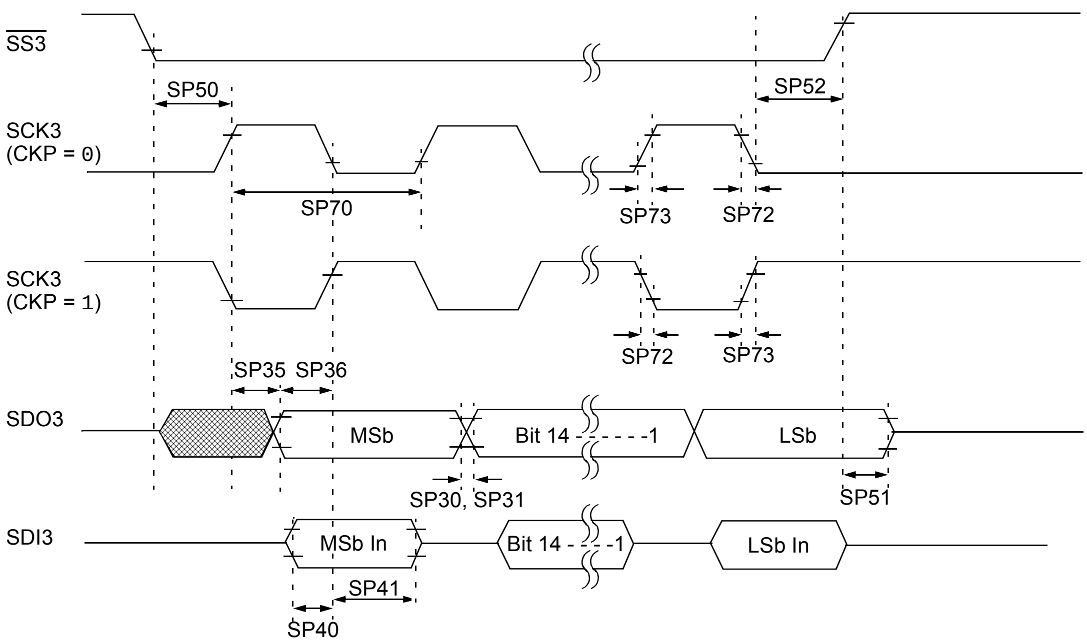
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCK3 is 66.7 ns. Therefore, the SCK3 clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPI3 pins.

5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-26: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING CHARACTERISTICS[1,2]



Note 1: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

2: Refer to Figure 30-1 for load conditions.

TABLE 30-46: SPI3 SLAVE MODE (FULL-DUPLEX, CKE = 0, CKP = 0, SMP = 0) TIMING REQUIREMENTS[5]

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param.	**Symbol**	**Characteristic[1]**	**Min.**	**Typ.[2]**	**Max.**	**Units**	**Conditions**
SP70	FscP	Maximum SCK3 Input Frequency	—	—	25	MHz	(Note 3)
SP72	TscF	SCK3 Input Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP73	TscR	SCK3 Input Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP30	TdoF	SDO3 Data Output Fall Time	—	—	—	ns	See Parameter DO32 (Note 4)
SP31	TdoR	SDO3 Data Output Rise Time	—	—	—	ns	See Parameter DO31 (Note 4)
SP35	TscH2doV, TscL2doV	SDO3 Data Output Valid after SCK3 Edge	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 Data Output Setup to First SCK3 Edge	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	Setup Time of SDI3 Data Input to SCK3 Edge	20	—	—	ns	
SP41	TscH2diL, TscL2diL	Hold Time of SDI3 Data Input to SCK3 Edge	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ to SCK3 ↑ or SCK3 ↓ Input	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ to SDO3 Output High-Impedance	10	—	50	ns	(Note 4)
SP52	TscH2ssH, TscL2ssH	$\overline{SS3}$ ↑ after SCK1 Edge	1.5 TCY + 40	—	—	ns	(Note 4)

Note 1: These parameters are characterized, but are not tested in manufacturing.

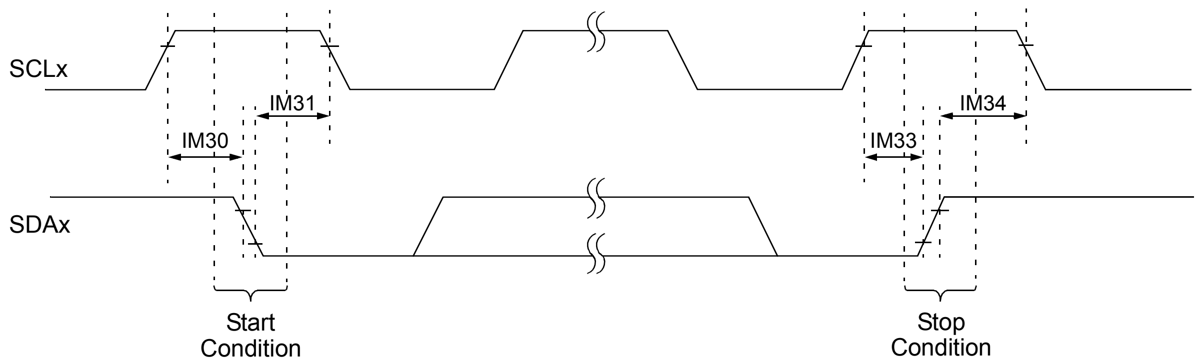
2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated.

3: The minimum clock period for SCK3 is 91 ns. Therefore, the SCK3 clock generated by the master must not violate this specification.

4: Assumes 50 pF load on all SPI3 pins.

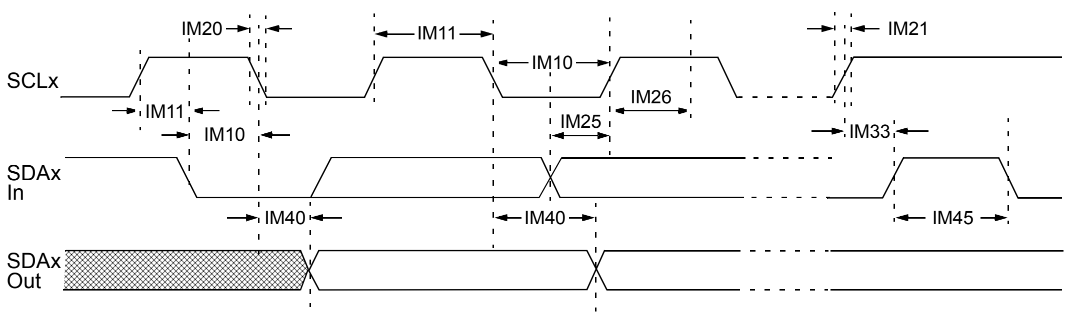
5: For dsPIC33EPXXXGSX06 and dsPIC33EPXXXGSX08 devices with a fixed SCK3 pin.

FIGURE 30-27: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (MASTER MODE)



Note: Refer to Figure 30-1 for load conditions.

FIGURE 30-28: I2Cx BUS DATA TIMING CHARACTERISTICS (MASTER MODE)



Note: Refer to Figure 30-1 for load conditions.

TABLE 30-47: I2Cx BUS DATA TIMING REQUIREMENTS (MASTER MODE)

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Param No.	Symbol	Characteristic(4)		Min.(1)	Max.	Units	Conditions
IM10	TLO:SCL	Clock Low Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	
			400 kHz mode	TCY/2 (BRG + 2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM11	THI:SCL	Clock High Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	
			400 kHz mode	TCY/2 (BRG + 2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	100	ns	
IM21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode(2)	—	300	ns	
IM25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	
			400 kHz mode	100	—	ns	
			1 MHz mode(2)	40	—	ns	
IM26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	μs	
			400 kHz mode	0	0.9	μs	
			1 MHz mode(2)	0.2	—	μs	
IM30	TSU:STA	Start Condition Setup Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	TCY/2 (BRG + 2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM31	THD:STA	Start Condition Hold Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	After this period, the first clock pulse is generated
			400 kHz mode	TCY/2 (BRG +2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM33	TSU:STO	Stop Condition Setup Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	
			400 kHz mode	TCY/2 (BRG + 2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM34	THD:STO	Stop Condition Hold Time	100 kHz mode	TCY/2 (BRG + 2)	—	μs	
			400 kHz mode	TCY/2 (BRG + 2)	—	μs	
			1 MHz mode(2)	TCY/2 (BRG + 2)	—	μs	
IM40	TAA:SCL	Output Valid from Clock	100 kHz mode	—	3500	ns	
			400 kHz mode	—	1000	ns	
			1 MHz mode(2)	—	400	ns	
IM45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
			1 MHz mode(2)	0.5	—	μs	
IM50	CB	Bus Capacitive Loading		—	400	pF	
IM51	TPGD	Pulse Gobbler Delay		65	390	ns	(Note 3)

Note 1: BRG is the value of the I^{2}C Baud Rate Generator.
2: Maximum Pin Capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).
3: Typical value for this parameter is 130 ns.
4: These parameters are characterized but not tested in manufacturing.

FIGURE 30-29: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)

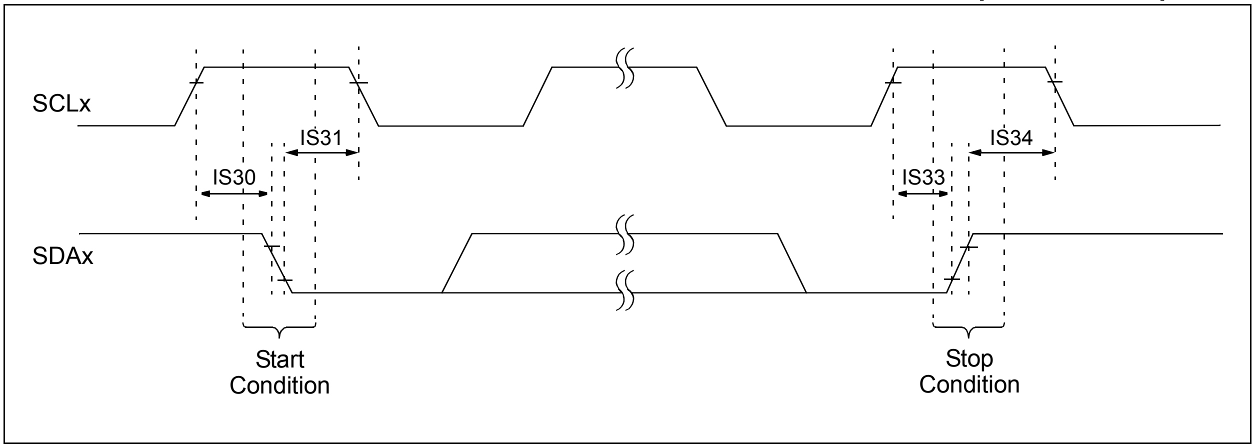


FIGURE 30-30: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

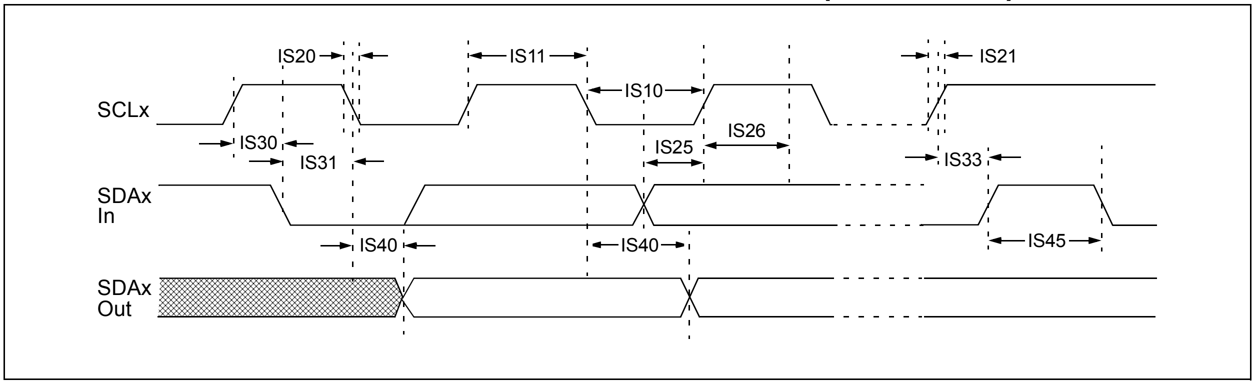


TABLE 30-48: I2Cx BUS DATA TIMING REQUIREMENTS (SLAVE MODE)

AC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Param No.	**Symbol**	**Characteristic[3]**		**Min.**	**Max.**	**Units**	**Conditions**
IS10	TLO:SCL	Clock Low Time	100 kHz mode	4.7	—	μs	
			400 kHz mode	1.3	—	μs	
			1 MHz mode[1]	0.5	—	μs	
IS11	THI:SCL	Clock High Time	100 kHz mode	4.0	—	μs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	μs	Device must operate at a minimum of 10 MHz
			1 MHz mode[1]	0.5	—	μs	
IS20	TF:SCL	SDAx and SCLx Fall Time	100 kHz mode	—	300	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode[1]	—	100	ns	
IS21	TR:SCL	SDAx and SCLx Rise Time	100 kHz mode	—	1000	ns	CB is specified to be from 10 to 400 pF
			400 kHz mode	20 + 0.1 CB	300	ns	
			1 MHz mode[1]	—	300	ns	
IS25	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	
			400 kHz mode	100	—	ns	
			1 MHz mode[1]	100	—	ns	
IS26	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	μs	
			400 kHz mode	0	0.9	μs	
			1 MHz mode[1]	0	0.3	μs	
IS30	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	μs	
			1 MHz mode[1]	0.25	—	μs	
IS31	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	μs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
			1 MHz mode[1]	0.25	—	μs	
IS33	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
			1 MHz mode[1]	0.6	—	μs	
IS34	THD:STO	Stop Condition Hold Time	100 kHz mode	4	—	μs	
			400 kHz mode	0.6	—	μs	
			1 MHz mode[1]	0.25		μs	
IS40	TAA:SCL	Output Valid from Clock	100 kHz mode	0	3500	ns	
			400 kHz mode	0	1000	ns	
			1 MHz mode[1]	0	350	ns	
IS45	TBF:SDA	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
			1 MHz mode[1]	0.5	—	μs	
IS50	CB	Bus Capacitive Loading		—	400	pF	
IS51	TPGD	Pulse Gobbler Delay		65	390	ns	(Note 2)

Note 1: Maximum Pin Capacitance = 10 pF for all I2Cx pins (for 1 MHz mode only).

