PCA9629

Fm+ I²C-bus stepper motor controller

Rev. 1 — 29 February 2012 Product data sheet

1. General description

The PCA9629 is an I²C-bus controlled low-power CMOS device that provides all the logic and control required to drive a four phase stepper motor. PCA9629 is intended to be used with external high current drivers to drive the motor coils. The PCA9629 supports three stepper motor drive formats: one-phase (wave drive), two-phase, and half-step. In addition, when used as inputs, four General Purpose Input/Outputs (GPIOs) allow sensing of logic level output from optical interrupter modules and generate active LOW interrupt signal on the $\overline{\text{INT}}$ pin of PCA9629. This is a useful feature in sensing home position of motor shaft or reference for step pulses. Upon interrupt, the PCA9629 can be programmed to automatically stop the motor or reverse the direction of rotation of motor.

Output wave train is programmable using control registers. The control registers are programmed via the I²C-bus. Features built into the PCA9629 provide highly flexible control of stepper motor, off-load bus master/micro and significantly reduce I²C-bus traffic. These include control of step size, number of steps per single command, number of full rotations and direction of rotation. A ramp-up on start and/or ramp-down on stop is also provided.

The PCA9629 is available in a 16-pin TSSOP package and is specified over the −40 °C to +85 °C industrial temperature range.

2. Features and benefits

- Generate motor coil drive phase sequence signals with four outputs for use with external high current drivers to off-load CPU
- Four balanced push-pull type outputs capable of sinking 25 mA or sourcing 25 mA for glueless connection to external high current drivers needed to drive motor coils
  - Up to 1000 pF loads with 100 ns rise and fall times
- Built-in oscillator requires no external components
- Stepper motor drive control logic
- One-phase (wave drive), two-phase, and half-step drive format logic level outputs
- Programmable step rate: 344.8 kpps to 0.3 pps with ±5 % accuracy
- Programmable ramp-up on start and ramp-down to stop
- Programmable steps and rotation control
- Sensor enabled drive control: linked to interrupt from I/O pins
- Direction control of motor shaft
- Selectable active hold, power off or released states for motor shaft

- Four general purpose I/Os:
  - Configured to sense logic level outputs from optical interrupter photo transistor circuit
  - Configured as outputs to drive (source/sink) LEDs or other loads up to 25 mA
  - Programmable interrupt Mask Control for input pins
- 4.5 V to 5.5 V operation
- 1 MHz Fast-mode Plus (Fm+) I²C-bus serial interface with 30 mA high drive capability on SDA output for driving high capacitive buses
- Compliant with I²C-bus Standard-mode (100 kHz) and Fast-mode (400 kHz) speeds
- Active LOW open-drain interrupt output
- Active LOW reset ($\overline{\text{RESET}}$) input pin resets device to power-up default state: can be used to recover from bus stuck condition
- Programmable watchdog timer
- All Call address allows programming of more than one device at the same time with the same parameters
- 16 programmable slave addresses using two address pins
- –40 °C to +85 °C operation
- ESD protection exceeds 2000 V HBM per JESD22-A114 and 1000 V CDM per JESD22C101
- Latch-up testing is done to JEDEC Standard JESD78 which exceeds 100 mA
- Package offered: TSSOP16

3. Applications

- Amusement machines
- Gaming and slot machines
- Consumer home appliances or toys
- Industrial automation
- HVAC and building climate control systems
- Robotics

4. Ordering information

Table 1. Ordering information

Type number	Package		
	Name	Description	Version
PCA9629PW	TSSOP16	plastic thin shrink small outline package; 16 leads; body width 4.4 mm	SOT403-1

4.1 Ordering options

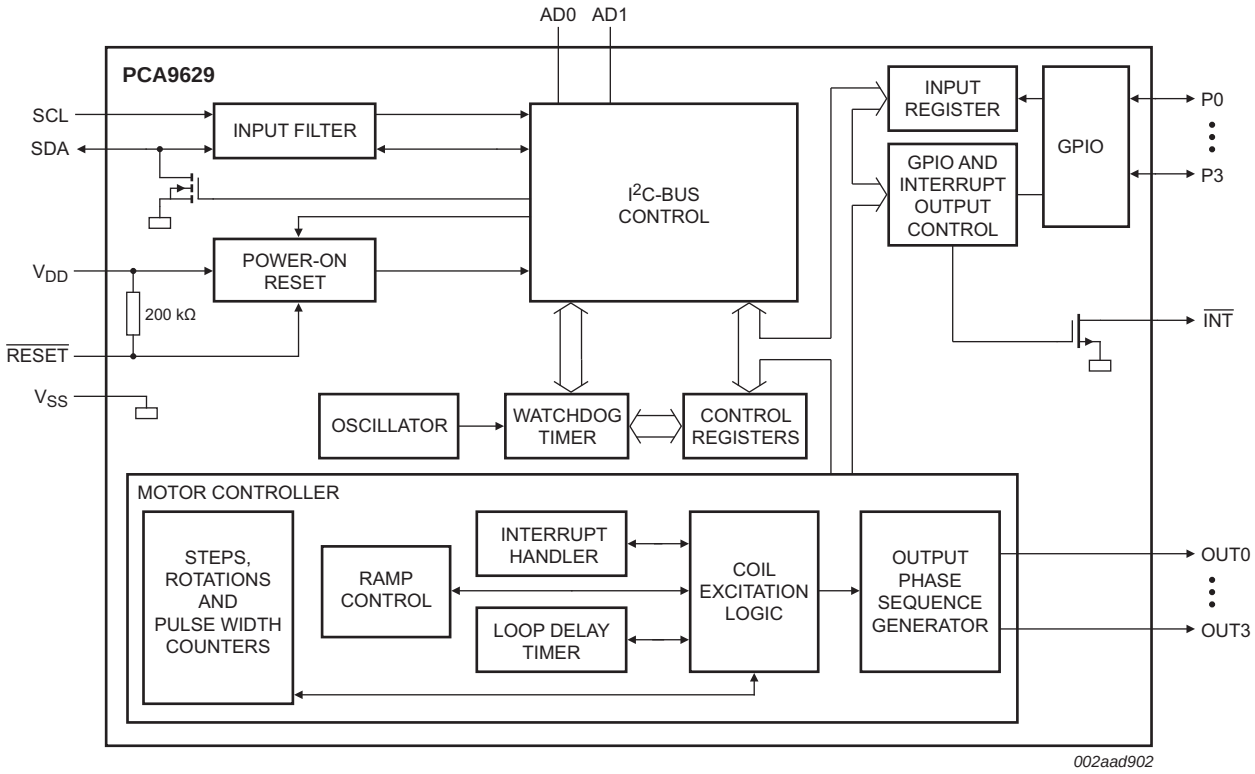
Table 2. Ordering options

Type number	Topside mark	Temperature range
PCA9629PW	PCA9629	T_{amb} = –40 °C to +85 °C

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

5. Block diagram



Remark: All I/Os are set to inputs at power-up and reset.

Fig 1. PCA9629 block diagram

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

6. Pinning information

6.1 Pinning

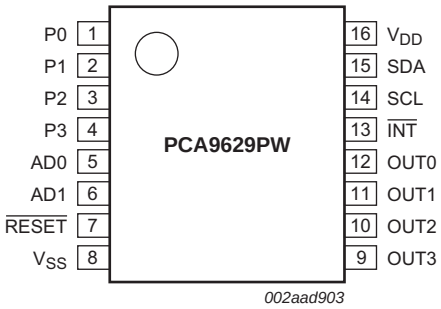


Fig 2. Pin configuration for TSSOP16

6.2 Pin description

Table 3. Pin description

Symbol	Pin	Type	Description
P0	1	I/O	input/output 0 (output is 25 mA push-pull)
P1	2	I/O	input/output 1 (output is 25 mA push-pull)
P2	3	I/O	input/output 2 (output is 25 mA push-pull)
P3	4	I/O	input/output 3 (output is 25 mA push-pull)
AD0	5	I	address input 0
AD1	6	I	address input 1
$\overline{\text{RESET}}$	7	I	active LOW reset input with 1 μs filter
V_{SS}	8	ground	supply ground
OUT3	9	O	control 25 mA push-pull output 3
OUT2	10	O	control 25 mA push-pull output 2
OUT1	11	O	control 25 mA push-pull output 1
OUT0	12	O	control 25 mA push-pull output 0
$\overline{\text{INT}}$	13	O	active LOW interrupt output; open-drain
SCL	14	I	serial clock line
SDA	15	I/O	serial data line; open-drain capable of sinking 30 mA
V_{DD}	16	power supply	supply voltage

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7. Functional description

Refer to Figure 1 “PCA9629 block diagram”.

7.1 Device address

Following a START condition, the bus master must send the target slave address followed by a read or write operation. The slave address of the PCA9629 is shown in Figure 3. Slave address pins AD1 and AD0 choose one of 16 slave addresses. To conserve power, no internal pull-up resistors are incorporated on AD1 and AD0. Table 4 shows all 16 slave addresses by connecting the AD0 and AD1 to V_{DD}, V_{SS}, SCL or SDA.

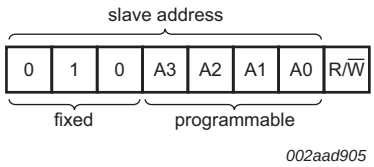


Fig 3. PCA9629 device address

The last bit of the first byte defines the reading from or writing to the PCA9629. When set to logic 1 a read is selected, while logic 0 selects a write operation.

Table 4. PCA9629 address map

AD1	AD0	Device family high-order address bits			Variable portion of address				Address
		A6	A5	A4	A3	A2	A1	A0	
V_{SS}	V_{SS}	0	1	0	0	0	0	0	40h
V_{SS}	V_{DD}	0	1	0	0	0	0	1	42h
V_{DD}	V_{SS}	0	1	0	0	0	1	0	44h
V_{DD}	V_{DD}	0	1	0	0	0	1	1	46h
V_{SS}	SCL	0	1	0	0	1	0	0	48h
V_{SS}	SDA	0	1	0	0	1	0	1	4Ah
V_{DD}	SCL	0	1	0	0	1	1	0	4Ch
V_{DD}	SDA	0	1	0	0	1	1	1	4Eh
SCL	V_{SS}	0	1	0	1	0	0	0	50h
SDA	V_{SS}	0	1	0	1	0	0	1	52h
SCL	V_{DD}	0	1	0	1	0	1	0	54h
SDA	V_{DD}	0	1	0	1	0	1	1	56h
SCL	SCL	0	1	0	1	1	0	0	58h
SCL	SDA	0	1	0	1	1	0	1	5Ah
SDA	SCL	0	1	0	1	1	1	0	5Ch
SDA	SDA	0	1	0	1	1	1	1	5Eh

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.2 Command register

Following the successful acknowledgement of the slave address and a write bit, the bus master sends a byte to the PCA9629. This byte is stored in the Command register.

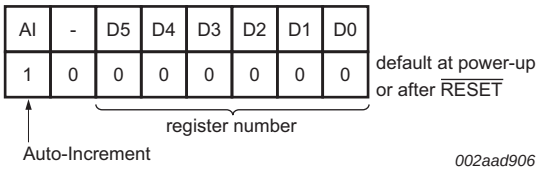


Fig 4. Command register

At power-up, the Command register defaults to 80h, with the AI bit set to '1' and the lowest seven bits set to '0'. The lowest six bits are used as a pointer to determine which register will be accessed. Only a command register code with the six least significant bits equal to the 39 allowable values as defined in Table 5 "Register summary" are acknowledged. Reserved or undefined command codes are not acknowledged.

