PMS

Pressure Monitoring Sensor

SP27

High integrated single-chip pressure sensor with a low power embedded micro-controller

SP27 Version A5

1300kPa

Datasheet

Revision 1.0, 2012-02-07

Sense & Control

CONFIDENTIAL

SP27 High integrated single-chip pressure sensor with a low power embedded micro-controller
CONFIDENTIAL
Revision History: 2012-02-07, Revision 1.0

Previous Revision: Revision 0.1

Page	Subjects (major changes since last revision)
Page 80	I2C Command - Measure Pressure Function Update

We Listen to Your Comments

Any information within this document that you feel is wrong, unclear or missing at all?
Your feedback will help us to continuously improve the quality of this document.
Please send your proposal (including a reference to this document) to:

sensors@infineon.com

Trademarks of Infineon Technologies AG

AURIX™, C166™, CanPAK™, CIPOS™, CIPURSE™, EconoPACK™, CoolMOS™, CoolSET™, CORECONTROL™, CROSSAVE™, DAVE™, EasyPIM™, EconoBRIDGE™, EconoDUAL™, EconoPIM™, EiceDRIVER™, eupec™, FCOS™, HITFET™, HybridPACK™, I²RF™, ISOFACE™, IsoPACK™, MIPAQ™, ModSTACK™, my-d™, NovalithIC™, OptiMOS™, ORIGA™, PRIMARION™, PrimePACK™, PrimeSTACK™, PRO-SIL™, PROFET™, RASIC™, ReverSave™, SatRIC™, SIEGET™, SINDRION™, SIPMOS™, SmartLEWIS™, SOLID FLASH™, TEMPFET™, thinQ!™, TRENCHSTOP™, TriCore™.

Other Trademarks

Advance Design System™ (ADS) of Agilent Technologies, AMBA™, ARM™, MULTI-ICE™, KEIL™, PRIMECELL™, REALVIEW™, THUMB™, µVision™ of ARM Limited, UK. AUTOSAR™ is licensed by AUTOSAR development partnership. Bluetooth™ of Bluetooth SIG Inc. CAT-iq™ of DECT Forum. COLOSSUS™, FirstGPS™ of Trimble Navigation Ltd. EMV™ of EMVCo, LLC (Visa Holdings Inc.). EPCOS™ of Epcos AG. FLEXGO™ of Microsoft Corporation. FlexRay™ is licensed by FlexRay Consortium. HYPERTERMINAL™ of Hilgraeve Incorporated. IEC™ of Commission Electrotechnique Internationale. IrDA™ of Infrared Data Association Corporation. ISO™ of INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. MATLAB™ of MathWorks, Inc. MAXIM™ of Maxim Integrated Products, Inc. MICROTEC™, NUCLEUS™ of Mentor Graphics Corporation. Mifare™ of NXP. MIPI™ of MIPI Alliance, Inc. MIPS™ of MIPS Technologies, Inc., USA. muRata™ of MURATA MANUFACTURING CO., MICROWAVE OFFICE™ (MWO) of Applied Wave Research Inc., OmniVision™ of OmniVision Technologies, Inc. Openwave™ Openwave Systems Inc. RED HAT™ Red Hat, Inc. RFMD™ RF Micro Devices, Inc. SIRIUS™ of Sirius Satellite Radio Inc. SOLARIS™ of Sun Microsystems, Inc. SPANSION™ of Spansion LLC Ltd. Symbian™ of Symbian Software Limited. TAIYO YUDEN™ of Taiyo Yuden Co. TEAKLITE™ of CEVA, Inc. TEKTRONIX™ of Tektronix Inc. TOKO™ of TOKO KABUSHIKI KAISHA TA. UNIX™ of X/Open Company Limited. VERILOG™, PALLADIUM™ of Cadence Design Systems, Inc. VLYNQ™ of Texas Instruments Incorporated. VXWORKS™, WIND RIVER™ of WIND RIVER SYSTEMS, INC. ZETEX™ of Diodes Zetex Limited.

Last Trademarks Update 2011-02-24

Table of Contents

List of Figures

List of Tables

Table 1   IPin Description 11
Table 2   SP27 - Operating Modes 13
Table 3   State Transitions in NORMAL Mode 17
Table 4   Status of SP27 Blocks in Different States 17
Table 5   Register Naming Convention Wakeup / Reset Value. 24
Table 6   Registers Overview 24
Table 7   Registers Access Types 26
Table 8   Register 0 to F 32
Table 9   GPIO Port Configuration 64
Table 10   External Wakeup Configuration 66
Table 11   I/O Port 1 - Alternative Functionality 66
Table 12   FLASH Write Line Parameters 75
Table 13   FLASH Erase Parameters 75
Table 14   FLASH Check Erase Parameters 76
Table 15   FLASH Check Erase Return values 76
Table 16   FLASH Read Line Parameters 77
Table 17   FLASH Read Line Return values 77
Table 18   FLASH Read Status Return values 78
Table 19   FLASH Check CRC Return values 79
Table 20   Measure Pressure Parameters 80
Table 21   Measure Pressure Return Values 81
Table 22   Measure Temperature Return values 82
Table 23   DEBUG mode SFRs 83
Table 24   Absolute Maximum Ratings 87
Table 25   Operating Range 89
Table 26   Pressure Sensor (1300 kPa variant) , V_{CC} = 3.3 V 90
Table 27   Temperature Sensor, V_{CC} = 3.3 V 91
Table 28   Supply Currents 92
Table 29   12 MHz RC HF Oscillator 93
Table 30   2 kHz RC LP Oscillator 93
Table 31   Interval Timer 94
Table 32   Voltage Regulator 95
Table 33   Power On Reset / Brown Out Reset 96
Table 34   FLASH Memory 97
Table 35   Watchdog Timer 98
Table 36   Digital I/O pins 98
Table 37   I^{2}C Interface 98

SP27

1 Introduction

1.1 Overview

The SP27 is a sensor for pressure measurements designed for PMS applications. With its microcontroller and integrated peripherals, the SP27 offers a single package solution for PMS applications. It requires only few external components. The SP27 features

- Pressure sensor for ambient pressure measurement
- Temperature sensor
- 8051 based microcontroller
- Advanced system controller to minimize power consumption

Measurements of pressure and temperature are performed under software control, and the data can be formatted and prepared to be sent by the microcontroller.

An intelligent wakeup mechanism is available to reduce power consumption. An Interval Timer controls the timing of measurements. The circuitry can be programmed to wakeup at regular intervals. Additionally, wakeup is possible by an external wakeup source connected to a General Purpose Input/Output (GPIO).

The integrated microcontroller is instruction set compatible to the standard 8051 processor.

On-chip FLASH memory is integrated to store:

- The customer specific application program code
- A unique ID-Number
- The calibration data for the sensors

Additional on-chip ROM memory is available that holds the ROM library functions (developed by Infineon) which covers standard tasks used by the application. The available ROM Library functions are described in a separate document (see [1]).

1.2 Features

Main features:

- Operating temperature range -40 to +125 °C
- Low supply current
- Pressure sensor for 1300kPa range
- Temperature sensor
- 8051 instruction set compatible microcontroller (cycle-optimized)
- 6 kByte FLASH memory (for application code)
- 16 kByte ROM (for ROM library functions)
- 256 Bytes RAM
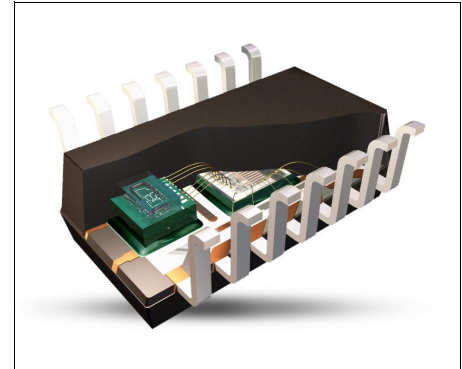
Product Name	Product Type	Ordering Code	Package
SP27	Pressure Monitoring Sensor	SP270-25-256-0	PG-DSOSP-14-6

- Wakeup from POWER DOWN state using the Interval Timer or an external wakeup source connected via a GPIO pin
- I^2C programming/debugging interface
- 16 Bit Hardware CRC generator
- 8 Bit Pseudo Random Number Generator
- Watchdog timer
- 3 bidirectional GPIO pins

2   Functional Description

2.1   General

2.2   Pin Configuration
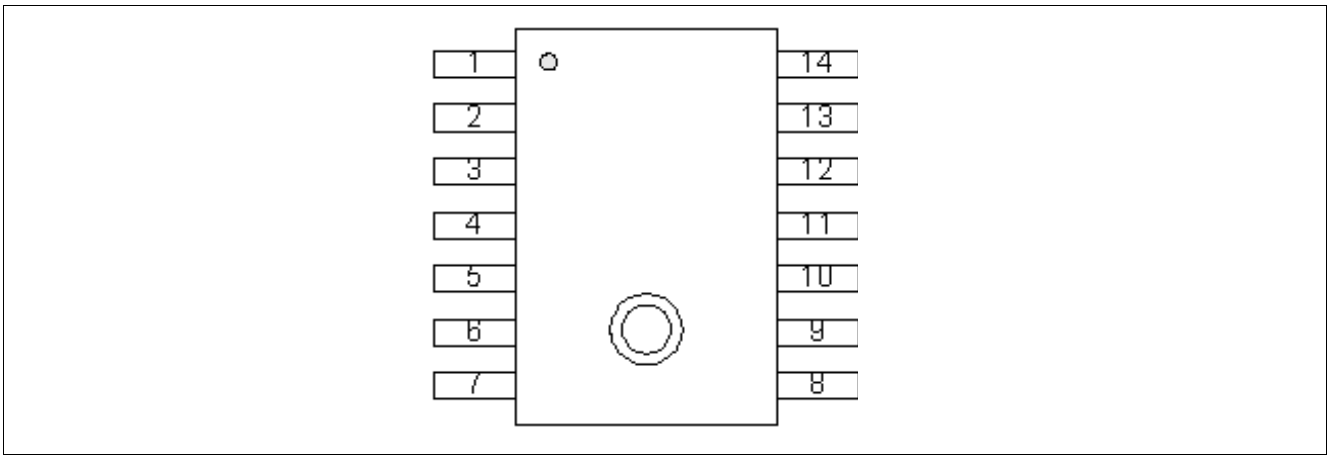
Figure 1   Pin Configuration PG-DSOSP-14-6 (Top View, Figure not to Scale)

2.3 Pin Description

Table 1   **IPin Description**

Pin No.	Name	Pin Type	Buffer Type	Function
1	PP0	Digital I/0		GPIO PP0 / I2C Clock / OpMode0 *Note: Internal pull-up/pull-down switchable*
2	PP1	Digital I/0		GPIO PP1 / I2C Data / OpMode1 *Note: Internal pull-up/pull-down switchable*
3	PP2	Digital I/0		GPIO PP2 *Note: Internal pull-up/pull-down switchable*
6	Vcc	Supply		**Supply Voltage**
9	VREG	Supply		**Internal voltage regulator output** *Note: Connect to decoupling capacitor (CBCAP=100nF)*
4	GND	Supply		**Ground**
5	GND	Supply		**Ground**
7	GND	Supply		**Ground**
8	GND	Supply		**Ground**
10	GND	Supply		**Ground**
11	GND	Supply		**Ground**
12	GND	Supply		**Ground**
13	Reserved	n.c.	n.c.	**Reserved**
14	GND	Supply		**Ground**

2.4 Block Diagram
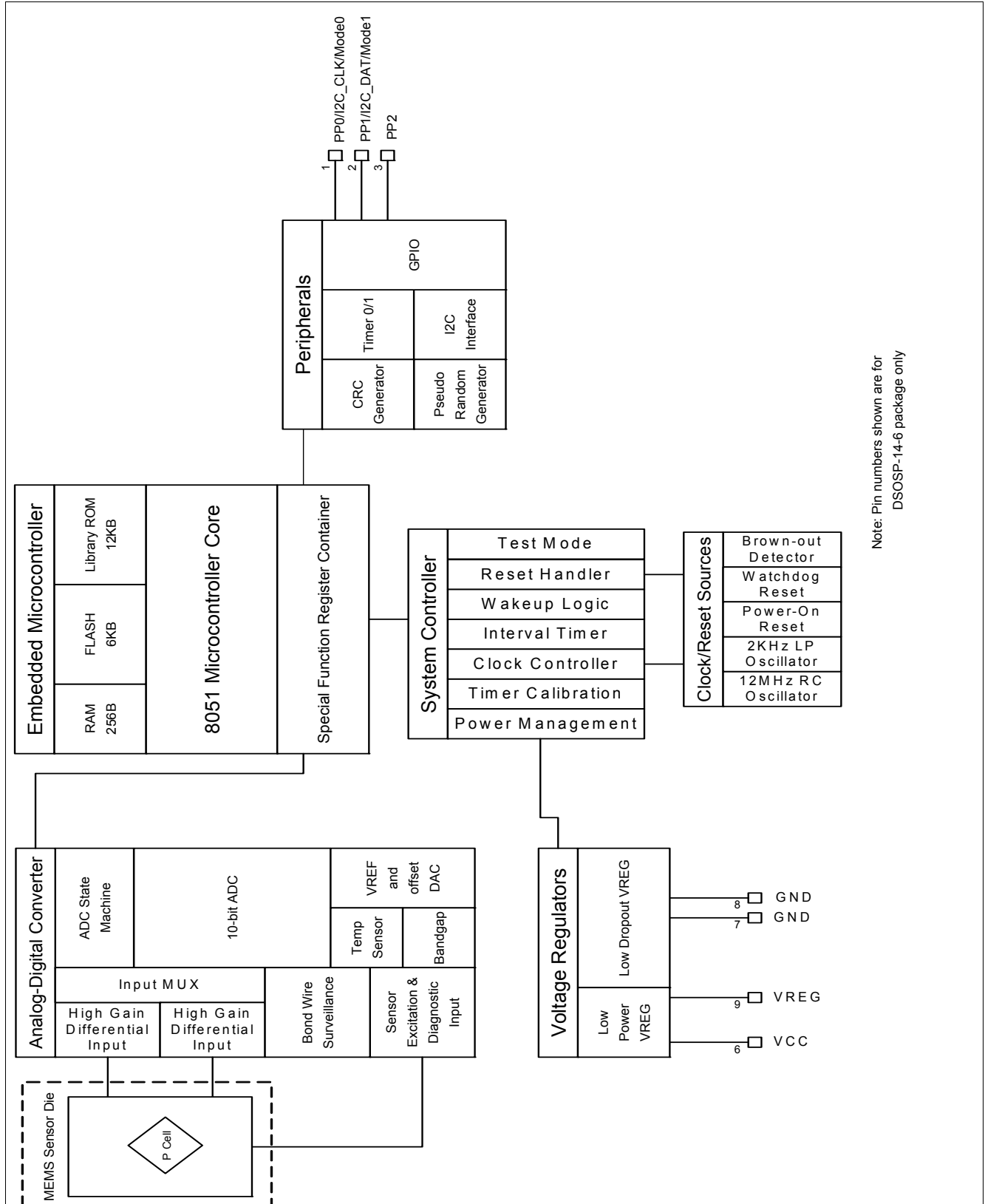
Figure 2   SP27 Block Diagram

2.5 Operating Modes and States

The SP27 can be operated in four different operating modes.

- NORMAL mode
- PROGRAMMING mode
- DEBUG mode
- (internal production TEST mode)

The operating mode selection is done at Power On Reset by setting the GPIO pins PP0 and PP1 according to the following table:

Table 2 SP27 - Operating Modes

PP0	PP1	Operating mode	Device controlled by
1	1	NORMAL mode	FLASH Program at 4000_H
0	1	PROGRAMMING mode	ROM firmware / external I^{2}C Master
1	0	DEBUG mode	ROM firmware / external I^{2}C Master
0	0	Internal production test mode[1]	ROM firmware / external I^{2}C Master

1) IMPORTANT: Do not enter this mode since unpredictable behavior of the device might result

Note: Since PP0 and PP1 have their internal pull-up resistors enabled at Power On Reset, the default startup mode is NORMAL mode if the PP0 and PP1 pins are left unconnected.

2.5.1   Operating Modes

The operating modes depend on the setting of Lockbyte 2, which protects the FLASH Sector 0 (Code Sector) against overwriting, erasing and read-out to prevent reverse engineering of the application code. **Figure 3** shows the mode diagram if the Lockbyte 2 is not set.
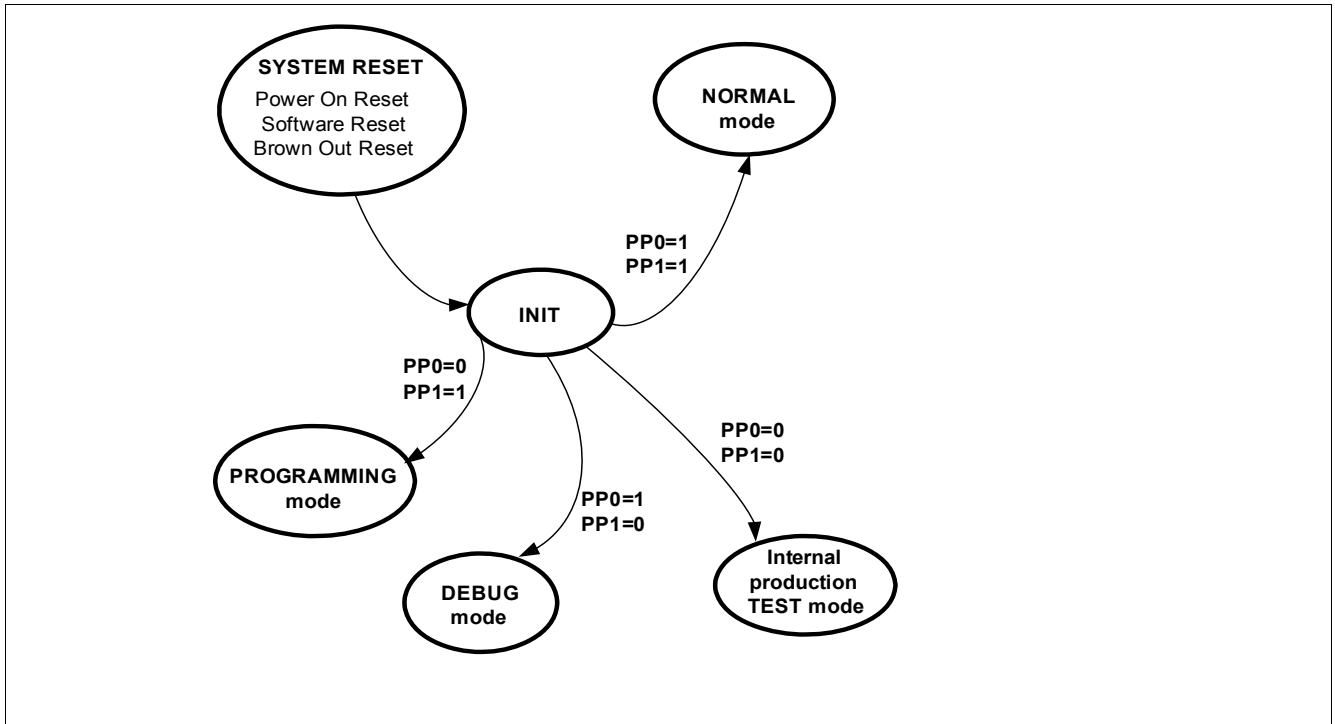
Figure 3   Available Operating Modes if Lockbyte 2 is not Set

For security reasons some operating modes are not accessible anymore after the Lockbyte 2 (see **"FLASH" on Page 22**) is set. **Figure 4** shows the behavior of the SP27 once the Lockbyte 2 is set. All mode selections (except the internal production TEST mode) lead to NORMAL mode.
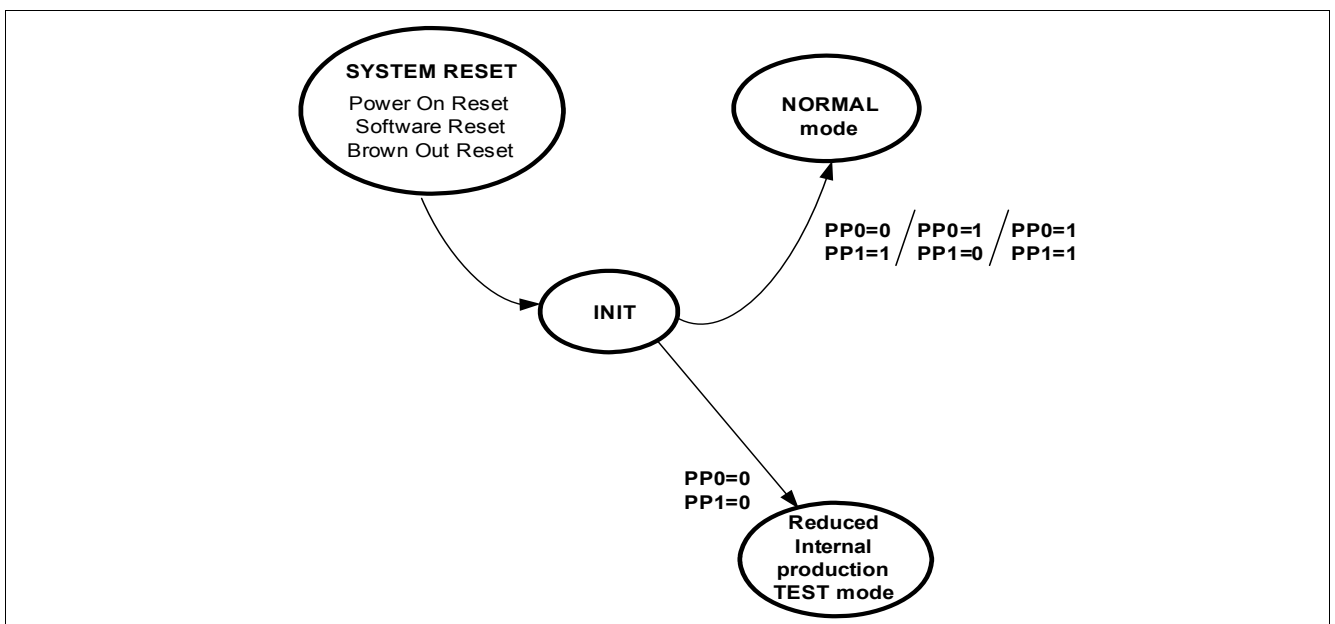
Figure 4   Available Operating Modes if Lockbyte 2 is Set

2.5.2 Resets and Operating Mode Selection
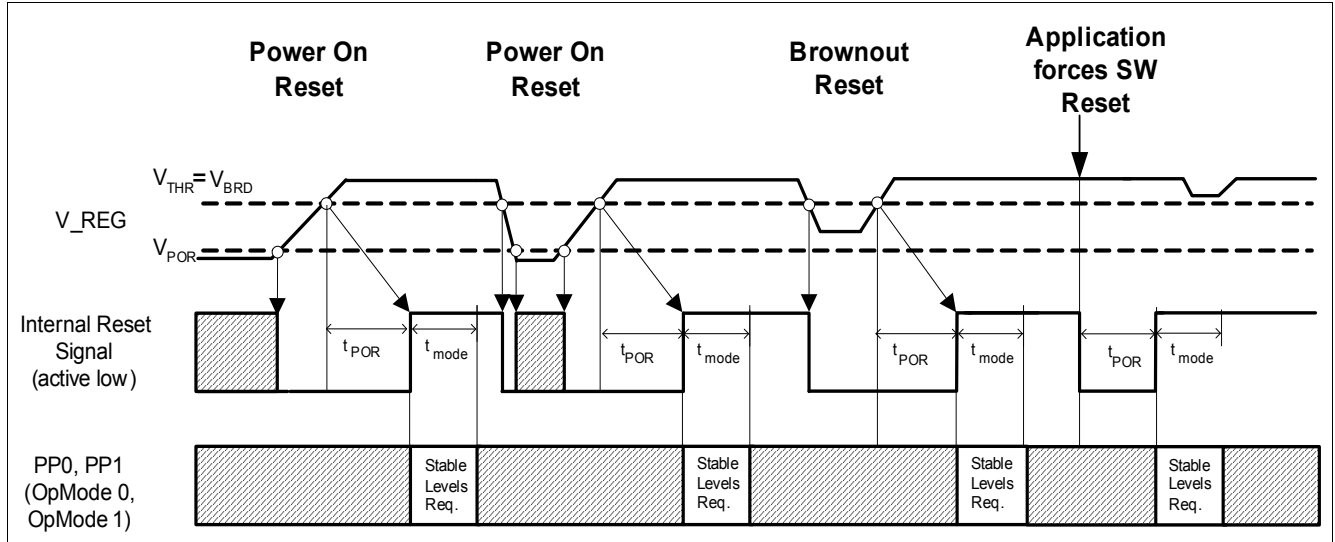
Figure 5   Power On Reset / Brown Out Reset / Operating Mode Selection

Three kinds of resets can occur which cause an operating mode selection:

- The Power On Reset circuit is activated if V_{REG} rises above V_{POR}. The internal blocks are held in RESET state until V_{REG} has risen above V_{THR}.
- The Brown Out Reset circuit is activated if V_{REG} drops below V_{BRD}. The internal blocks are held in RESET state until V_{REG} has risen above V_{BRD} again.
- The SP27's Software Reset can be forced by SP27 setting SRF bit CFG2.0[RESET].