2: Typical value for this parameter is 130 ns.

3: These parameters are characterized but not tested in manufacturing.

FIGURE 30-31: CANx MODULE I/O TIMING CHARACTERISTICS

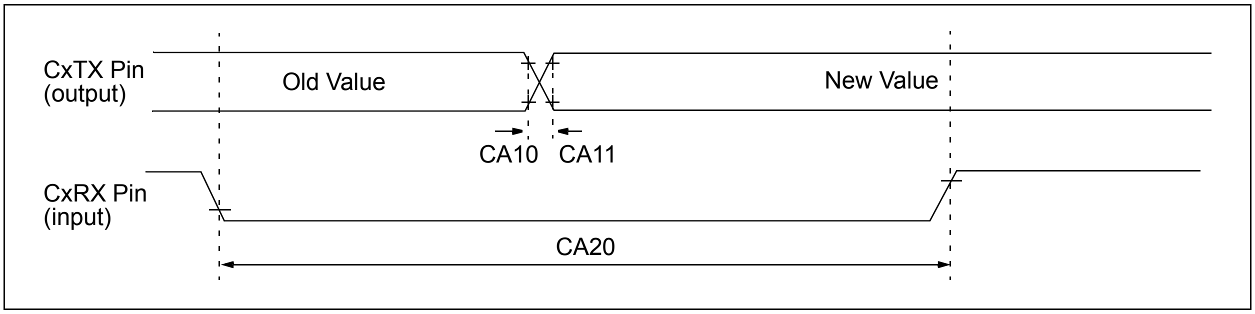


TABLE 30-49: CANx MODULE I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic[1]	Min.	Typ.[2]	Max.	Units	Conditions
CA10	TIOF	Port Output Fall Time	—	—	—	ns	See Parameter DO32
CA11	TIOR	Port Output Rise Time	—	—	—	ns	See Parameter DO31
CA20	TCWF	Pulse Width to Trigger CAN Wake-up Filter	120	—	—	ns	

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in “Typical” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

FIGURE 30-32: UARTx MODULE I/O TIMING CHARACTERISTICS

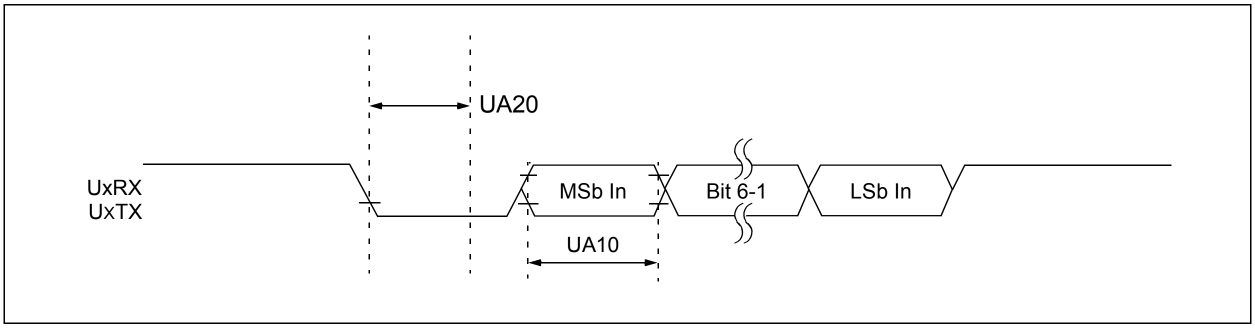


TABLE 30-50: UARTx MODULE I/O TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C				
Param No.	Symbol	Characteristic(1)	Min.	Typ.(2)	Max.	Units	Conditions
UA10	TUABAUD	UARTx Baud Time	66.67	—	—	ns	
UA11	FBAUD	UARTx Baud Frequency	—	—	15	Mbps	
UA20	TCWF	Start Bit Pulse Width to Trigger UARTx Wake-up	500	—	—	ns	

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in “Typ.” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 30-51: ANALOG CURRENT SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C				
Param No.	Symbol	Characteristic(1)	Min.	Typ.(2)	Max.	Units	Conditions
AVD01	IDD	Analog Modules Current Consumption	—	9	—	mA	Characterized data with the following modules enabled: APLL, 5 ADC Cores, 2 PGAs and 4 Analog Comparators

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in “Typ.” column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

TABLE 30-52: ADC MODULE SPECIFICATIONS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[5] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristics**	**Min.**	**Typical**	**Max.**	**Units**	**Conditions**
Device Supply							
AD01	AVDD	Module VDD Supply	Greater of: VDD – 0.3 or 3.0	—	Lesser of: VDD + 0.3 or 3.6	V	Within 300 mV of VDD at all times, including device power-up
AD02	AVSS	Module VSS Supply	VSS	—	VSS + 0.3	V	
Reference Inputs							
AD06	VREFL	Reference Voltage Low	—	AVSS	—	V	(Note 1)
AD07	VREF	Absolute Reference Voltage (VREFH – VREFL)	2.7	—	AVDD	V	(Note 3)
AD08	IREF	Reference Input Current	—	5	10	μA	ADC operating or in standby
Analog Input							
AD12	VINH-VINL	Full-Scale Input Span	AVSS	—	AVDD	V	
AD14	VIN	Absolute Input Voltage	AVSS – 0.3	—	AVDD + 0.3	V	
AD17	RIN	Recommended Impedance of Analog Voltage Source	—	100	—	Ω	For minimum sampling time (Note 1)
AD66	VBG	Internal Voltage Reference Source	—	1.2	—	V	
ADC Accuracy: Pseudodifferential Input							
AD20a	Nr	Resolution	12			bits	
AD21a	INL	Integral Nonlinearity	> -3	—	< 3	LSb	AVSS = 0V, AVDD = 3.3V
AD22a	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	AVSS = 0V, AVDD = 3.3V (Note 2)
AD23a	GERR	Gain Error (Dedicated Core)	> 0	8	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		Gain Error (Shared Core)	> 5	15	< 22	LSb	
AD24a	EOFF	Offset Error (Dedicated Core)	> 0	5	< 10	LSb	AVSS = 0V, AVDD = 3.3V
		Offset Error (Shared Core)	> 2	8	< 13	LSb	
AD25a	—	Monotonicity	—	—	—	—	Guaranteed

Note 1: These parameters are not characterized or tested in manufacturing.
2: No missing codes, limits based on characterization results.
3: These parameters are characterized but not tested in manufacturing.
4: Characterized with a 15 kHz sine wave.
5: The ADC module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is ensured, but not characterized.

TABLE 30-52: ADC MODULE SPECIFICATIONS (CONTINUED)

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[5] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristics**	**Min.**	**Typical**	**Max.**	**Units**	**Conditions**
ADC Accuracy: Single-Ended Input							
AD20b	Nr	Resolution	12			bits	
AD21b	INL	Integral Nonlinearity	> 5	—	< 5	LSb	AVSS = 0V, AVDD = 3.3V
AD22b	DNL	Differential Nonlinearity	> -1	—	< 1	LSb	AVSS = 0V, AVDD = 3.3V (Note 2)
AD23b	GERR	Gain Error (Dedicated Core)	> 0	8	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		Gain Error (Shared Core)	> 5	15	< 22	LSb	
AD24b	EOFF	Offset Error (Dedicated Core)	> 2	9	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		Offset Error (Shared Core)	> 5	17	< 22	LSb	
AD25b	—	Monotonicity	—	—	—	—	Guaranteed
Dynamic Performance							
AD31b	SINAD	Signal-to-Noise and Distortion	63	—	> 65	dB	(Notes 3, 4)
AD34b	ENOB	Effective Number of Bits	10.3	—	—	bits	(Notes 3, 4)

Note 1: These parameters are not characterized or tested in manufacturing.
2: No missing codes, limits based on characterization results.
3: These parameters are characterized but not tested in manufacturing.
4: Characterized with a 15 kHz sine wave.
5: The ADC module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is ensured, but not characterized.

TABLE 30-53: ANALOG-TO-DIGITAL CONVERSION TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[2] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristics	Min.	Typ.[1]	Max.	Units	Conditions
Clock Parameters							
AD50	TAD	ADC Clock Period	14.28	—	—	ns	
Throughput Rate							
AD51	FTP	SH0-SH3	—	—	3.25	Msps	70 MHz ADC clock, 12 bits, no pending conversion at time of trigger
		SH4	—	—	3.25	Msps	

Note 1: These parameters are characterized but not tested in manufacturing.

2: The ADC module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is ensured, but not characterized.

TABLE 30-54: HIGH-SPEED ANALOG COMPARATOR MODULE SPECIFICATIONS

AC/DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[2] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Comments
CM10	VIOFF	Input Offset Voltage	-35	±5	35	mV	
CM11	VICM	Input Common-Mode Voltage Range[1]	0	—	AVDD	V	
CM13	CMRR	Common-Mode Rejection Ratio	60	—	—	dB	
CM14	TRESP	Large Signal Response	—	15	—	ns	V+ input step of 100 mV while V- input is held at AVDD/2. Delay measured from analog input pin to PWMx output pin.
CM15	VHYST	Input Hysteresis	5	10	20	mV	Depends on HYSSEL<1:0>
CM16	TON	Comparator Enabled to Valid Output	—	—	1	µs	

Note 1: These parameters are for design guidance only and are not tested in manufacturing.

2: The comparator module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is tested, but not characterized.

TABLE 30-55: DACx MODULE SPECIFICATIONS

AC/DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[2] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
DA01	EXTREF	External Voltage Reference[1]	1	—	AVDD	V	
DA02	CVRES	Resolution	12			bits	
DA03	INL	Integral Nonlinearity Error	-16	-12	0	LSB	
DA04	DNL	Differential Nonlinearity Error	-1.8	±1	1.8	LSB	
DA05	EOFF	Offset Error	-8	3	15	LSB	
DA06	EG	Gain Error	-1.2	-0.5	0	%	
DA07	TSET	Settling Time[1]	—	700	—	ns	Output with 2% of desired output voltage with a 10-90% or 90-10% step

Note 1: Parameters are for design guidance only and are not tested in manufacturing.
2: The DACx module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is tested, but not characterized.

TABLE 30-56: DACx OUTPUT (DACOUTx PIN) SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[1] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
DA11	RLOAD	Resistive Output Load Impedance	10K	—	—	Ohm	
DA11a	CLOAD	Output Load Capacitance	—	—	35	pF	Including output pin capacitance
DA12	IOUT	Output Current Drive Strength	—	300	—	μA	Sink and source
DA13	VRANGE	Output Drive Voltage Range at Current Drive of 300 μA	AVSS + 250 mV	—	AVDD – 900 mV	V	
DA14	VLRANGE	Output Drive Voltage Range at Reduced Current Drive of 50 μA	AVSS + 50 mV	—	AVDD – 500 mV	V	
DA15	IDD	Current Consumed when Module is Enabled	—	—	1.3 x IOUT	μA	Module will always consume this current, even if no load is connected to the output
DA30	VOFFSET	Input Offset Voltage	—	±5	—	mV	

Note 1: The DACx module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is tested, but not characterized.