The most significant bit of the Command register is for Auto-Increment. If the Auto-Increment flag is set, the six low-order bits of the Control register are automatically incremented after a read or write. This allows the user to program the registers sequentially. The contents of these bits will roll over to '00 0000' after the last register (address = 26h) is accessed. Only the six least significant bits are affected by the AI flag. Unused bits must be programmed with zeroes.

7.3 Register definitions

Table 5. Register summary

Register number	D5	D4	D3	D2	D1	D0	Name	Type	Function
00h	0	0	0	0	0	0	MODE	read/write	Mode register
01h	0	0	0	0	0	1	SUBADR1	read/write	I²C-bus subaddress 1
02h	0	0	0	0	1	0	SUBADR2	read/write	I²C-bus subaddress 2
03h	0	0	0	0	1	1	SUBADR3	read/write	I²C-bus subaddress 3
04h	0	0	0	1	0	0	ALLCALLADR	read/write	All Call I²C-bus address
05h	0	0	0	1	0	1	WDTOI	read/write	Watchdog time-out interval register
06h	0	0	0	1	1	0	WDCNTL	read/write	Watchdog control register
07h	0	0	0	1	1	1	IP	read only	Input Port register
08h	0	0	1	0	0	0	INTSTAT	read only	Interrupt status register
09h	0	0	1	0	0	1	OP	read/write	Output Port register
0Ah	0	0	1	0	1	0	IOC	read/write	I/O Configuration register
0Bh	0	0	1	0	1	1	MSK	read/write	Mask interrupt register
0Ch	0	0	1	1	0	0	CLRINT	write only	Clear interrupts
0Dh	0	0	1	1	0	1	INTMODE	read/write	Interrupt mode register
0Eh	0	0	1	1	1	0	INT_ACT_SETUP	read/write	Interrupt action setup control register
0Fh	0	0	1	1	1	1	INT_MTR_SETUP	read/write	Interrupt motor setup control register

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Table 5. Register summary *...continued*

Register number	D5	D4	D3	D2	D1	D0	Name	Type	Function
10h	0	1	0	0	0	0	INT_ES_SETUP	read/write	Interrupt extra steps setup control register
11h	0	1	0	0	0	1	INT_AUTO_CLR	read/write	Interrupt auto clear control register
12h	0	1	0	0	1	0	SETMODE	read/write	Output state on STOP
13h	0	1	0	0	1	1	PHCNTL	read/write	Phase control register
14h	0	1	0	1	0	0	SROTNL	read/write	Steps per rotation low byte
15h	0	1	0	1	0	1	SROTNH	read/write	Steps per rotation high byte
16h	0	1	0	1	1	0	CWPWL	read/write	Step pulse width for CW rotation low byte
17h	0	1	0	1	1	1	CWPWH	read/write	Step pulse width for CW rotation high byte
18h	0	1	1	0	0	0	CCWPWL	read/write	Step pulse width for CCW rotation low byte
19h	0	1	1	0	0	1	CCWPWH	read/write	Step pulse width for CCW rotation high byte
1Ah	0	1	1	0	1	0	CWSCOUNTL	read/write	Number of steps CW low byte
1Bh	0	1	1	0	1	1	CWSCOUNTH	read/write	Number of steps CW high byte
1Ch	0	1	1	1	0	0	CCWSCOUNTL	read/write	Number of steps CCW low byte
1Dh	0	1	1	1	0	1	CCWSCOUNTH	read/write	Number of steps CCW high byte
1Eh	0	1	1	1	1	0	CWRCOUNTL	read/write	Number of rotations CW low byte
1Fh	0	1	1	1	1	1	CWRCOUNTH	read/write	Number of rotations CW high byte
20h	1	0	0	0	0	0	CCWRCOUNTL	read/write	Number of rotations CCW low byte
21h	1	0	0	0	0	1	CCWRCOUNTH	read/write	Number of rotations CCW high byte
22h	1	0	0	0	1	0	EXTRASTEPS0	read/write	Count value for extra steps or rotations for INTP0
23h	1	0	0	0	1	1	EXTRASTEPS1	read/write	Count value for extra steps or rotations for INTP1
24h	1	0	0	1	0	0	RMPCNTL	read/write	Ramp control register
25h	1	0	0	1	0	1	LOOPDLY	read/write	Loop delay time register
26h	1	0	0	1	1	0	MCNTL	read/write	Control start/stop motor
27h to FFh	-	-	-	-	-	-	-	-	Reserved

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.1 MODE — Mode register

Table 6. MODE - Mode register (address 00h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
00h	MODE	7	-	0*	not used
		6	-	0*	not used
		5	R/W	1	Disable $\overline{\text{INT}}$ output pin
				0*	Enable $\overline{\text{INT}}$ output pin
		4	R/W	1	outputs change on I²C-bus ACK
				0*	outputs change on I²C-bus STOP command
		3	R/W	1	PCA9629 responds to I²C-bus subaddress 1
				0*	PCA9629 does not respond to I²C-bus subaddress 1
		2	R/W	1	PCA9629 responds to I²C-bus subaddress 2
				0*	PCA9629 does not respond to I²C-bus subaddress 2
		1	R/W	1	PCA9629 responds to I²C-bus subaddress 3
				0*	PCA9629 does not respond to I²C-bus subaddress 3
		0	R/W	1*	PCA9629 responds to All Call I²C-bus address
				0	PCA9629 does not respond to All Call I²C-bus address

7.3.1.1 Disable interrupt output pin (bit 5)

This feature is useful when the host/micro/master does not want the $\overline{\text{INT}}$ pin to toggle when interrupts occur. Within PCA9629, when interrupts are enabled and interrupt event occurs, the actions related to the interrupt event are still carried out. However, if bit 5 = 1, the $\overline{\text{INT}}$ pin does not show the activation of interrupt because the pin is disabled. If bit 5 = 0, the micro sees the actual status of the $\overline{\text{INT}}$ pin.

The only exception to this rule is when the watchdog timer is enabled in the ‘Interrupt and Reset’ mode (see Section 7.3.4.2). In this case, the interrupt line toggles when the watchdog timer times out (even though bit 5 of this register is a ‘1’). This is because in the ‘Interrupt and Reset mode’ the part gets reset (and hence bit 5 is cleared) when the timer times out.

7.3.1.2 Outputs change on STOP (bit 4)

This feature can be used to synchronize the starting of the motor across multiple PCA9629 devices on the bus at approximately the same time (within few microseconds of one another). The host controller can program all the PCA9629s on the bus and then issue the I²C-bus STOP command. Upon receiving the STOP command, all the PCA9629 devices on the bus start generating pulse sequences required to turn the motor. This feature is applicable only to the motor coil outputs of the device namely, OUT0 to OUT3. It is **not** applicable to the general purpose I/Os (P0 to P3).

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.2 SUBADR1 to SUBADR3 — I²C-bus subaddress 1 to 3

Table 7. SUBADR1 to SUBADR3 - I²C-bus subaddress registers 1 to 3 (addresses 01h, 02h 03h) bit description

*Legend: * default value.*

Address	Register	Bit	Symbol	Access	Value	Description
01h	SUBADR1	7:1	A1[7:1]	R/W	1110 001*	I²C-bus subaddress 1
		0	A1[0]	R only	0*	reserved
02h	SUBADR2	7:1	A2[7:1]	R/W	1110 010*	I²C-bus subaddress 2
		0	A2[0]	R only	0*	reserved
03h	SUBADR3	7:1	A3[7:1]	R/W	1110 100*	I²C-bus subaddress 3
		0	A3[0]	R only	0*	reserved

Subaddresses are programmable through the I²C-bus. Default power-up values are E2h, E4h, E8h, and the device(s) will not acknowledge these addresses right after power-up (the corresponding bits [3:1] in MODE register is equal to 0).

Once subaddresses have been programmed to their right values, bits [3:1] (MODE register) must be set to logic 1 in order to have the device acknowledging these addresses. Only the seven MSBs representing the I²C-bus subaddress are valid. The LSB in SUBADRx register is a read-only bit (0). When subaddress control bits [3:1] in MODE register is set to logic 1, the corresponding I²C-bus subaddress can be used during either an I²C-bus read or write sequence.

7.3.3 ALLCALLADR — All Call I²C-bus address

Table 8. ALLCALLADR - All Call I²C-bus address register (address 04h) bit description

*Legend: * default value.*

Address	Register	Bit	Symbol	Access	Value	Description
04h	ALLCALLADR	7:1	AC[7:1]	R/W	1110 000*	ALLCALL I²C-bus address register
		0	AC[0]	R only	0*	reserved

The All Call I²C-bus address allows all the PCA9629s on the bus to be programmed at the same time (bit 0 in register MODE must be equal to 1 (power-up default state)). This address is programmable through the I²C-bus and can be used during either an I²C-bus read or write sequence. Only the seven MSBs representing the All Call I²C-bus address are valid. The LSB in ALLCALLADR register is a read-only bit (0). If bit 0 in MODE register = 0, the device does not acknowledge the address programmed in register ALLCALLADR.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.4 Watchdog timer

The purpose of the watchdog timer is to recover the PCA9629 if the system it is used in enters an erroneous state. When the timer times out, the watchdog generates an interrupt to the host controller and, if programmed for reset, resets PCA9629 if the user program fails to 'feed' the watchdog. To feed the watchdog, the user simply addresses the PCA9629 ([START + slave address + START] or [START + slave address + STOP]) within the watchdog time-out interval. Only this sequence resets the watchdog.

Watchdog timer features:

- Can be programmed to reset the PCA9629 to POR state if it is not periodically addressed
- Enabled by software, but requires a hardware reset or a watchdog reset to be disabled
- Flag to indicate watchdog reset
- Programmable 8-bit timer with internal prescaler
- Selectable time period from one second to 255 seconds

The watchdog timer should be used in the following manner:

- Set the time-out interval value in WDTOI register
- Set the mode of operation (interrupt only or interrupt and reset) and enable the watchdog using the WDCNTL register
- Watchdog should be fed by periodically addressing PCA9629 before the watchdog timer underflows to prevent reset/interrupt
- Watchdog control register, WDCNTL, can be read at any time to determine the status of the watchdog operation

7.3.4.1 WDTOI — WatchDog Time-Out Interval register

The watchdog time-out interval should be programmed in this register. The default value is FFh, which indicates a 255 second time-out interval. The smallest value for the time-out interval is 01h, which indicates a one-second time-out interval. Watchdog operation cannot be enabled with a zero second time-out interval. If user writes a zero value to this register, the timer does not start.

Table 9. WDTOI - Watchdog time-out interval register (address 05h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
05h	WDTOI	7:0	R/W	FFh*	Watchdog time-out interval

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.4.2 WDCNTL — WatchDog Control register

Table 10. WDMOD - Watchdog control register (address 06h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
06h	WDCNTL	7:5	read only	000*	Reserved.
		4	write only	1	Clear WDINT flag.
				0*	Read value.
		3	read only	1	WDINT: watchdog interrupt flag set.[1]
				0*	WDINT: watchdog interrupt flag not set.
		2	read only	1	WDRST: watchdog reset flag.[2]
				0*	WDRST: watchdog reset flag not set.
		1	R/W	1	WDMOD: watchdog interrupt and reset mode (set only).
				0*	WDMOD: watchdog interrupt only mode.
		0	R/W	1	WDEN: watchdog enabled (set only).
				0*	WDEN: watchdog disabled.

[1] Use bit 4 to clear this bit.

[2] Reading WDCNTL register clears this bit.

This register controls the operation of the watchdog timer. Watchdog timer can be enabled by setting the WDEN bit of this register. WDEN is a set-only bit. Once set (enabled), this bit cannot be cleared by software. It can be cleared only with a hardware reset or watchdog reset.