When the Internal Reset state is released (after t_{POR} is elapsed), a further period t_{MODE} is required for reading the states applied to PP0 and PP1 to determine the operation mode of the device according to **Table 2 "SP27 - Operating Modes" on Page 13**. The levels on these pins must be stable during the whole t_{MODE} period. After t_{MODE} has elapsed, the device starts operation in the selected mode.

The Watchdog Reset is a special case and it does not result in a mode selection. The Watchdog Reset affects only the CPU core and forces a program restart.

2.5.3   NORMAL Mode Operation

After a startup in NORMAL mode the system controller handles the different states as shown in the state transition diagram below.
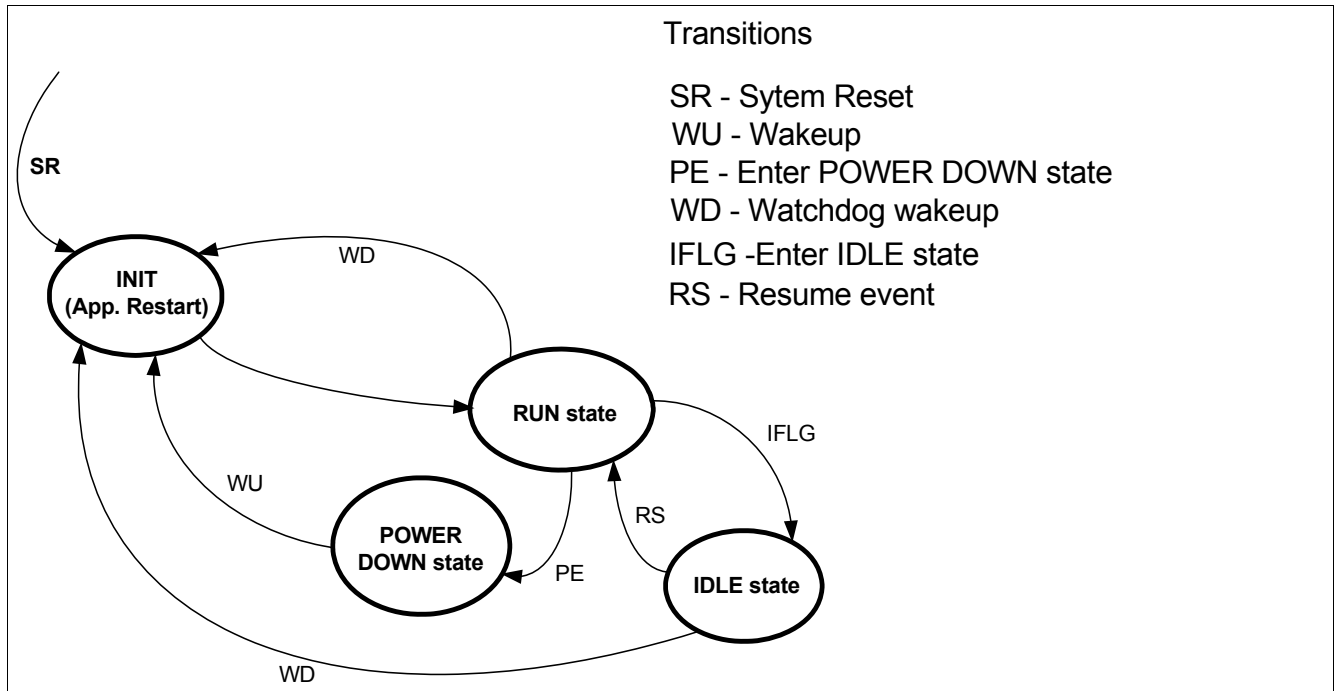
Figure 6   NORMAL Mode State Diagram

2.5.3.1   INIT State

The INIT state is entered after every System Reset (either Power On Reset, Brown Out Reset or Software Reset) to determine the desired operation mode (see **"Operating Modes" on Page 14**) and to initialize the SP27. The INIT state is entered as well if a wakeup or watchdog timeout occurs, but no operation mode selection is done after a wakeup and RUN state is entered immediately.

2.5.3.2   RUN State

In RUN state, the microcontroller is executing the application program from the FLASH. In this state the watchdog is active to prevent software-deadlocks. The microcontroller (CPU) clock source is always based on the 12MHz RC HF Oscillator.

2.5.3.3   IDLE State

In IDLE state the microcontroller clock is stopped to reduce the current consumption, while the peripherals Timer 0, ADC can continue normal operation.

After a resume event is triggered by one of the enabled peripherals, the microcontroller continues the operation where it was interrupted. The resume event source can be identified by reading SFR REF (see **"Resume Event Flag Register" on Page 42**).

2.5.3.4   POWER DOWN State

In POWER DOWN state, the system controller takes control of the SP27 microcontroller and most peripherals are switched off.

In POWER DOWN state the active peripherals (Interval Timer...) are clocked by the 2kHz RC LP Oscillator.

After a wakeup occurred, the wakeup source can be identified by reading SFR WUF (see **"Wakeup Flag Register" on Page 39**).

2.5.3.5　State Transitions

With reference to **Figure 6 "NORMAL Mode State Diagram" on Page 16**, the following state transitions can occur:

Table 3　State Transitions in NORMAL Mode

State transition	Description
RUN state => IDLE state	The application program can set SFR bit CFG0.5 [IDLE] to enter IDLE state. (see **"Configuration Register 0" on Page 34**) Note: If no peripheral that can create a RESUME event is active, IDLE state will not be entered and the application will continue un-interrupted.
IDLE state => RUN state	A peripheral unit (Timer 0, ADC) creates a resume event. The application automatically resumes where it was interrupted when entering IDLE state (see **"Resume Event Flag Register" on Page 42**)
IDLE state => INIT state RUN state => INIT state	Overflow of the watchdog timer. The application will restart. The watchdog wakeup is indicated in the SFR WUF. **"Wakeup Flag Register" on Page 39**
RUN state => POWER DOWN state	The application program can call a ROM Library function to enter POWER DOWN state.
INIT state => RUN state	This state change is initiated automatically by the system controller as soon as the initialization is finished.

2.5.3.6　Status of SP27 Blocks in Different States

Depending on the current state in NORMAL mode the internal blocks of the SP27 are active, inactive or are not supplied with supply voltage to minimize the current consumption. The following table gives an overview over of the individual blocks in the different states.

Table 4　Status of SP27 Blocks in Different States

Unit	Run state	IDLE state	POWER DOWN state
Power On Reset	Active[1]	Active	Active
Brown Out Detector	Active	Active	Inactive
Voltage Regulator (V_{REG})	Active	Active	Active
System controller	Active	Active	Active
Microcontroller	Active	Inactive	No supply[2]
Timer	Active	Active	No supply
Peripheral modules CRC, I2C, Pseudo Random Number Generator	Selectable inactive or active	Inactive	No supply
Watchdog timer	Active	Active	No supply
Upper 128Bytes RAM	Active	Inactive	No supply

Table 4   Status of SP27 Blocks in Different States (cont'd)

Unit	Run state	IDLE state	POWER DOWN state
Lower 128Bytes RAM	Active	Inactive	Selectable No supply or inactive
FLASH	Active	Inactive	No supply
ROM	Active	Inactive	No supply
2kHz RC LP Oscillator	Active	Active	Active
12MHz RC HF Oscillator	Active	Active	No supply
Interval Timer	Active	Active	Active
Sensor	Handled by ROM Library functions	Handled by ROM Library functions	No supply

1) Active: block is powered, active and keeps its register contents.

2) No supply: block is not powered, cannot be used and all register content is lost.

2.6   Fault Protection

The SP27 features multiple fault protections which prevent the application from unexpected behavior and deadlocks. This chapter gives a brief overview of the available fault protections. Detailed explanation of the usage can be found later in this document and in **[1]**.

2.6.1   Watchdog Timer

For operation security a watchdog timer is available to avoid application software deadlocks. The watchdog timer is only active in NORMAL mode and DEBUG mode and must be reset periodically by the application, otherwise the timer generates a wakeup and forces a restart of SP27 application program. Setting SFR bit CFG2.1[WDRES] resets the watchdog timer (see **"Configuration Register 2" on Page 36**)

The watchdog timeout period is fixed (see **Table 35 "Watchdog Timer" on Page 98**). The accuracy depends on the accuracy of the 2kHz RC LP Oscillator which is used to clock the watchdog timer.

Upon wakeup the watchdog timer is automatically reset. The watchdog timer is not, however, automatically reset upon entry into IDLE state. Therefore care must be taken so that the application does not remain in IDLE state longer than the minimum watchdog timeout period.

2.6.2   ADC Measurement Overflow & Underflow

The ROM Library functions which perform measurements will return the over/underflow status in a status byte with the measurement result. (see **[1]**)

2.6.3   ADC Selftest

A dedicated ROM Library function is able to perform a selftest of the ADC (see **[1]**).

2.6.4   Bond Wire Surveillance

The continuity of the bond wire connection between the ASIC die and the sensor die is checked as part of every pressure measurement. The ROM library routines which perform the measurements will return the bond wire status in a status byte with the measurement result (see **[1]**).

2.7 Functional Block Description

2.7.1 Sensors and Data Acquisition

The SP27 has two sensors to acquire environmental data:

2.7.1.1 Pressure Sensor

The pressure sensor consists of a single-crystal silicon, bulk micro machined membrane with an integrated full Wheatstone piezo-resistive bridge. The piezo-resistors are placed inside a vacuum reference chamber, whilst the pressure media to be measured in the application is applied to the opposite side of the membrane. This gives good long-term properties as the measurement bridge is protected from the environment. Pressure measurement is performed by a dedicated ROM library function (see [1]).

2.7.1.2 Temperature Sensor

The temperature sensor is placed on the ASIC. This is read by the ADC referenced to a fixed (band gap) voltage. Temperature measurement is performed by a dedicated ROM library function (see [1]).

2.7.1.3 Data Acquisition

The analog data is acquired and digitized by the internal 10 Bit ADC.

Measurement routines for acquiring the environmental data are available within the ROM library functions that are described in [1].

Characteristic of the individual sensors can be found in **"Characteristics" on Page 90**.

2.7.2 Memory Organization
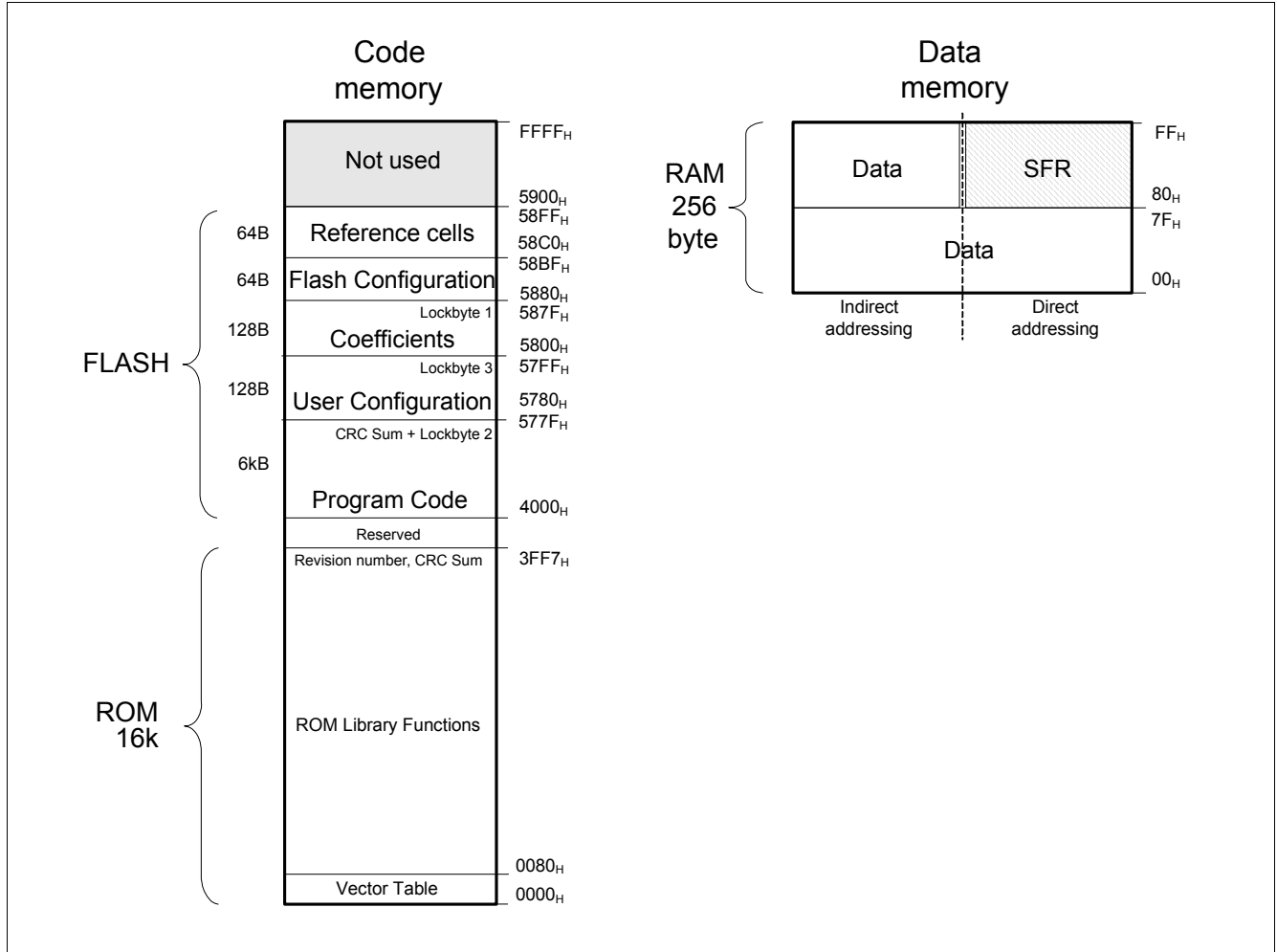
Figure 7   Memory Map

The following memory blocks are implemented:

- 16 kByte ROM memory
- 6 kByte FLASH memory
- 256 Byte RAM memory
- 128 Byte SFR register

2.7.2.1   ROM

A 16 kB ROM memory is located in the address range 0000_H to $3FF7_H$.

2.7.2.1.1   ROM Library Functions and Reset/Wakeup Handlers

The ROM contains the reset handler, operation mode handler, wakeup handler, internal test and debug routines and the ROM Library functions (see in [1]).

2.7.2.1.2   ROM Protections

To protect the ROM code against readout a hardware mechanism is implemented, thus no read operations can be performed on the ROM.

Direct jumping into the ROM area is prevented by a hardware mechanism, thus access to the ROM library functions is granted only via a vector table at the bottom of the ROM address space.

2.7.2.2 FLASH

2.7.2.2.1 FLASH Organization

The FLASH is divided into five sectors. Each sector can be erased and written individually.

Sectors 0 and 1 are accessible for customer usage.

Sectors 2, 3 and 4 are written in the Infineon production site and cannot be erased or re-written by the customer.

- 4000_H -- $577F_H$ **(6016 Bytes) Code sector (0):** The code sector contains the application software including a CRC16 Checksum (to be written to $577D_H$ -- $577E_H$) and the Lockbyte 2 (to be written to $577F_H$).
- 5780_H -- $57FF_H$ **(128 Bytes) User Configuration sector (1):** The User Configuration sector can store individual device configuration data. It also contains the Lockbyte 3 (to be written to $57FF_H$).
- 5800_H -- $587F_H$ **(128 Bytes) Coefficients sector (2):** This sector is written during the sensor calibration process and contains calibration coefficients, the unique Sensor ID and Lockbyte 1 (to be written to $587F_H$).
- 5880_H -- $58BF_H$ **(64 Bytes) FLASH Configuration sector (3):** This sector contains the FLASH driver parameters and other device configuration parameters.
- $58C0_H$ -- $58FF_H$ **(64 Bytes) Reference Cell sector (4):** This sector contains the reference cells for FLASH reading.

2.7.2.2.2 FLASH Protection

To protect the FLASH against unauthorized access three FLASH Lockbytes are available.

Note: The Lockbytes are set, if the value in the appropriate FLASH address is programmed to $D1_H$. Setting the Lockbyte to 00_H will result in a unlocked FLASH area. Any other value must not be written to these locations. After programming a Lockbyte, the SP27 has to be reset before the FLASH lock takes effect.

- **Lockbyte 1** ($587F_H$)

This Lockbyte protects the FLASH sectors 2, 3 and 4 against overwriting and erasing. This Lockbyte is programmed at the Infineon production site.

- **Lockbyte 2** ($577F_H$)

This Lockbyte protects the FLASH sector 0 (Code Sector) against overwriting, erasing (except reduced internal production test mode) and read-out to prevent reverse engineering of the application code.

This Lockbyte has to be set at the end of the programming sequence of the Code Sector via the I²C Interface (when writing the highest FLASH Line starting at 5760_H). Once it is set, the available operating modes are reduced according to **Figure 4 "Available Operating Modes if Lockbyte 2 is Set" on Page 14**.

- **Lockbyte 3** ($57FF_H$)

This Lockbyte protects the FLASH sector 1 (User Configuration Sector) against overwriting and erasing (except reduced internal production test mode).

The Lockbyte can be set either via I²C in PROGRAMMING mode in the same programming sequence as Lockbyte 2 is set, or by using a dedicated ROM Library function in NORMAL mode by the application software (see **[1]**).

2.7.2.3   RAM

The RAM is available as volatile data storage for the application program. Some RAM locations are required by the ROM Library Functions and therefore not freely available for use by the application program. For more details please refer to the ROM Library function guide **[1]**.

The upper 128 bytes of RAM are switched off in POWER DOWN state and lose their contents.

The lower 128 bytes of RAM can be powered during POWER DOWN state. This is selectable using SFR bit CFG2.4[PDLMB].

If not powered in these states, this RAM loses the content, otherwise it can be used as battery buffered storage like the General Purpose Registers (see **"General Purpose Registers" on Page 32**).

Note: The RAM is not reset at a System Reset or watchdog timeout.

   After a Brown Out Reset this feature may be used to possibly recover data.

   After Power On Reset the application has to initialize the RAM if needed.

3 Special Function Registers

Special Function Registers (SFR) are used to control and monitor the state of the SP27 and its peripherals.

The following table shows the naming convention for the SFR descriptions that are used throughout this document

Table 5   Register Naming Convention Wakeup / Reset Value

State	Symbol	Description
Low	0	Register value is 0_B
High	1	Register value is 1_B
Undefined	X	Register value is undefined
Unchanged	U	Register value is unchanged

Table 6 provides links within this document to detailed description of the application relevant SFRs. In addition to register names and offset addresses, this table indicates how each SFR behaves after wakeup and reset events. The Wakeup Value column applies in the case of a wakeup event, which includes a watchdog timeout. The Reset Value column applies in the case of a Power On Reset, Brownout Reset or Software Reset event.

Table 6   Registers Overview

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
Microcontroller				
ACC	Accumulator	$E0_H$	00_H	00_H
B	Register B	$F0_H$	00_H	00_H
DPH	Data Pointer (high)	83_H	00_H	00_H
DPL	Data Pointer (low)	82_H	00_H	00_H
PSW	Program Status Word	$D0_H$	00_H	00_H
SP	Stack Pointer	81_H	07_H	07_H
General Purpose Registers[1]				
GPR0	General Purpose Register 0	$B8_H$	UU_H	XX_H
GPR1	General Purpose Register 1	$B0_H$	UU_H	XX_H
GPR2	General Purpose Register 2	$A8_H$	UU_H	XX_H
GPR3	General Purpose Register 3	$F1_H$	UU_H	XX_H
GPR4	General Purpose Register 4	$F2_H$	UU_H	XX_H
GPR5	General Purpose Register 5	$F3_H$	UU_H	XX_H
GPR6	General Purpose Register 6	$F5_H$	UU_H	XX_H
GPR7	General Purpose Register 7	$F6_H$	UU_H	XX_H
GPR8	General Purpose Register 8	$F7_H$	UU_H	XX_H
GPR9	General Purpose Register 9	$F9_H$	UU_H	XX_H
GPRA	General Purpose Register 10	FA_H	UU_H	XX_H
GPRB	General Purpose Register 11	FB_H	UU_H	XX_H

Table 6   Registers Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
GPRC	General Purpose Register 12	FC_H	UU_H	XX_H
GPRD	General Purpose Register 13	FD_H	UU_H	XX_H
GPRE	General Purpose Register 14	FE_H	UU_H	XX_H
GPRF	General Purpose Register 15	FF_H	UU_H	XX_H
System Configuration Registers				
CFG0	Configuration Register 0	$F8_H$	$0000U000_B$	00_H
CFG1	Configuration Register 1	$E8_H$	$000U000U_B$	00_H
CFG2	Configuration Register 2	$D8_H$	$000U1000_B$	18_H
DSR	Diagnosis and Status Register	$D9_H$	$0XUU00XU_B$	$0XXX0000_B$
System Controller				
WUF	Wakeup Flag Register	$C0_H$	XX_H	00_H
WUM	Wakeup Mask Register	$C1_H$	UU_H	FF_H
REF	Resume Event Flag Register	$D1_H$	00_H	00_H
Interval Timer				
ITPR	Interval Timer Period Register	BC_H	UU_H	01_H
Interval Timer Precounter / Calibration				
ITPH	Interval Timer Precounter Register (High Byte)	BB_H	$0000UUUU_B$	03_H
ITPL	Interval Timer Precounter Register (Low Byte)	BA_H	UU_H	$E8_H$
Clock Controller				
DIVIC	Internal Clock Divider	$B9_H$	$000000UU_B$	00_H
16 Bit CRC (Cyclic Redundancy Check) Generator/Checker				
CRCC	CRC Control Register	$A9_H$	02_H	02_H
CRCD	CRC Data Register	AA_H	00_H	00_H
CRC0	CRC Preload/Result Register 0 (low byte)	AC_H	00_H	00_H
CRC1	CRC Preload/Result Register 1 (high byte)	AD_H	00_H	00_H
Pseudo Random Number Generator				
RNGD	Random Number Generator Data Register	AB_H	UU_H	55_H
Timer Unit				
TCON	Timer Control Register	88_H	00_H	00_H
TMOD	Timer Mode Register	89_H	00_H	00_H
TH0	Timer 0 Register High Byte	$8C_H$	00_H	00_H
TL0	Timer 0 Register Low Byte	$8A_H$	00_H	00_H
TH1	Timer 1 Register High Byte	$8D_H$	00_H	00_H
TL1	Timer 1 Register Low Byte	$8B_H$	00_H	00_H
General Purpose Input/Output (GPIO)				
P1DIR	IO-Port 1 Direction Register	91_H	UU_H	FF_H

Table 6   Registers Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
P1IN	IO-Port 1 Data In Register	92_H	$00000XXX_B$	$00000XXX_B$
P1OUT	IO-Port 1 Data Out Register	90_H	UU_H	FF_H
P1SENS	IO-Port 1 Sensitivity Register	93_H	$00000UUU_B$	00_H
I²C Interface				
I2CD	I2C Data Register	$9A_H$	00_H	00_H
I2CS	I2C Status Register	$9B_H$	00_H	00_H
Debug Special Function Register				
DBCH0	Debug Compare Register 0 (high byte)	95_H	00_H	00_H
DBCL0	Debug Compare Register 0 (low byte)	94_H	00_H	00_H
DBCH1	Debug Compare Register 1 (high byte)	$9D_H$	00_H	00_H
DBCL1	Debug Compare Register 1 (low byte)	$9C_H$	00_H	00_H
DBTH0	Debug Target Register 0 (high byte)	97_H	00_H	00_H
DBTL0	Debug Target Register 0 (low byte)	96_H	00_H	00_H
DBTH1	Debug Target Register 1 (high byte)	$9F_H$	00_H	00_H
DBTL1	Debug Target Register 1 (low byte)	$9E_H$	00_H	00_H

1) Reset Value for GRP0 - GPRF is typically undefined (X) except in the case of Software Reset, which leaves the GPR content unchanged (U).

The register is addressed bytewise.