TABLE 30-57: PGAx MODULE SPECIFICATIONS

AC/DC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[1] Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**		**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
PA01	VIN	Input Voltage Range		AVSS – 0.3	—	AVDD + 0.3	V	
PA02	VCM	Common-Mode Input Voltage Range		AVSS	—	AVDD – 1.6	V	
PA03	VOS	Input Offset Voltage		-10	—	10	mV	
PA04	VOS	Input Offset Voltage Drift with Temperature		—	±15	—	µV/°C	
PA05	RIN+	Input Impedance of Positive Input		—	>1M || 7 pF	—	Ω|| pF	
PA06	RIN-	Input Impedance of Negative Input		—	10K || 7 pF	—	Ω|| pF	
PA07	GERR	Gain Error		-2	—	2	%	Gain = 4x, 8x
				-3	—	3	%	Gain = 16x
				-4	—	4	%	Gain = 32x, 64x
PA08	LERR	Gain Nonlinearity Error		—	—	0.5	%	% of full scale, Gain = 16x
PA09	IDD	Current Consumption		—	2.0	—	mA	Module is enabled with a 2-volt P-P output voltage swing
PA10a	BW	Small Signal Bandwidth (-3 dB)	G = 4x	—	10	—	MHz	
PA10b			G = 8x	—	5	—	MHz	
PA10c			G = 16x	—	2.5	—	MHz	
PA10d			G = 32x	—	1.25	—	MHz	
PA10e			G = 64x	—	0.625	—	MHz	
PA11	OST	Output Settling Time to 1% of Final Value		—	0.4	—	µs	Gain = 16x, 100 mV input step change
PA12	SR	Output Slew Rate		—	40	—	V/µs	Gain = 16x
PA13	TGSEL	Gain Selection Time		—	1	—	µs	
PA14	TON	Module Turn On/Setting Time		—	—	10	µs	

Note 1: The PGAx module is functional at VBORMIN < VDD < VDDMIN, but with degraded performance. Unless otherwise stated, module functionality is tested, but not characterized.

TABLE 30-58: CONSTANT-CURRENT SOURCE SPECIFICATIONS

DC CHARACTERISTICS				Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated)[1] Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended			
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
CC01	I_{DD}	Current Consumption	—	30	—	µA	
CC02	I_{REG}	Regulation of Current with Voltage On	—	±3	—	%	
CC03	I_{OUT}	Current Output at Terminal	—	10	—	µA	

Note 1: The constant-current source module is functional at V_{BORMIN} < V_{DD} < V_{DDMIN}, but with degraded performance. Unless otherwise stated, module functionality is tested, but not characterized.

TABLE 30-59: DMA MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended				
Param No.	**Characteristic**	**Min.**	**Typ.[1]**	**Max.**	**Units**	**Conditions**
DM1	DMA Byte/Word Transfer Latency	1 T_{CY}[2]	—	—	ns	

Note 1: These parameters are characterized, but not tested in manufacturing.

2: Because DMA transfers use the CPU data bus, this time is dependent on other functions on the bus.

NOTES:

31.0 DC AND AC DEVICE CHARACTERISTICS GRAPHS

Note: The graphs provided following this note are a statistical summary based on a limited number of samples and are provided for only. The performance characteristics listed herein are not tested or guaranteed. In some graphs, the data presented may be outs range (e.g., outside specified power supply range) and therefore, outside the warranted range.