The WDMOD bit determines the mode of operation. This bit is a set-only bit. There are two modes of operation:

- Interrupt only mode: This is the default mode of operation. In this mode, when the watchdog timer times out, the interrupt flag is set (WDINT) and an interrupt is generated to the host controller.
- Interrupt and reset mode: In this mode, when the watchdog timer times out, the reset flag is set (WDRST) and an interrupt is generated to host controller and resets the chip to POR state.

WDINT flag: This flag can be cleared by writing a ‘1’ to bit 4 of this register.

WDRST flag: This flag indicates that a watchdog reset has occurred. This flag does not get cleared by the watchdog reset. After a watchdog reset event, the host controller can read this bit to determine if a reset had occurred. The WDRST flag gets cleared after it is read or after an external reset is applied.

Before enabling the watchdog timer, the watchdog flags (interrupt flag and reset flag) **must** be cleared (if they are set). The interrupt flag is cleared by using bit 4 of the WDCNTL register and the reset flag is cleared just by reading the WCNTL register.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.5 GPIOs and interrupts

7.3.5.1 IP — Input Port register

This register is read-only. They reflect the incoming logic levels of the port pins P0 to P3, regardless of whether the pin is defined as an input or an output by the I/O configuration register. Writes to this register have no effect.

Table 11. IP - Input Port register (address 07h) bit description

*Legend: * default value 'X' is determined by the externally applied logic level.*

Address	Register	Bit	Access	Value	Description
07h	IP	7:4	read only	0h*	reserved
		3:0	read only	Xh*	reflects incoming logic levels of I/O P0 to P3

7.3.5.2 INTSTAT — Interrupt Status register

This register reflects the status of an interrupt. INTSTAT is a read-only register.

INTP0 to INTP3 interrupt caused by input port pins P0 to P3, respectively.

Table 12. INTSTAT - Interrupt status register (address 08h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
08h	INTSTAT	7:4	-	0*	reserved
		3:0	read only	1	INTP3 flag set
				0*	INTP3 flag clear
				1	INTP2 flag set
				0*	INTP2 flag clear
				1	INTP1 flag set
				0*	INTP1 flag clear
				1	INTP0 flag set
				0*	INTP0 flag clear

Upon power-up or activation of hardware reset by $\overline{\text{RESET}}$ pin, INTSTAT register bits [3:0] are cleared (= 0), thus clearing the interrupt flags. Change in logic level at GPIO pins P0 to P3 configured as inputs will cause generation of interrupt when not masked using MSK register. The corresponding flag bit in this register is set and latched until cleared.

7.3.5.3 OP — Output Port register

This register is an output-only port. It reflects the outgoing logic levels of the pins defined as outputs by IOC register. Bit values in this register have no effect on pins defined as inputs. In turn, reads from this register reflect the value that is in the flip-flop controlling the output selection, not the actual pin value. Only the lower four bits are used and P0 to P3 are affected by this register.

Table 13. OP - Output Port register (address 09h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
09h	OP	7:4	-	0000*	reserved
		3:0	R/W	0000*	reflects outgoing logic levels of I/O P0 to P3

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.5.4 IOC — I/O Configuration register

The lower four bits of this register configures the direction of the I/O pins P0 to P3. If a bit in [3:0] is set (written with logic 1), the corresponding port pin is enabled as an input with high-impedance output driver. If the bit is cleared (written with logic 0), the corresponding port pin is enabled as an output. At reset, the device's ports P0 to P3 are inputs.

Table 14. IOC - I/O configuration register (address 0Ah) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Ah	IOC	7:4	-	0*	reserved
		3	R/W	1*	P3 will be configured as input
				0	P3 will be configured as output
		2	R/W	1*	P2 will be configured as input
				0	P2 will be configured as output
		1	R/W	1*	P1 will be configured as input
				0	P1 will be configured as output
		0	R/W	1*	P0 will be configured as input
				0	P0 will be configured as output

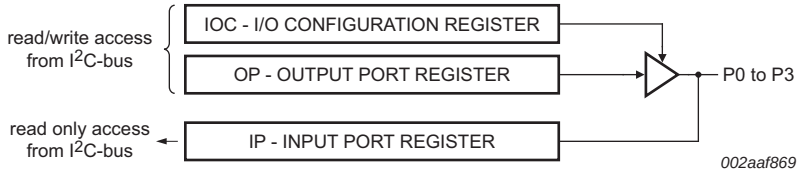


Fig 5. Simplified schematic for GPIO control

7.3.5.5 MSK — Mask interrupt register

Upon power-up, all the internal interrupt latches are reset and interrupt flags cleared and interrupt mask bits [3:0] are set to logic 1, thus disabling interrupts from input ports P0 to P3. Interrupts may be enabled by setting corresponding mask bits to logic 0.

Table 15. MSK - Interrupt mask register (address 0Bh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Bh	MSK	7:4	-	0*	reserved
		3	R/W	1*	disables interrupt for I/O P3
				0	enables interrupt for I/O P3
		2	R/W	1*	disables interrupt for I/O P2
				0	enables interrupt for I/O P2
		1	R/W	1*	disables interrupt for I/O P1
				0	enables interrupt for I/O P1
		0	R/W	1*	disables interrupt for I/O P0
				0	enables interrupt for I/O P0

An additional control to enable or disable the $\overline{\text{INT}}$ pin is provided by MODE control register bit 5 (MODE[5]). Refer to Table 6.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.5.6 CLRINT — Clear Interrupts register

Interrupt flags can be cleared by bits [3:0] when set to logic 1.

Table 16. CLRINT - Clear interrupts register (address 0Ch) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Ch	CLRINT	7:4	-	0*	reserved
		3	write only	1	clear INTP3 flag
				0*	read value
		2	write only	1	clear INTP2 flag
				0*	read value
		1	write only	1	clear INTP1 flag
				0*	read value
		0	write only	1	clear INTP0 flag
				0*	read value

7.3.5.7 INTMODE — Interrupt Mode register

When interrupt(s) are enabled, bits [3:0] determine whether rising edge or falling edge of signal at P0 to P3 causes the interrupt to be generated. Interrupts are latched and flag(s) are set in the corresponding bits of INTSTAT register. When interrupts are masked using MSK register, these bits have no effect.

Table 17. INTMODE - Interrupt mode register (address 0Dh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Dh	INTMODE	7:4	-	0*	reserved
		3	R/W	1	interrupt occurs on falling edge for P3
				0*	interrupt occurs on rising edge for P3
		2	R/W	1	interrupt occurs on falling edge for P2
				0*	interrupt occurs on rising edge for P2
		1	R/W	1	interrupt occurs on falling edge for P1
				0*	interrupt occurs on rising edge for P1
		0	R/W	1	interrupt occurs on falling edge for P0
				0*	interrupt occurs on rising edge for P0

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

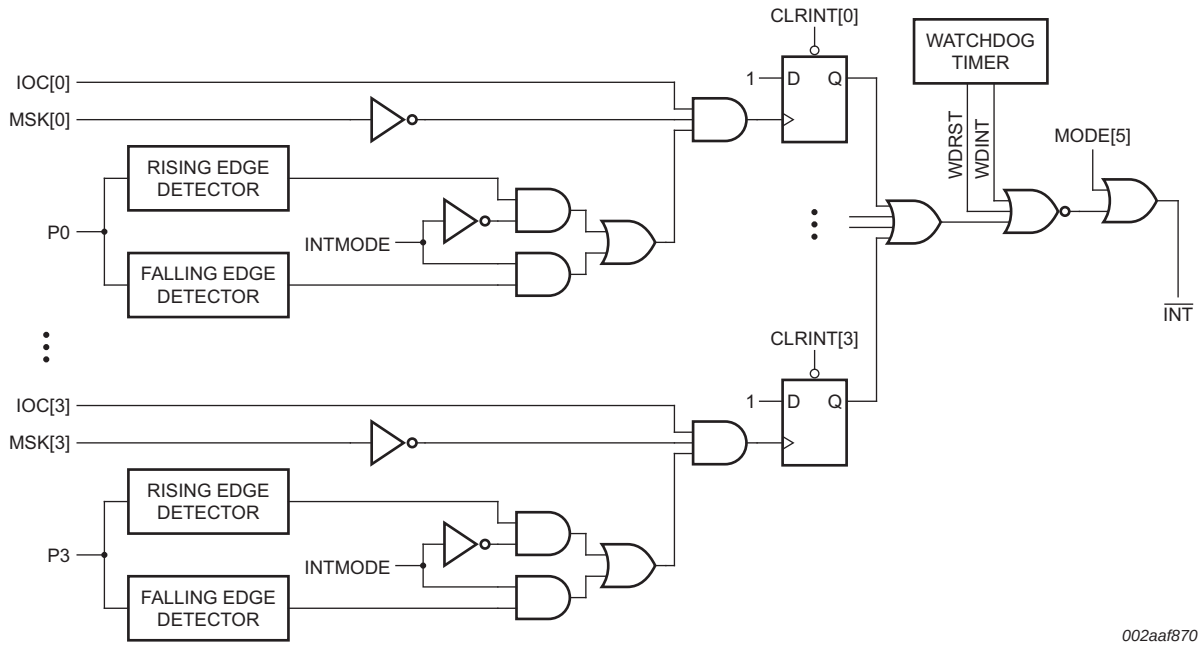


Fig 6. PCA9629 interrupt logic

7.3.6 Interrupt based motor control

Interrupt mechanisms from GPIOs 0 and 1 (INTP0 and INTP1) can be used to control the motor operation. Interrupts from GPIOs 2 and 3 are not used for motor control. They behave as normal GPIO interrupts. In the following sections, the word interrupt refers only to INTP0 and INTP1. The following actions can be performed upon the occurrence of an interrupt:

- Stop the motor
- Reverse the direction of motion
- Move extra steps/rotations and then, stop the motor or reverse its direction.

Only interrupts that occurred after the motor was started are acted upon. When an interrupt occurs, it is latched and the programmed action is performed. The microcontroller has to clear the interrupt before another occurrence of the same interrupt otherwise the second occurrence will not be acted upon. The following four registers, INT_ACT_SETUP, INT_MTR_SETUP, INT_ES_SETUP and INT_AUTO_CLR are used to program the various interrupt based control features of the motor. To enable the interrupt based control of the motor, bit 0 of the INT_ACT_SETUP register must be set.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.6.1 INT_ACT_SETUP — Interrupt Action Setup control register

Table 18. INT_ACT_SETUP - Interrupt action setup control register (address 0Eh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Eh	INT_ACT_SETUP	7:5	-	-	not used
		4	R/W	1	unit for EXTRASTEPS for both P0 and P1 counter is number of full rotations
				0*	unit for EXTRASTEPS for both P0 and P1 counter is number of steps
		3:1	-	-	not used
		0	R/W	1	enable interrupt based control of motor
				0*	disable interrupt based control of motor

If the interrupt based control is disabled, then values programmed in the following three registers (INT_MTR_SETUP, INT_ES_SETUP and INT_AUTO_CLR) have no effect on the motor operation.

Bit 4 of this register determines whether the values programmed in EXTRASTEPS0 and EXTRASTEPS1 registers represent the number of steps or number of rotations (see Section 7.3.16).

7.3.6.2 INT_MTR_SETUP — Interrupt Motor Setup control register

Table 19. INT_MTR_SETUP - Interrupt motor setup control register (address 0Fh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
0Fh	INT_MTR_SETUP	7:2	R	-	reserved
		1:0	R/W	11	Reverse motor on INT caused by P0 or P1
				10	Stop motor on INT caused by P0 or P1
				01	Stop motor on INT caused by P1
				00*	Stop motor on INT caused by P0

When an interrupt occurs, if the motor is programmed to stop on that interrupt, the following sequence of events takes place in the given order:

1. If extra steps feature is enabled for that interrupt (see INT_ES_SETUP, Section 7.3.6.3) then extra steps (/rotations) will occur.
2. If ramp down is enabled (see RMPCNTL, Section 7.3.17), the motor starts ramping down.
3. Motor stops.