Table 7   Registers Access Types

Mode	Symbol	Description Hardware (HW)	Description Software (SW)
Basic Access Types			
write/read	wr	Register is used as input for the HW	Register is read and writable by SW
read only	r	Register is written by HW	Value written by SW is ignored by HW; that is, SW may write any value to this field without affecting HW behavior
write only	w	Register is written by software and affects hardware behavior with every write by software.	Register is writable by SW. When read, the register does not return the value that has been written previously, but the reset value instead.
UNUSED	-	Register is not used by HW.	Value written by SW is ignored by HW; that is, SW may write any value to this field without affecting HW behavior.
Reserved		Register is used as input for internal HW; Access Type is not documented.	Value must be kept by SW. SW read or write any value to this field affecting HW behavior.
Special Access Types[1)]			

Table 7   Registers Access Types (cont'd)

Mode	Symbol	Description Hardware (HW)	Description Software (SW)
Read self clearing	rc	Register is used as input for the HW, the register will be cleared due to a HW mechanism.	Reading from the register generates a strobe signal for the HW. Register is readable by SW.
Write self clearing	wc	Register is used as input for the HW, the register will be cleared due to a HW mechanism.	Writing to the register generates a strobe signal for the HW. Register is writable by SW.

1) Optional types

3.1   Microcontroller

The SP27 incorporates an 8051 instruction set compatible microcontroller. It offers an 8-bit data path, several addressing modes (direct, register, register indirect, immediate, index), and accesses the built-in peripherals through SFRs. To handle the sequential nature of PMS applications efficiently, wakeup and resume mechanisms are implemented instead of an interrupt controller.

The microcontroller incorporates the following basic SFRs: Accumulator (ACC), Register B (B) and Program Status Word (PSW) are bit addressable registers used to perform arithmetical and logical operations. The Stack Pointer (SP) and Data Pointer (DPTR) are included to allow basic programming structures. The Data Pointer (DPTR) is determined by SFR DPH and SFR DPL.

SFR PSW holds the status of basic arithmetic operations.

Accumulator

ACC	**Offset**	**Wakeup Value**	**Reset Value**
Accumulator	**E0**$_H$	**00**$_H$	**00**$_H$

7							0
			ACC				

wr
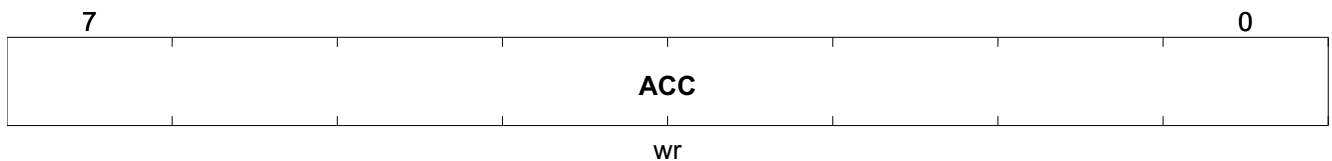
Field	Bits	Type	Description
ACC	7:0	wr	**Accumulator** Reset: 00$_H$

Register B

B	**Offset**	**Wakeup Value**	**Reset Value**
Register B	**F0**$_H$	**00**$_H$	**00**$_H$

7							0
			B				

wr
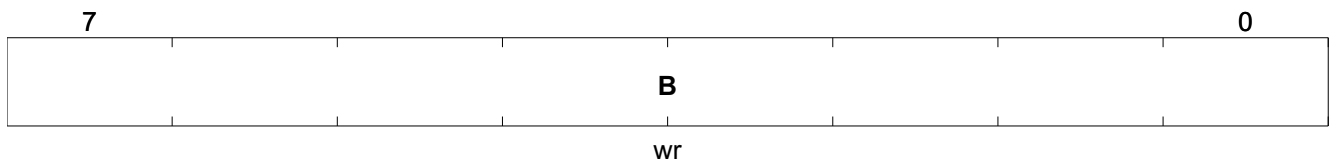
Field	Bits	Type	Description
B	7:0	wr	**Register B 7-0** Reset: 00$_H$

Data Pointer (high)

DPH				**Offset**	**Wakeup Value**	**Reset Value**
Data Pointer (high)				**83**$_H$	**00**$_H$	**00**$_H$

7   0

DPH

wr
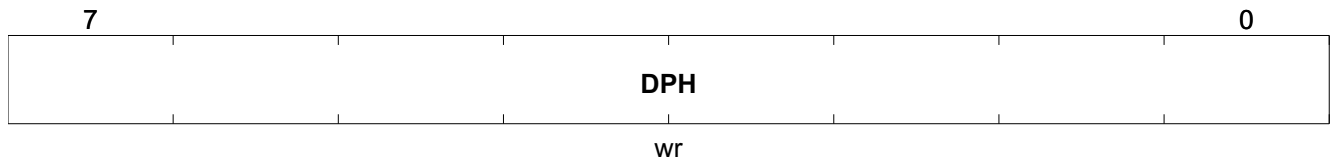
Field	Bits	Type	Description
DPH	7:0	wr	**Data Pointer (high)** Reset: 00$_H$

Data Pointer (low)

DPL				**Offset**	**Wakeup Value**	**Reset Value**
Data Pointer (low)				**82**$_H$	**00**$_H$	**00**$_H$

7   0

DPL

wr
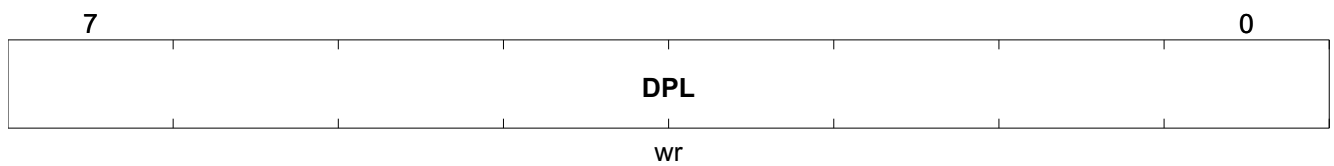
Field	Bits	Type	Description
DPL	7:0	wr	**Data Pointer (low)** Reset: 00$_H$

CONFIDENTIAL

Program Status Word

PSW	Offset	Wakeup Value	Reset Value
Program Status Word	**D0$_H$**	**00$_H$**	**00$_H$**

7	6	5	4	3	2	1	0
CY	**AC**	**F0**	**RS**		**OV**	**F1**	**P**
wr	wr	wr	wr		wr	wr	r

Field	Bits	Type	Description
CY	7	wr	**Carry** Reset: 0$_H$
AC	6	wr	**Auxiliary Carry** Carry-out for BCD operations Reset: 0$_H$
F0	5	wr	**Flag 0** Available for general purpose use. Reset: 0$_H$
RS	4:3	wr	**Register Bank Select** 00$_B$ Bank 0 (00$_H$ - 07$_H$) 01$_B$ Bank 1(08$_H$ - 0F$_H$) 10$_B$ Bank 2(10$_H$ - 17$_H$) 11$_B$ Bank 3(18$_H$ - 1F$_H$) Reset: 0$_H$
OV	2	wr	**Overflow** Reset: 0$_H$
F1	1	wr	**Flag 1** Available for general purpose use. Reset: 0$_H$
P	0	r	**Parity** Set or cleared each instruction cycle to indicate an odd or even number of 1 bits in the accumulator Reset: 0$_H$

Stack Pointer

SP		Offset	Wakeup Value	Reset Value
Stack Pointer		81$_H$	07$_H$	07$_H$

7							0
			SP				

wr
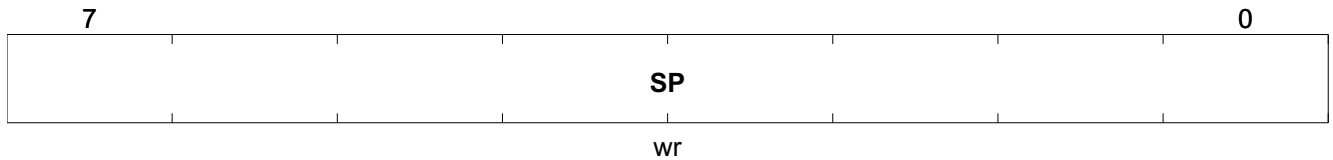
Field	Bits	Type	Description
SP	7:0	wr	**Stack Pointer (SP)** SP is incremented before data is pushed and decremented after data is popped. SP always points to the last valid stack byte. Reset: 07$_H$

3.2   General Purpose Registers

The SP27 incorporates 16 general purpose registers that can be used by the application to store data beyond a POWER DOWN state period. The GPR Registers are not cleared after a System Reset. After a Power On Reset, the GPR contents will be undefined.

General Purpose Register 0

GPR0	Offset	Wakeup Value	Reset Value
General Purpose Register 0	B8$_H$	UU$_H$	XX$_H$

7							0
			GPR0				

wr

Field	Bits	Type	Description
GPR0	7:0	wr	**General Purpose** Reset: XX$_H$

General Purpose Registers 0 - F are freely available for application program use. GPR0, GPR1, GPR2 are located at Offset Addresses that make them well suited for bit manipulation. Placing bit variables in these GPRs will result in more efficient CPU operation.

Table 8   Register 0 to F[1]

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
GPR0	General Purpose Register 0	B8$_H$	UU$_H$	XX$_H$
GPR1	General Purpose Register 1	B0$_H$	UU$_H$	XX$_H$
GPR2	General Purpose Register 2	A8$_H$	UU$_H$	XX$_H$
GPR3	General Purpose Register 3	F1$_H$	UU$_H$	XX$_H$
GPR4	General Purpose Register 4	F2$_H$	UU$_H$	XX$_H$
GPR5	General Purpose Register 5	F3$_H$	UU$_H$	XX$_H$
GPR6	General Purpose Register 6	F5$_H$	UU$_H$	XX$_H$
GPR7	General Purpose Register 7	F6$_H$	UU$_H$	XX$_H$
GPR8	General Purpose Register 8	F7$_H$	UU$_H$	XX$_H$
GPR9	General Purpose Register 9	F9$_H$	UU$_H$	XX$_H$
GPRA	General Purpose Register 10	FA$_H$	UU$_H$	XX$_H$
GPRB	General Purpose Register 11	FB$_H$	UU$_H$	XX$_H$
GPRC	General Purpose Register 12	FC$_H$	UU$_H$	XX$_H$
GPRD	General Purpose Register 13	FD$_H$	UU$_H$	XX$_H$

Table 8   Register 0 to F[1) (cont'd)

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
GPRE	General Purpose Register 14	FE$_H$	UU$_H$	XX$_H$
GPRF	General Purpose Register 15	FF$_H$	UU$_H$	XX$_H$

1) Reset Value for GRP0 - GPRF is typically undefined (X) except in the case of Software Reset, which leaves the GPR content unchanged (U).

3.3 System Configuration Registers

The system configuration registers can be used for:

- Initiating state transitions (SFR CFG0)
- Enabling or disabling peripherals (SFR CFG1 and SFR CFG2)
- Monitoring the operation mode, the system state and peripherals (SFR DSR)

Configuration Register 0

CFG0		Offset	Wakeup Value	Reset Value
Configuration Register 0		**F8$_H$**	**0000U000$_B$**	**00$_H$**

Field	Bits	Type	Description
PDWN	7	wr	**Enter POWER DOWN State** If set to 1 by software the POWER DOWN state is entered; This bit is automatically reset to 0 by the system controller after a wakeup. Entering POWER DOWN state is handled by a ROM Library function. It is not recommended to set this bit manually. Reset: 0$_H$
Res	6	-	**Reserved** This bit must be set to 0$_B$. Reset: 0$_H$
IDLE	5	wr	**Enter IDLE State** If set to 1 by software the IDLE state is entered; This bit is automatically reset to 0 by the system controller after a resume event occurs. Reset: 0$_H$
Res	4	-	**Reserved** This bit must be set to 0$_B$ Reset: 0$_H$
FTM	3	wr	**Enable Functional Test Mode** This mode is used only during internal device testing. 0$_B$ FTM disabled 1$_B$ FTM enable Reset: 0$_H$
Res	2		**Reserved** This bit must be set to 0$_B$. Reset: 0$_H$
UNUSED	1	-	**UNUSED** Reset: 0$_H$

Field	Bits	Type	Description
Res	0		**Reserved** This bit must be set to 0_B. Reset: 0_H

Configuration Register 1

CFG1				Offset		Wakeup Value		Reset Value
Configuration Register 1				$E8_H$		$000U000U_B$		00_H

7	6	5	4	3	2	1	0
Res	I2CEN	RNGEN	Res	Res	ITRD	ITINIT	ITEN
	wr	wc			wr	r	wr

Field	Bits	Type	Description
Res	7		**Reserved** This bit must be set to 0_B. Reset: 0_H
I2CEN	6	wr	**I2C Enable** 0_B Standard I/O Port functionality 1_B I2C functionality on Pins PP0/SCL and PP1/SDA Reset: 0_H
RNGEN	5	wc	**Random Number Generator Enable** 0_B Cleared automatically after random number is generated 1_B Initiates generation of a new pseudo random number Reset: 0_H
Res	4	-	**Reserved** This bit must be set to 0_B Reset: 0_H
Res	3		**Reserved** This bit must be set to 0_B. Reset: 0_H
ITRD	2	wr	**Interval Timer Read Enable** To safely read SFR ITPR, this bit should be set to 1 prior to reading ITPR. After the ITPR contents are read, this bit should be checked. 0_B SFR ITPR read result is not valid 1_B SFR ITPR read result is valid Reset: 0_H

Field	Bits	Type	Description
ITINIT	1	r	**Interval Timer Initialization** When the wakeup interval is changed by programming ITPR, ITPL or ITPH with a new value, this bit is set to 1 until the new value has taken effect. The application should not leave RUN state while this bit is 1 otherwise the ITPR setting does not take effect. This bit is automatically cleared after initialization is complete. 0_B Interval Timer Initialization complete 1_B Interval Timer Initialization in progress Reset: 0_H
ITEN	0	wr	**Interval Timer Enable** Interval Timer is always enabled in NORMAL mode, setting or clearing ITEN bit has no effect. 0_B Disable Interval Timer (TEST/DEBUG mode only) 1_B Enable Interval Timer Reset: 0_H

Configuration Register 2

CFG2				Offset	Wakeup Value		Reset Value
Configuration Register 2				$D8_H$	$000U1000_B$		18_H

7	6	5	4	3	2	1	0
UNUSED	**I2CGCEN**	**UNUSED**	**PDLMB**	Res	**UNUSED**	**WDRES**	**RESET**
-	wr	-	wr		-	wc	wc

Field	Bits	Type	Description
UNUSED	7	-	**UNUSED** Reset: 0_H
I2CGCEN	6	wr	**I2C General Call Enable** 0_B Disabled 1_B Enabled Reset: 0_H
UNUSED	5	-	**UNUSED** Reset: 0_H
PDLMB	4	wr	**Lower RAM Memory Block (00_H-$7F_H$) Power Control** 0_B Lower RAM retains power in POWER DOWN state 1_B Lower RAM is not powered in POWER DOWN state Reset: 1_H
Res	3		**Reserved** This bit must be set to 1_B. Reset: 1_H
UNUSED	2	-	**UNUSED** Reset: 0_H

Field	Bits	Type	Description
WDRES	1	wc	**Reset Watchdog Counter** 0_B Cleared automatically after reset of watchdog counter 1_B Watchdog counter is reset Reset: 0_H
RESET	0	wc	**System Reset** 0_B Cleared automatically after software reset is performed 1_B A software reset is performed Reset: 0_H

Diagnosis and Status Register

DSR		Offset	Wakeup Value	Reset Value
Diagnosis and Status Register		**D9$_H$**	**0XUU00XU$_B$**	**0XXX0000$_B$**

7	6	5	4	3	2	1	0
Res	Res	OPMODE		Res		WUPEND	Res
		r				r	

Field	Bits	Type	Description
Res	7		**Reserved** This bit must be set to 0$_B$. Reset: 0$_H$
Res	6		**Reserved**
OPMODE	5:4	r	**Operating Mode** 00$_B$ TEST Mode 01$_B$ DEBUG Mode 10$_B$ PROGRAMMING Mode 11$_B$ NORMAL Mode Reset: XX$_B$
Res	3:2		**Reserved** These bits must be set to 00$_B$. Reset: 0$_H$
WUPEND	1	r	**Wakeup Pending in SFR WUF** 0$_B$ SFR WUF contents have not changed since the last read 1$_B$ SFR WUF contents have changed since the last read Reset: 0$_H$
Res	0		**Reserved** This bit must be set to 0$_B$. Reset: 0$_H$

3.4 System Controller

While the microcontroller controls the SP27 in RUN state, the system controller takes over control in POWER DOWN state and IDLE state.

The system controller handles wakeup / resume events and system resets. It is clocked by the 2kHz RC LP Oscillator.

Difference between System Reset and Wakeup:

- **System Reset** - The digital circuit is reset. Program execution starts at address 0000_H to perform reset initialization routines (including operation mode selection) and will jump to the FLASH at address 4000_H to execute the application program.
- **Wakeup** - Only the program counter of the microcontroller and its peripheral units are reset. Program Execution starts at address 0000_H to perform wakeup initialization routines and jumps to the FLASH at 4000_H to execute the application program.

Wakeup Event Handling

Whenever a wakeup occurs, the SP27 leaves POWER DOWN state and enters RUN state to execute the application program. This transition can be initiated from various sources. The wakeup source can be identified by reading SFR WUF.

The wakeup sources can be enabled or disabled by setting the appropriate bits in SFR WUM. The bits WDOG and ITIM are always enabled. Setting these bits have no effect in NORMAL/DEBUG modes.

The bits in SFR WUF are cleared upon read (read-clear). If subsequent wakeup source activity is detected, the SFR DSR.WUPEND bit and the corresponding SFR WUF bit(s) will be set. Note that wakeup sources that have a mask bit set (disabled) in SFR WUM will always be cleared (inactive) in SFR WUF.

The SFR WUF is not automatically cleared when entering the POWER DOWN state. For this reason it is a good practice to have the application software read WUF and handle any pending wakeup events before entering POWER DOWN. Otherwise, any flags in SFR WUF will remain marked as pending upon the next device wakeup. Note that only new wakeup source activity, occurring after entering POWER DOWN, will cause a device wakeup to occur.

Watchdog Wakeup

A watchdog wakeup occurs after the watchdog timer has elapsed.
See **"Watchdog Timer" on Page 18** for details about the watchdog timer.

External PP2 Wakeup Event

I/O Port PP2 can be configured to wakeup the SP27 from POWER DOWN state by an external source.

PP2 has to be configured according to **"External Wakeup on PP2" on Page 66** for this feature.

Interval Timer Wakeup Event

When the Interval Timer elapses, a wakeup event is generated and POWER DOWN state is left. The wakeup can be identified by the application software reading SFR bit WUF.0[ITIM].
The Interval Timer is reloaded automatically with the actual value from SFR ITPR and immediately restarted. The Interval Timer is also counting during RUN state which leads to accurate wakeup intervals even if RUN state periods vary in execution time.

Wakeup Flag Register

WUF		Offset	Wakeup Value	Reset Value
Wakeup Flag Register		C0$_H$	XX$_H$	00$_H$

7	6	5	4	3	2	1	0
WDOG_FL AG	UNUSED	UNUSED		UNUSED		EXT_FLA G	ITIM_FL AG
rc	rc	rc	rc	rc	rc	rc	rc

Field	Bits	Type	Description
WDOG_FLAG	7	rc	**Watchdog Wakeup** Reset: 0$_H$
UNUSED	6	-	**Unused** Reset: 0$_H$
UNUSED	5	-	**Unused** Reset: 0$_H$
UNUSED	4	-	**Unused** Reset: 0$_H$
UNUSED	3	-	**Unused** Reset: 0$_H$
UNUSED	2	rc	**Unused** Reset: 0$_H$
EXT_FLAG	1	rc	**I/O-Port PP2 External Wakeup** Reset: 0$_H$
ITIM_FLAG	0	rc	**Interval Timer Wakeup** Reset: 0$_H$

Wakeup Mask Register

WUM			Offset	Wakeup Value	Reset Value
Wakeup Mask Register			C1$_H$	UU$_H$	FF$_H$

7	6	5	4	3	2	1	0
WDOG_MASK	Res	Res	Res	Res	Res	EXT_MASK	ITIM_MASK
wr	wr					wr	wr

Field	Bits	Type	Description
WDOG_MASK	7	wr	**Disable Watchdog Reset** Watchdog is always enabled, setting this bit has no effect in NORMAL/DEBUG modes. Reset: 1$_H$
Res	6	-	**Reserved** This bit must be set to 1$_B$. Reset: 1$_H$
Res	5	-	**Reserved** This bit must be set to 1$_B$ Reset: 1$_H$
Res	4	-	**Reserved** This bit must be set to 1$_B$ Reset: 1$_H$
Res	3	-	**Reserved** This bit must be set to 1$_B$ Reset: 1$_H$
Res	2	-	**Reserved** This bit must be set to 1$_B$ Reset: 1$_H$
EXT_MASK	1	wr	**Disable I/O-Port PP2 External Wakeup** 0$_B$ Enable I/O-Port PP2 wakeup 1$_B$ Disable I/O-Port PP2 wakeup Reset: 1$_H$
ITIM_MASK	0	wr	**Disable Interval Timer Wakeup** Interval Timer Wakeup is always enabled in NORMAL mode, setting this bit has no effect in NORMAL/DEBUG modes. Reset: 1$_H$

Resume Event Flag Register

Note: SFR DIVIC must be 00b in order to reliably read SFR REF

REF	Offset	Wakeup Value	Reset Value
Resume Event Flag Register	**D1$_H$**	**00$_H$**	**00$_H$**

7	6	5	4	3	2	1	0
UNUSED		READC	UNUSED	UNUSED	UNUSED	UNUSED	RET0
-		rc				-	rc
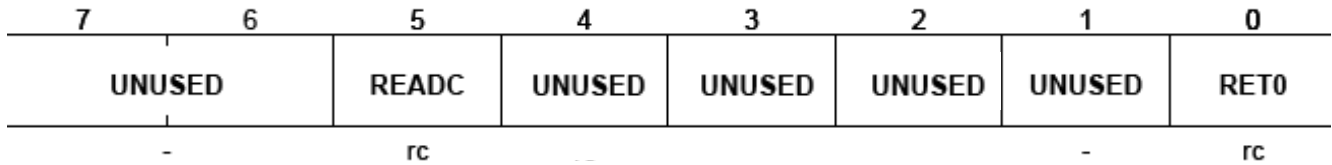
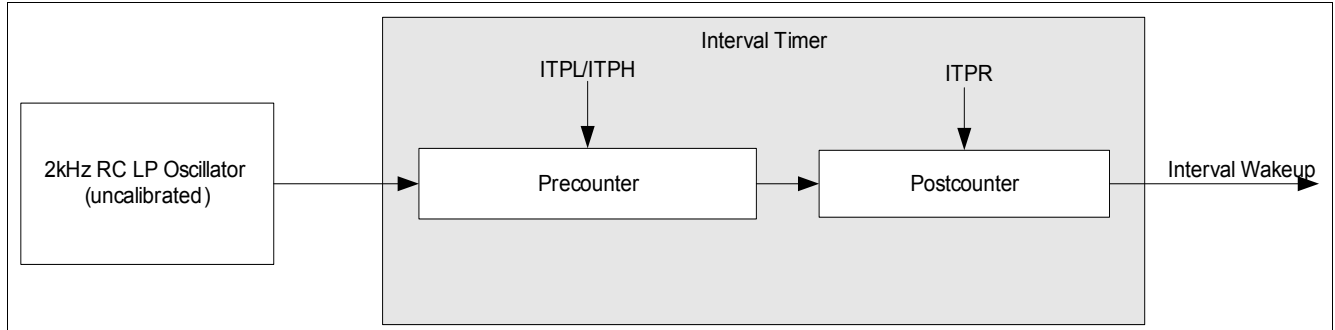| Field	Bits	Type	Description
UNUSED	7:6	-	**UNUSED** Reset: 0$_H$
READC	5	rc	**A/D Conversion complete** This bit is for use by ROM Library functions. Reset: 0$_H$
UNUSED	4	-	**UNUSED** Reset: 0$_H$
UNUSED	3	-	**UNUSED** Reset: 0$_H$
UNUSED	2	-	**UNUSED** Reset: 0$_H$
UNUSED	1	-	**UNUSED** Reset: 0$_H$
RET0	0	rc	**Timer 0 Underflow** Reset: 0$_H$

3.5 Interval Timer

Figure 8   Interval Timer Block Diagram

The Interval Timer is responsible to wakeup the SP27 from the POWER DOWN state after a predefined time interval. It is clocked by the 2kHz RC LP Oscillator and incorporates two dividers:

- Precounter: can be calibrated and represents the timebase; please refer to **"Interval Timer Precounter / Calibration" on Page 45**
- Postcounter: configures the Interval Timer duration in multiples of the timebase.

The Precounter (ITPH/L) is a 16 bit register with 12 significant bits. The precounter values 0001_H up to $0FFF_H$ corresponds to 1_{dec} up to 4095_{dec}. The maximum precounter value 0000_H corresponds to 4096_D.