FIGURE 31-1: VOH – 4x DRIVER PINS

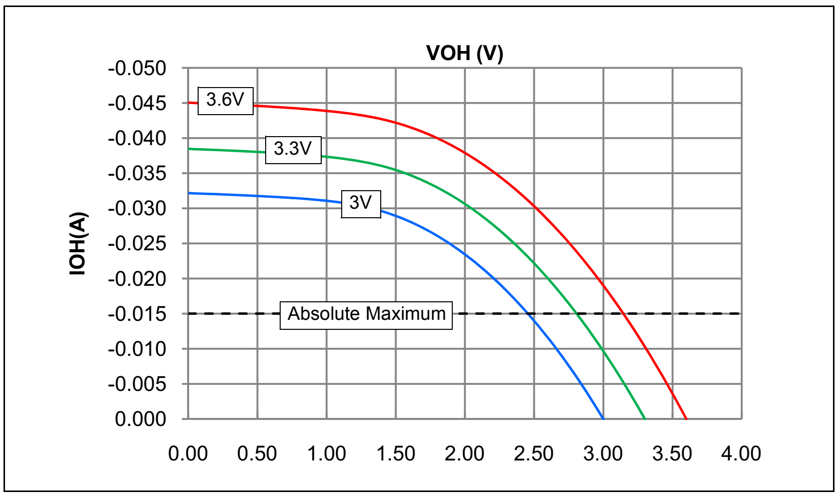


FIGURE 31-2: VOH – 8x DRIVER PINS

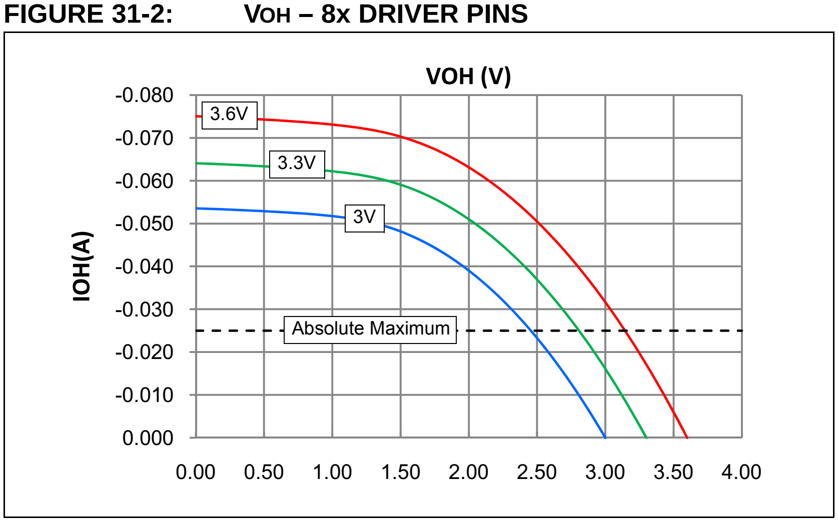


FIGURE 31-3: VOL – 4x DRIVER PINS

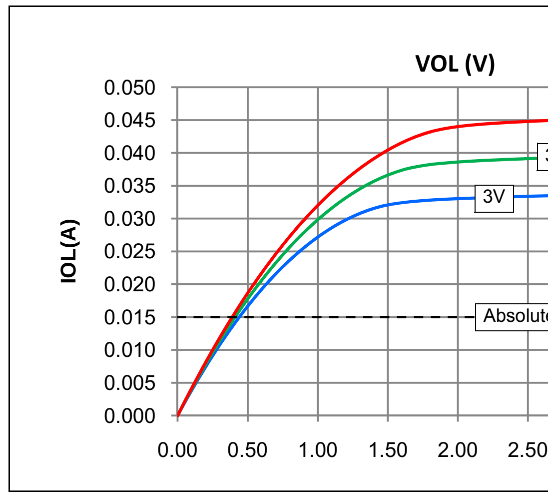


FIGURE 31-4: VOL – 8x DRIVER PINS

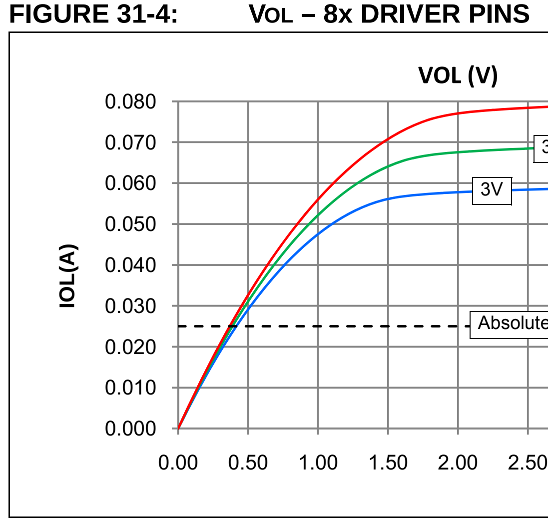


FIGURE 31-5: TYPICAL I_{PD} CURRENT @ V_{DD} = 3.3V

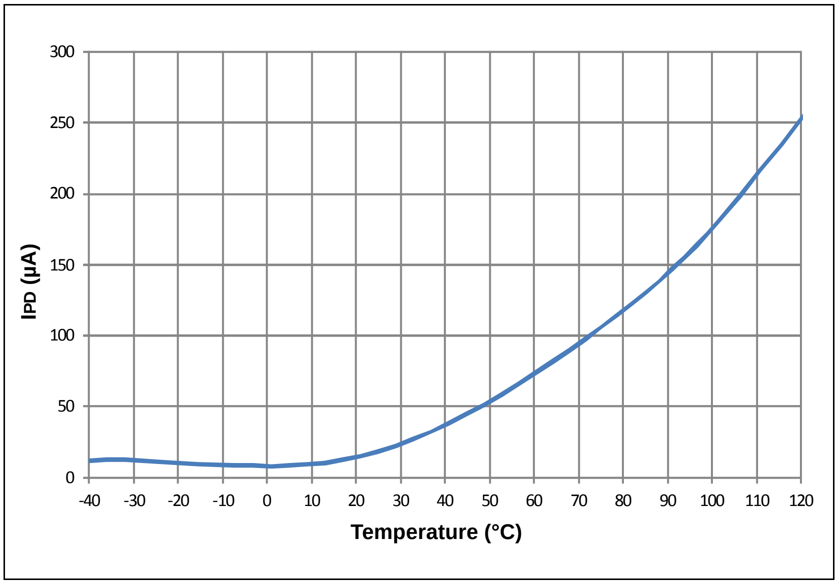


FIGURE 31-6: TYPICAL I_{DD} CURRENT @ V_{DD} = 3.3V, +25°C

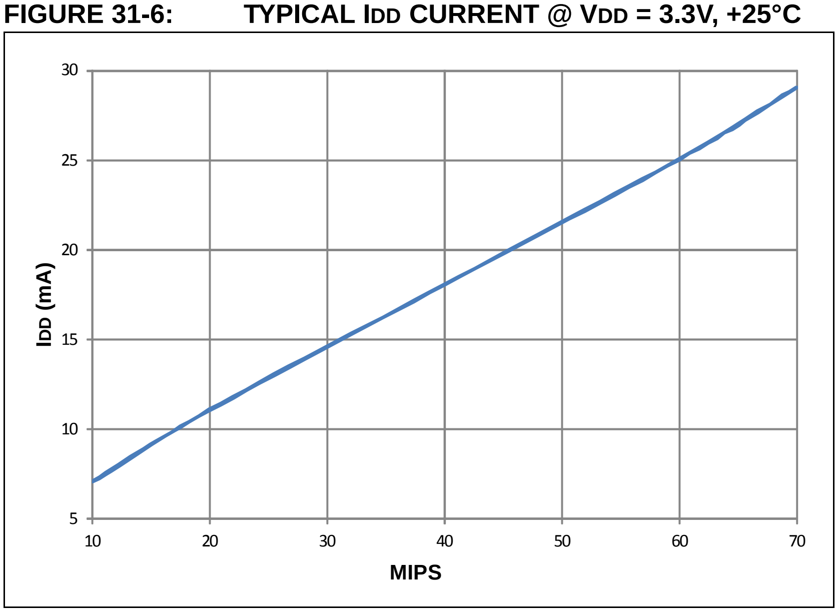


FIGURE 31-7: TYPICAL I_{DOZE} CURREN

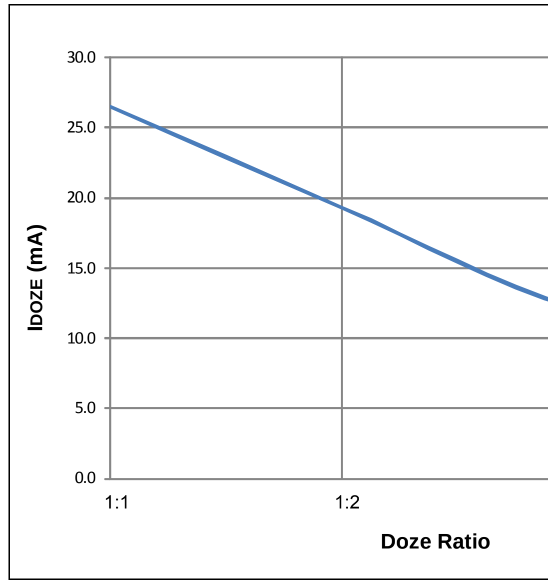


FIGURE 31-8: TYPICAL I_{IDLE} CURREN

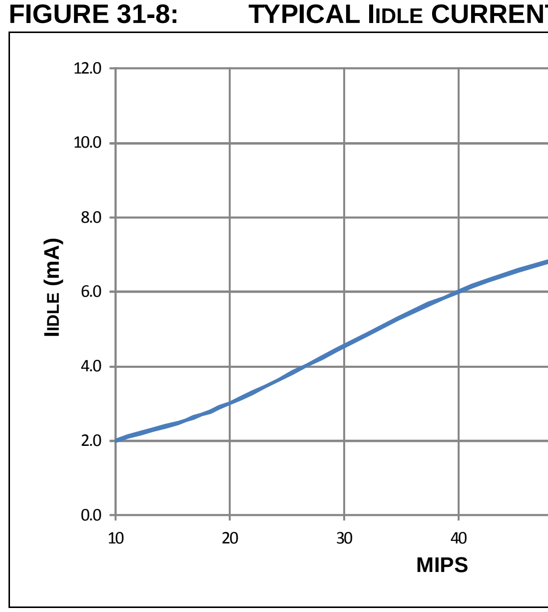


FIGURE 31-9: TYPICAL FRC FREQUENCY @ VDD = 3.3V

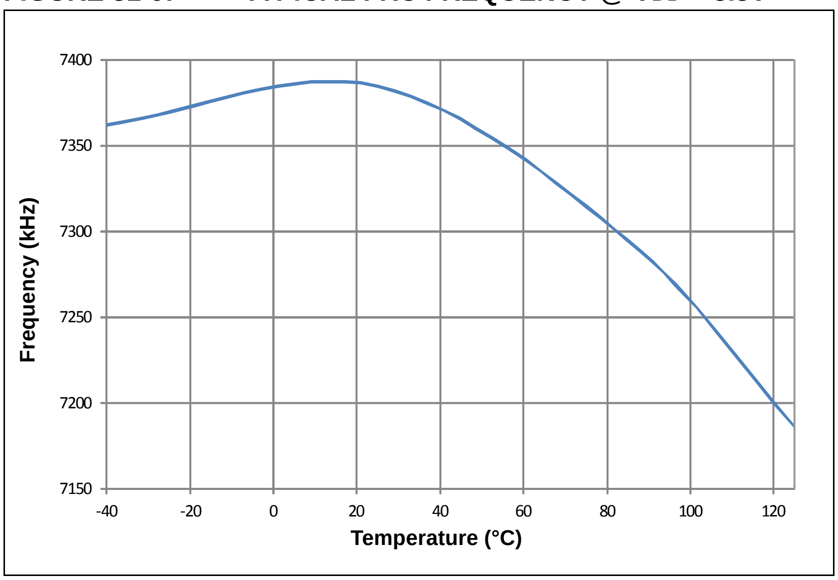
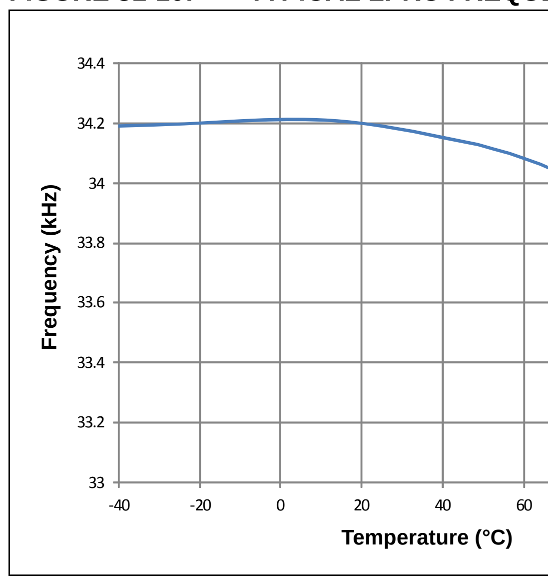


FIGURE 31-10: TYPICAL LPRC FREQUE

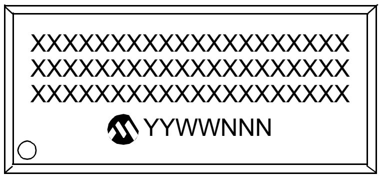


NOTES:

32.0 PACKAGING INFORMATION

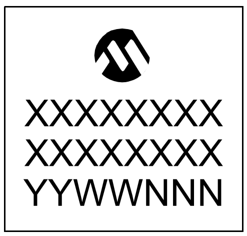
32.1 Package Marking Information

28-Lead SOIC (7.50 mm)



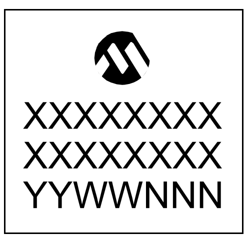
Example

28-Lead UQFN (6x6x0.55 mm)



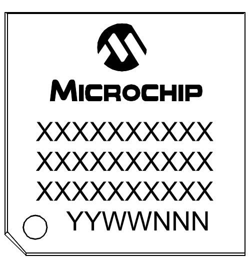
Example

28-Lead QFN-S (6x6x0.9 mm)

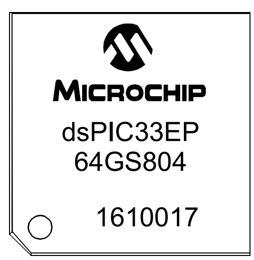


Example

44-Lead TQFP (10x10x1 mm)



Example

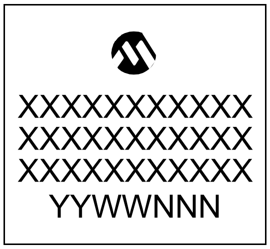


Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

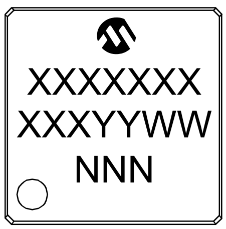
32.1 Package Marking Information (Continued)

44-Lead QFN (8x8 mm)



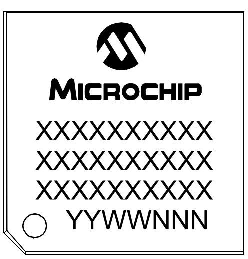
Example

48-Lead TQFP (7x7x1.0 mm)

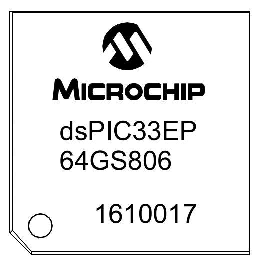


Example

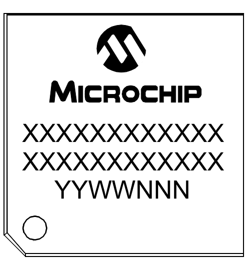
64-Lead TQFP (10x10x1 mm)



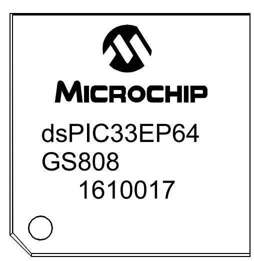
Example



80-Lead TQFP (12x12x1 mm)



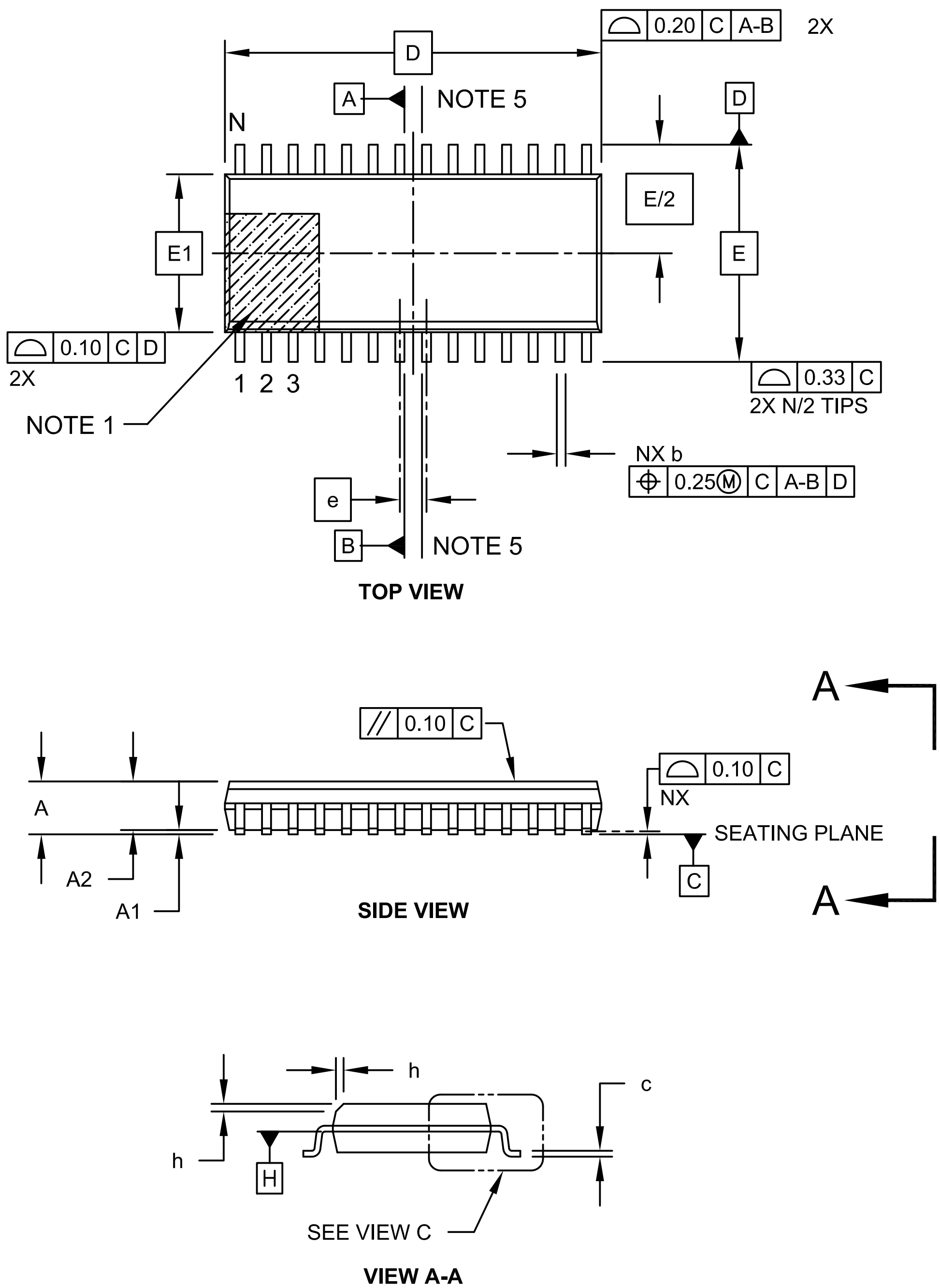
Example



32.2 Package Details

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

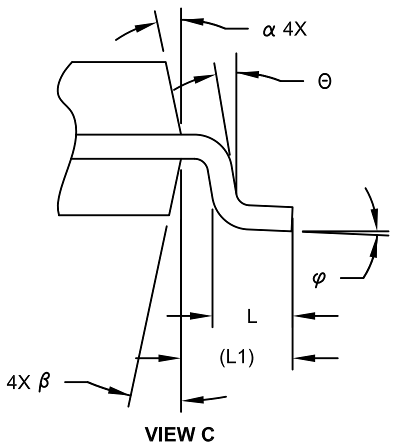
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging

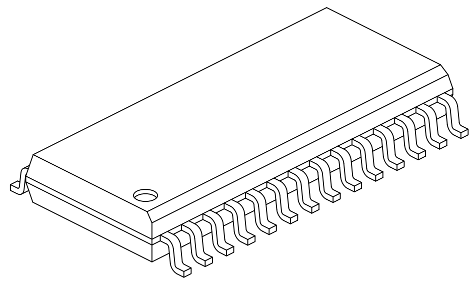


Microchip Technology Drawing C04-052C Sheet 1 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging





Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	1.27 BSC		
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1	1.40 REF		
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

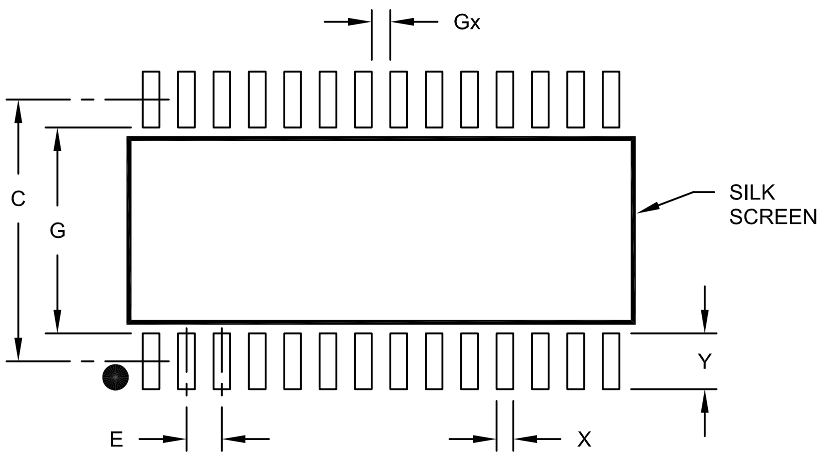
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