When an interrupt occurs, if the motor is programmed to reverse direction on that interrupt, the following sequence of events takes place:

1. If extra steps feature is enabled for that interrupt (see INT_ES_SETUP, Section 7.3.6.3) then extra steps (/rotations) occurs in the current direction of motion.
2. The motor stops for the amount of time specified in the LOOPDLY timer register.
3. Motor reverses its direction of rotation.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.6.3 INT_ES_SETUP — Interrupt Extra Steps Setup control register

Table 20. INT_ES_SETUP - Interrupt extra steps setup control register (address 10h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
10h	INT_ES_SETUP	7:2	R	0000 00	reserved
		1:0	R/W	11	Enable EXTRASTEPS on both INTP0 and INTP1
				10	Enable EXTRASTEPS only on INTP1
				01	Enable EXTRASTEPS only on INTP0
				00*	Disable EXTRASTEPS for both INTP0 and INTP1

This register can be used to enable / disable the extra steps feature for each interrupt. Extra steps feature is used to make the motor rotate a specified amount of steps/rotations from the point of an interrupt occurrence.

7.3.6.4 INT_AUTO_CLR — Interrupt Auto Clear register

This register provides a mechanism to clear the two interrupts (INTP0 and INTP1) automatically without the occurrence of one interrupt clears the other without the microcontroller. The auto clear feature is disabled by default.

Table 21. INT_AUTO_CLR - Interrupt auto clear register (address 11h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
11h	INT_AUTO_CLR	7:2	-	0*	reserved
		1:0	R/W	11	INTP0 auto clears INTP1
				10	INTP1 auto clears INTP0
				01	INTP0 auto clears INTP1; INTP1 auto clears INTP0
				00*	INT auto clear for INTP0, INTP1 disabled

This feature is only available for interrupts that directly affect the operation of the motor as defined by the INT_MTR_SETUP register (see Section 7.3.6.2). For example, if INTP0 is used to stop the motor then it can be automatically cleared by its pair INTP1. However INTP1 should be manually cleared (through I²C-bus write to the CLRINT register). If both the interrupts are used to control the motor operation (INT_MTR_SETUP = 10 or 11), then all options of this register are valid. Any interrupt that is not automatically cleared by its pair should be manually cleared through I²C-bus write.

The auto clear mechanism can be used to create various motor movement patterns without being supervised by the microcontroller. For example, consider an application where the direction of motor rotation must be automatically reversed based on signals from two sensors placed apart from each other (sometimes referred to as ‘HOME’ positions) in a continuous manner without involving the microcontroller. The following example shows how to program the device for such an operation.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Example: This example assumes that two position sensors are located spaced apart and a drive mechanism is needed to move an object back and forth between these two sensors. Figure 7 shows this application use case. Driving the stepper motor causes movement of the object toward one of the sensors. Logic level output of one sensor is connected to input pin P0 and the other to P1. P0 and P1 are configured as **inputs**.

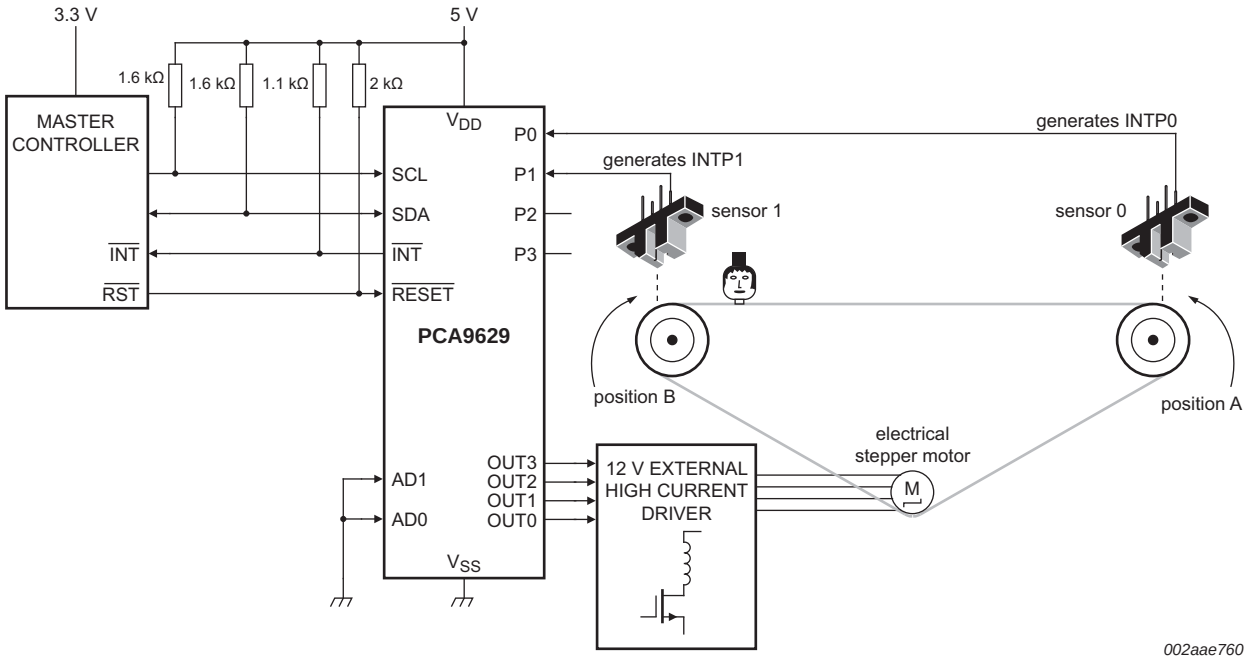


Fig 7. Example of controlling doll head movement between two home positions

At power-up, INTP0 to INTP3 flags INTSTAT[3:0] are clear (= 0).

Set INT_ACT_SETUP[0] = 1, enable interrupt based motor control.

Set INT_MTR_SETUP[1:0] = 11, Reverse motor on interrupt caused by P0 or P1.

Set INT_AUTO_CLR[1:0] = 01, INTP0 clears INTP1; INTP1 clears INTP0.

Start motor by writing MCNTL register and after some time, position sensor causes input logic at P0 to toggle.

When the input logic level at P0 changes, the interrupt caused by P0 is latched; INTP0 flag in INTSTAT is set (= 1).

Since INT_ACT_SETUP[0] = 1 and INT_MTR_SETUP[1:0] = 11 (reverse motor on interrupt caused by P0 or P1), the motor direction is reversed and the INTP1 flag is cleared (since INTP0 clears INTP1). This allows interrupt generation at the end of reverse movement by sensor at P1.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.7 SETMODE — output state on STOP control register

This register determines the condition of motor output pins when STOPPED, one of logic 0 or Hold (last state).

Table 22. SETMODE - Output state on STOP control register (address 12h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
12h	SETMODE	7:2	R/W	-	reserved
		1	R/W	1	outputs = HOLD after CCW STOP
				0*	outputs = logic 0 after CCW STOP
		0	R/W	1	outputs = HOLD after CW STOP
				0*	outputs = logic 0 after CW STOP

7.3.8 PHCNTL — Phase Control register

This register is used to configure the phase of the output waveforms at the output ports OUT0 to OUT3 to drive the motor coils (with external high current drivers). One of the following three modes of drive method can be selected using these bits:

- One-phase drive (wave drive)
- Two-phase drive
- Half-step drive

Table 23. PHCNTL - Phase control register (address 13h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
13h	PHCNTL	7:2	-	0*	reserved
		1:0	R/W	11 or 10	half-step drive outputs
				01	two-phase drive outputs
				00*	one-phase drive outputs

The phase drive can be changed at any time by writing to PHCNTL[1:0] bits.

7.3.9 SROTNL, SROTNH — Steps per rotation registers

This register determines how many steps are needed to execute one full turn of motor shaft (360°). This register should have a non-zero value if the requested operation is rotations (see Section 7.3.19).

Remark: If the motor has built-in gear, the number of steps needed to complete one full turn at the output shaft depends on the gear ratio used.

Table 24. SROTNL, SROTNH - Steps per rotation control registers (address 14h, 15h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
14h	SROTNL	7:0	R/W	00h*	number of steps per one rotation, low byte
15h	SROTNH	7:0	R/W	00h*	number of steps per one rotation, high byte

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.10 CWPWL, CWPWH — Clockwise step pulse width register

This register determines the step pulse width used for the phase sequence output waveforms during ClockWise (CW) rotation.

Table 25. CWPWL, CWPWH - Clockwise step pulse width control register (address 16h, 17h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
16h	CWPWL	7:0	R/W	00h*	step pulse width, low byte
17h	CWPWH	7:0	R/W	00h*	step pulse width, high byte

This register sets the pulse width value between 3 μs and 3145 ms (±5 %).

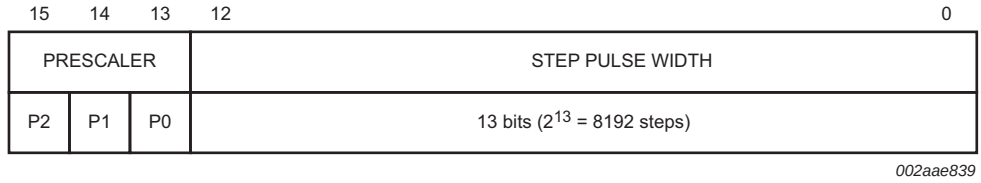


Fig 8. Step pulse width

The upper three bits of the register are the prescaler that determines the dynamic range for the step pulse width. Table 26 shows the range for each setting of the prescaler.

Table 26. Prescaler range settings

Prescaler [P2:P0]	Decimal value (D)	2^D	Range
000	0	1	3 μs to 24.576 ms
001	1	2	6 μs to 49.152 ms
010	2	4	12 μs to 98.304 ms
011	3	8	24 μs to 196.608 ms
100	4	16	48 μs to 393.216 ms
101	5	32	96 μs to 786.432 ms
110	6	64	192 μs to 1572.864 ms
111	7	128	384 μs to 3145.728 ms

Remark: The values given in Table 26 are based on nominal 1 MHz internal clock.

This method gives the user access to the entire range with the smallest pulse width (fastest speed) of 3 μs at the lower end, and the largest pulse width (slowest speed) of 3145 ms at the higher end.

The prescaler value defines the range of the ramp control. The ramp-up starts from its maximum pulse width and ramp-down ends at same maximum pulse width. The top speed of the ramp control is defined by both PRESCALER and STEP_PULSE_WIDTH values.

Final (top) speed = (minimum pulse width in the range defined by PRESCALER[15:13]) × (STEP_PULSE_WIDTH[12:0] + 1).

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.11 CCWPWL, CCWPWH — Counter-clockwise step pulse width register

This register determines the step pulse width used for the phase sequence output waveforms during Counter-ClockWise (CCW) rotation.

Table 27. CCWPWL, CCWPWH - Counter-clockwise step pulse width control register (address 18h, 19h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
18h	CCWPWL	7:0	R/W	00h*	step pulse width, low byte
19h	CCWPWH	7:0	R/W	00h*	step pulse width, high byte

The 16-bit value sets the pulse width between 3 μs and 3145 ms (±5 %).