To increase the timer accuracy, it is recommended to use a ROM library function which calibrates the precounter dependent on the actual frequency of the 2kHz RC LP Oscillator. See **[1]** for details.

The Postcounter (ITPR) is an 8 bit register. 01_H up to FF_H corresponds to a multiplication of the timebase with 1_{dec} up to 255_{dec}. The maximum postcounter value 00_H corresponds to 256_D.

The Interval Timer duration is determined by the SFR ITPR. The desired value can be calculated by using the following equation

$$Interval\,timer\,period\,[s] = timebase\,[s] \bullet postcounter\,[ITPR]$$

(1)

$$timebase\,[s] = \frac{precounter\,[ITPL/H]}{f_{2kHzRCLPOscillator}\left[\frac{l}{s}\right]}$$

(2)

Writing to the Postcounter (ITPR) establishes the counter reload value.

Reading from the Postcounter (ITPR) returns the counter value, not the counter reload value. The Interval Timer is asynchronous from the CPU, so care must be taken when reading its contents. To safely read the ITPR, the SFR CFG1 ITRd bit should be set to one prior to reading ITPR. After the ITPR contents are read, the SFR CFG1 ITRd bit should be checked: If the ITPR read result is valid, the SFR CFG ITRd bit will still be set. If the SFR CFG1 ITRd bit is cleared during ITPR read, the ITPR read result is not valid.

Interval Timer Period Register

ITPR			Offset	Wakeup Value	Reset Value
Interval Timer Period Register			BC$_H$	UU$_H$	01$_H$

7							0
			ITPR				

wr
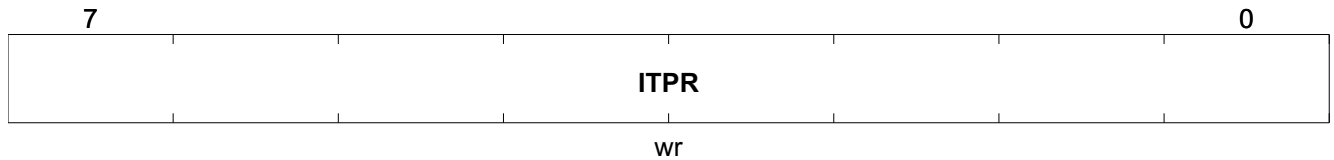
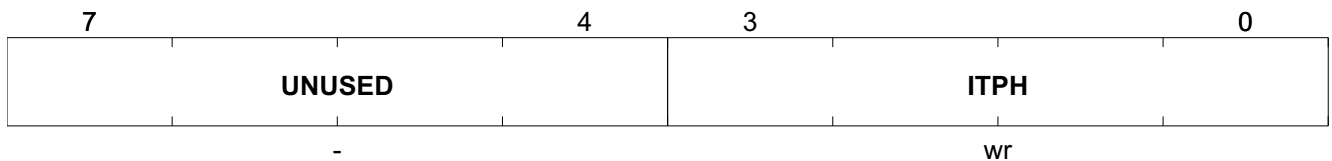| Field	Bits	Type	Description
ITPR	7:0	wr	**Interval Timer Period Register** To safely read ITPR, SFR CFG1.2 should be set to 1 prior to reading ITPR. If the ITPR read was successful, SFR CFG1.2 will remain set to 1. Reset: 01$_H$

3.6 Interval Timer Precounter / Calibration

The calibration is performed automatically by a ROM library function (see [1]).

Notes

1. *SFR ITPL and SFR ITPH can be modified manually for using other (uncalibrated) precounter values than the ones determined by the ROM Library function.*
2. *After writing SFR ITPR, SFR ITPL or SFR ITPH some time is needed to activate the new setting.*
   SFR bit CFG1.1[ITInit] is cleared automatically when the new setting takes effect.
3. *Reading SFR ITPL or SFR ITPH returns the low byte or the high byte of the Precounter reload value, respectively.*
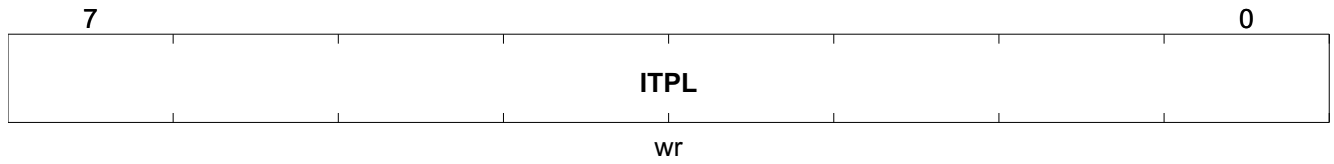
Interval Timer Precounter Register (High Byte)

ITPH		Offset	Wakeup Value	Reset Value
Interval Timer Precounter Register (High Byte)		BB_H	$0000UUUU_B$	03_H

7		4	3		0
UNUSED			ITPH		
-			wr		

Field	Bits	Type	Description
UNUSED	7:4	-	**UNUSED** Reset: 0_H
ITPH	3:0	wr	**Interval Timer Precounter Register (High Byte)** Reset: 3_H

Interval Timer Precounter Register (Low Byte)

ITPL		Offset	Wakeup Value	Reset Value
Interval Timer Precounter Register (Low Byte)		**BA$_H$**	**UU$_H$**	**E8$_H$**

7							0
			ITPL				

wr

Field	Bits	Type	Description
ITPL	7:0	wr	**Interval Timer Precounter Register (Low Byte);** Reset: E8$_H$

3.7 Clock Controller

The Clock Controller for internal clock management is part of the system controller.

In SP27 the microcontroller (CPU) clock source is always based on the 12 MHz RC HF Oscillator (system clock). to provide minimum current consumption. The internal clock divider (SFR DIVIC) may be used to slow down the speed of the microcontroller and to reduce the current consumption further. **Figure 9** shows the clocking scheme for the different SP27 blocks.
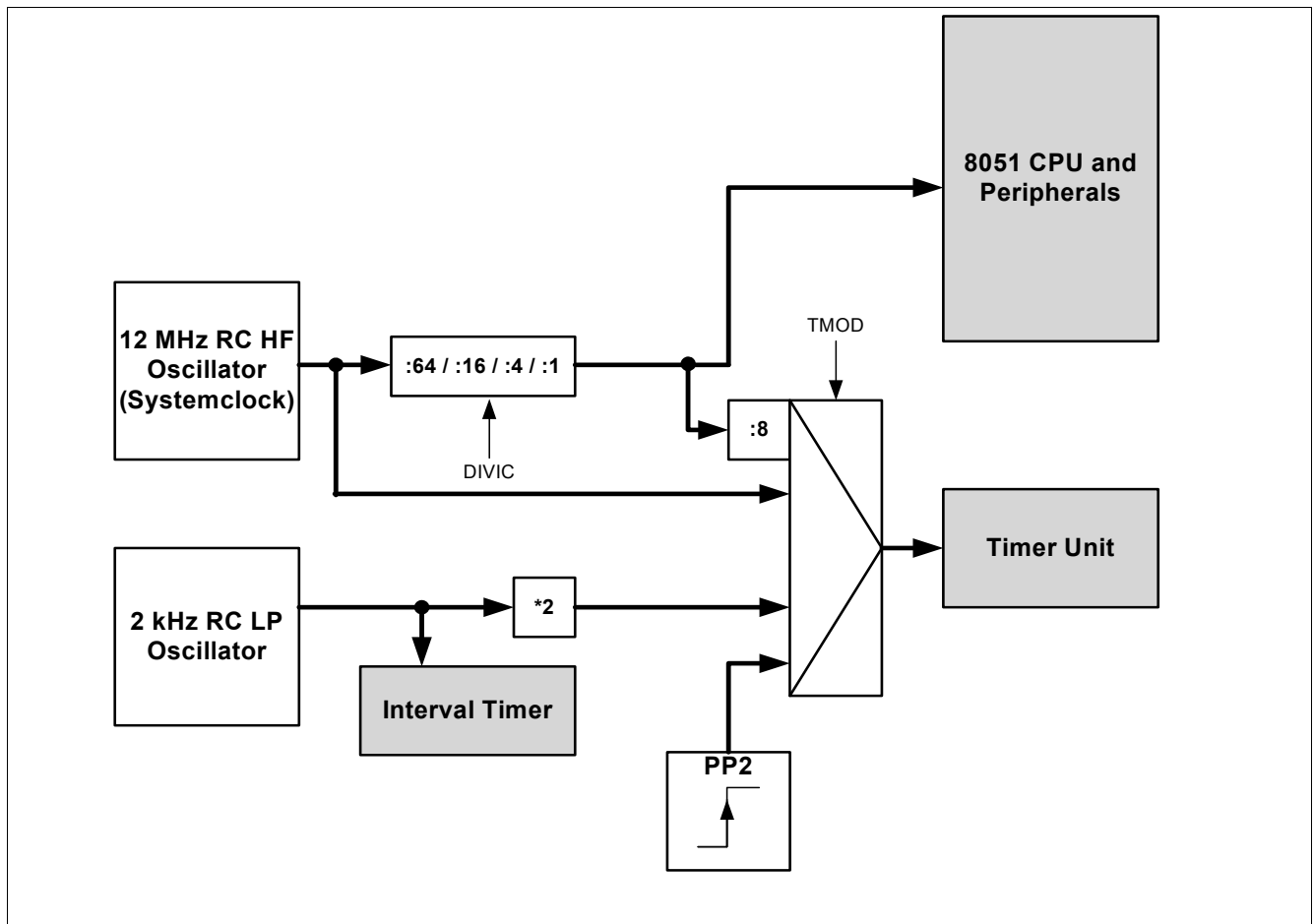

Figure 9   SP27 Clock Concept

2kHz RC LP Oscillator (Low Power)

The 2 kHz RC LP Oscillator stays active throughout all operating states. The typical oscillator frequency is 2kHz. Characteristics can be found in **Table 30 "2 kHz RC LP Oscillator" on Page 93**.

12MHz RC HF Oscillator (High Frequency)

The 12 MHz RC HF Oscillator runs at typical 12 MHz and is used as system clock for the SP27 in RUN state. Characteristics can be found in **Table 29 "12 MHz RC HF Oscillator" on Page 93**.

Internal Clock Divider

DIVIC		Offset	Wakeup Value	Reset Value
Internal Clock Divider		**B9**$_H$	**000000UU**$_B$	**00**$_H$

7				2	1	0
UNUSED					DIVIC	
-					wr	

Field	Bits	Type	Description
UNUSED	7:2	-	**UNUSED** Reset: 00$_H$
DIVIC	1:0	wr	**SystemClock Divider** 00$_B$ Duvide by 1 01$_B$ Divide by 4 10$_B$ Divide by 16 11$_B$ Divide by 64 Reset: 0$_H$

3.8 16 Bit CRC (Cyclic Redundancy Check) Generator/Checker
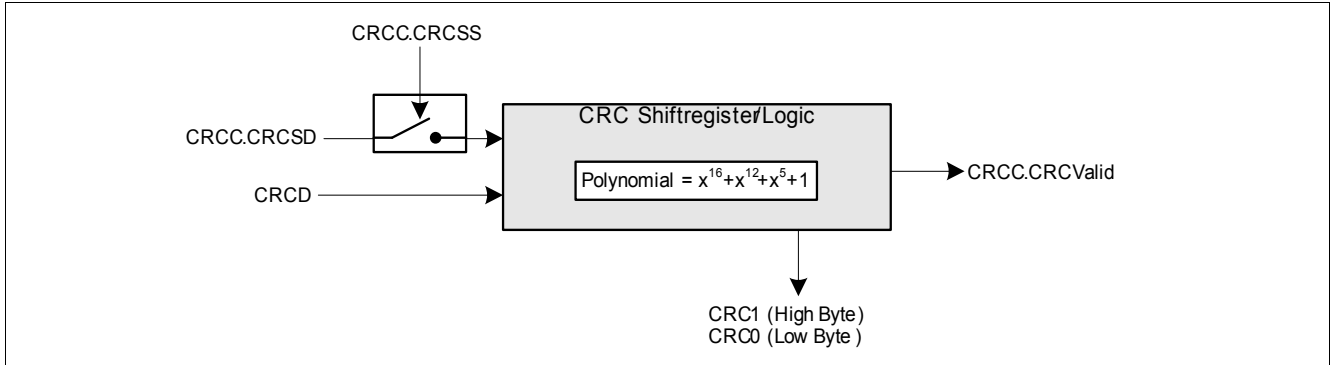

Figure 10 CRC (Cyclic Redundancy Check) Generator/Checker

CRC is a powerful method to detect errors in data packets that have been transmitted over a distorted connection. The CRC Generator/Checker divides each byte of a data packet by a polynomial, leaving the remainder which represents the checksum. The CRC-Generator/Checker is using the 16 Bit CCITT polynomial x16+x12+x5+1. The 16 bit start value is determined by the initial contents of SFR CRC0 and SFR CRC1.

CRC Generation and Checking

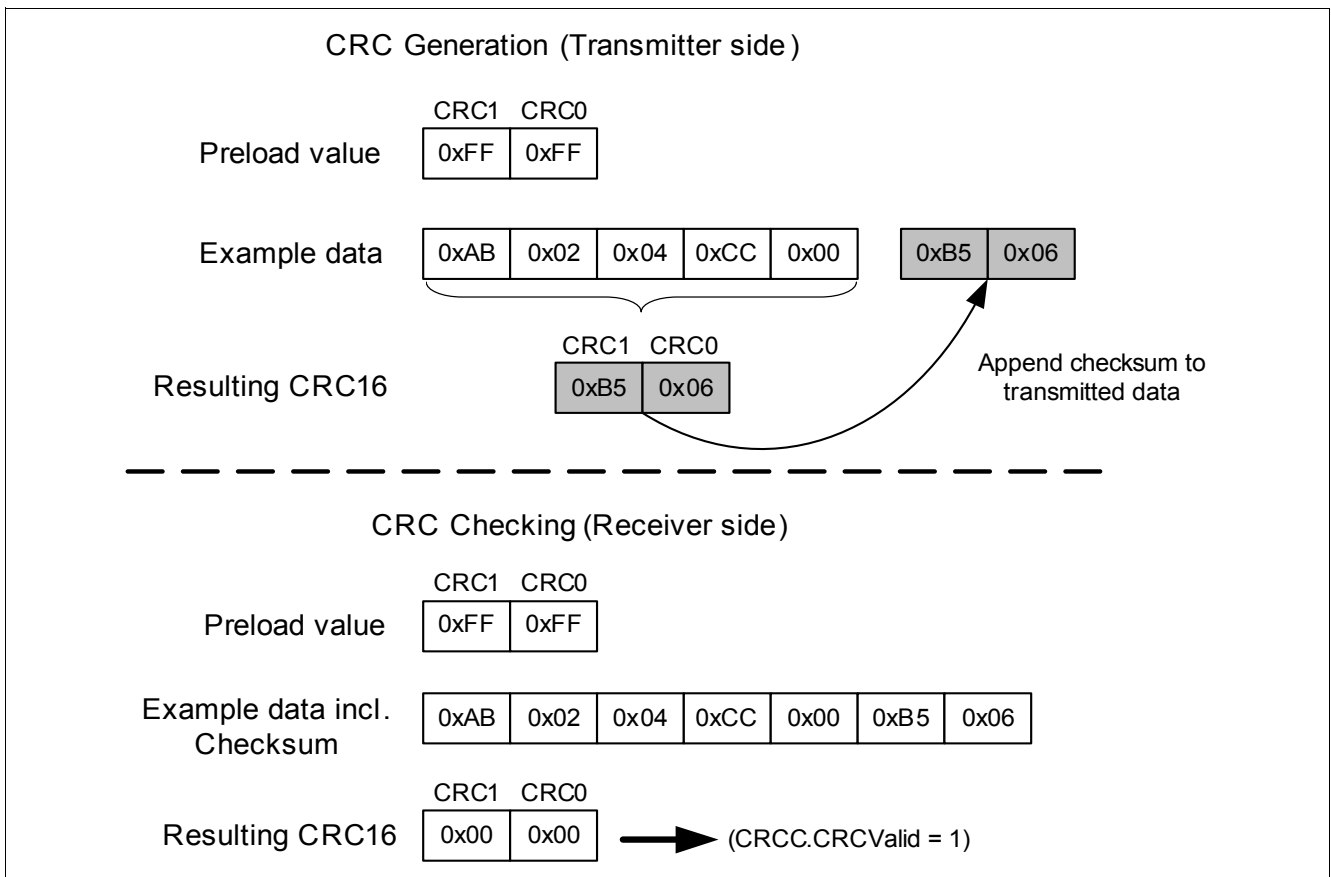**Figure 11** shows the basic usage of the CRC16.

Figure 11 Example for CRC16 usage

Byte aligned CRC Generation

CRC generation is done by executing the following steps:

- The CRC shift register has to be initialized by writing FF_H (or if desired another start value) to both SFR CRC0 and SFR CRC1.
- The databytes which are to be used for the CRC Generation have to be shifted one after the other into the SFR CRCD.The process of CRC Generation is automatically invoked when data bytes are written to the SFR CRCD.

Note: Processing the data in SFR CRCD takes 3 instruction cycles, therefore the application has to assure that there are no consecutive write instructions without one extra instruction cycle between.

- The resulting checksum is available in the CRC Result Register SFR CRC0 and SFR CRC1 after the last data byte has been processed.

Byte aligned CRC Checking

CRC checking is done by executing the following steps:

- The CRC shift register has to be initialized by writing FF_H (or if desired another start value) to both SFR CRC0 and SFR CRC1.
- The databytes which should be checked have to be shifted one after the other into the SFR CRCD. It is important that the order (MSB-LSB) is the same as it was during the CRC Generation. In addition to the data the CRC16 has to be shifted into SFR CRCD as well (first the high byte then the low byte - see also **Figure 11**). The process of CRC Checking is automatically invoked when data bytes are written to the SFR CRCD.

Note: Processing the data in SFR CRCD takes 3 instruction cycles, therefore the application has to assure that there are no consecutive write instructions without one extra instruction cycle between.

Note: One instruction cycle corresponds to 6 system clock cycles.

- After the last byte is processed the SFR bit CRCC.1[CRCValid] indicates the correctness of the CRC calculation after the last data byte has been processed and both - SFR CRC0 and SFR CRC1 are 0.

Serial bitstream CRC Generation/Checking

The CRC Generator/Checker features a serial mechanism to perform CRC generation and checking of non byte-aligned data streams. In this case SFR bit CRCC.5[CRCSS] and SFR bit CRCC.6[CRCSD] are used instead of SFR CRCD.

The data stream is written bit by bit into SFR bit CRCC.6[CRCSD]. Each bit is processed by setting the flag SFR bit CRCC.5[CRCSS].

The following figure shows an example for the usage of SFR bit CRCC.5[CRCSS] and SFR bit CRCC.6[CRCSD].

Note: No time limitations apply when using serial data, so the software can shift and strobe without any wait states between processing consecutive bits.
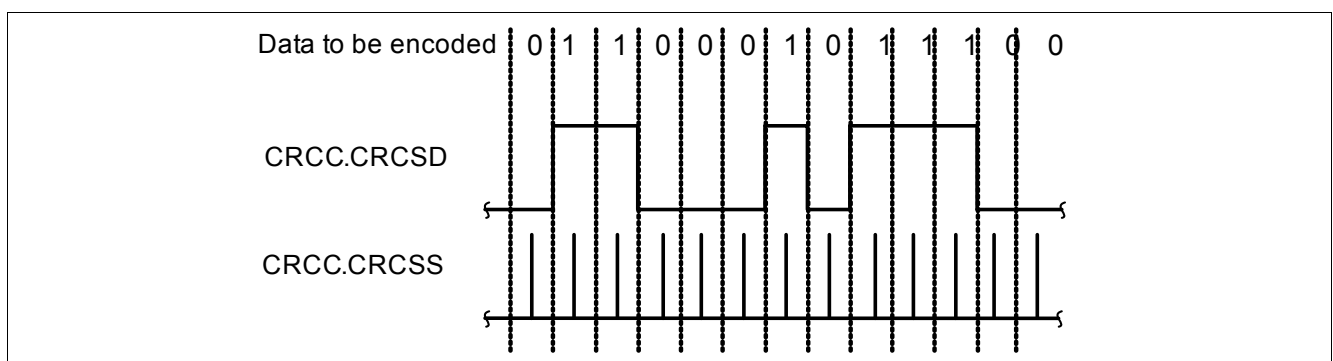

Figure 12   Example for serial CRC generation/checking

Note: The serial and byte-aligned generation/checking mechanism is interchangeable within the same generation/checking process.
Example: If a data packet consists of 18 bits, 16 bits can be processed byte-aligned via SFR CRCD and the two remaining bits can be processed bit-aligned by using SFR bit CRCC.5[CRCSS] and SFR bit CRCC.6[CRCSD].
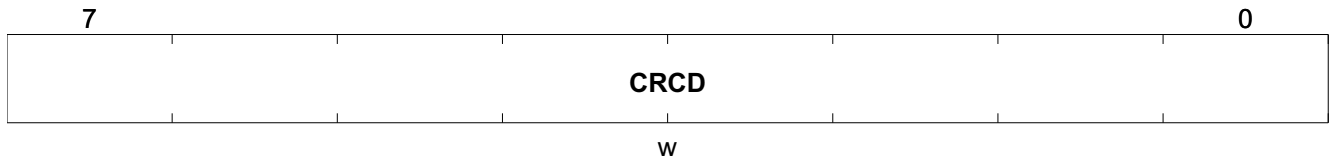
CRC Control Register

CRCC			**Offset**		**Wakeup Value**		**Reset Value**
CRC Control Register			**A9$_H$**		**02$_H$**		**02$_H$**

7	6	5	4		2	1	0
UNUSED	CRCSD	CRCSS	UNUSED			CRCVALID	UNUSED
-	wr	w	-			r	-

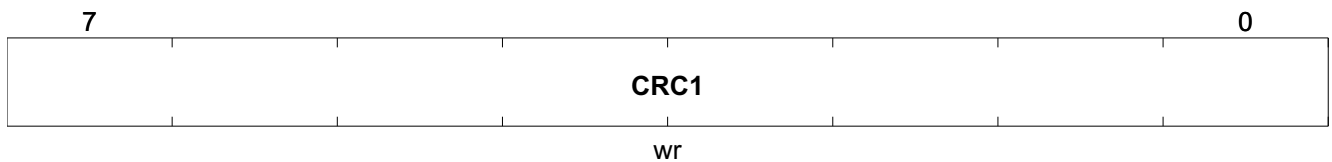| Field	Bits	Type	Description
UNUSED	7	-	**UNUSED** Reset: 0$_H$
CRCSD	6	wr	**CRC Serial Data** Reset: 0$_H$
CRCSS	5	w	**CRC Serial Data Strobe** By setting this bit the data bit from CRCSD is strobed into CRC generator/checker Reset: 0$_H$
UNUSED	4:2	-	**UNUSED** Reset: 0$_H$
CRCVALID	1	r	**CRC Valid** This bit is set automatically when all CRC result bits are zero. 0$_B$ CRC check failed 1$_B$ CRC check successful Reset: 1$_H$
UNUSED	0	-	**UNUSED** Reset: 0$_H$

CRC Data Register

CRCD			Offset	Wakeup Value	Reset Value
CRC Data Register			**AA$_H$**	**00$_H$**	**00$_H$**

```
7                                                    0
┌─────────────────────────────────────────────────────┐
│                        CRCD                          │
└─────────────────────────────────────────────────────┘
                          w
```

Field	Bits	Type	Description
CRCD	7:0	w	**CRC Data Register** Reset: 00$_H$

CRC Shift Register 1 (high byte)

CRC1			Offset	Wakeup Value	Reset Value
CRC Preload/Result Register 1 (high byte)			**AD$_H$**	**00$_H$**	**00$_H$**

```
7                                                    0
┌─────────────────────────────────────────────────────┐
│                        CRC1                          │
└─────────────────────────────────────────────────────┘
                          wr
```

Field	Bits	Type	Description
CRC1	7:0	wr	**CRC Preload/Result Register 1 (high byte)** Reset: 00$_H$

CRC Shift Register 0 (low byte)

CRC0		Offset	Wakeup Value	Reset Value
CRC Preload/Result Register 0 (low byte)		**AC$_H$**	**00$_H$**	**00$_H$**

7							0
			CRC0				

wr
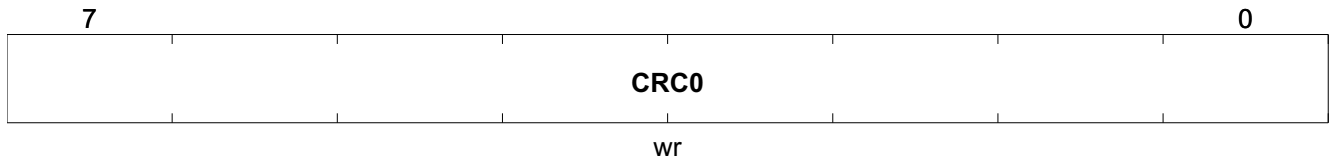
Field	Bits	Type	Description
CRC0	7:0	wr	**CRC Preload/Result Register 0 (low byte)** Reset: 00$_H$

CONFIDENTIAL

3.9 Pseudo Random Number Generator

The SP27 offers a Pseudo Random Number Generator. It consists of a Maximum Length Linear Feedback Shift Register (MLFSR) as built in hardware unit which creates a new pseudo random number everytime SFR bit CFG1.5[RNGEn] is set.
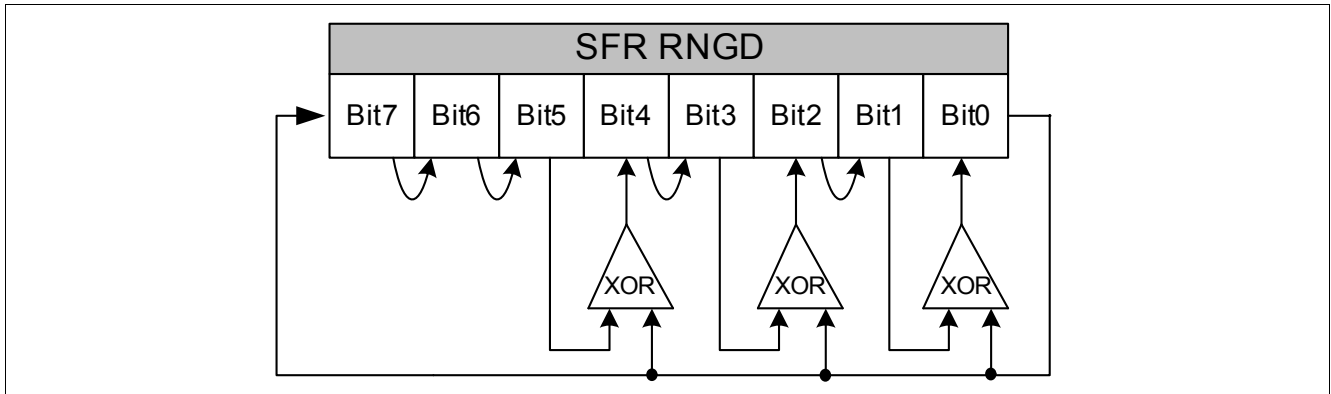

Figure 13   Shift Register Implementation

A user-defined start value (except 00_H) can be written to SFR RNGD. The default value is 55_H.