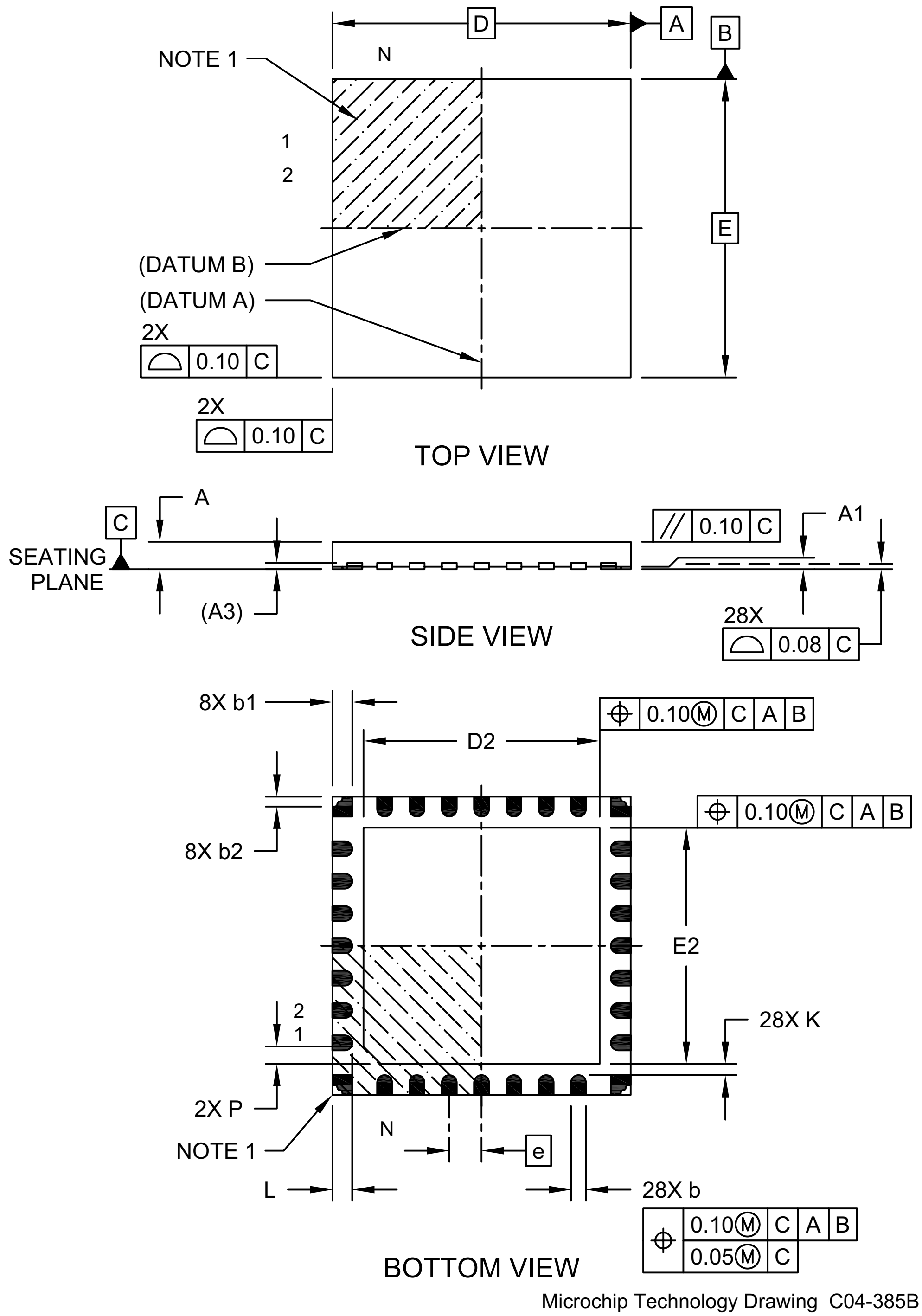
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

28-Lead Ultra Thin Plastic Quad Flat, No Lead Package (2N) - 6x6x0.55 mm Body [UQFN] With 4.65x4.65 mm Exposed Pad and Corner Anchors

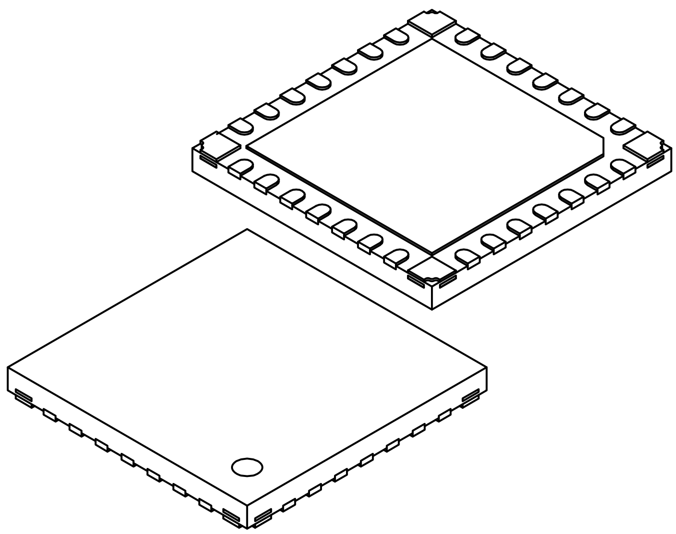
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-385B Sheet 1 of 2

28-Lead Ultra Thin Plastic Quad Flat, No Lead Package (2N) - 6x6x0.55 mm Body [UQFN] With 4.65x4.65 mm Exposed Pad and Corner Anchors

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units	MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.45	0.50	0.55
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.127 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	4.55	4.65	4.75
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	4.55	4.65	4.75
Exposed Pad Corner Chamfer	P	-	0.35	-
Terminal Width	b	0.25	0.30	0.35
Corner Anchor Pad	b1	0.35	0.40	0.43
Corner Pad, Metal Free Zone	b2	0.15	0.20	0.25
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

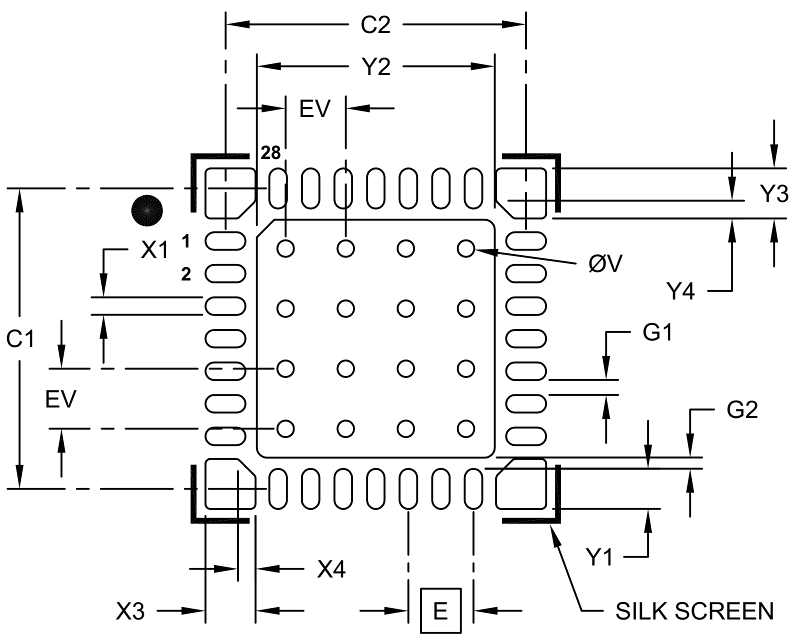
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-385B Sheet 2 of 2

28-Lead Ultra Thin Plastic Quad Flat, No Lead Package (2N) - 6x6x0.55 mm Body [UQFN] With 4.65x4.65 mm Exposed Pad and Corner Anchors

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	X2			4.75
Optional Center Pad Length	Y2			4.75
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.35
Contact Pad Length (X28)	Y1			0.80
Corner Anchor (X4)	X3			1.00
Corner Anchor (X4)	Y3			1.00
Corner Anchor Chamfer (X4)	X4			0.35
Corner Anchor Chamfer (X4)	Y4			0.35
Contact Pad to Pad (X28)	G1	0.20		
Contact Pad to Center Pad (X28)	G2	0.20		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

Notes:

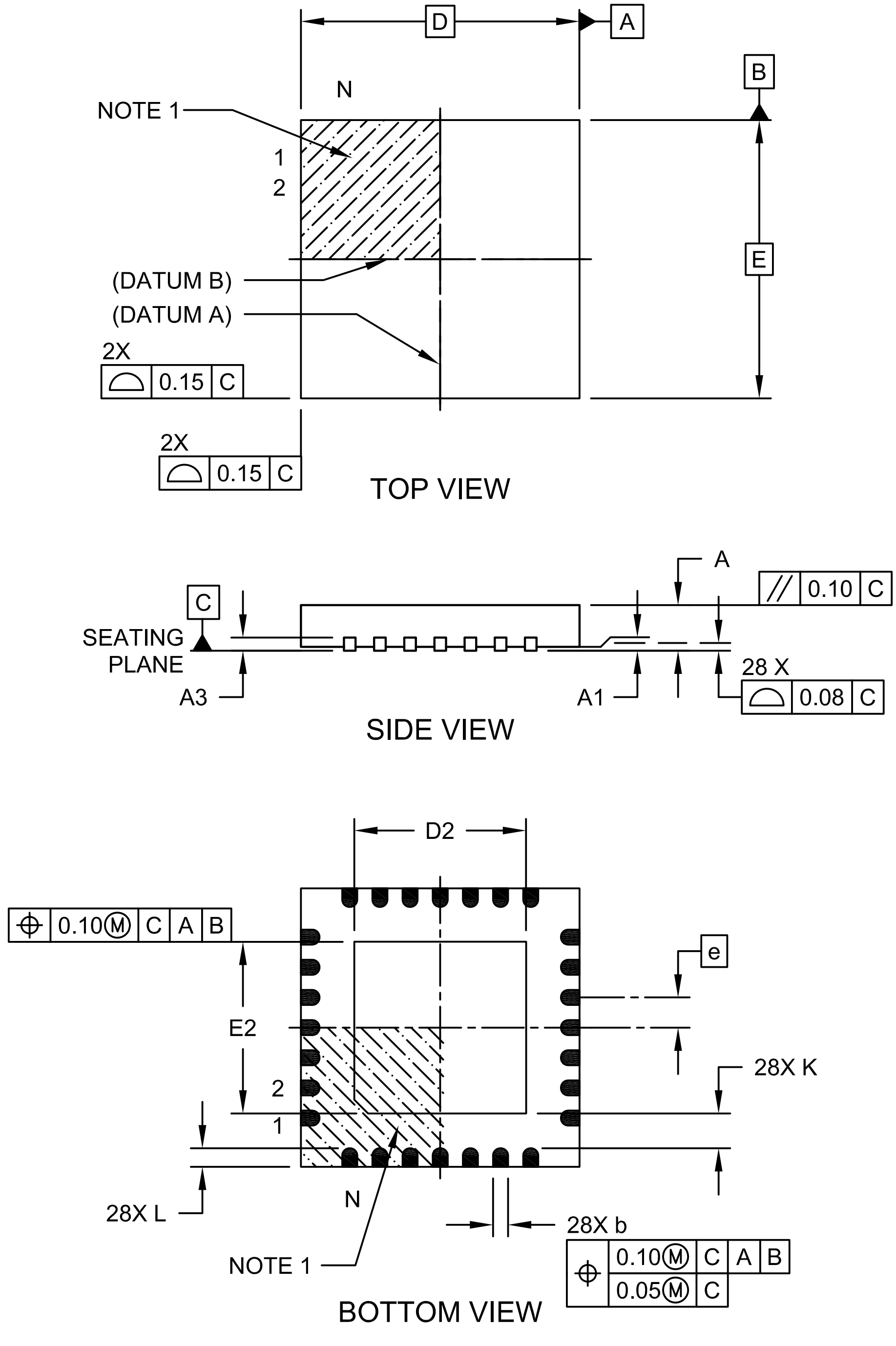
1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2385B

Note: Corner anchor pads are not connected internally and are designed as mechanical features when the package is soldered to the PCB.

28-Lead Plastic Quad Flat, No Lead Package (MM) - 6x6x0.9mm Body [QFN-S]
With 0.40 mm Terminal Length

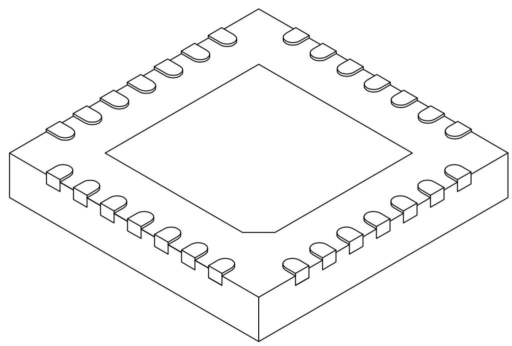
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-124C Sheet 1 of 2

28-Lead Plastic Quad Flat, No Lead Package (MM) - 6x6x0.9mm Body [QFN-S] With 0.40 mm Terminal Length

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.20 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.70
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.70
Terminal Width	b	0.23	0.30	0.35
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

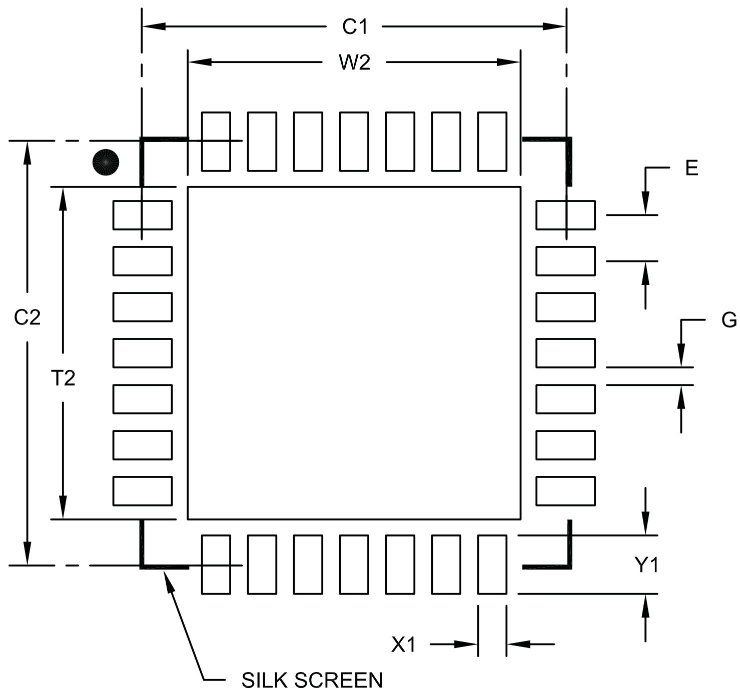
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-124C Sheet 2 of 2