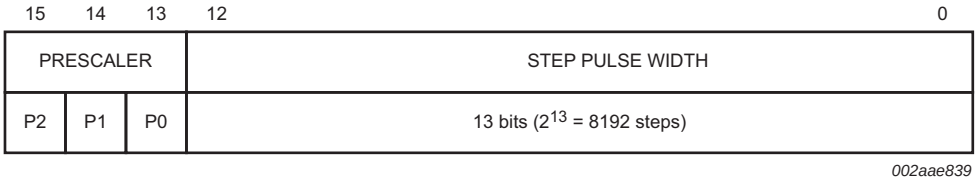


Fig 9. Step pulse width

The upper three bits of the register are the prescaler that determines the dynamic range for the step pulse width. Table 28 shows the range for each setting of the prescaler.

Table 28. Prescaler range settings

Prescaler [P2:P0]	Decimal value (D)	2^D	Range
000	0	1	3 μs to 24.576 ms
001	1	2	6 μs to 49.152 ms
010	2	4	12 μs to 98.304 ms
011	3	8	24 μs to 196.608 ms
100	4	16	48 μs to 393.216 ms
101	5	32	96 μs to 786.432 ms
110	6	64	192 μs to 1572.864 ms
111	7	128	384 μs to 3145.728 ms

Remark: The values given in Table 28 are based on nominal 1 MHz internal clock.

This method gives the user access to the entire range with the smallest pulse width (fastest speed) of 3 μs at the lower end, and the largest pulse width (slowest speed) of 3145 ms at the higher end.

The prescaler value defines the range of the ramp control. The ramp-up is started from its maximum pulse width and ramp-down ends at same maximum pulse width. The top speed of the ramp control is defined by both PRESCALER and STEP_PULSE_WIDTH values.

Final (top) speed = (minimum pulse width in the range defined by PRESCALER[15:13]) × (STEP_PULSE_WIDTH[12:0] + 1).

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.12 CWSCOUNTL, CWSCOUNTH — Number of clockwise steps register

This register determines the number of steps the motor should turn in clockwise direction.

Table 29. CWSCOUNTL, CWSCOUNTH - Number of clockwise steps count register (address 1Ah, 1Bh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
1Ah	CWSCOUNTL	7:0	R/W	00h*	number of clockwise steps, low byte
1Bh	CWSCOUNTH	7:0	R/W	00h*	number of clockwise steps, high byte

7.3.13 CCWSCOUNTL, CCWSCOUNTH — Number of counter-clockwise steps register

This register determines the number of steps the motor should turn in counter-clockwise direction.

Table 30. CCWSCOUNTL, CCWSCOUNTH - Number of counter-clockwise steps count register (address 1Ch, 1Dh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
1Ch	CCWSCOUNTL	7:0	R/W	00h*	number of counter-clockwise steps, low byte
1Dh	CCWSCOUNTH	7:0	R/W	00h*	number of counter-clockwise steps, high byte

7.3.14 CWRCOUNTL, CWRCOUNTH — Number of clockwise rotations register

This register determines the number of full rotations the motor should turn in clockwise direction.

Table 31. CWRCOUNTL, CWRCOUNTH - Number of clockwise rotations count register (address 1Eh, 1Fh) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
1Eh	CWRCOUNTL	7:0	R/W	00h*	number of clockwise rotations, low byte
1Fh	CWRCOUNTH	7:0	R/W	00h*	number of clockwise rotations, high byte

7.3.15 CCWRCOUNTL, CCWRCOUNTH — Number of counter-clockwise rotations register

This register determines the number of full rotations the motor should turn in counter-clockwise direction.

Table 32. CCWRCOUNTL, CCWRCOUNTH - Number of counter-clockwise rotations count register (address 20h, 21h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
20h	CCWRCOUNTL	7:0	R/W	00h*	number of counter-clockwise rotations, low byte
21h	CCWRCOUNTH	7:0	R/W	00h*	number of counter-clockwise rotations, high byte

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.16 EXTRASTEPS0, EXTRASTEPS1 — Extra steps count for INTP0, INTP1 control register

Table 33. EXTRASTEPS0, EXTRASTEPS1 - Extra steps count for INTP0, INTP1 register (address 22h, 23h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
22h	EXTRASTEPS0	7:0	R/W	00h*	count value for EXTRASTEPS (steps or rotations) for INTP0
23h	EXTRASTEPS1	7:0	R/W	00h*	count value for EXTRASTEPS (steps or rotations) for INTP1

This register has no effect if the interrupt based motor control is disabled or if the EXTRASTEPS feature for that interrupt is disabled.

When EXTRASTEPS feature is selected using INT_ES_SETUP register bits [1:0], the 8-bit value in this register is used to determine the number of steps or rotations to be overdriven. Direction of rotation of motor is maintained. If the count value in this register = 0, no EXTRASTEPS occurs. Whether the count indicates the number of extra steps or number of full rotations depends on the value of INT_ACT_SETUP control register bit 4.

If INT_ACT_SETUP[4] = 0 (default value), then EXTRASTEPSn value indicates number of extra steps that will occur after the corresponding interrupt.

If INT_ACT_SETUP[4] = 1, then EXTRASTEPSn value indicates number of full rotations that will occur after the corresponding interrupt.

7.3.17 RMPCNTL — Ramp control register

Table 34. RMPCNTL - Ramp control register (address 24h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
24h	RMPCNTL	7:6	R only	00*	reserved
		5	R/W	1	enable ramp-up during start
				0*	disable ramp-up during start
		4	R/W	1	enable ramp-down to stop
				0*	disable ramp-down to stop
		3:0	R/W	0000*	ramp step multiplication factor

The multiplication factor has a decimal range from 1 to 8192 as shown in Table 35.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Table 35. Multiplication factor value for ramp-up, ramp-down control

Register value [3:0]	Decimal value (D)	Ramp step multiplication factor (2^D)
0000	0	1
0001	1	2
0010	2	4
0011	3	8
0100	4	16
0101	5	32
0110	6	64
0111	7	128
1000	8	256
1001	9	512
1010	10	1024
1011	11	2048
1100	12	4096
1101	13	8192
1110, 1111	14, 15	reserved and do not use

RMPCNTL[5:4] enables/disables the speed ramp-up during starting of the motor and speed ramp-down during stopping of the motor.

The RMPCNTL[3:0] defines the acceleration/decelerating rate of the ramp control. If the value is small, the PWM width decrement (accelerating)/increment (decelerating) is slower.

The pulse width decrement and increment step is 'smallest_pulse_step × RMPCNTL[3:0]'. The smallest_pulse_step is defined by prescaler value of CWPWH and CCWPWH. Each prescaler setting's smallest_pulse_step is given in Table 26 and Table 28 (the minimum value of the range).

The ramp control will start and end in speed of maximum_pulse_step, which is the maximum value of the range given in Table 26 and Table 28.

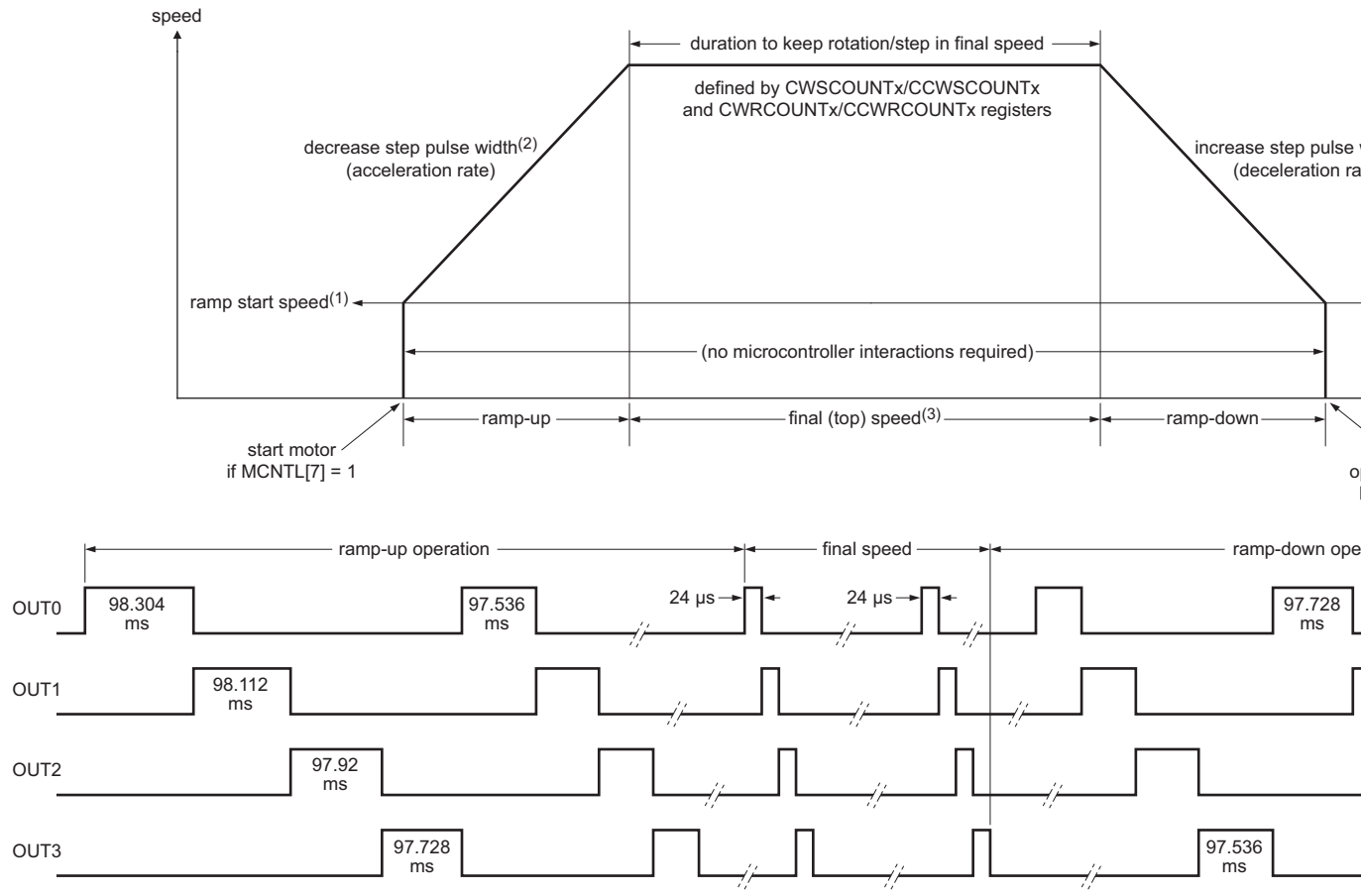
The ramp-up is completed when the pulse width gets the width that is set by CWPWL/CWPWH or CCWPWL/CCWPWH registers.

During ramp-up, the step pulse width is automatically decremented (from the maximum value for step pulse width in the chosen range) until the value in CWPW or the CCWPW register is reached, depending on the direction of rotation. See Figure 10.

During ramp-down, the step pulse width is automatically incremented from the current value in CWPW or the CCWPW, depending on the direction of rotation, until it reaches the maximum value for step pulse width in the chosen range. See Figure 10.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.



Example shown is one-phase drive for clockwise rotation.

(1) The ramp start or ramp end speed is defined as the maximum value of the range given in Table 26 and Table 28 based on prescaler bit CWPWH/CCWPWH registers. For example, the ramp start or ramp end speed is 98.304 ms if the CWPWH/CCWPWH[15:13] = 010.