The generation of a new random number is initiated by setting SFR bit CFG1.5[RNGEn]. The random number generation requires one systemclock cycle, thus the application can read the generated number in the next instruction without any extra wait states.

Random Number Generator Data Register

RNGD	Offset	Wakeup Value	Reset Value
Random Number Generator Data Register	AB_H	UU_H	55_H

7							0
			RNGD				

wr

Field	Bits	Type	Description
RNGD	7:0	wr	**Random Number Generator Data Register** Reset: 55_H

3.10   Timer Unit

The SP27 comprises two independent 16 bit timers which operate as down counters. Different timer modes are available for extended functionality.

Basic Timer Configuration

Timer 0 and Timer 1 comprise two fully programmable 16-bit timers, which can be used for time measurements as well as generating time delays. The clock source is selectable in order to enlarge the timer runtime.SFR TMOD is used to select the clock source and the desired timer mode.

Setting the SFR bit TCON.0[T0Run] (respectively SFR bit TCON.4[T1Run]) starts Timer 0 (resp. Timer 1). The Timer counts down from the start values SFR TH/L0 (resp. SFR TH/L1) using the selected counter clock (see SFR TMOD) until the timer is elapsed. SFR bit TCON.1[T0Full] (resp. SFR bit TCON.5[T1Full]) is set.

Note: The timer full flag is set when the timer would underflow from 0000_H ==> FFFFH. To count e.g. 10 events with timer 0 SFR TL0 has to be configured to 9H instead of 0AH.

After a timer is elapsed there are two possibilities that can occur (dependant on the selected timer mode):

• **Reload:** If the selected timer mode uses timer reload, the timer is automatically reloaded and restarted.
• **Stop:** If the selected timer mode doesn't use timer reload, the timer is stopped and SFR bit TCON.0[T0Run] (resp. SFR bit TCON.4[T1Run]) is cleared.

Timer Modes

Timer mode 0

Comprises:

• 16 bit timer with reload

The timer unit is configured as one 16 bit reloadable timer.

SFR TL0 and SFR TH0 hold the start value.

When SFR bit TCON.0[T0Run] is set, the timer starts counting down.

SFR bit TCON.1[T0Full] is set when the timer is elapsed.

The timer value is reloaded from SFR TL1 and SFR TH1 and the timer is restarted automatically.

SFR bit TCON.1[T0Full] has to be cleared by the application software. It is not cleared automatically on a read-access.
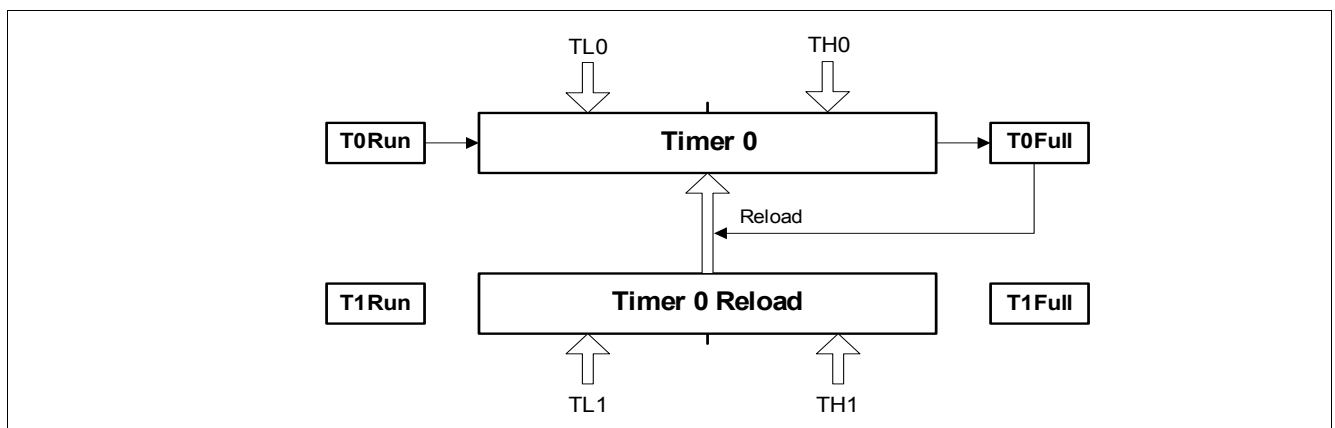

Figure 14   Timer Mode 0

Note: In this mode both SFR bit TCON.4[T1Run] and SFR bit TCON.5[T1Full] are not used.

Timer Mode 1

Comprises:

- 16 bit timer without reload
- 8 bit timer with reload

Timer 0 operates as a 16 bit timer with the start value in SFR TL0 and SFR TH0, timer run bit SFR bit TCON.0[T0Run] and timer elapsed indicator SFR bit TCON.1[T0Full].

If the timer is elapsed, it is stopped, SFR bit TCON.1[T0Full] is set and the timer run bit SFR bit TCON.0[T0Run] is cleared.

Timer 1 sets up a reloadable 8 bit timer holding the startup value in SFR TL1, timer reload value in SFR TH1, timer run bit in SFR bit TCON.4[T1Run] and timer elapsed indicator in SFR bit TCON.5[T1Full].
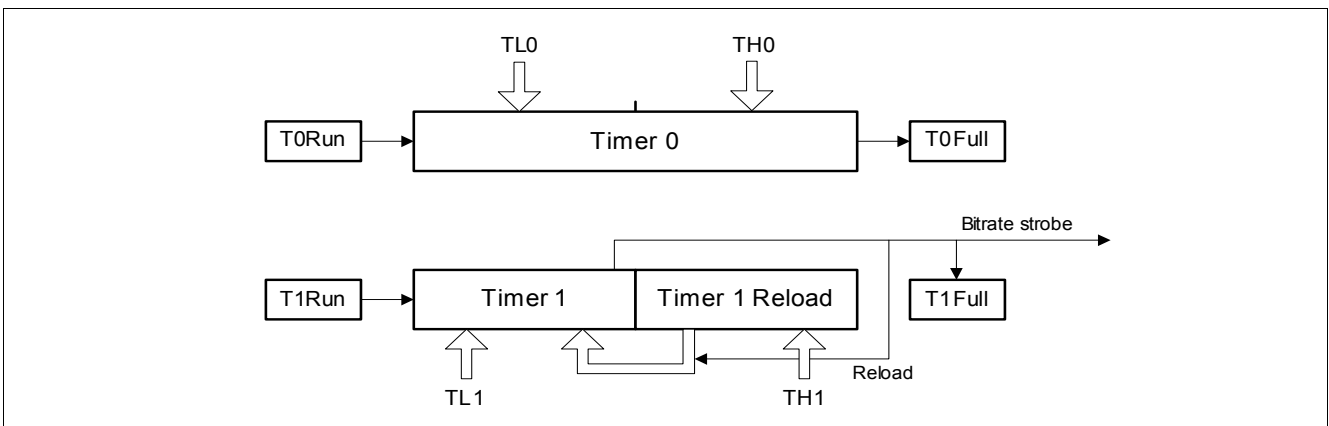

Figure 15   Timer Mode 1

Timer Mode 2

Comprises:

- 8 bit timer with reload
- 8 bit timer with reload

Timer 0 sets up a reloadable 8 bit timer holding the start value SFR TL0, timer reload value SFR TH0, timer run bit SFR bit TCON.0[T0Run] and timer elapsed indicator SFR bit TCON.1[T0Full].

Timer 1 sets up a reloadable 8 bit timer holding the start value SFR TL1, timer reload value SFR TH1, timer run bit SFR bit TCON.4[T1Run] and timer elapsed indicator SFR bit TCON.5[T1Full].
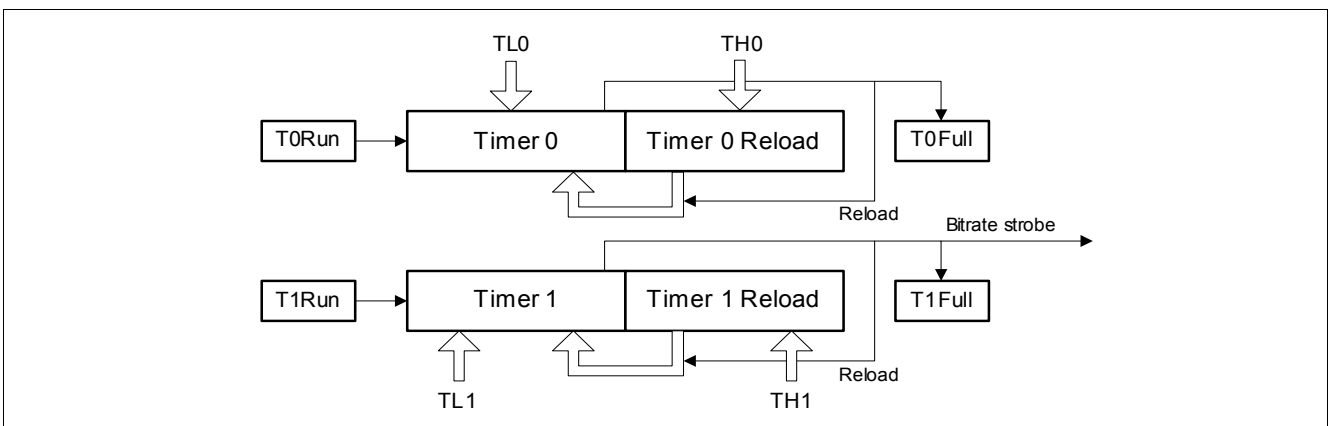

Figure 16   Timer Mode 2

Timer Mode 3

Comprises:

*   8 bit timer without reload (1)
*   8 bit timer without reload (2)
*   8 bit timer with reload

Timer 0 (1) uses SFR TL0 as start value.

Setting SFR bit TCON.0[T0Run] starts the timer.

SFR bit TCON.1[T0Full] is set when the timer is elapsed.

SFR bit TCON.0[T0Run] is cleared automatically when the timer is elapsed.

Timer 0 (2) uses SFR TH0 as start value.

Setting SFR bit TCON.4[T1Run] starts the timer.

SFR bit TCON.5[T1Full] is set when the timer is elapsed.

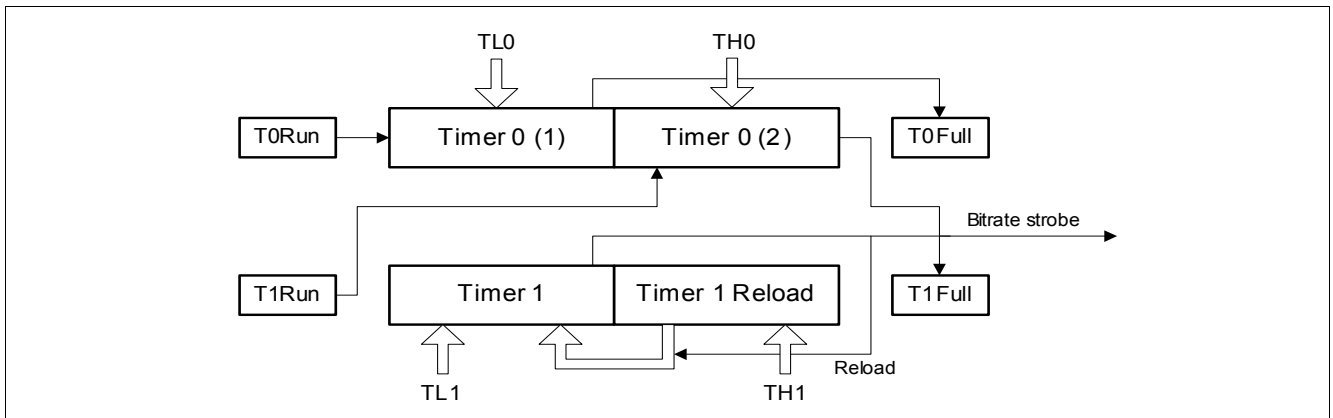SFR bit TCON.4[T1Run] is cleared automatically when the timer is elapsed.

Figure 17   Timer Mode 3

Timer mode 4

This time mode is not availavle for application usage.

Timer Mode 5

Comprises:

- 8 bit timer with reload
- 16 bit timer without reload

Timer 0 sets up a reloadable 8 bit timer holding the start value in SFR TL0, timer reload value in SFR TH0, timer run bit SFR bit TCON.0[T0Run] and timer elapsed indicator in SFR bit TCON.1[T0Full].

Timer 1 operates as a 16 bit timer with the start value in SFR TL1 and SFR TH1, timer run bit SFR bit TCON.4[T1Run] and timer elapsed indicator SFR bit TCON.5[T1Full]. If the timer is elapsed the timer is stopped, SFR bit TCON.5[T1Full] is set and the timer run bit SFR bit TCON.4[T1Run] is cleared.

Figure 18   Timer Mode 5

Timer Mode 6

Comprises:

• 16 bit timer without reload
• 16 bit timer without reload

Timer 0 operates as a 16 bit timer with the start value in SFR TL0 and SFR TH0, timer run bit SFR bit TCON.0[T0Run] and timer elapsed indicator SFR bit TCON.1[T0Full].

If the timer is elapsed the timer is stopped, SFR bit TCON.1[T0Full] is set and the timer run bit SFR bit TCON.0[T0Run] is cleared.

Timer 1 operates as a 16 bit timer with the start value in SFR TL1 and SFR TH1, timer run bit SFR bit TCON.4[T1Run] and timer elapsed indicator SFR bit TCON.5[T1Full].

If the timer is elapsed the timer is stopped, SFR bit TCON.5[T1Full] is set and the timer run bit SFR bit TCON.4[T1Run] is cleared.
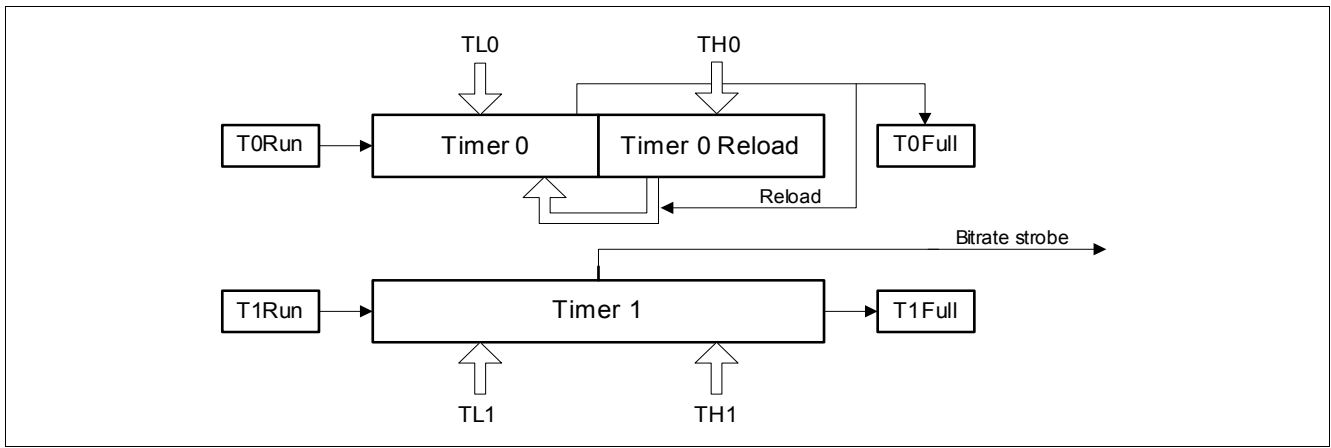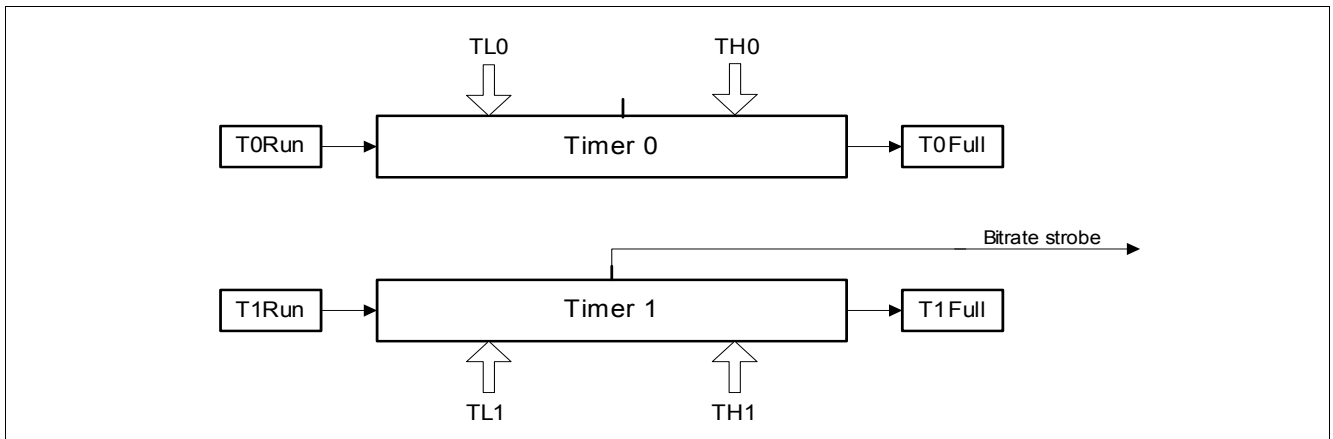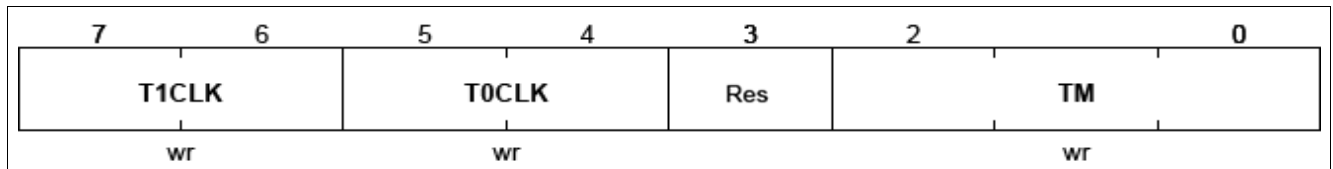

Figure 19   Timer Mode 6

Timer Mode 7

This timer mode is not available for application usage. It is used by the ROM library functions for calibration purpose.

Timer Mode Register

TMOD	Offset	Wakeup Value	Reset Value
Timer Mode Register	89_H	00_H	00_H

7	6	5	4	3	2	0
T1CLK		T0CLK		Res	TM	
wr		wr			wr	

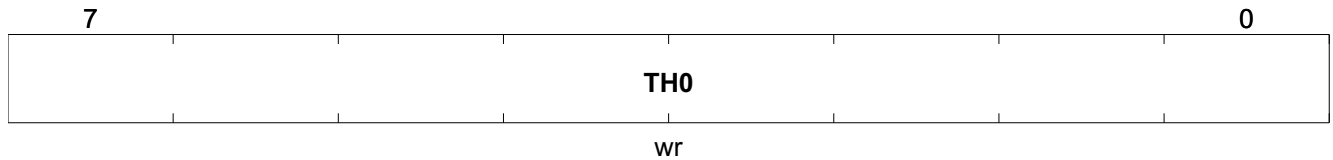| Field	Bits	Type	Description
T1CLK	7:6	wr	**Timer 1 Clock Source Select** 00_B System clock 01_B System clock divided by SFR DIVIC and 8 10_B 4kHz (2*2kHz LP RC Oscillator) 11_B PP2 event count (rising edge) Reset: 0_H
T0CLK	5:4	wr	**Timer 0 Clock Source Select** 00_B System clock 01_B System clock divided by SFR DIVIC and 8 10_B 4kHz (2*2kHz LP RC Oscillator) 11_B Timer 1 underflow event count Reset: 0_H
Res	3	-	**Reserved** This bit must be set to 0_B. Reset: 0_H
TM	2:0	wr	**Timer Mode** 000_B Mode 0: 16 bit timer w/ reload and no baudrate strobe 001_B Mode 1: 16 bit timer w/o reload and 8 bit timer w/ reload 010_B Mode 2: two 8 bit timers w/ reload 011_B Mode 3: two 8 bit timers w/o reload and 8 bit timer w/ reload 100_B Mode 4: Not available for application use 101_B Mode 5: 8 bit timer w/ reload and 16 bit timer w/o reload 110_B Mode 6: two 16 bit timer w/o reload 111_B Mode 7: Not available for application use Reset: 0_H

Timer Control Register

TCON				Offset	Wakeup Value		Reset Value
Timer Control Register				88$_H$	00$_H$		00$_H$

7	6	5	4	3	2	1	0
UNUSED		T1FULL	T1RUN	UNUSED		T0FULL	T0RUN
-		wr	wr	-		wr	wr
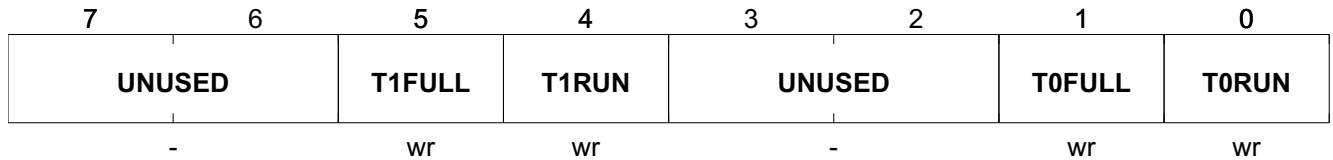
Field	Bits	Type	Description
UNUSED	7:6	-	**UNUSED** Reset: 0$_H$
T1FULL	5	wr	**Timer 1 Full Flag** Set if a Timer 1 underflow has occurred. Reset: 0$_H$
T1RUN	4	wr	**Timer 1 Run** 0$_B$ Timer 1 is stopped 1$_B$ Timer 1 counts downward Reset: 0$_H$
UNUSED	3:2	-	**UNUSED** Reset: 0$_H$
T0FULL	1	wr	**Timer 0 Full Flag** Set if a Timer 0 underflow has occurred. Reset: 0$_H$
T0RUN	0	wr	**Timer 0 Run** 0$_B$ Timer 0 is stopped 1$_B$ Timer 0 counts downward Reset: 0$_H$

Timer 0 Register High Byte

TH0			Offset	Wakeup Value	Reset Value
Timer 0 Register High Byte			$8C_H$	00_H	00_H

7							0
			TH0				

wr

Field	Bits	Type	Description
TH0	7:0	wr	**Timer 0 High Byte** Timer 0 [15:8] Reset: 00_H

Timer 0 Register Low Byte

TL0			Offset	Wakeup Value	Reset Value
Timer 0 Register Low Byte			$8A_H$	00_H	00_H

7							0
			TL0				

wr
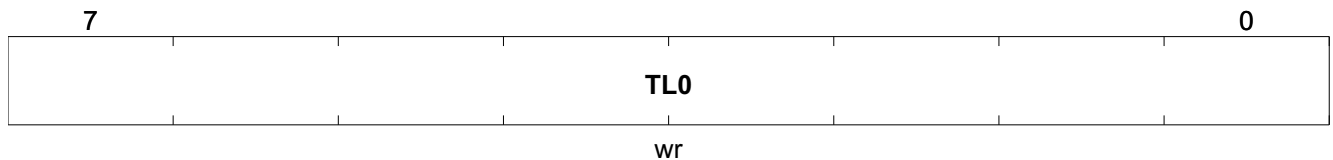
Field	Bits	Type	Description
TL0	7:0	wr	**Timer 0 Low Byte** Timer 0 [7:0] Reset: 00_H

Timer 1 Register High Byte

TH1		**Offset**	**Wakeup Value**	**Reset Value**
Timer 1 Register High Byte		$8D_H$	00_H	00_H

7   0

TH1

wr
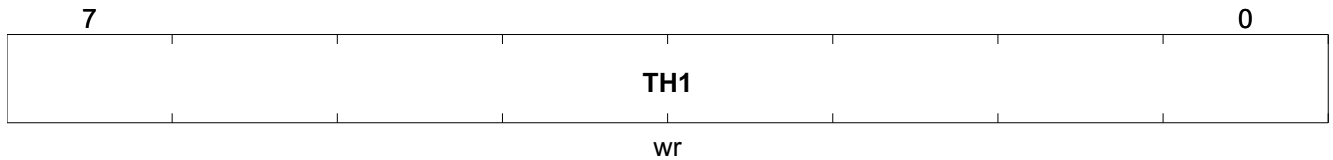
Field	Bits	Type	Description
TH1	7:0	wr	**Timer 1 High Byte** Timer 1 [15:8] Reset: 00_H

Timer 1 Register Low Byte

TL1		**Offset**	**Wakeup Value**	**Reset Value**
Timer 1 Register Low Byte		$8B_H$	00_H	00_H

7   0

TL1

wr
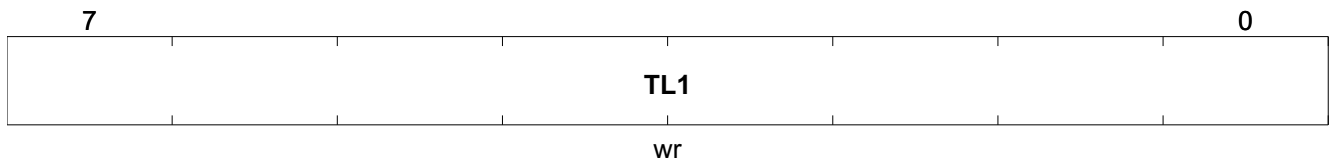
Field	Bits	Type	Description
TL1	7:0	wr	**Timer 1 Low Byte** Timer 1 [7:0] Reset: 00_H

3.11 General Purpose Input/Output (GPIO)

The SP27 has three GPIO pins that are used for:

• General purpose by the application software
• Dedicated digital peripherals (**"Alternative Port Functionality" on Page 66**).
• Operating mode selection (**"Resets and Operating Mode Selection" on Page 15**)
• If configured for general purpose usage, the GPIO pins can be accessed directly by the CPU. All GPIO pins have selectable pull-up and pull-down resistors.