28-Lead Plastic Quad Flat, No Lead Package (MM) – 6x6x0.9 mm Body [QFN-S] with 0.40 mm Contact Length

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			4.70
Optional Center Pad Length	T2			4.70
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.40
Contact Pad Length (X28)	Y1			0.85
Distance Between Pads	G	0.25		

Notes:

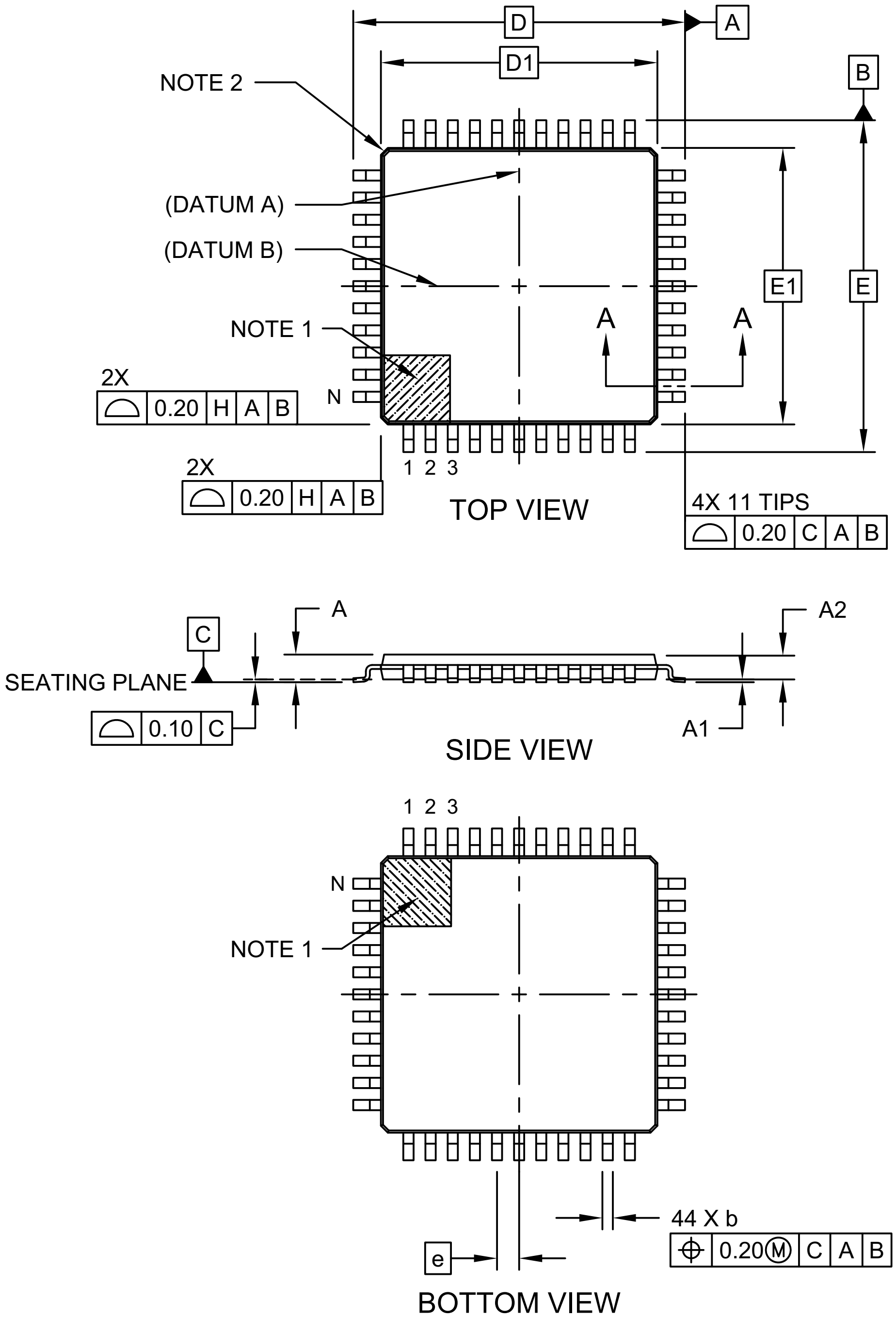
1. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2124A

44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [TQFP]

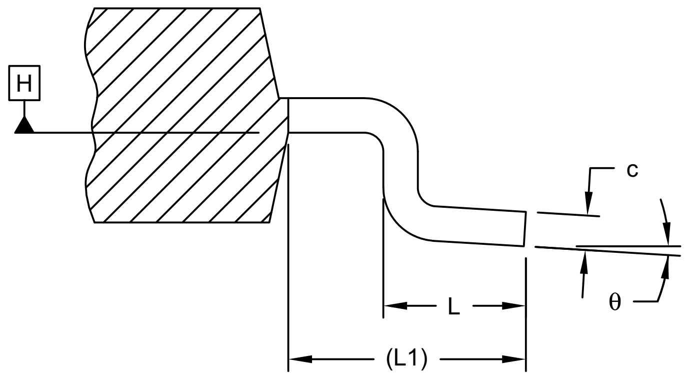
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



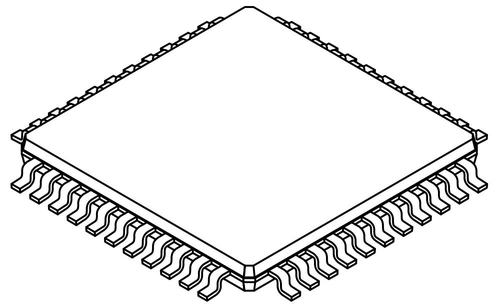
Microchip Technology Drawing C04-076C Sheet 1 of 2

44-Lead Plastic Thin Quad Flatpack (PT) - 10x10x1.0 mm Body [TQFP]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



SECTION A-A



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Number of Leads	N	44		
Lead Pitch	e	0.80 BSC		
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Overall Width	E	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Width	b	0.30	0.37	0.45
Lead Thickness	c	0.09	-	0.20
Lead Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	θ	0°	3.5°	7°

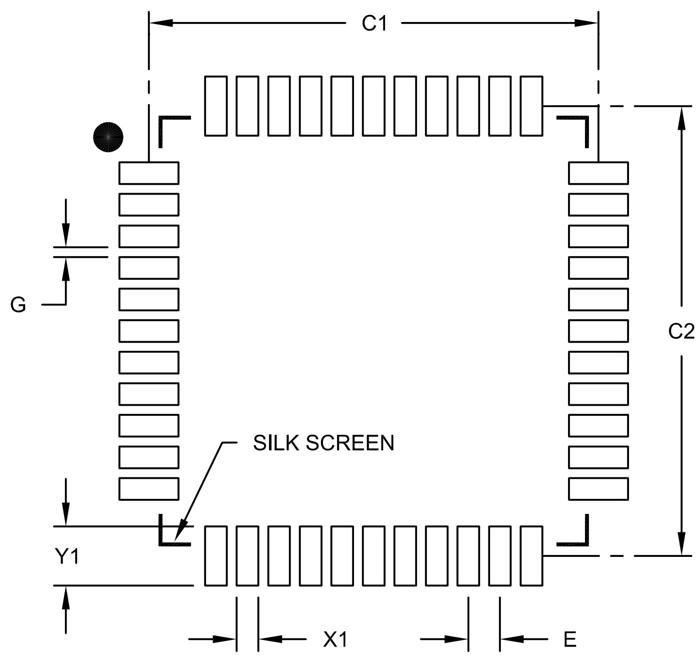
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Exact shape of each corner is optional.
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076C Sheet 2 of 2

44-Lead Plastic Thin Quad Flatpack (PT) 10X10X1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

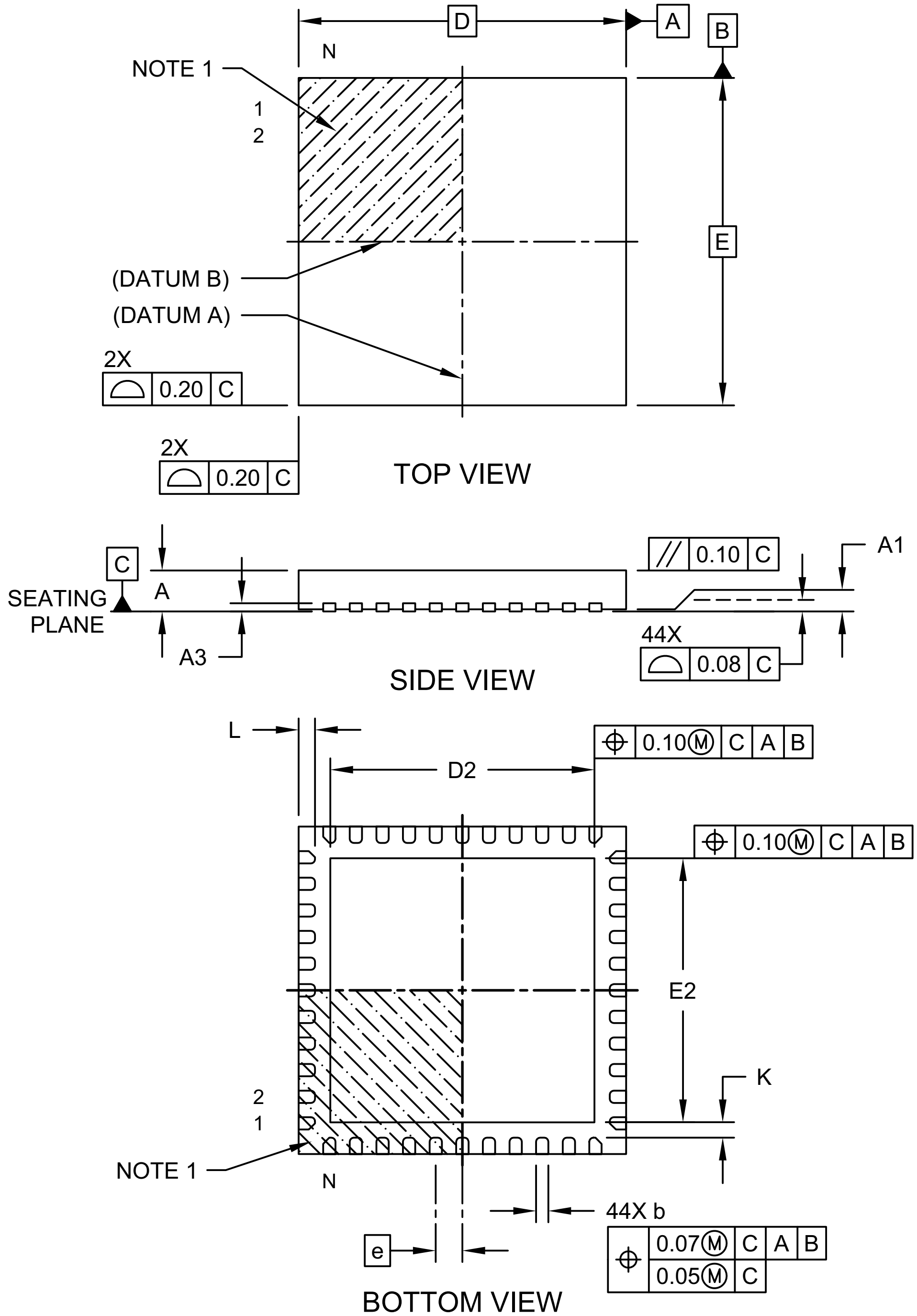
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076B

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN or VQFN]

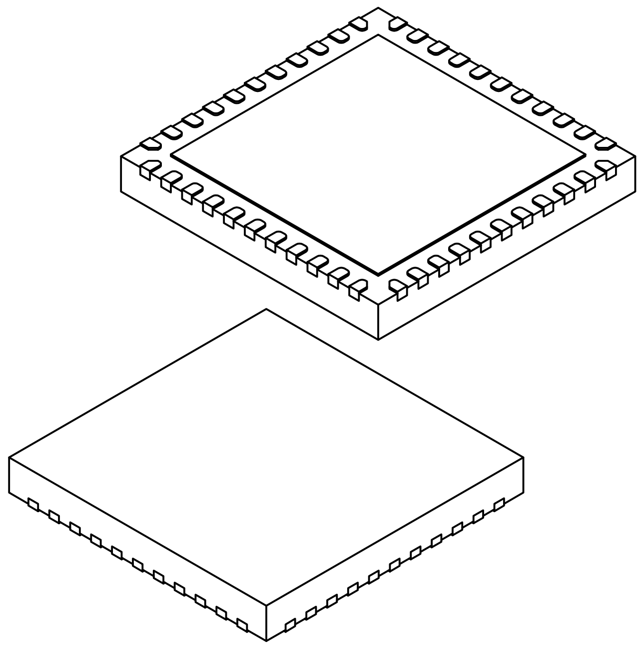
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-103D Sheet 1 of 2