(2) The decrease/increase step pulse width is defined as the minimum value of the range given in Table 26 and Table 28 based on prescal CWPWH/CCWPWH registers times the ramp step multiplication factor bits [3:0] in RMPCNTL register. For example, the decrease/incre 192 µs (12 µs × 16) if the CWPWH/CCWPWH[15:13] = 010 (minimum value 12 µs) and RMPCNTL[3:0] = 0100 (multiplication factor 16

(3) The ramp-up final speed is defined as the minimum value of the range given in Table 26 and Table 28 based on prescaler bit s [15:13] t bits [12:0] plus 1 in CWPWH/L and CCWPWH/L registers. For example, the ramp-up final speed is 24 µs (12 µs × 2) if the CWPWH/CC value 12 µs) and the CWPWH/L or CCWPWH/L = 0x0001 (1 + 1).

Fig 10. PCA9629 operation model for ramp-up (acceleration) and ramp-down (deceleration)

All information provided in this document is subject to legal disclaimers.
© NXP B.V. 2012. All rights reserved.

During ramp-up and ramp-down phase of operation, the interrupt based controls do not affect the motor run. An interrupt can happen during ramp-up or ramp-down and it gets registered in the chip. Once the ramp-up operation is finished, then the interrupt is acted upon. A stop request from the microcontroller (writing MCNTL[7] to '0') is the only event that affects the motor operation during ramp-up and ramp-down.

During ramp-up, the micro can issue a stop request. The following sequence of events takes place in the given order:

1. If hard stop is enabled, the motor stops immediately (even if ramp-down is enabled) - Priority 1.
2. If hard stop is disabled but ramp-down is enabled, then the motor starts to ramp down to a stop - Priority 2.
3. If hard stop is disabled and ramp-down is disabled, then motor stops immediately - Priority 3.

During ramp-down, the micro can issue a stop request. The following sequence of events takes place in the given order:

1. If hard stop is enabled, the motor stops immediately (it does not finish ramping down) - Priority 1.
2. If hard stop is disabled but ramp-down is enabled, then the motor continues to ramp down to a stop - Priority 2.

In the duration between end of ramp-up and beginning of ramp-down, the interrupt based controls (if enabled) can affect the operation of the motor. In this region, Section 7.3.6 gives the priority of events when both interrupt based control and ramp control are enabled together.

7.3.18 LOOPDLY — Loop delay timer register

This feature is used to make the motor wait for a certain amount of time before reversing its direction of rotation. There are two situations in which the motor must reverse its direction of rotation:

- The user requests both clockwise and counter clockwise rotation (also known as auto reversal mode).
- On an interrupt (also known as interrupt reversal mode). This register holds the wait time value in seconds. 00h = 0 second wait time. FFh = 255 seconds wait time.

Remark: LOOPDLY has an accuracy of ±5 %.

Table 36. LOOPDLY - Loop delay timer control register (address 25h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
25h	LOOPDLY	7:0	R/W	00h*	loop delay counter

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.19 MCNTL — Motor control register

This register acts like the master control panel for driving the motor. It determines the type of motor operation and controls the starting/stopping of the motor. The registers from address 0Eh (INT_ACT_SETUP) to 25h (LOOPDLY) are referred to as the motor parameter registers. The user must first program the motor parameter registers that are required for the current run of the motor. After that, this register should be programmed with the type of operation required. The motor starts when bit 7 of this register is set.

Table 37. MCNTL - Motor control register (address 26h) bit description

*Legend: * default value.*

Address	Register	Bit	Access	Value	Description
26h	MCNTL	7	R/W	1	start motor
				0*	stop motor
		6	R only	0*	reserved
		5	R/W	1	hard stop enabled
				0*	hard stop disabled
		4	R/W	1	perform the actions specified in bits [3:0] continuously
				0*	perform the actions specified in bits [3:0] once
		3:2	R/W	11 or 10	step pulses and then rotations
				01	rotate for specified number of rotations
				00*	send specified number of step pulses
		1:0	R/W	11	rotate counter-clockwise first, then clockwise
				10	rotate clockwise first, then counter-clockwise
				01	rotate counter-clockwise
				00*	rotate clockwise

7.3.19.1 MCNTL[7]: start/stop motor

This bit indicates the state of the motor. A ‘1’ indicates that the motor is running and ‘0’ indicates that the motor is in the stopped state.

To start the motor, write ‘1’ to this bit. Once the motor is started, any changes to the motor parameter registers do not affect the current run of the motor except for phase changes. Only phase changes (using the PHCNTL register) are allowed during motor operation. Similarly, bits [6:0] of the MCNTL register cannot be changed during motor operation. The only bit that can be changed in the MCNTL register while the motor is running is this start/stop bit. Also, any restart command (writing ‘1’ to this bit when it is already set), before the completion of the current operation are ignored.

When the current operation is completed, the motor stops and this bit is cleared. The completion of motor operation can be checked by reading this bit. After the motor has stopped, the motor parameter registers can be updated and the motor can be started again.

The microcontroller can stop the motor at any time by writing ‘0’ to this bit (this is referred to as a stop request). Once the motor stops, this bit is cleared. Stop request issued when the motor is already in the stopped state is ignored.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.3.19.2 MCNTL[5]: hard stop

The ‘hard stop’ feature is only applicable for stop requests issued by the micro. It does not affect the interrupt based stop mechanism. This feature is used to stop the motor immediately when the micro issues a stop request. Hard stop feature has a higher priority over ramp down. So even if ramp down is enabled, if the micro issues a stop request, the motor stops immediately and does not ramp down to stop. The micro should decide how the part should handle its stop request and accordingly enable/disable this feature. The priority of events during a stop request is:

- If hard stop is enabled (MCNTL[5]), then the motor stops immediately.
- If ramp down is enabled (RMPCNTL[4]), the motor starts ramping down to a stop.

7.3.19.3 MCNTL[4]: continuous operation

This bit determines if the operation specified in bits [3:0] of this register is executed once or continuously. If continuous operation is enabled, the motor can be stopped either by issuing a stop request or if an interrupt happens and the motor is programmed to stop on that interrupt. If continuous operation is not enabled then, the motor stops automatically after finishing the current operation once.

7.3.19.4 MCNTL[3:2]: steps and/or rotations

These two bits determine how many steps and/or rotations are executed by the motor in the current run. Based on clockwise or counter-clockwise direction (MCNTL[1:0]), the clockwise registers (CWSCOUNT, CWRCOUNT) or counter-clockwise registers (CCWSCOUNT, CCWRCOUNT) are used to determine the number steps/rotations. The following rules should be observed while programming these bits:

- Requested operation is steps (MCNTL[3:2] = 00): Number of steps should be non zero in the direction of operation (CW or CCW). In auto/interrupt based reversal modes (CW and CCW), the number of steps in both directions should be non zero. If this condition is not satisfied, the motor does not start.
- Requested operation is rotations (MCNTL[3:2] = 01): Number of rotations should be non zero in the direction of operation (CW or CCW). In auto/interrupt based reversal modes (CW and CCW), the number of rotations in both directions should be non zero. If this condition is not satisfied, the motor does not start.
- Requested operation is steps and rotations (MCNTL[3:2] = 10 or 11): At least one of the parameters, steps or rotations, should be a non zero value in the direction of operation. In auto/interrupt based reversal modes (CW and CCW), the same rule applies to both directions. If this condition is not satisfied, the motor does not start.

7.3.19.5 MCNTL[1:0]: clockwise (CW) / counter-clockwise (CCW)

These two bits are used to program the direction of the motor for current operation. Options 10 and 11 are called auto reversal modes (to differentiate it from interrupt based reversal). In these modes, the motor starts rotating in one direction and after completing the required steps/rotations reverses the direction of rotation. If continuous mode of operation is programmed with auto reversal, then the motor keeps repeating the operation continuously.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.4 Motor coil excitation

Initially, after a power-up of the device, when the motor is started for the first time, the first coil that is energized is OUT0 (if the motor is turning in the clockwise direction), or OUT3 (if the motor is turning in the counter clockwise direction). This very first step (after a power-up) is not counted towards the number steps the motor is required to move (it is the reference step). All subsequent steps are all counted. This applies only for the very first time the motor is started after the device is powered up.

For all subsequent starting of the motor, the first coil that is energized is the same coil where it had stopped. For example, consider the motor running in clockwise direction in the one-phase drive mode. If the last coil that was energized before the motor stopped was OUT2, then when the motor is started again OUT2 is energized first and after the pulse width time elapses the next coil in sequence, that is, OUT3 is energized.

7.5 Power-on reset

When power is applied to V_{DD}, an internal Power-On Reset (POR) holds the PCA9629 in a reset condition until V_{DD} has reached V_{POR}. At that point, the reset condition is released and the PCA9629 registers and state machine initialize to their default states. The power-on reset typically completes the reset and enables the part by the time the power supply is above V_{POR}. However, when it is required to reset the part by lowering the power supply, it is necessary to lower it below 2 V typical.

Remark: The system level reset pulse should be > 4 μs for the chip to guarantee reset condition.

7.6 $\overline{\text{RESET}}$ input

A reset can be accomplished by holding the $\overline{\text{RESET}}$ pin LOW for a minimum of $t_{w(rst)}$. The PCA9629 registers and I²C-bus state machine are held in their default state until the $\overline{\text{RESET}}$ input is once again HIGH. The $\overline{\text{RESET}}$ input has a 200 kΩ internal pull-up to V_{DD} pin.

The maximum wait time after RESET pin is released is 1 ms (typical).

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.7 Software reset

The Software Reset Call allows all the devices in the I²C-bus to be reset to the power-up state value through a specific formatted I²C-bus command. To be performed correctly, it implies that the I²C-bus is functional and that there is no device hanging the bus.

The maximum wait time after software reset is 1 ms (typical).

The SWRST Call function is defined as the following:

1. A START command is sent by the I²C-bus master.
2. The reserved General Call I²C-bus address '0000 000' with the R/$\overline{W}$ bit set to '0' (write) is sent by the I²C-bus master.
3. The PCA9629 device(s) acknowledge(s) after seeing the General Call address '0000 0000' (00h) only. If the R/$\overline{W}$ bit is set to '1' (read), no acknowledge is returned to the I²C-bus master.
4. Once the General Call address has been sent and acknowledged, the master sends one byte. The value of the byte must be equal to 06h. The PCA9629 acknowledges this value only. If the byte is not equal to 06h, the PCA9629 does not acknowledge it. If more than one byte of data is sent, the PCA9629 does not acknowledge anymore.
5. Once the right byte has been sent and correctly acknowledged, the master sends a STOP command to end the software reset sequence: the PCA9629 then resets to the default value (power-up value) and is ready to be addressed again within the specified bus free time. If the master sends a Repeated START instead, no reset is performed. The I²C-bus master must interpret a non-acknowledge from the PCA9629 (at any time) as a 'Software Reset Abort'. The PCA9629 does not initiate a software reset.

7.8 Interrupt output

The open-drain active LOW interrupt is activated by the following two mechanisms:

- **Watchdog timer:** If the watchdog timer is enabled and the timer times out, then an interrupt is generated and the watchdog interrupt flag bit [3] is set in the watchdog control register (WDCNTL).
- **GPIOs:** One or more of pins P0 to P3 can generate an interrupt if the following conditions are met:
  - The pin is configured as an input in the I/O configuration register (IOC).
  - The interrupt from that pin is enabled in the mask interrupt register (MSK).
  - The pin's state change (rising edge or falling edge) is programmed to generate an interrupt in the interrupt mode register (INTMODE).

The interrupt $\overline{INT}$ pin output can be enabled or disabled using MODE register bit [5] (0 = enable; 1 = disable). The interrupt flag bit is set in the INTSTAT register when one of the interrupts is generated from P0 to P3.