Note: The GPIO pins are configured as input with the pull-up resistor enabled after reset. In POWER DOWN state the GPIOs keep their configuration.

Peripheral Port Configuration

The following table shows the different possible configurations for the GPIO pins.

The 'x' in the table has to be replaced by either 0,1 or 2 (for PP0, PP1 or PP2).

Table 9   GPIO Port Configuration

P1DIR.x	P1OUT.x	P1SENS.x	I/O	Pull-up/ pull-down	Comment
0	0	-	Output	No	LOW (sink)
0	1	-	Output	No	HIGH (source)
1	0	-	Input	No	high-Z (Tri-State)
1	1	0	Input	Pull-up	Weak-High
1	1	1	Input	Pull-down	Weak-Low

Note: The GPIO pin PP2 can be configured as external wakeup. For the required register settings please refer to **"External Wakeup on PP2" on Page 66***.*

Spike Suppression on Input Pins

To avoid unstability when reading the GPIO pins, a synchronization stage with a two-stage spike filter is included.

Due to the synchronization stage the following conditions might occur:

• Signal duration (T_{SIGNAL}) < 1 systemclock[1] period (1 T_{CLK}): Signal is suppressed
• 1 T_{CLK} < T_{SIGNAL} < 2 T_{CLK}: undefined if suppressed or passed
• T_{SIGNAL} > 2 T_{CLK}: Signal appears in SFR P1IN

Figure 20 shows examples of different input signals and how they are processed by the the synchronization stage and the spike filter.

---

1) For PP0, PP1 and PP2 the synchronization stage uses the undivided systemclock (SFR DIVIC) as clocksource, so SFR DIVIC has no influence.
   The synchronization stage is disabled in POWER DOWN state, so it is not used for an external wakeup (see **"External Wakeup on PP2" on Page 66**).
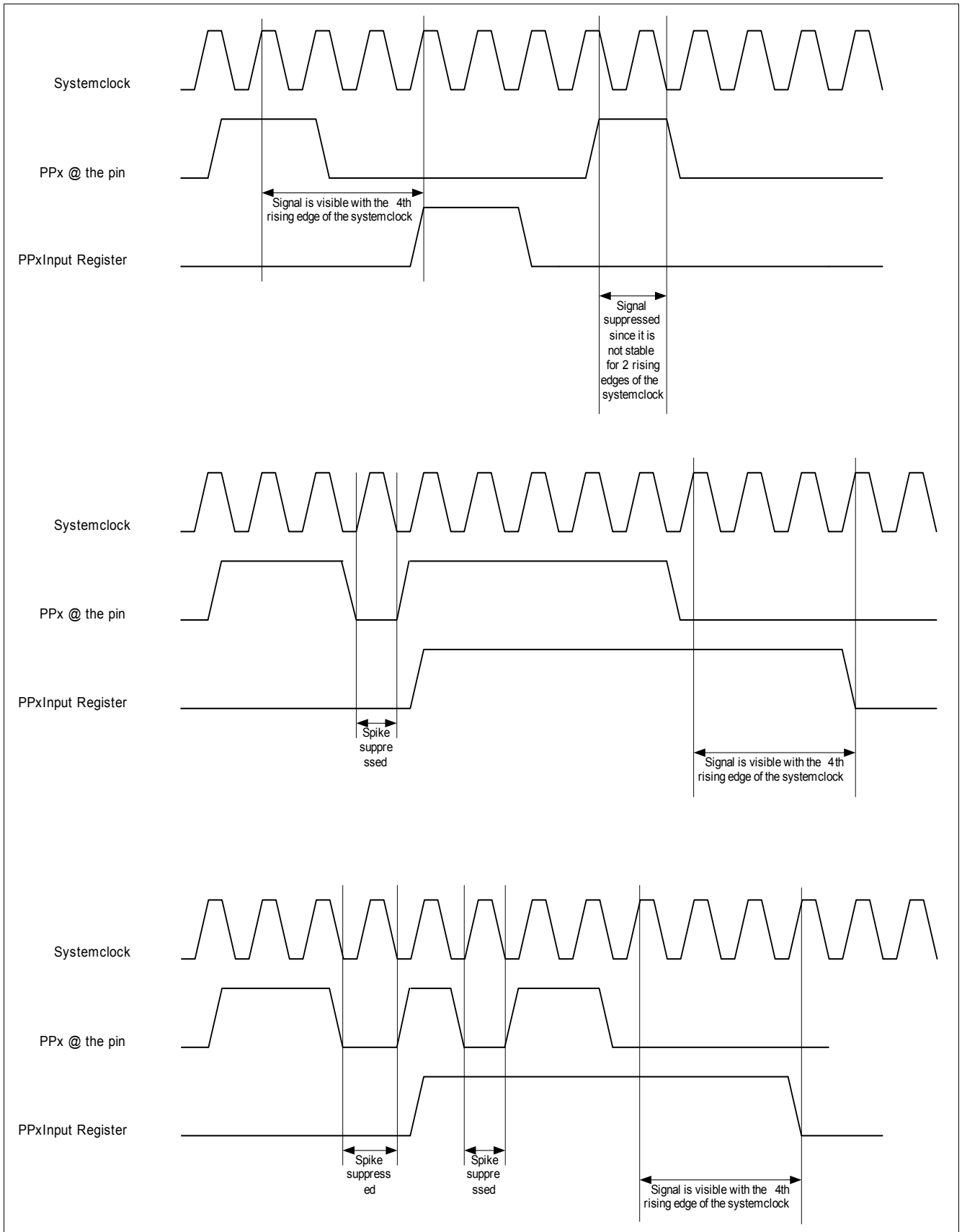
Figure 20   Synchronization Stage & Spike Suppression Examples

External Wakeup on PP2

PP2 can be used as an external wakeup source if enabled by the Wakeup-Mask SFR bit WUM.1[EXT1] and configured as input pin by setting SFR bit P1DIR.2[PPD2].

The internal pull-up/pull-down resistor is enabled if SFR bit P1OUT.2[PPO2] is set. SFR bit P1SENS.2[PPS2] selects the sensitivity (high active/low active):

Table 10   External Wakeup Configuration

SFR Settings	Description
SFR bit P1DIR.2[PPD2] = 1 SFR bit P1OUT.2[PPO2]=1 SFR bitP1SENS.2[PPS2] = 0 SFR bit WUM.1[EXT] = 0	PP2 configured as Input, pull-up enabled, Wakeup occurs if PP2 is forced to LOW externally.
SFR bit P1DIR.2[PPD2] = 1 SFR bit P1OUT.2[PPO2] =1 SFR bitP1SENS.2[PPS2] = 1 SFR bit WUM.1[EXT] = 0	PP2 configured as Input, pull-down enabled, Wakeup occurs if PP2 is forced to HIGH externally.

**Alternative Port Functionality**

In **Table 11**, the alternative port functionality which has higher priority than standard I/O port functionality is shown.

Table 11   I/O Port 1 - Alternative Functionality

Pin	Function	I/O	Description
PP0	I2C-SCL	I	**I2C Serial Clock Line** Configured to I2C clock pin if SFR bit CFG1.6 [I2CEN] is set. Weak-High has to be provided by an external pull-up resistor or by the I2C master device.
	Port Pin I/O	I/O	Standard I/O port functionality
PP1	I2C-SDA	I/O	I2C Serial Data Configured to I2C data pin if bit CFG1.6 [I2CEN] is set. Weak-High has to be provided either by the internal pull-up resistor, by an external pull-up resistor or by the I2C master device.
	Port Pin I/O	I/O	Standard I/O port functionality
PP2	Port Pin I/O	I/O	Standard I/O port functionality / External Wakeup

IO-Port 1 Direction Register

P1DIR	**Offset**	**Wakeup Value**	**Reset Value**
IO-Port 1 Direction Register	**91$_H$**	**UU$_H$**	**FF$_H$**

7				3	2	1	0
Res					PPD2	PPD1	PPD0
					wr	wr	wr

Field	Bits	Type	Description
Res	7:3		**Reserved** These bits must be set to 11111$_B$. Reset: 1F$_H$
PPD2	2	wr	**PP2 I/O Port Direction** 0$_B$ Output 1$_B$ Input Reset: 1$_H$
PPD1	1	wr	**PP1 I/O Port Direction** 0$_B$ Output 1$_B$ Input Reset: 1$_H$
PPD0	0	wr	**PP0 I/O Port Direction** 0$_B$ Output 1$_B$ Input Reset: 1$_H$

IO-Port 1 Data In Register

P1IN			Offset	Wakeup Value	Reset Value
IO-Port 1 Data In Register			92$_H$	00000XXX$_B$	00000XXX$_B$

7				3	2	1	0
UNUSED					PPI2	PPI1	PPI0
			-		r	r	r
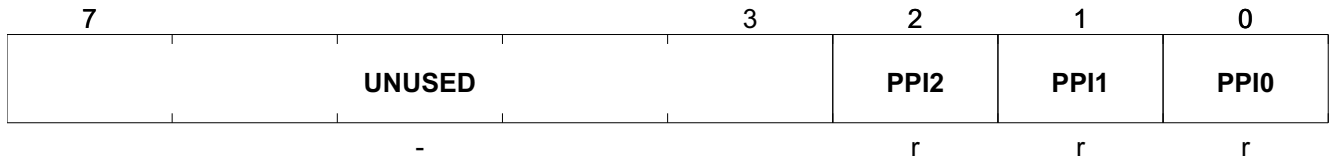
Field	Bits	Type	Description
UNUSED	7:3	-	**UNUSED** Reset: 00$_H$
PPI2	2	r	**PP2 I/O-Port Input Data** 0$_B$ LOW Level 1$_B$ HIGH Level Reset: X$_B$
PPI1	1	r	**PP1 I/O-Port Input Data** 0$_B$ LOW Level 1$_B$ HIGH Level Reset: X$_B$
PPI0	0	r	**PP0 I/O-Port Input Data** 0$_B$ LOW Level 1$_B$ HIGH Level Reset: X$_B$

IO-Port 1 Data Out Register

P1OUT	Offset	Wakeup Value	Reset Value
IO-Port 1 Data Out Register	90_H	UU_H	FF_H

7				3	2	1	0
Res					PPO2	PPO1	PPO0
					wr	wr	wr

Field	Bits	Type	Description
Res	7:3		**Reserved** These bits must be set to 11111_B. Reset: $1F_H$
PPO2	2	wr	**PP2 I/O-Port Output Data / Pulling Resistor** When SFR P1DIR.PPD2 is clear (output), then PPO2 defines the output port state. When SFR P1DIR.PPD2 is set (input), then PPO2 controls the input pulling resistor. 0_B Output low / Tristate 1_B Output high / Pulling Resistor enabled Reset: 1_H
PPO1	1	wr	**PP1 I/O-Port Output Data / Pulling Resistor** When SFR P1DIR.PPD1 is clear (output), then PPO1 defines the output port state. When SFR P1DIR.PPD1 is set (input), then PPO1 controls the input pulling resistor. 0_B Output low / Tristate 1_B Output high / Pulling Resistor enabled Reset: 1_H
PPO0	0	wr	**PP0 I/O-Port Output Data / Pulling Resistor** When SFR P1DIR.PPD0 is clear (output), then PPO0 defines the output port state. When SFR P1DIR.PPD0 is set (input), then PPO0 controls the input pulling resistor. 0_B Output low / Tristate 1_B Output high / Pulling Resistor enabled Reset: 1_H

IO-Port 1 Sensitivity Register

P1SENS	**Offset**	**Wakeup Value**	**Reset Value**
IO-Port 1 Sensitivity Register	93$_H$	00000UUU$_B$	00$_H$

7				3	2	1	0
		UNUSED			PPS2	PPS1	PPS0
		-			wr	wr	wr

Field	Bits	Type	Description
UNUSED	7:3	-	**UNUSED** Reset: 00$_H$
PPS2	2	wr	**PP2 I/O-Port Sensitivity** Only applies when SFR P1DIR.PPD2 is set (input) and SFR P1OUT.PPO2 is set (enable pulling). 0$_B$ Pull-up Resistor 1$_B$ Pull down Resistor Reset: 0$_H$
PPS1	1	wr	**PP1 I/O-Port Sensitivity** Only applies when SFR P1DIR.PPD1 is set (input) and SFR P1OUT.PPO1 is set (enable pulling). 0$_B$ Pull-up Resistor 1$_B$ Pull down Resistor Reset: 0$_H$
PPS0	0	wr	**PP0 I/O-Port Sensitivity** Only applies when SFR P1DIR.PPD0 is set (input) and SFR P1OUT.PPO0 is set (enable pulling). 0$_B$ Pull-up Resistor 1$_B$ Pull down Resistor Reset: 0$_H$

3.12   I^{2}C Interface

For communication between a host and the SP27 a I2C slave interface is implemented.

- PP1 is used as serial data line (SDA)
- PP0 is used as serial clock line (SCL)
- SP27 responds to I^{2}C Address $6C_H$ or a general call (if enabled) by addressing slave address 00_H. General call can be enabled by setting SFR bit CFG2.6[I2CGCEN].
- The data transfer rate is specified in **"I^{2}C Interface" on Page 98**.

Note: In PROGRAMMING mode and DEBUG mode the I^{2}C Interface is handled by the mode handlers located in the ROM automatically. Manual access to those registers is only required if the I^{2}C Interface is utilized in NORMAL mode

Using the slave I^{2}C interface in NORMAL mode

To enable the I2C interface the SFR bit CFG1.6[I2CEN] has to be set. Once the I2C interface has been enabled, the SP27 waits for a start condition to occur.

After the SP27 has received a start condition, the following received 8 bits are compared to the device address. If the address matches, the hardware automatically generates an acknowledge, sets SFR bit I2CS.7[AM] and configures SFR bit I2CS.3[RNW] according to the received address which determines if the I^{2}C access is read or write.

Dependant on SFR bit I2CS.3[RNW] the following two actions has to be performed by the application software:

(SFR bit I2CS.3[RnW] == 0) - Receive I^{2}C-data

- If SFR bit I2CS.0[RBF] is set, one byte has been received and can be read by the application software via SFR I2CD.
  Each byte is acknowledged automatically as long as no receive buffer overflow (SFR bit I2CS.5[OV]) occurs.
- If SFR bit I2CS.4[S] is set, a stop condition has occurred; the transmission was closed by the I^{2}C master device.
- If SFR bit I2CS.7[AM] is set, a restart condition has been initiated. In case of a write access (SFR bit I2CS.3[RNW] == 1) a branch to the transmit subroutine has to be performed.

(SFR bit I2CS.3[RnW] == 1) - Transmit I^{2}C-data

- Data to be transmitted has to be written to SFR I2CD.
  SFR bit I2CS.1[TBF] cleared when the data written to SFR I2CD is taken over by the shift-register. If the I^{2}C master device requests more data bytes it creates an acknowledge (SFR bit I2CS.2[RACK]) and subsequently new data may be written to SFR I2CD. If no data is provided, the I^{2}C-interface automatically sets line SCL to low until data is written to SFR I2CD (slave device gains access over the I^{2}C clock line).
- If SFR bit I2CS.4[S] is set, the transmission process has been terminated by the I^{2}C master device and the transmission subroutine can be left.

To control I^{2}C slave interface the following registers are implemented:
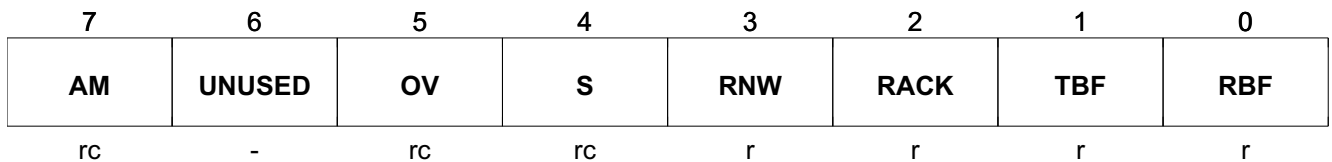
I2C Data Register

I2CD			Offset	Wakeup Value	Reset Value
I2C Data Register			**9A**$_H$	**00**$_H$	**00**$_H$

7							0
			I2CD				

wr
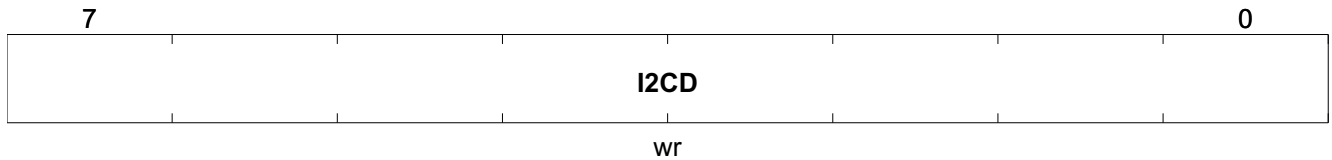
Field	Bits	Type	Description
I2CD	7:0	wr	**I2C Data Register** Reset: 00$_H$

I2C Status Register

I2CS			Offset	Wakeup Value	Reset Value
I2C Status Register			**9B**$_H$	**00**$_H$	**00**$_H$

7	6	5	4	3	2	1	0
AM	UNUSED	OV	S	RNW	RACK	TBF	RBF
rc	-	rc	rc	r	r	r	r

Field	Bits	Type	Description
AM	7	rc	**I2C Address Match** This bit will be set when the device address or general call matches with received address byte. It is automatically cleared by hardware after I2CS is read. Reset: 0$_H$
UNUSED	6	-	**UNUSED** Reset: 0$_H$
OV	5	rc	**I2C Overflow** This bit will be set when a received byte is not read out before a new byte is received. In this case the old byte value is kept, the new byte is rejected. Overflow also occurs if a new byte is written to register I2CD prior to transmitting the previous byte. In this case the old byte value is overwritten with the new byte. This bit is automatically cleared by hardware after I2CS is read. Reset: 0$_H$

CONFIDENTIAL

Field	Bits	Type	Description
S	4	rc	**I2C Stop Condition** This bit will be set when a stop condition is detected. It is automatically cleared by hardware after I2CS is read. Reset: 0_H
RNW	3	r	**I2C Read/Write Information** RnW reflects the state of the Read/Write bit received within the I2C address byte. 0_B I2C Write Command 1_B I2C Read Command Reset: 0_H
RACK	2	r	**I2C Acknowledge** 0_B Not acknowledged 1_B Acknowledged Reset: 0_H
TBF	1	r	**I2C Transmit Buffer Full** Bit is automatically cleared after the data byte is transferred to the shift register. 0_B Ready to accept new byte 1_B Byte transmission is pending Reset: 0_H
RBF	0	r	**I2C Receive Buffer Full** Bit is automatically cleared by hardware after I2CD is read. 0_B No new received data 1_B Received data byte is available Reset: 0_H

CONFIDENTIAL

3.12.1 Programming mode Operation

In PROGRAMMING mode the SP27 is only accessible via the I²C Interface.

The device is operating using the internal 12 MHz RC HF Oscillator.

If started up in PROGRAMMING mode (see **"Resets and Operating Mode Selection" on Page 15**), the SP27 waits until an I²C commands is received.
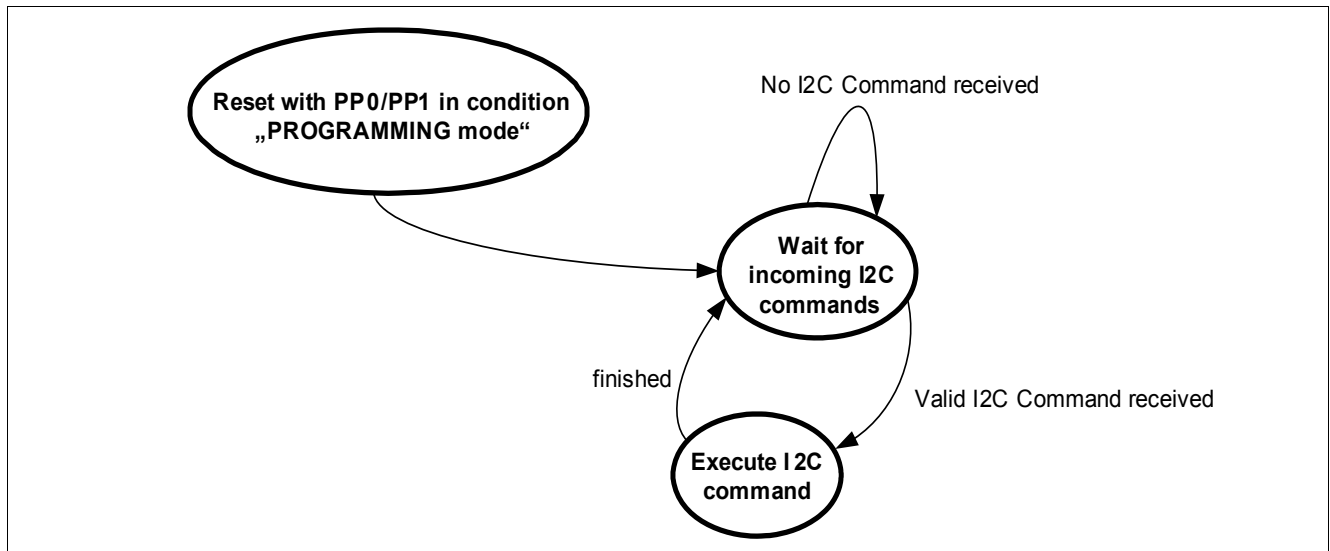
Figure 21   PROGRAMMING Mode State Diagram

*Note: To avoid programming failures all PROGRAMMING mode commands are protected by a 16 bit CRC at the end of each command (**"16 Bit CRC (Cyclic Redundancy Check) Generator/Checker" on Page 49** shows details about the used CRC polynomial).*
The checksum has to be calculated over all bytes in the command excluding the SP27 I²C device address.