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN or VQFN]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		44	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.20 REF	
Overall Width	E		8.00 BSC	
Exposed Pad Width	E2	6.25	6.45	6.60
Overall Length	D		8.00 BSC	
Exposed Pad Length	D2	6.25	6.45	6.60
Terminal Width	b	0.20	0.30	0.35
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

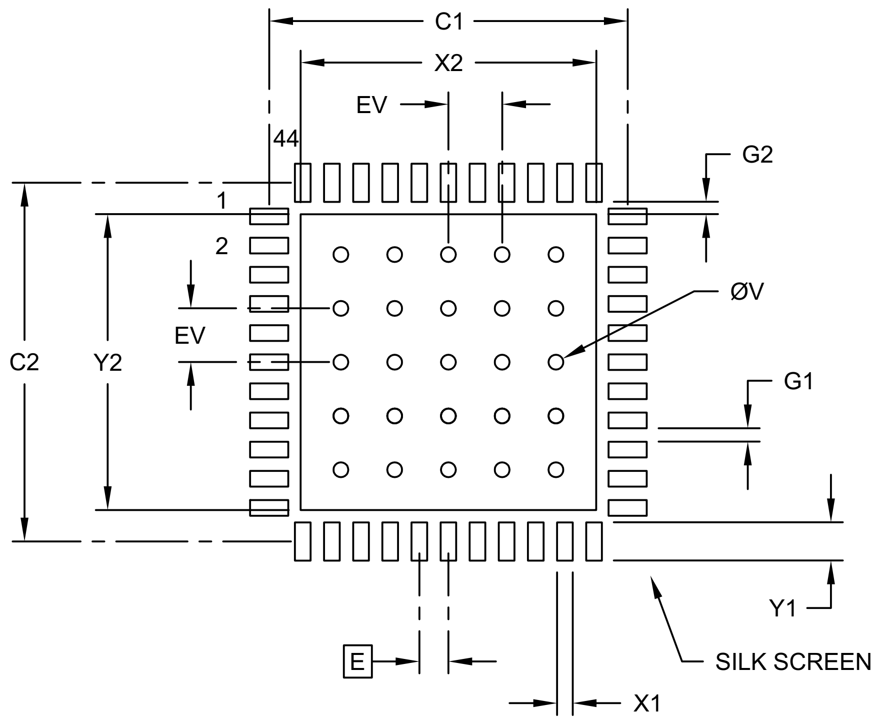
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103D Sheet 2 of 2

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN or VQFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	X2			6.60
Optional Center Pad Length	Y2			6.60
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.85
Contact Pad to Contact Pad (X40)	G1	0.30		
Contact Pad to Center Pad (X44)	G2	0.28		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

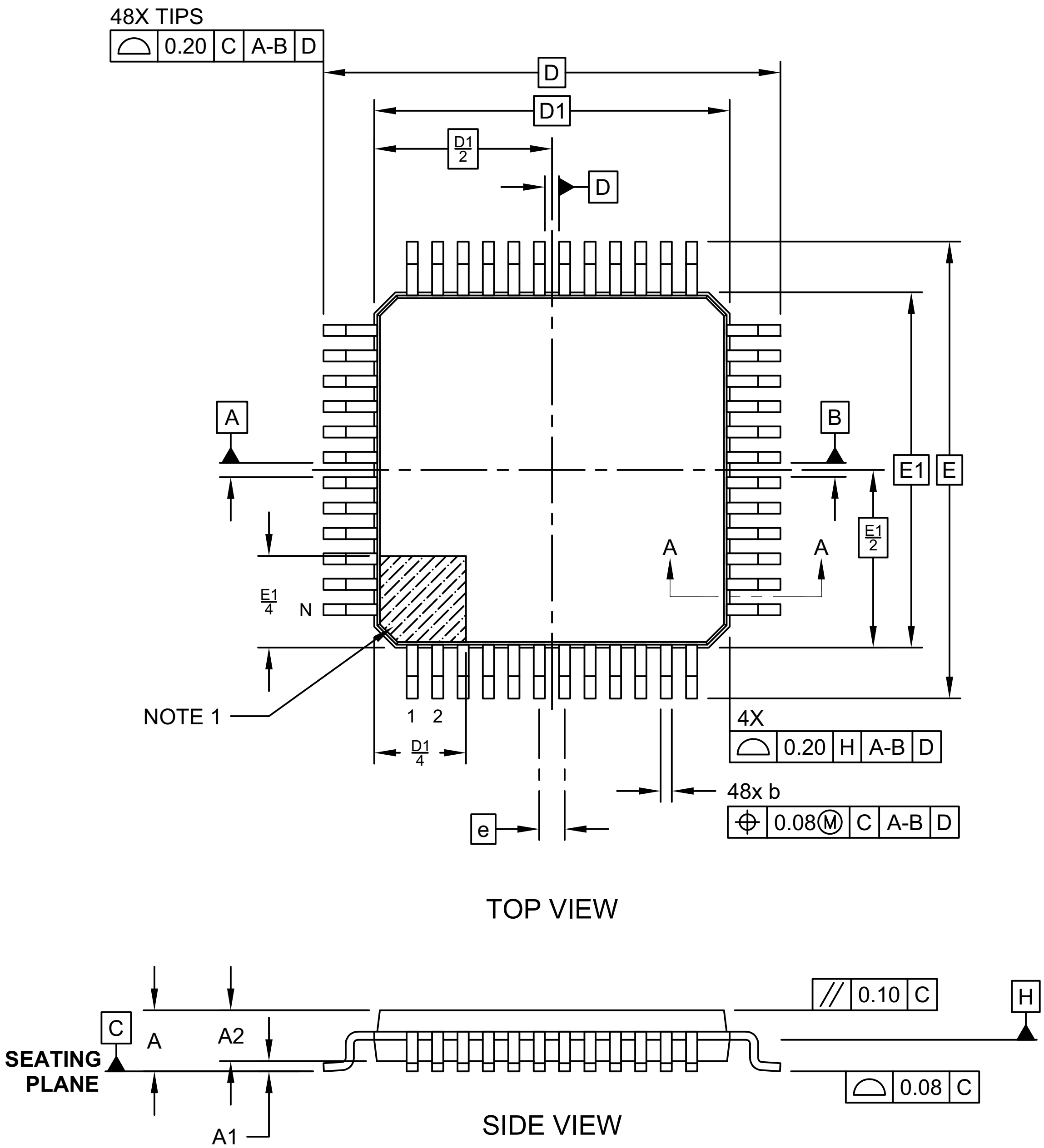
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing No. C04-2103C

48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

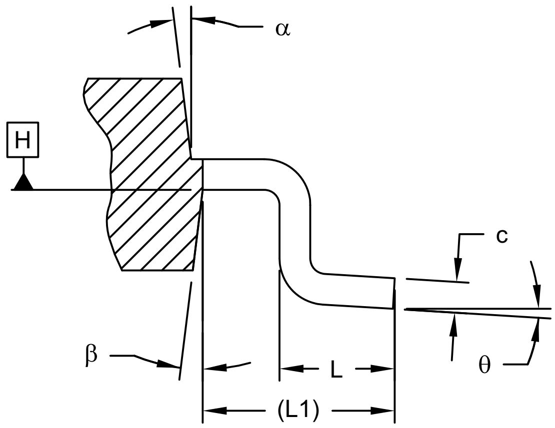
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



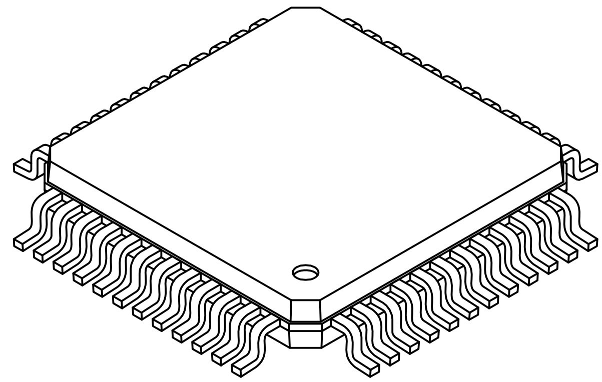
Microchip Technology Drawing C04-300-PT Rev A Sheet 1 of 2

48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



SECTION A-A



	Units	MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		48	
Lead Pitch	e		0.50 BSC	
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	φ	0°	3.5°	7°
Overall Width	E		9.00 BSC	
Overall Length	D		9.00 BSC	
Molded Package Width	E1		7.00 BSC	
Molded Package Length	D1		7.00 BSC	
Lead Thickness	c	0.09	-	0.16
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

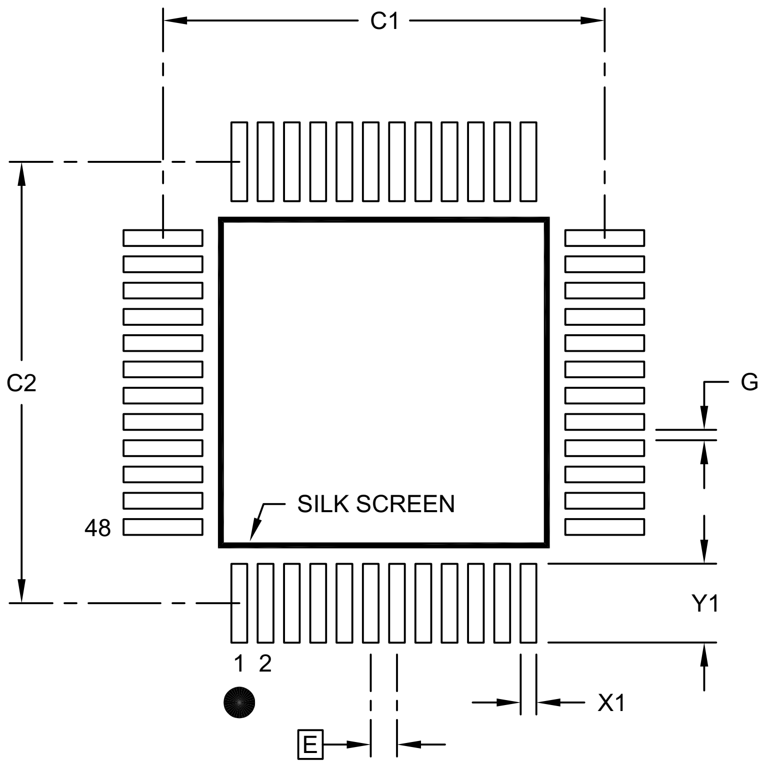
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A-B and D to be determined at center line between leads where leads exit plastic body at datum plane H

Microchip Technology Drawing C04-300-PT Rev A Sheet 2 of 2

48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		8.40	
Contact Pad Spacing	C2		8.40	
Contact Pad Width (X48)	X1			0.30
Contact Pad Length (X48)	Y1			1.50
Distance Between Pads	G	0.20		

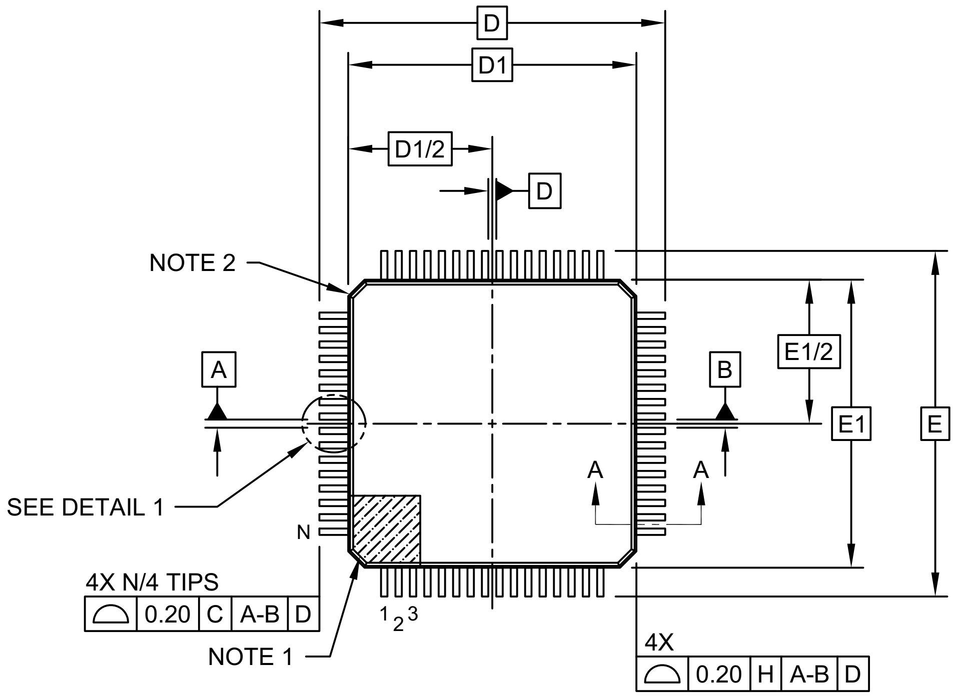
Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