Remark: If the state of the pin does not match the contents of the Input port register, changing an I/O from an output to an input may cause a false interrupt to occur.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.9 Phase sequence generator

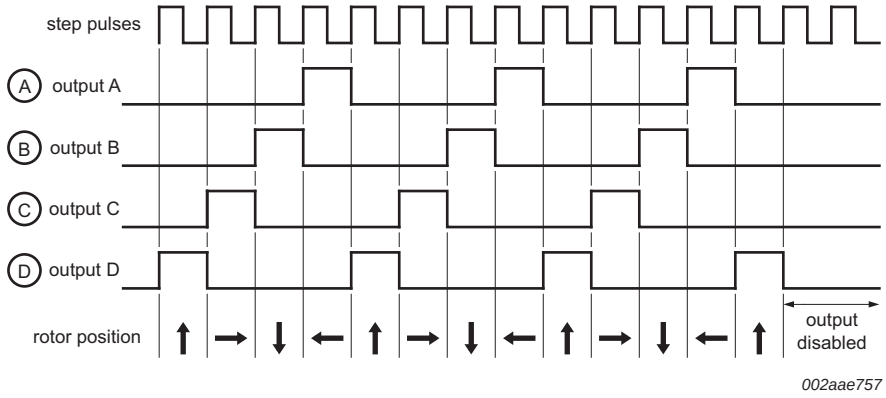
The PCA9629 phase sequence generator uses the on-chip oscillator and control logic to generate logic waveforms needed to support the following three types of stepper motor drive formats:

- One-phase drive, also called ‘wave drive’
- Two-phase drive
- Half-step drive

These logic level outputs are used to drive high current power driver stages to provide required drive current to the stepper motor coils.

7.9.1 One-phase drive (wave drive)

In one-phase drive method, only one winding is energized at any given time. The advantage of wave drive mode is its simplicity. The disadvantage of wave drive mode is that in the unipolar wound motor only 25 %, and in the bipolar motor only 50 % of the total motor winding are used at any given time. This means that maximum torque output from the motor is not made available. Since only one winding is energized, holding torque and working torque are reduced by 30 %. This can, within limits, be compensated by increasing supply voltage. The advantage of this form of drive is higher efficiency, but at the cost of reduced step accuracy.



Number of steps shown = 4 for simplicity.

Fig 11. Wave drive step sequence waveforms

Table 38. Logic output sequence for wave drive

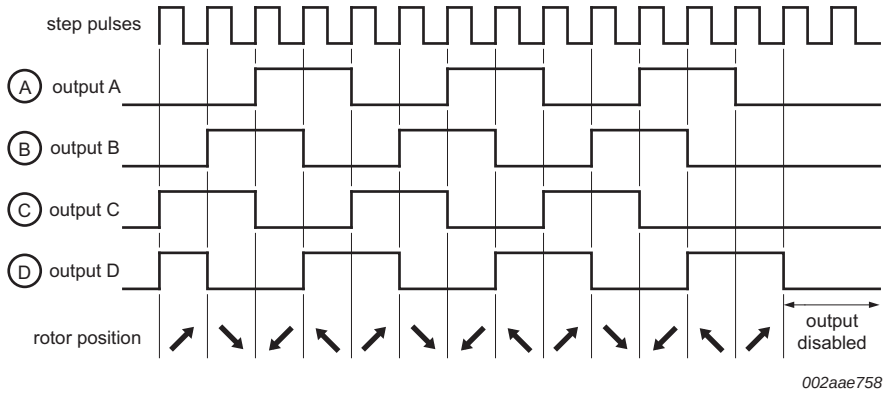
Winding	Step							
	1	2	3	4	5	6	7	8
Winding D	1	0	0	0	1	0	0	0
Winding C	0	1	0	0	0	1	0	0
Winding B	0	0	1	0	0	0	1	0
Winding A	0	0	0	1	0	0	0	1

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

7.9.2 Two-phase drive

In two-phase drive method, two windings are energized at any given time. In case of two-phase drive, the torque output of the unipolar wound motor is lower than the bipolar motor (for motors with the same winding parameters) since the unipolar motor uses only 50 % of the available winding, while the bipolar motor uses the entire winding.



Number of steps shown = 4 for simplicity.

Fig 12. Two-phase drive step sequence waveforms

Table 39. Logic output sequence for two-phase drive

Winding	Step							
	1	2	3	4	5	6	7	8
Winding D	1	0	0	1	1	0	0	1
Winding C	1	1	0	0	1	1	0	0
Winding B	0	1	1	0	0	1	1	0
Winding A	0	0	1	1	0	0	1	1

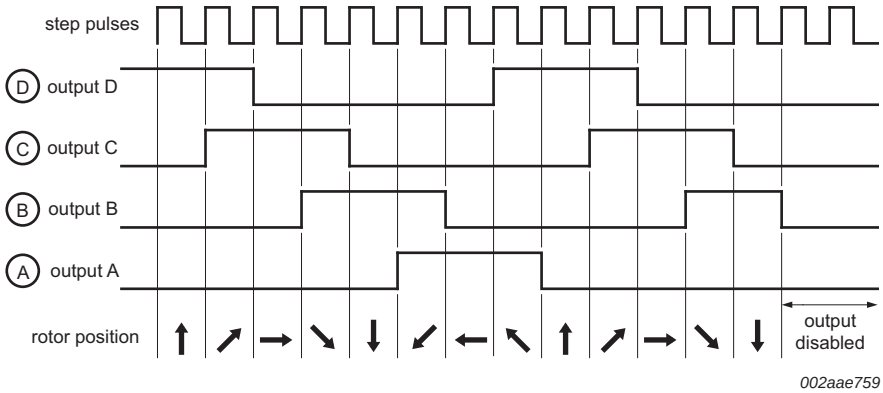
7.9.3 Half-step drive (one-phase and two-phase on)

'Half-step drive' combines both wave and two-phase (one-phase and two-phase on) drive modes. This results in angular movements that are half of those in 1- or 2-phases-on drive modes. Half-stepping can reduce a phenomenon referred to as resonance, which can be experienced in 1- or 2-phases-on drive modes.

As the name implies, in this mode it is possible to step a motor in a half-step sequence, thus producing half steps, for example 3.75° steps from a 7.5° motor. A possible drawback for some applications is that the holding torque is alternately strong and weak on successive motor steps. This is because on full steps only one phase winding is energized, while on the half-steps two stator windings are energized. Also, because current and flux paths differ on alternate steps, accuracy is worse than when full stepping.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.



Four step stepper motor run with half-step waveforms increases the number of steps to eight.

Fig 13. Half-step drive sequence waveforms

Table 40. Logic output sequence for half-step drive

Winding	Step							
	1	2	3	4	5	6	7	8
Winding D	1	1	0	0	0	0	0	1
Winding C	0	1	1	1	0	0	0	0
Winding B	0	0	0	1	1	1	0	0
Winding A	0	0	0	0	0	1	1	1

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

8. Characteristics of the I²C-bus

The I²C-bus is for two-way, two-line communication between different ICs or modules. The two lines are a serial data line (SDA) and a serial clock line (SCL). Both lines must be connected to a positive supply via a pull-up resistor when connected to the output stages of a device. Data transfer may be initiated only when the bus is not busy.

8.1 Bit transfer

One data bit is transferred during each clock pulse. The data on the SDA line must remain stable during the HIGH period of the clock pulse as changes in the data line at this time will be interpreted as control signals (see Figure 14).

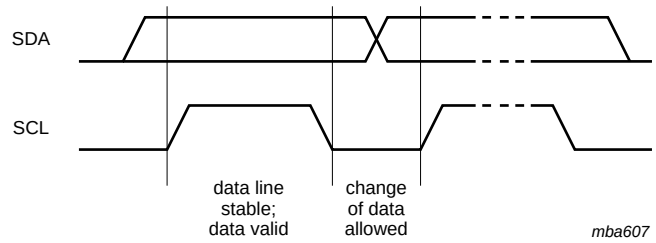


Fig 14. Bit transfer

8.1.1 START and STOP conditions

Both data and clock lines remain HIGH when the bus is not busy. A HIGH-to-LOW transition of the data line while the clock is HIGH is defined as the START condition (S). A LOW-to-HIGH transition of the data line while the clock is HIGH is defined as the STOP condition (P) (see Figure 15).

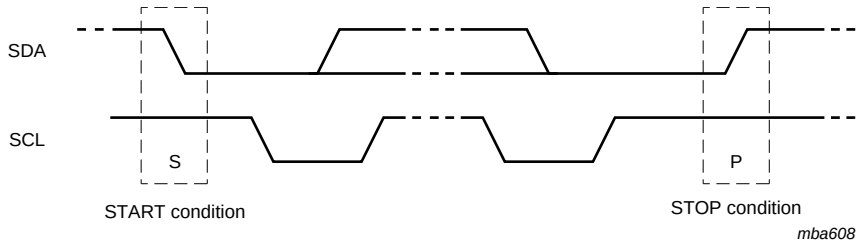


Fig 15. Definition of START and STOP conditions

8.2 System configuration

A device generating a message is a ‘transmitter’; a device receiving is the ‘receiver’. The device that controls the message is the ‘master’ and the devices which are controlled by the master are the ‘slaves’ (see Figure 16).

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

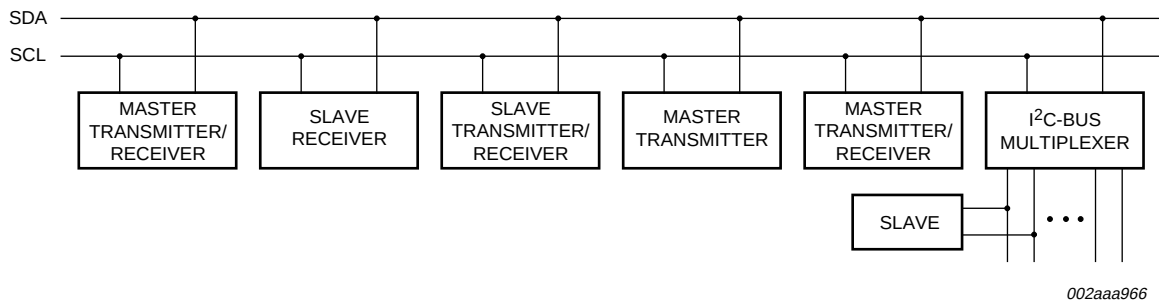


Fig 16. System configuration

8.3 Acknowledge

The number of data bytes transferred between the START and the STOP conditions from transmitter to receiver is not limited. Each byte of eight bits is followed by one acknowledge bit. The acknowledge bit is a HIGH level put on the bus by the transmitter, whereas the master generates an extra acknowledge related clock pulse.

A slave receiver which is addressed must generate an acknowledge after the reception of each byte. Also a master must generate an acknowledge after the reception of each byte that has been clocked out of the slave transmitter. The device that acknowledges has to pull down the SDA line during the acknowledge clock pulse, so that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse; set-up time and hold time must be taken into account.

A master receiver must signal an end of data to the transmitter by not generating an acknowledge on the last byte that has been clocked out of the slave. In this event, the transmitter must leave the data line HIGH to enable the master to generate a STOP condition.