PROGRAMMING Mode Commands

- **"FLASH Write Line" on Page 75**
- **"FLASH Erase" on Page 75**
- **"FLASH Check Erase Status" on Page 76**
- **"FLASH Read Line" on Page 77**
- **"Read Status" on Page 78**
- **"FLASH Check CRC" on Page 79**
- **"FLASH Set User Configuration Sector Lock" on Page 79**
- **"Measure Pressure" on Page 80**
- **"Measure Temperature" on Page 82**

Figure 22 shows the nomenclature for the I2C commands.

■	from master to slave	**S**	start condition	**nA**	not acknowledge
□	from slave to master	**P**	stop condition	**A**	acknowledge
SR	repeated start condition,may be replaced by StopStart condition				

Figure 22   I²C Legend - PROGRAMMING Mode

3.12.1.1   FLASH Write Line

The FLASH Write Line command writes 32 bytes to the FLASH. The start address has to be a multiple of 20_H.

This command should only be used if the FLASH line is fully erased. If an already programmed FLASH line gets overwritten (without being erased before) the resulting data is undefined.

Note: After the Stop condition (P) is received the data is programmed into the FLASH. During the programming time incoming I²C commands are not acknowledged. This programming time is specified in **Table 34**.

| S | 0x6C | A | AdrHi | A | AdrLo | A | Data0 | A | ... | A | Data31 | A | CRCH | A | CRCL | A | P |
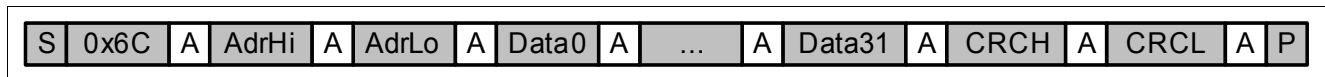
Figure 23   FLASH Write Line I2C Command

Table 12   FLASH Write Line Parameters

Parameter	Description
AdrHi	High Byte of FLASH start address
AdrLo	Low Byte of FLASH start address Must be a multiple of 20_H
Data0 - Data 31	32 Databytes that shall be written to the FLASH line indicated with AdrHi&AdrLo
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.2   FLASH Erase

The FLASH Erase command erases the individual sectors of the FLASH.

Note: After the Stop condition (P) is received the selected FLASH sectors are being erased. During the erase time incoming I²C commands are not acknowledged. This erase time is specified in **Table 34**.
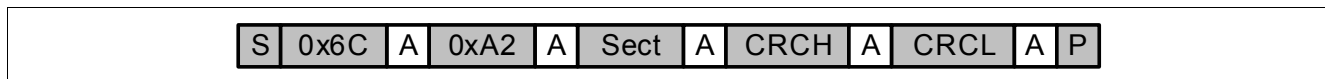
| S | 0x6C | A | 0xA2 | A | Sect | A | CRCH | A | CRCL | A | P |

Figure 24   FLASH Erase Command

Table 13   FLASH Erase Parameters

Parameter	Description
Sect	**Selects which sectors to erase** *Note: Set individual bits to '1' to erase or to '0' to NOT erase* Bit7:2 protected FLASH area -don't care Bit1 FLASH User Configuration Sector Bit0 FLASH Code Sector
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.3 FLASH Check Erase Status

This function returns the status of the selected FLASH sector(s).

Note: After the first Stop condition (P) is received the selected FLASH sectors are checked. During this time incoming I²C commands are not acknowledged.
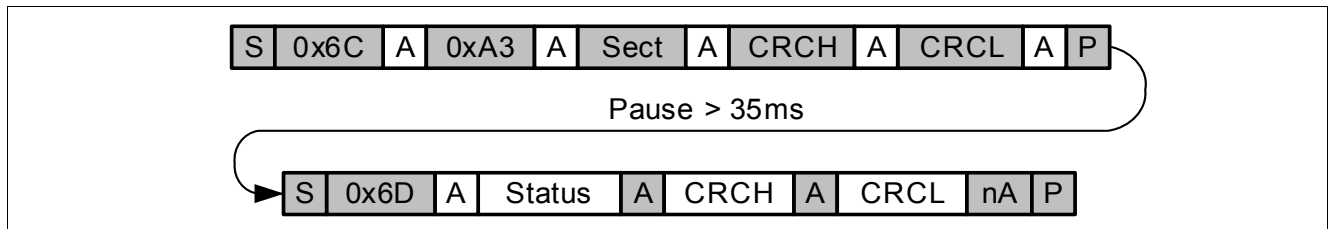

Figure 25   FLASH Check Erase Status Command

Table 14   FLASH Check Erase Parameters

Parameter	Description
Sect	**Selects which sectors to erase** *Note: Set individual bits to '1' to check or to '0' to NOT check.* Bit7:2 protected FLASH area -don't care Bit1 FLASH User Configuration Sector Bit0 FLASH Code Sector
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

Table 15   FLASH Check Erase Return values

Parameter	Description
Status	**Returns the Status of the Check** *Note: '0' means Sector is erased or unchecked '1' means that at least one bit is not erased in the checked Sector.* Bit7:2 protected FLASH area -don't care Bit1 FLASH User Configuration Sector Bit0 FLASH Code Sector
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.4 FLASH Read Line

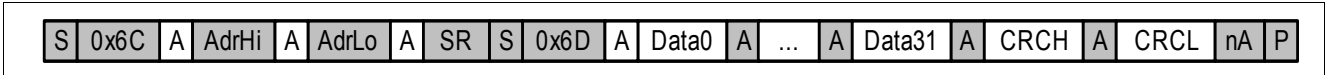The contents of the FLASH memory can be read out using the following command.

| S | 0x6C | A | AdrHi | A | AdrLo | A | SR | S | 0x6D | A | Data0 | A | ... | A | Data31 | A | CRCH | A | CRCL | nA | P |

Figure 26   FLASH Read Line Command

Table 16   FLASH Read Line Parameters

Parameter	Description
AdrHi	High Byte of FLASH start address
AdrLo	Low Byte of FLASH start address Must be a multiple of 20_H

Table 17   FLASH Read Line Return values

Parameter	Description
Data0 - Data 31	32 Databytes Read from FLASH starting at address AdrHi&AdrLo
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.5   Read Status

This function is intended to read out the status of previous I^{2}C commands. It can be called whenever desired to verify if errors occurred since the last Read Status call.
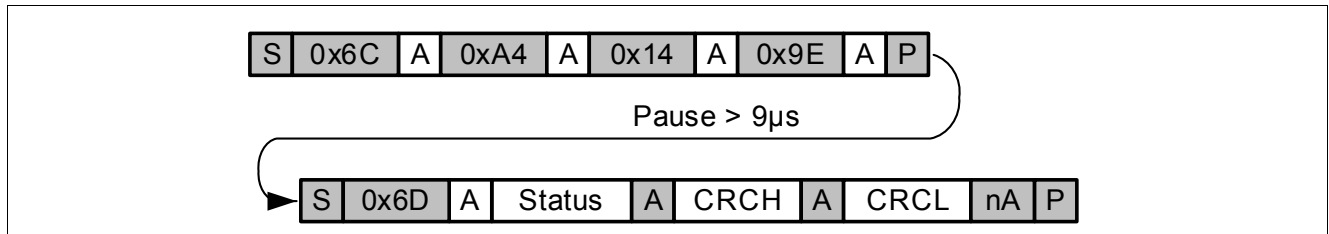

Figure 27   FLASH Read Status Command

Table 18   FLASH Read Status Return values

Parameter	Description
Status	**Status byte** Bit7:4 CmdCnt Counter indicating the number of executed commands since first detected error *Note: 0000b = no error occurred since the last call* *0001b = 1 command* *...* *1111b = 15 commands or more* Bit3:2 ErrCntInvCmdL Counter of erroneous events since last call *Note: 00b = no error* *01b = one error* *10b = two errors* *11b = three or more errors* Bit1 InvCmdL Flag indicating if there was a invalid command or execution failure since the last call. *Note: 0b = no failure found* *1b = failure found* Bit0 CRCFail CRC Failure detected since the last call *Note: 0b = no CRC Error detected since the last call* *1b = one or more CRC Error detected since last call*
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.6 FLASH Check CRC

This function performs a CRC check on the locked FLASH Code sector. The CRC-Generator/Checker is using the 16 bit CCITT polynomial $x16+x12+x5+1$ with a preload value of $FFFF_H$. The resulting CRC checksum (CRCHi, CRCLo) is compared with value stored at the FLASH addresses $577D_H$ and $577E_H$ in the FLASH Code sector. For this function to operate correctly the CRC of the entire code sector, including memory locations otherwise unused by the application software, must be considered.
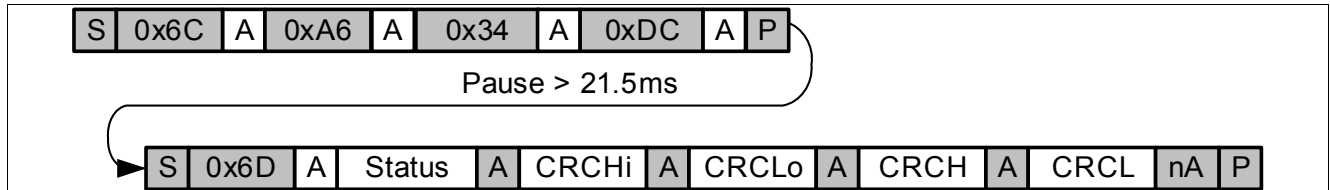
Figure 28   FLASH Check CRC

Table 19   FLASH Check CRC Return values

Parameter	Description
Status	**Status Byte** Returns the pass/fail information of the FLASH Check CRC *Note: 00_H = CRC Check passed, no ECC error detected* *FF_H = CRC Check failed* *FE_H = CRC Check passed, but ECC error detected (correctable error)*
CRCHi	High Byte of calculated CRC16 (not the CRCH Byte stored in the FLASH address $577D_H$)
CRCLo	Low Byte of calculated CRC16 (not the CRCL Byte stored in the FLASH address $577E_H$)
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.7 FLASH Set User Configuration Sector Lock

This command may be issued in order to lock the User Configuration Sector from any further writing or reading. This command causes Lockbyte 3 to become set. The structure for this command is as follows:
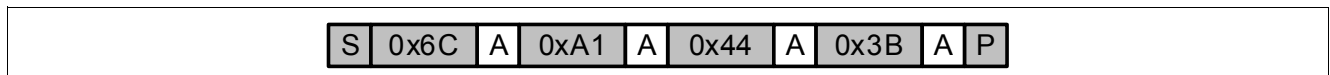

Figure 29   FLASH Set User Configuration Sector Lock

3.12.1.8 Measure Pressure

This function performs a pressure sensor measurement.

The result can either:

- Compensated for sensitivity, offset and temperature
- Output as raw value without performing the compensation

The function can measure up to 64 samples with a specified sample rate and can average them in order to compensate for noise.
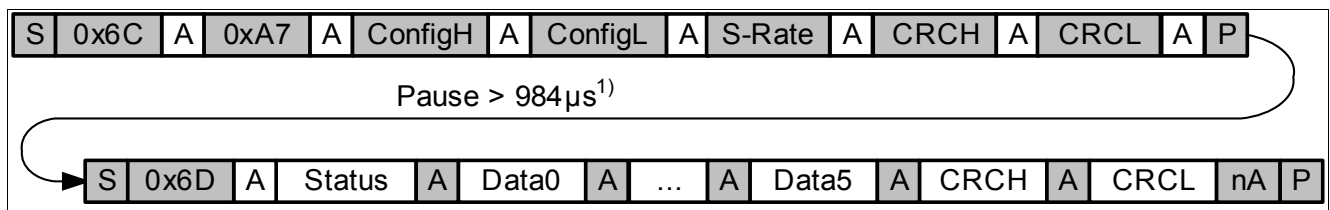
| S | 0x6C | A | 0xA7 | A | ConfigH | A | ConfigL | A | S-Rate | A | CRCH | A | CRCL | A | P |

Pause > 984 µs[1]

| S | 0x6D | A | Status | A | Data0 | A | ... | A | Data5 | A | CRCH | A | CRCL | nA | P |

Figure 30   Measure Pressure

Table 20   Measure Pressure Parameters

Parameter	Description
ConfigH	**Config Parameters** Bit7:1 Reserved, it should be set to 0_B Bit0: Selects the source of raw temperature data for compensation *Note: 0_B = Perform new raw temperature measurement* *1_B = Use raw temperature data from previous measurement*

ConfigL	**Config Parameters**
	Bit7 Select Pressure Measurement, set to 1_B
	Bit6 Defines if the RAW ADC result is compensated
	
	Note: 0_B = temperature compensation is performed. Returns Compensated & RAW value
	       1_B = no compensation is performed. Returns only RAW value
	
	Bit5:3 Reserved, should set to 0_B
	Bit2:0 Number of ADC measurements that are taken and averaged
	
	Note: 000b = 1 Samples (default)
	       001b = 2 Samples
	       010b = 4 Samples
	       011b = 8 Samples
	       100b = 16 Samples
	       101b = 32 Samples
	       110b = 64 Samples
	       111b = 64 Samples

S-Rate	Bit7:0 Number of systemclock cycles divided by 8 between two consecutive samples (only applicable if more than one sample is taken and averaged)
	
	Note: 00_H = No delay (fastest possible samplerate)
	       $01_H..4F_H$ = Not allowed
	       $50_H..FF_H$ = 1/samplerate (samplerate in systemclocks divided by 8)

CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

Table 21   Measure Pressure Return Values

Parameter	Description

Status	**Status Byte**
	Bit7 Reserved
	Bit6 Reserved
	Bit5 Reserved
	Bit4 Reserved
	Bit3 Reserved
	Bit2 Sensor Fault Wire Bond Check
	
	Note: 0_B = Wire Bond Check successful
	     1_B = Wire Bond Check Failed
	
	Bit1 Overflow of ADC Result
	
	Note: 0_B= No overflow of the ADC Result
	     1_B= Overflow of the ADC Result
	
	Bit0 Underflow of the ADC Result
	
	Note: 0_B= No underflow of the ADC Result
	     1_B= underflow of the ADC Result

Data0	Compensated Pressure High Byte
Data1	Compensated Pressure Low Byte
Data2	RAW Pressure High Byte
Data3	RAW Pressure Low Byte
Data4	RAW Temperature High Byte
Data5	RAW Temperature Low Byte
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.1.9 Measure Temperature

This function performs a temperature measurement. The result is compensated for sensitivity and offset.

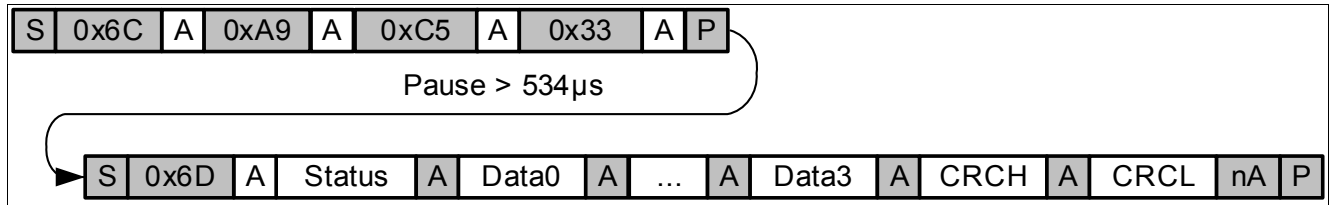**Figure 31   Measure Temperature**

Table 22   Measure Temperature Return values

| Parameter	Description
Status	**Status Byte** Bit7 Reserved Bit6 Reserved Bit5 Reserved Bit4 Reserved Bit3 Reserved Bit2 Reserved Bit1 Overflow of ADC Result *Note: 0b = No overflow of the ADC Result* *1b = Overflow of the ADC Result* Bit0 Underflow of ADC Result *Note: 0b = No underflow of the ADC Result* *1b = Underflow of the ADC Result*
Data0	Compensated Temperature High Byte
Data1	Compensated Temperature Low Byte
Data2	RAW Temperature High Byte
Data3	RAW Temperature Low Byte
CRCH	High Byte of CRC16
CRCL	Low Byte of CRC16

3.12.2 DEBUG Mode Operation

Note: DEBUG mode operation is automatically handled by the development environment provided by Infineon. Manual changes of the SFRs (see Table 23) and/or usage of the Debugger Commands (see "Debugger Commands" on Page 84) by other tools may result in undefined operation.

3.12.2.1 Debug Special Function Register

The SP27 incorporates 8 debug registers that are cleared after Wakeup and Reset.

CPU Debug Compare Register 0 (high)

DBCH0	Offset	Wakeup Value	Reset Value
CPU Debug Compare Register 0 (high)	95_H	00_H	00_H

7						0
			DBCH0			

wr
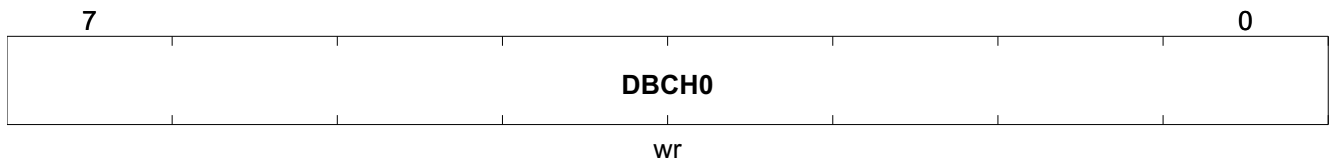
Field	Bits	Type	Description
DBCH0	7:0	wr	**CPU Debug Compare Register 0 (high)** Reset: 00_H

Table 23   DEBUG mode SFRs

Register Short Name	Register Long Name	Offset Address	Wakeup Value	Reset Value
DBCH0	Debug Compare Register 0 (highbyte)	95_H	00_H	00_H
DBCL0	Debug Compare Register 0 (lowbyte)	94_H	00_H	00_H
DBCH1	Debug Compare Register 1 (highbyte)	$9D_H$	00_H	00_H
DBCL1	Debug Compare Register 1(lowbyte)	$9C_H$	00_H	00_H
DBTH0	Debug Target Register 0 (highbyte)	97_H	00_H	00_H
DBTL0	Debug Target Register 0 (lowbyte)	96_H	00_H	00_H
DBTH1	Debug Target Register 1 (highbyte)	$9F_H$	00_H	00_H
DBTL1	Debug Target Register 1 (lowbyte)	$9E_H$	00_H	00_H

3.12.2.2   Debugging Functionality

During program execution the Program Counter (PC) of the microcontroller is continuously compared with the contents of the DBCHx + DBCLx registers.

In case of a match the PC is automatically set to the address given in DBTHx + DBTLx to handle exceptions (e.g. breakpoints).

The debugger consists of a debug handler and a single stepper. The debug handler processes the I2C communication and debug command interpretation. The debug commands **SetSFR, ReadSFR, SetMemory, ReadMemory** and **SetPC, ReadPC** are executed directly by the debug handler. The debug commands **Single Step, Run Interruptible** and **Run to next Breakpoint** are executed by the single stepper.

3.12.2.3   Debugger Commands

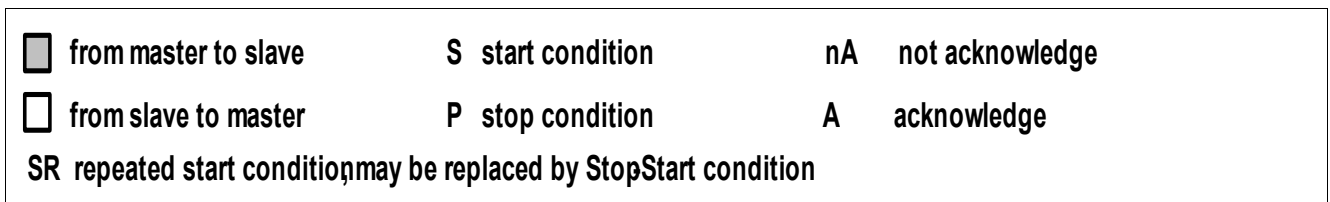The following figure shows the I^{2}C nomenclature for the commands.

▢ from master to slave	**S** start condition	**nA**	not acknowledge
☐ from slave to master	**P** stop condition	**A**	acknowledge
SR repeated start condition, may be replaced by StopStart condition			

Figure 32   I^{2}C Legend - DEBUG Mode

3.12.2.3.1   SetSFR - Set an SFR to a User-defined Value

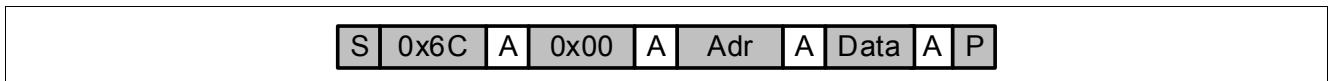| S	0x6C	A	0x00	A	Adr	A	Data	A	P

Figure 33   DEBUG SetSFR Command

Addr: represents the address of the SFR to be set.

Data: this value has to be put into the SFR address specified by Addr.

3.12.2.3.2   ReadSFR - Read the Value of One SFR

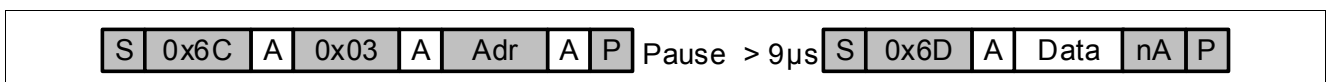| S	0x6C	A	0x03	A	Adr	A	P	Pause > 9µs	S	0x6D	A	Data	nA	P

Figure 34   DEBUG ReadSFR Command

Addr: represents the address of the SFR to be read.

Data: this value was read on the SFR address specified by Addr.

3.12.2.3.3 SetMemory - Set one Byte in RAM to a User-defined Value

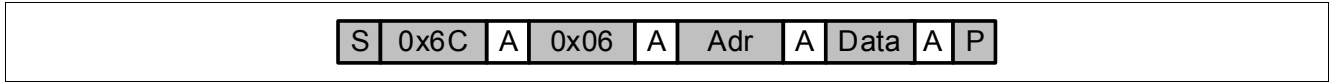| S | 0x6C | A | 0x06 | A | Adr | A | Data | A | P |

Figure 35 DEBUG SetMemory Command

Addr: represents the address of the internal data memory to be set.

Data: this value that has to be written into the internal data memory byte specified by Addr.

3.12.2.3.4 ReadMemory - Read One Byte of the RAM

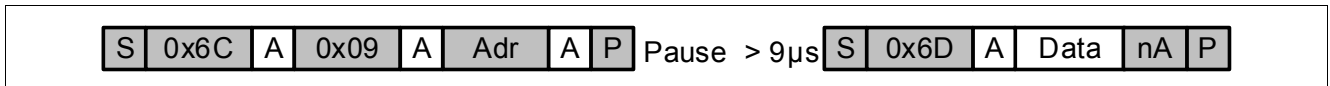| S | 0x6C | A | 0x09 | A | Adr | A | P | Pause > 9µs | S | 0x6D | A | Data | nA | P |

Figure 36 DEBUG ReadMemory Command

Addr: represents the address of the Internal data memory location to be read.

Data: this value was read from the internal data memory address specified by Addr.

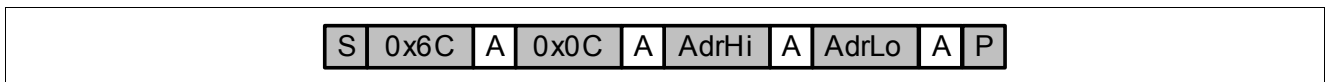### 3.12.2.3.5 SetPC - Set the Program Counter to a user-defined value

| S | 0x6C | A | 0x0C | A | AdrHi | A | AdrLo | A | P |

Figure 37 DEBUG SetPC Command

AdrHi: MSB of the new Program Counter.

AdrLo: LSB of the new Program Counter.

3.12.2.3.6 ReadPC - Read the Program Counter

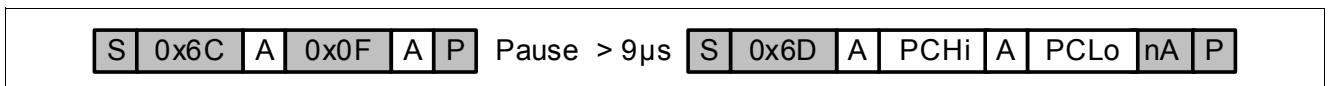| S | 0x6C | A | 0x0F | A | P | Pause > 9µs | S | 0x6D | A | PCHi | A | PCLo | nA | P |

Figure 38 DEBUG ReadPC Command

PCHi: MSB of the Program Counter.