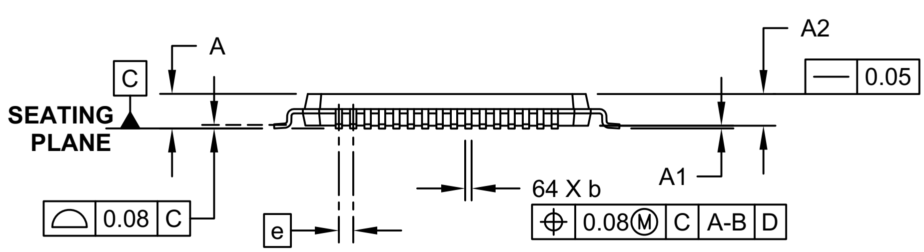
Microchip Technology Drawing C04-2300-PT Rev A

64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW

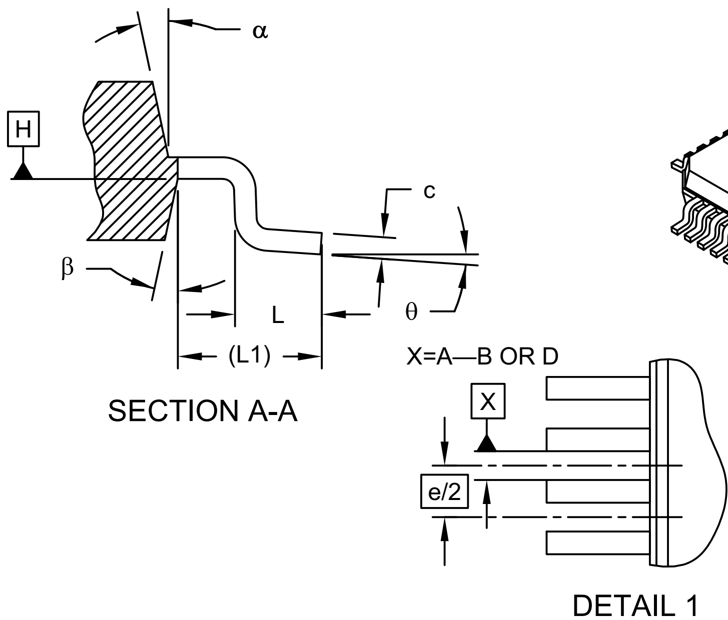


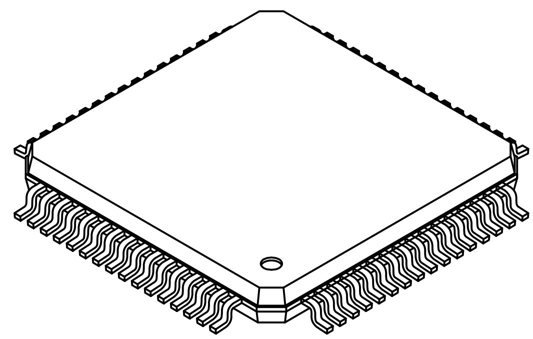
SIDE VIEW

Microchip Technology Drawing C04-085C Sheet 1 of 2

64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging





Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		64	
Lead Pitch	e		0.50 BSC	
Overall Height	A	-	-	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	-	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	φ	0°	3.5°	7°
Overall Width	E		12.00 BSC	
Overall Length	D		12.00 BSC	
Molded Package Width	E1		10.00 BSC	
Molded Package Length	D1		10.00 BSC	
Lead Thickness	c	0.09	-	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

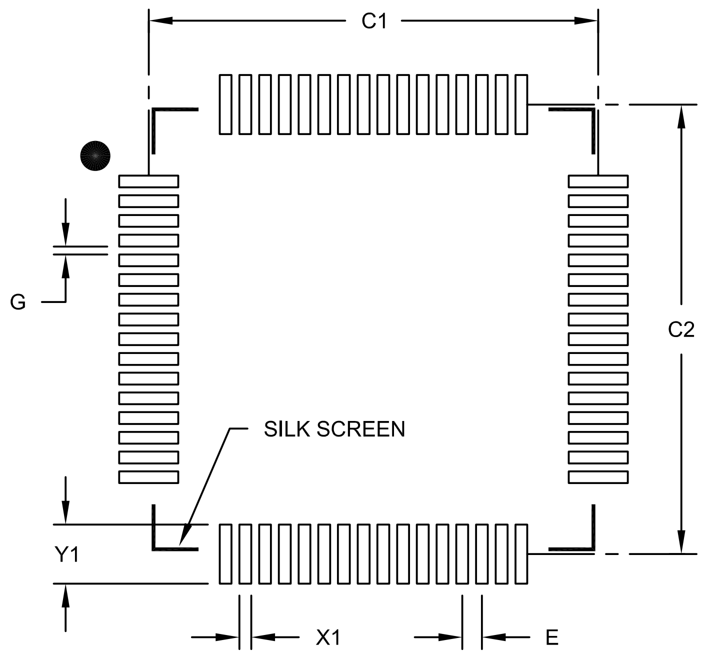
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085C Sheet 2 of 2

64-Lead Plastic Thin Quad Flatpack (PT) 10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

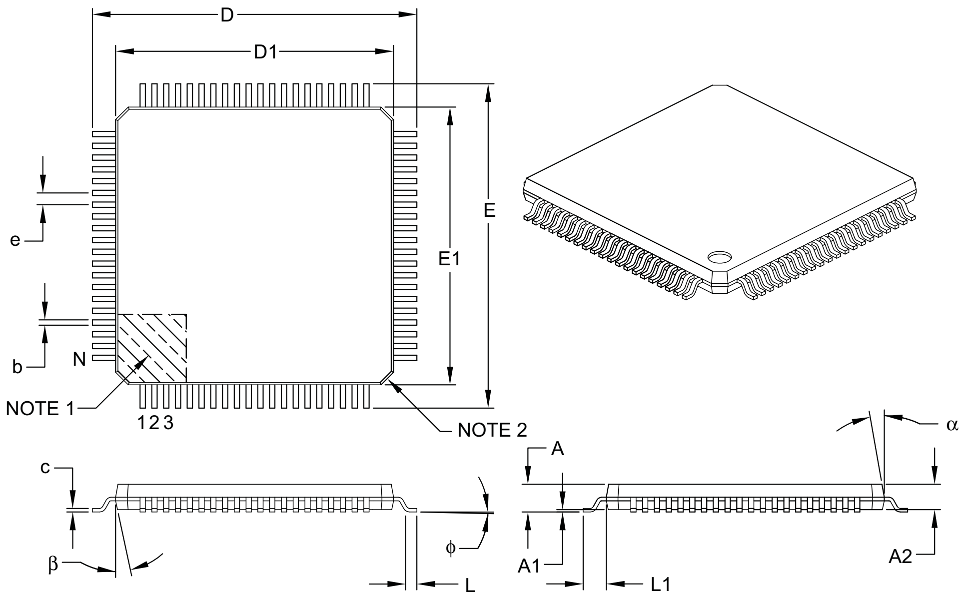
1. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2085B

80-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		80	
Lead Pitch	e		0.50 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	φ	0°	3.5°	7°
Overall Width	E		14.00 BSC	
Overall Length	D		14.00 BSC	
Molded Package Width	E1		12.00 BSC	
Molded Package Length	D1		12.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

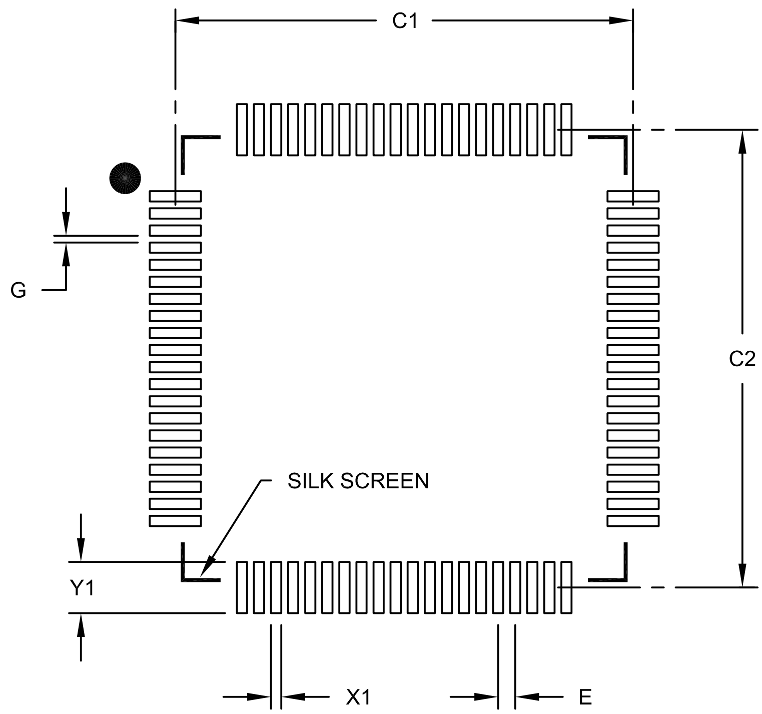
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-092B

80-Lead Plastic Thin Quad Flatpack (PT) - 12x12x1mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X80)	X1			0.30
Contact Pad Length (X80)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2092B

NOTES:

APPENDIX A: REVISION HISTORY

Revision A (May 2016)

This is the initial version of the document.

Revision B (January 2017)

- Sections:
  - Updates Note 1 in **Section 5.0 “Flash Program Memory”**.
- Tables:
  - Updates the device description table on page 2.
  - Updates Table 1-1, Table 4-2, Table 4-11, Table 7-1, Table 8-1, Table 11-11, Table 11-13, Table 17-1, Table 30-3, Table 30-4, Table 30-6, Table 30-7, Table 30-8, Table 30-9, Table 30-10, Table 30-11, Table 30-52, Table 30-54 and Table 30-55.
  - Adds Table 11-6, Table 11-7, Table 11-8, Table 11-9 and Table 11-10.
- Figures:
  - Updates the Pin Function tables in the Pin Diagram figures on pages 5 through 8.
  - Updates Figure 4-1, Figure 17-1, Figure 18-1 and Figure 18-2.
- Registers:
  - Updates Register 3-3, Register 16-5, Register 17-11, Register 18-1 and Register 19-2.
  - Adds Register 11-1, Register 11-2, Register 11-3, Register 11-4, Register 11-5, Register 11-6, Register 11-7 and Register 11-8.

NOTES:

INDEX

A

B

S

T

U

V

W

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

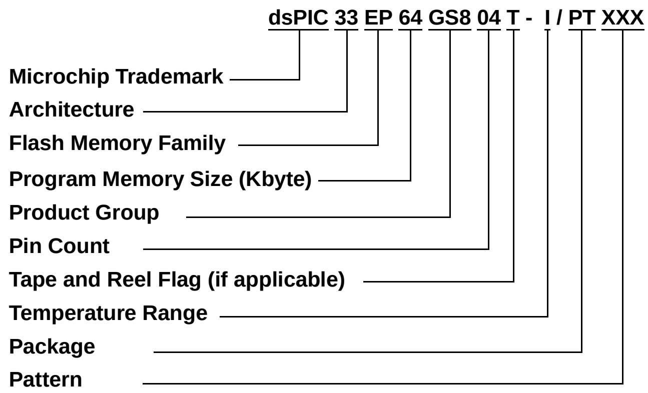
Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

NOTES:

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



Architecture:	33	=	16-Bit Digital Signal Controller
Flash Memory Family:	EP	=	Enhanced Performance
Product Group:	GS	=	SMPS Family
Pin Count:	02	=	28-pin
	04	=	44-pin
	05	=	48-pin
	06	=	64-pin
	08	=	80-pin
Temperature Range:	I	=	-40°C to +85°C (Industrial)
	E	=	-40°C to +125°C (Extended)
Package:	ML	=	Plastic Quad, No Lead Package – (44-pin) 8x8 mm body (QFN)
	MM	=	Plastic Quad, No Lead Package – (28-pin) 6x6 mm body (QFN-S)
	2N	=	Plastic Quad Flat, No Lead Package – (28-pin) 6x6 mm body (UQFN)
	PT	=	Plastic Thin Quad Flatpack – (44-pin) 10x10 mm body (TQFP)
	PT	=	Plastic Thin Quad Flatpack – (48-pin) 7x7 mm body (TQFP)
	PT	=	Plastic Thin Quad Flatpack – (64-pin) 10x10 mm body (TQFP)
	PT	=	Plastic Thin Quad Flatpack – (80-pin) 12x12 mm body (TQFP)
	SO	=	Plastic Small Outline, Wide – (28-pin) 7.50 mm body (SOIC)

Examples:

dsPIC33EP64GS804-I/PT:
dsPIC33, Enhanced Performance, 64-Kbyte Program Memory, SMPS, 44-Pin, Industrial Temperature, TQFP Package.

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-1251-9

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-3326-8000
Fax: 86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

France - Saint Cloud
Tel: 33-1-30-60-70-00

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820