PCLo: LSB of the Program Counter.

3.12.2.3.7 SingleStep

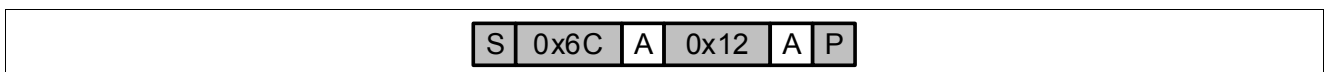Execute one Opcode Instruction and return to the debug handler.

| S | 0x6C | A | 0x12 | A | P |

Figure 39 DEBUG SingleStep Command

3.12.2.3.8   Run Interruptible

The function executes consecutive single steps until any I²C command is received on the bus. Compared to running the program in real-time this function has a slower execution speed by a factor of roughly 1/50, thus it cannot be used for debugging time critical application code.
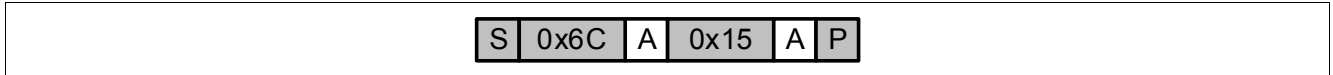
| S | 0x6C | A | 0x15 | A | P |

Figure 40   DEBUG Run Interruptible Command

3.12.2.3.9   Run to Next Breakpoint

The debugged program is executed without single steps in real-time. This enables debugging of runtime critical functions like RF transmission or LF data receiving. The execution is stopped when the PC matches one of the two hardware breakpoints. The second Breakpoint (Register 1) can be set if required using the SetSFR command.

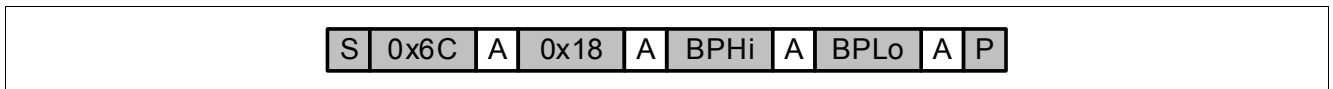*Note: If none of these breakpoints is hit the communication to the debugger is lost.*

| S | 0x6C | A | 0x18 | A | BPHi | A | BPLo | A | P |

Figure 41   DEBUG Run to Next Breakpoint Command

BPHi: MSByte of the Breakpoint Register 0.

BPLo: LSByte of the Breakpoint Register 0.

4 Specification

4.1 Absolute Maximum Ratings

Table 24   Absolute Maximum Ratings

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Max. Supply Voltage	V_{ccmax}	-0.3	–	+4.0	V		∎	
Operating Temperature	T_j	-40	–	+150	°C	Max 24 hrs accumulated over lifetime between 125°C and 150°C. Device powered V_{CC}=3.6 V	∎	1.2
		-55		-40	°C	Temperature measurement, operation of interval timer	∎	1.3
Operating temperature (transient)	T_{trans}	+150	–	+175	°C	Max 3 min., 10 times over lifetime device powered V_{CC}=3.6 V	∎	1.4
Storage temperature	$T_{storage}$	-40°C	–	+150	°C	Max 1000 hours accumulated over lifetime between 125°C and 150°C. Device not powered	∎	1.5
ESD robustness HBM	$V_{ESD,HBM}$		–	5000	V	All pins According to EIA/JESD22-A114-B	∎	1.6
ESD robustness CDM	$V_{ESD,CDM}$		–	500	V	All pins (According to ESDA STM 5.3.1)	∎	1.7
			–	750	V	Corner pins (According to ESDA STM 5.3.1)	∎	1.8
Latch up	I_{LU}	-100	–	+100	mA	AEC-Q100 (transient current)	∎	1.9
Input voltage	$V_{In,Digital}$	-0,3	–	V_{CC} + 0,3	V	Pin PP0, PP1, PP2	∎	1.10
Input and Output current (digital I/O pins)	$I_{IO,Digital}$	–	–	4	mA	Pin PP0, PP1, PP2	∎	1.13
Input Pressure range	p_{In}	0	–	2000	kPa	–	∎	1.15
		2000	–	2500	kPa	Max. 2 sec. 5 times over lifetime	∎	1.16

Attention: Test ∎ means that the parameter is not subject to production test. It was verified by design/characterization.

Attention: Stresses above the max. values listed here may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Maximum ratings are absolute ratings; exceeding only one of these values may cause irreversible damage to the integrated circuit.

4.2   Operating Range

The following operating conditions must not be exceeded in order to ensure correct operation of the SP27.

All parameters specified in the following sections refer to these operating conditions unless otherwise mentioned.

Table 25   Operating Range

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Supply Voltage	Vcc1	2.1	–	3.6	V	Measurement of pressure or temperature.	■	2.9
	Vcc2	1.9	–	3.6	V	Microcontroller, FLASH reading	■	2.10
Ambient Temperature	$T_{\text{Operating}}$	-40	–	125	°C	Normal Operation	■	2.11
	T_{Flash}	0	–	60	°C	FLASH programming/erasing	■	2.12

Attention: Test ■ means that the parameter is not subject to production test.
         It was verified by design/characterization.

4.3 Characteristics

Product characteristics involve the spread of values within the specified voltage and ambient temperature range.

Supply voltage: Vcc = 3.3 V, unless otherwise specified

Ambient temperature: Tamb = -40°C ... +125°C, unless otherwise specified

4.3.1 Pressure Sensor (1300kPa variant)

Table 26 Pressure Sensor (1300 kPa variant)[1] , V_{CC} = 3.3 V

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Minimum Input Pressure	$p_{in, min}$	–	–	100	kPa	T= -40...125°C		4.1
Maximum Input Pressure[2]	$p_{in, max}$	1300	–	–	kPa	T= -40...125°C		4.2
Input Pressure Range	$p_{in,100-500}$	100	–	500	kPa	T= -40...125°C		4.3
Measurement Error	$p_{Error,100-500}$	-21	–	+21	kPa	T= 25...80°C		4.4
		-46	–	+46	kPa	T= -40...125°C		4.5
Input Pressure Range	$p_{in,500-1300}$	500	–	1300	kPa	T= -40...125°C		4.8
Measurement Error	$p_{Error,500-1300}$	-31	–	+31	kPa	T= 25...80°C		4.9
		-60	–	+60	kPa	T= -40...125°C		4.10
RAW LSB resolution	$p_{LSB,RAW}$	0.9	–	2.0	kPa	T= -40°C	■	4.19
		1.12	–	2.3	kPa	T= 25°C	■	4.20
		1.3	–	2.75	kPa	T= 125°C	■	4.21
Pressure Measurement Stability Range	p_{sta}	-4.7		4.7	kPa	Minimum 95% of the measurements	■	4.22

1) This table is based on the average of 2 ADC samples

2) The maximum input pressure is equal or greater than 1300 kPa for +40°C< T < 125°C. For -40°C < T < + 40°C, the maximum input pressure is allowed to decrease below 1300 kPa but shall respect the following law:
Pin max >= 1300kPa + 4.15 * (T - 40°C)

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.2 Temperature Sensor

Table 27 Temperature Sensor[1], V$_{CC}$ = 3.3 V

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	**Typ.**	**Max.**				
Measurement Error	T_{Error}	-3	–	+3	°C	T= -20...70°C		6.1
		-5	–	+5	°C	T= -40..-20°C T= 70..125°C		6.2
Temperature Measurement Stability Range	T_{sta}	-1		+1	°C	Minimum 95% of the measurements	■	6.3

1) This table is based on the average of 2 ADC samples

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.3 Supply Currents

Table 28 Supply Currents

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
POWER DOWN current RAM lower memory block powered down SFR CFG2.PDLMB:1$_b$	$I_{PD,3.0V,25°C}$	–	–	700	nA	V_{CC}= 3.0 V, T= 25°C		8.13
	$I_{PD,3.0V,90°C}$	–	–	3.5	µA	V_{CC}= 3.0 V, T= 90°C	■	8.14
	$I_{PD,3.0V,125°C}$	–	–	19.5	µA	V_{CC}= 3.0 V, T= 125°C		8.15
POWER DOWN current RAM lower memory block kept powered SFR CFG2.PDLMB:0$_b$	$I_{PD\_RAMen,3.0V,25°C}$	–	–	750	nA	V_{CC}= 3.0 V, T= 25°C	■	8.16
	$I_{PD\_RAMen,3.0V,90°C}$	–	–	3.9	µA	V_{CC}= 3.0 V, T= 90°C	■	8.17
	$I_{PD\_RAMen,3.0V,125°C}$	–	–	21	µA	V_{CC}= 3.0 V, T= 125°C	■	8.18
IDLE current SFR DIVIC: 00$_B$ Timer0 active	$I_{Idle,3.0V,25°C}$	–	–	1	mA	V_{CC}= 3.0 V, T= 25°C		8.21
	$I_{Idle,3.0V,125°C}$	–	–	1.1	mA	V_{CC}= 3.0 V, T= 125°C	■	8.22
RUN current (Peripheral units in active state) SFR DIVIC: 00$_B$	$I_{Run,3.0V,25°C}$	–	–	2.1	mA	V_{CC}= 3.0 V, T= 25°C		8.23
	$I_{Run,3.0V,125°C}$	–	–	2.4	mA	V_{CC}= 3.0 V, T= 125°C	■	8.24

Attention: Test ■ means that the parameter is not subject to production test. It was verified by design/characterization.

4.3.4 12 MHz RC HF Oscillator

Table 29   12 MHz RC HF Oscillator

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Operating frequency	$f_{RC,HF}$	11.04	12.00	12.96	MHz			13.1

4.3.5 2 kHz RC LP Oscillator

Table 30   2 kHz RC LP Oscillator

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Operating frequency	$f_{RC,LP}$	1.3	2	2.7	kHz	V_{CC}= 3.0 V, T= 25°C		14.1
Frequency drift	$df_{RC,LP}$	-7	–	+7	%	–		14.2

4.3.6 Interval Timer

Table 31   Interval Timer

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	**Typ.**	**Max.**				
Interval Timer Precounter calibration error	df_{ITP}	-0.083	–	0	%/Hz	Error is dependent upon Interval Timer timebase specified to "Interval Timer Calibration" ROM Library function. This error does not include reference clock (12 MHz RC oscillator) and 2 kHz RC LP oscillator drift errors.	■	15.1

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.7   Voltage Regulator

Table 32   Voltage Regulator[1]

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Regulated output voltage in RUN state	V_{REG}	1.9	2.1	2.4	V	Vcc= 2.1 V - 3.6 V		17.1
Regulated output voltage in Programming mode	V_{REG}	2.3	2.5	2.75	V	V_{bat}= 2.5 V - 3.6 V		17.4
Regulated output voltage in POWERDOWN	$V_{REG,PD}$	1.7	–	2.5	V	–		17.5
External Capacitance at Vreg Pin	C_{VREG}	60	100	–	nF	Maximum ESR 15 Ohm	■	17.6

1) The voltage regulator is designed to supply only the internal blocks of the SP27 and not designed to drive any external circuitry, thus only the decoupling cap is allowed to be connected to pin V_{reg}. A 100nF decoupling cap with an maximum ESR of 15 Ohm is recommended for proper operation.

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.8 Power On Reset / Brown Out Reset

Table 33   Power On Reset / Brown Out Reset

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Power On Reset level	V_{POR}	0.2	0.4	1.7	V	Min. supply voltage level measured at Pin V_{REG} for a valid logic LOW at Power On Reset circuit	■	16.1
Power On Reset release level	V_{THR}	1.7	–	1.8	V	Measured at Pin V_{REG}		16.3
Power On reset time	t_{POR}	0.25	–	10	ms	–		16.4
Brown Out detect level in RUN state	$V_{RUN,BRD}$	1.7	–	1.8	V	Measured at Pin V_{REG}		16.5
Brown Out detect level in POWERDOWN	$V_{PD,BRD}$	0.7	–	1.7	V	Measured at Pin V_{REG}		16.8
Mode selection time	t_{mode}	–	–	2.5	ms	–	■	16.10
Minimum detectable Brown Out glitch in RUN state	$t_{Brownout,RUM}$	–	–	1	µs	–	■	16.11
Minimum detectable Brown Out glitch in POWERDOWN	$t_{Brownout,PD}$	–	–	100	µs	–	■	16.12

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.9 FLASH Memory

Table 34   FLASH Memory

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	**Typ.**	**Max.**				
Erase/Program temperature	T_{FL}	0	–	+60	°C	–	■	19.1
Erase/Program supply voltage	Vcc3	2.5	–	3.6	V	–	■	19.2
Endurance	En_{Flash}	100	–	–	cycles	Programming/erase cycles per wordline	■	19.3
Erase time	t_{Erase}	–	102	–	ms	Min/max can be derived by adding the tolerance and drift of the 12 MHz RC HF Osc. (see **Table 29**)	■	19.4
Write time per FLASH line	$t_{Program}$	–	2.2	–	ms	FLASH line = 32 Byte min/max can be derived by adding the tolerance and drift of the 12 MHz RC HF Osc. (see **Table 29**)	■	19.5

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.10   Watchdog Timer

Table 35   Watchdog Timer

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Watchdog timeout	$t_{Watchdog}$	0.6	1	1.7	s	Min/max is derived by considering the tolerance and drift of the 2 kHz RC LP Osc. (see **Table 30**)	■	21.1

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.11   Digital I/O pins

Table 36   Digital I/O pins

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
Input LOW voltage	V_{IL}	-0.2	–	0.4	V	–		22.1
Input HIGH voltage	V_{IH}	Vcc-0.4	–	Vcc+0.2	V	–		22.2
Output LOW voltage	V_{OL}	–	–	0.5	V	IOL= 1.6 mA		22.3
Output HIGH voltage	V_{OH}	Vcc-0.5	–	–	V	IOH=-1.6mA		22.4
Output transition time	$t_{HL, LH}$	–	–	30	ns	20 pF load, 10% ... 90%	■	22.5
Parasitic capacitance	C_{pad}	–	–	5	pF	–	■	22.6
Internal pullup / pulldown resistor	$R_{up/down}$	35	50	70	kOhm	Pin PP0, PP1		22.7
	$R_{up/down}$	175	250	335	kOhm	Pin PP2		22.8

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.

4.3.12   I²C Interface

Table 37   I²C Interface

Parameter	Symbol	Values			Unit	Note / Test Condition	Test	Number
		Min.	Typ.	Max.				
I²C bitrate	DR_{I2C}	–	–	400	kbit/s	SFR DIVIC[1:0]: 00$_B$	■	23.1

Attention: Test ■ means that the parameter is not subject to production test.
It was verified by design/characterization.
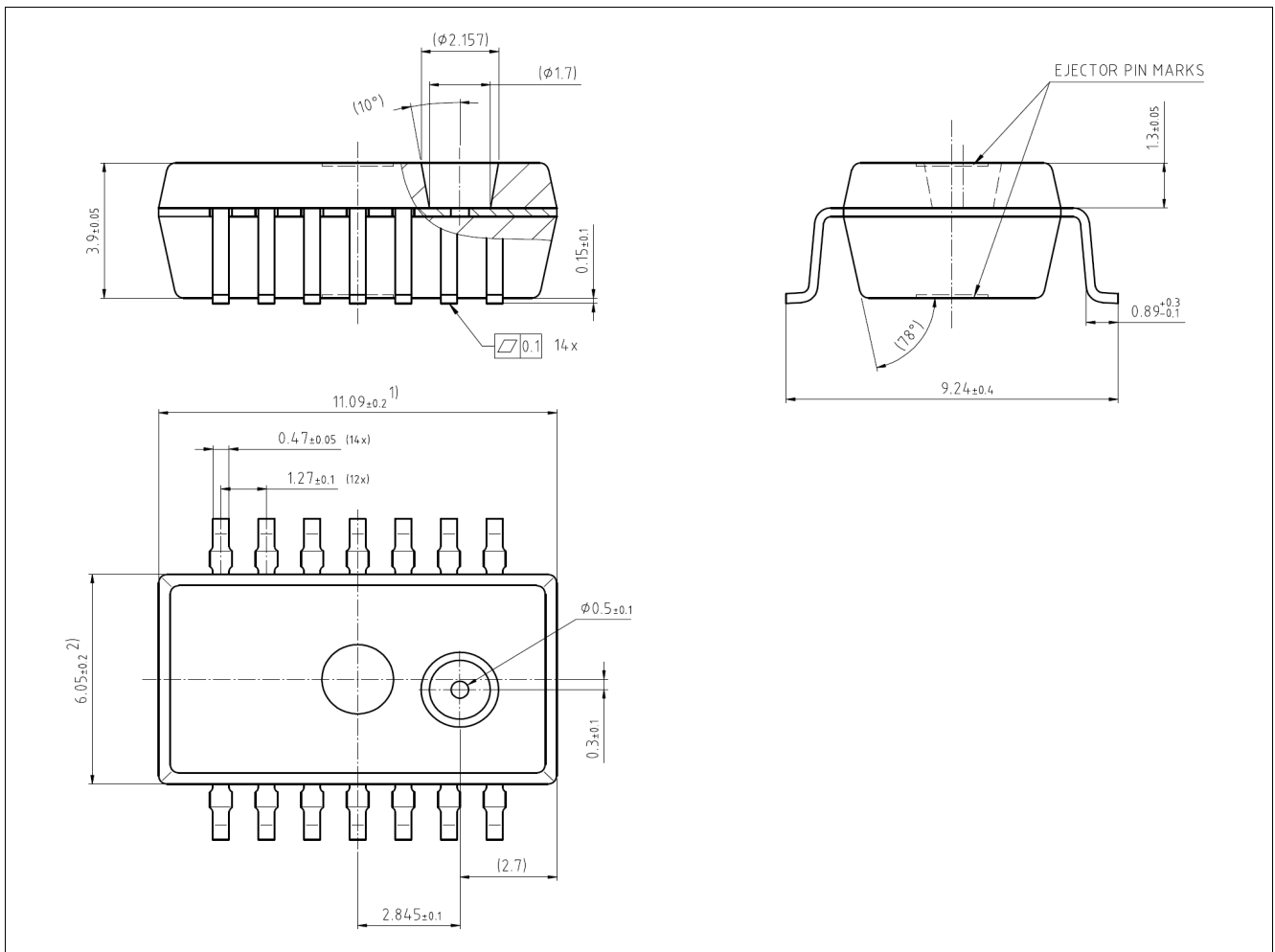
5 Package Information

5.1 Package Outline

Figure 42   PG-DSOSP-14-6 Package Dimensions

Dimensions in [mm][1)2)]

---

1) Dimension does not include mold flash, protrusions or gate burrs. Mold flash, protrusions and gate burrs do not exceed 0.15mm (0.006 inch) per side.

2) Dimension does not include inter-lead flash or protrusions. Inter-lead flash and protrusions do not exceed 0.25mm (0.010 inch) per side.
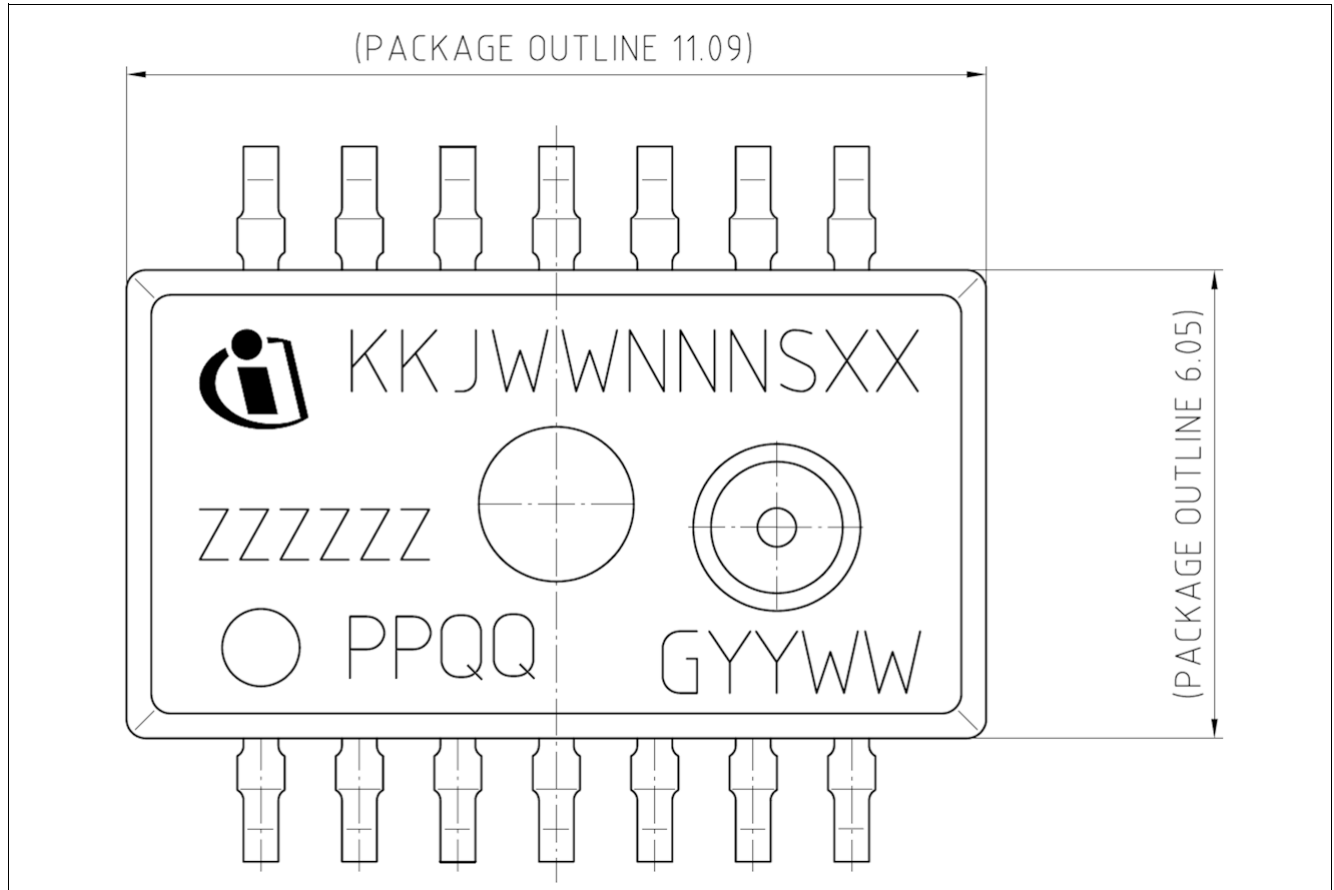
5.2   Identification Code

Figure 43   PG-DSOSP-14-6 marking

The marking for the SP27 is on the topside of the package.

5.2.1   Identification Code Definition

KKJWWNNNSXX: Infineon Lot Number

ZZZZZ: Product Identification (SP270)

O: Pin 1 Marking

PPQQ: Manufacturer Version (PP = 25 Villach Sensor) and Sensor Information Code (QQ = 25)

G: Green Package Indicator[1]

YYWW: Date Code (YY = Year, WW = week)

---

1)   The Green Package fulfils the solder condition for Pb-Free Assembly according to IPC/JEDEC J-STD-020C.

References

This section contains documents used for cross-reference throughout this document.

[1]   SP27 ROM Library function guide

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Board Mount Pressure Sensors *category:*

Click to view products by Infineon *manufacturer:*

Other Similar products are found below :

6407-249V-17343P 6407-250V-09273P 80527-25.0H2-05 80541-B00000150-01 80541-B00000200-05 80554-00700100-05 80568-00300050-01 93.631.4253.0 93.731.4353.0 93.932.4553.0 136PC150G2 136PC15A1 142PC95AW71 142PC05DW70 15PSI-G-4V 1805-01A-L0N-B 26PCBKT 26PCCFA6D26 26PCCFS2G 26PCCVA6D 93.632.7353.0 93.731.3653.0 93.931.4853.0 93.932.4853.0 SCDA120-XSC05DC 185PC30DH 20INCH-G-MV-MINI 26PCAFJ3G 26PCCEP5G24 26PCDFA3G 26PCJEU5G19 ASCX15AN-90 TSCSAAN001PDUCV DCAL401DN DCAL401GN XZ202798SSC XZ203676HSC 6407-249V-09343P 6407-250V-17343P SP370-25-116-0 81794-B00001200-01 HSCDLNN100PGAA5 82681-B00000100-01 81618-B00000040-05 SSCDJNN015PAAA5 TSCDLNN100MDUCV TSCSAAN100PDUCV NBPDANN015PGUNV NBPLLNS150PGUNV 142PC100D