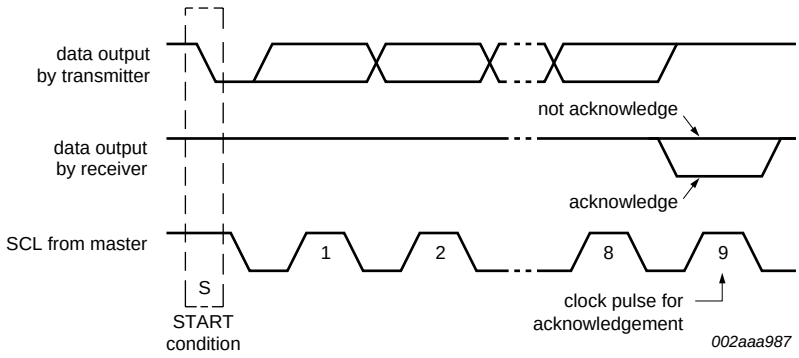


Fig 17. Acknowledgement on the I²C-bus

9. Bus transactions

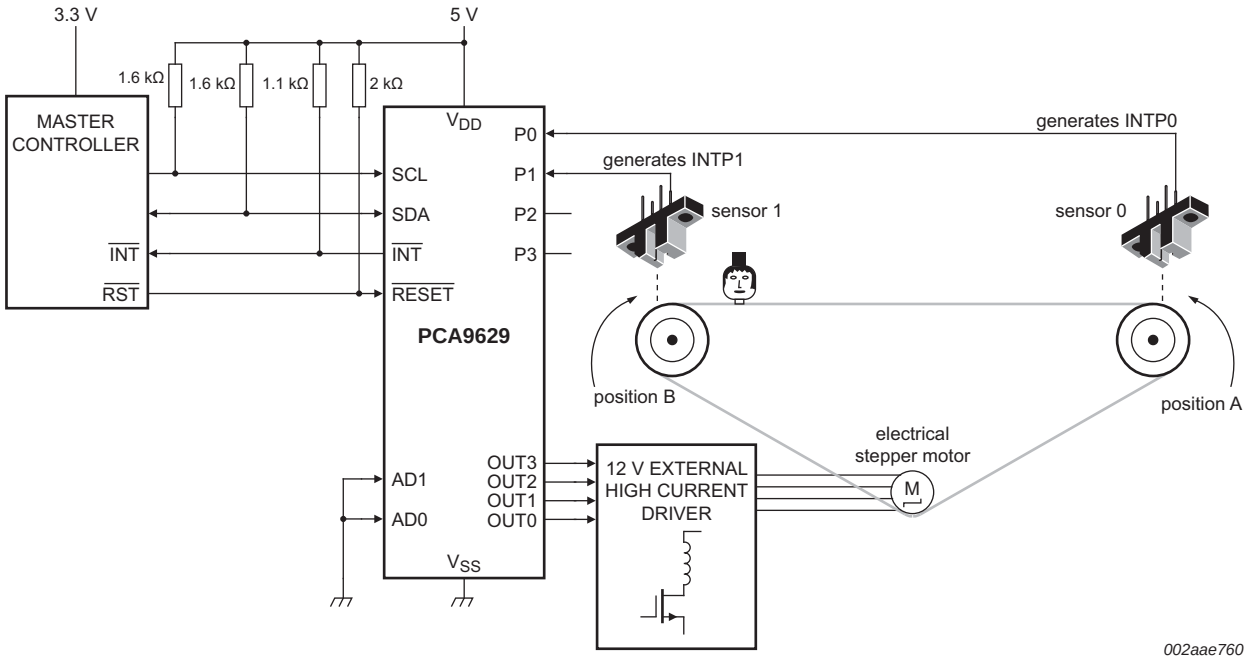
Data is transmitted to the PCA9629 registers using ‘Write Byte’ transfers.

Data is read from the PCA9629 registers using ‘Read Byte’ transfers.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

10. Application design-in information



Device address configured as 0100 0000b for this example.

Fig 18. Typical application

10.1 Stepper motor coil driver considerations

When choosing a motor and coil driver circuit for an application, it is necessary to choose the coil driver such that the minimum expected drive strength of the coil driver over the anticipated operating conditions exceeds the minimum coil current in the application. For the NMOS FETs, the gate voltage affects the FET drive strength, so it is necessary to evaluate the FET with its gate at the minimum V_{DD} planned for the PCA9629 application, because the PCA9629 cannot drive the gate higher than the V_{DD}.

For example, in most applications a 5 V power supply would have a specification like 5 V ± 10 % or 5 V ± 20 %, so it would be necessary to verify that the ON-resistance or current sinking capability of the FET with a gate voltage of 4.75 V or 4.5 V, whichever applies, is capable of sinking all of the current that the motor might require. Since FETs present a capacitive load to the outputs of the PCA9629, the output asymptotically approaches the V_{DD} of the part, so eventually the full V_{DD} appears at the output. However, for Darlington bipolar coil drivers the input current represents a static current load that reduces the V_{OH}. So depending upon the input current of the Darlington bipolar coil driver, the PCA9629 output voltage will always be less than V_{DD}. This in turn reduces the input current and also reduces the available drive current from the Darlington bipolar coil driver, so the lowest gain for the driver and the input current gain product must be considered in verifying that the maximum motor current can be sunk by the driver.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

10.2 Considerations when using GPIO pins P0 to P3 as inputs

For proper operation of GPIO pins as inputs, the signals at the inputs must be free from any glitches or noise. The signals must be logic level inputs.

For example, outputs from sensors must provide logic level signals at the input pins of PCA9629. This may require signal conditioning at the outputs of sensors. Another example is when using P0 to P3 for key switch sensing. The inputs of PCA9629 do not provide key de-bouncing. This is external to PCA9629 and is user-defined and supplied.

10.3 Priority of ramp control, interrupt-based control, loop delay and hard stop

During ramp-up and ramp-down phases of operation, the interrupt-based controls do not affect the motor run. Interrupts that occur during ramp-up or ramp-down are ignored. Once the ramp-up operation is finished (when the motor is running at the final speed), then the interrupts that occur are acted upon. A stop request from the microcontroller (writing 0 to MCNTL[7]) is the only event that affects the motor operation during ramp-up and ramp-down.

During ramp-up, the microcontroller can issue a stop request. The following sequence of events takes place in the given order:

1. If hard stop is enabled, the motor stops immediately (even if ramp-down is enabled); Priority 1.
2. If hard stop is disabled but ramp-down is enabled, then the motor starts to ramp down to a stop; Priority 2.
3. If hard stop is disabled and ramp-down is disabled, then motor stops immediately; Priority 3.

During ramp-down, the microcontroller can issue a stop request. The following sequence of events takes place in the given order:

1. If hard stop is enabled, the motor stops immediately (it does not finish ramping down); Priority 1.
2. If hard stop is disabled but ramp-down is enabled, then the motor continues to ramp down to a stop; Priority 2.

In the duration between end of ramp-up and beginning of ramp-down, the interrupt-based controls (if enabled) can affect the operation of the motor. In this region, Section 7.3.6.2 gives the priority of events when both interrupt-based control and ramp control are enabled together. Consider the following example (the motor is programmed to reverse rotation on an interrupt):

- Motor programmed for CW rotations; ramp-up and ramp-down enabled; reverse rotation on interrupt P0/P1.
- When motor is started, it starts ramping up and when ramp-up is completed it rotates at the final speed.
- If interrupt P0 happens, then it reverses rotation right away and start rotating in the CCW direction for the specified number of rotations.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

- Before the specified number of rotations is completed in the CCW direction, if interrupt P1 happens, then it again reverses its rotation right away and start rotating in the CW direction for the specified number of rotations.
- If no other interrupt happens, the motor finishes executing the specified number of rotations in the CW direction and then starts to ramp down.

In the above example, if extra steps are enabled for interrupts P0 and P1, then when the interrupts happen the motor executes the extra steps in the current direction of rotation and then reverses its direction.

11. Limiting values

Table 41. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions	Min	Max	Unit
V_{DD}	supply voltage		−0.5	+6.0	V
$V_{I/O}$	voltage on an input/output pin		V_{SS} − 0.5	5.5	V
$I_{I/O}$	input/output current	Pn, OUTn, $\overline{INT}$, SCL, SDA	-	±50	mA
I_I	input current		-	±20	mA
I_{SS}	ground supply current		-	210	mA
P_{tot}	total power dissipation		-	400	mW
T_{stg}	storage temperature		−65	+150	°C
T_{amb}	ambient temperature		−40	+85	°C

12. Static characteristics

Table 42. Static characteristics

V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V; T_{amb} = −40 °C to +85 °C; unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
Supply							
V_{DD}	supply voltage			4.5	-	5.5	V
I_{DD}	supply current	operating mode; no load; f_{SCL} = 1 MHz; V_{DD} = 5.5 V		-	6	10	mA
I_{stb}	standby current	no load; f_{SCL} = 0 kHz; $V_I = V_{DD}$ or V_{SS}; V_{DD} = 5.5 V		-	1	2	mA
V_{POR}	power-on reset voltage	no load; $V_I = V_{DD}$ or V_{SS}		-	2.3	-	V
V_{PDR}	power-down reset voltage	no load; $V_I = V_{DD}$ or V_{SS}	[1]	-	2.0	-	V
Input SCL; input/output SDA							
V_{IL}	LOW-level input voltage			−0.5	-	$+0.3V_{DD}$	V
V_{IH}	HIGH-level input voltage			$0.7V_{DD}$	-	5.5	V
I_{OL}	LOW-level output current	V_{OL} = 0.4 V; V_{DD} = 5.0 V		30	40	-	mA
I_L	leakage current	$V_I = V_{DD}$ or V_{SS}		−1	-	+1	μA
C_i	input capacitance	$V_I = V_{SS}$		-	6	10	pF

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Table 42. Static characteristics *...continued*

V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V; T_{amb} = −40 °C to +85 °C; unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
OUT0 to OUT3 outputs							
I_{OL}	LOW-level output current	V_{OL} = 0.5 V; V_{DD} = 4.5 V	[2]	25	28	-	mA
$I_{OL(tot)}$	total LOW-level output current	V_{OL} = 0.5 V; V_{DD} = 4.5 V	[2]	-	-	120	mA
V_{OH}	HIGH-level output voltage	I_{OH} = −10 mA; V_{DD} = 4.5 V	[3]	4.0	-	-	V
P0 to P3 I/Os							
I_{OL}	LOW-level output current	V_{OL} = 0.5 V; V_{DD} = 4.5 V	[2]	25	28	-	mA
$I_{OL(tot)}$	total LOW-level output current	V_{OL} = 0.5 V; V_{DD} = 4.5 V	[2]	-	-	120	mA
V_{OH}	HIGH-level output voltage	I_{OH} = −10 mA; V_{DD} = 4.5 V	[3]	4.0	-	-	V
I_{OZ}	OFF-state output current	3-state; V_{OH} = V_{DD} or V_{SS}		−10	-	+10	μA
C_{io}	input/output capacitance	3-state pins as inputs		-	5	8	pF
Address inputs							
V_{IL}	LOW-level input voltage			−0.5	-	+0.3V_{DD}	V
V_{IH}	HIGH-level input voltage			0.7V_{DD}	-	5.5	V
I_{LI}	input leakage current			−1	-	+1	μA
C_i	input capacitance			-	3	5	pF
RESET input							
V_{IL}	LOW-level input voltage			−0.5	-	+0.3V_{DD}	V
V_{IH}	HIGH-level input voltage			0.7V_{DD}	-	5.5	V
I_{LI}	input leakage current			−1	-	+1	μA
C_i	input capacitance			-	3	5	pF
I_{LIL}	LOW-level input leakage current	V_I = V_{SS}		−7	-	−45	μA
INT output							
I_{OL}	LOW-level output current	V_{OL} = 0.5 V; V_{DD} = 4.5 V		24	28	-	mA
I_{OH}	HIGH-level output current	open-drain; V_{OH} = V_{DD}		−10	-	+10	μA
C_o	output capacitance			-	5	-	pF

[1] In order to reset part, V_{DD} must be lowered to 1.4 V.

[2] Each bit must be limited to a maximum of 25 mA and the total package limited to 210 mA due to internal busing limits.

[3] For I_{OH} = −25 mA, the minimum V_{OH} = V_{DD} − 0.7 V with V_{DD} = 4.5 V to 5.5 V.

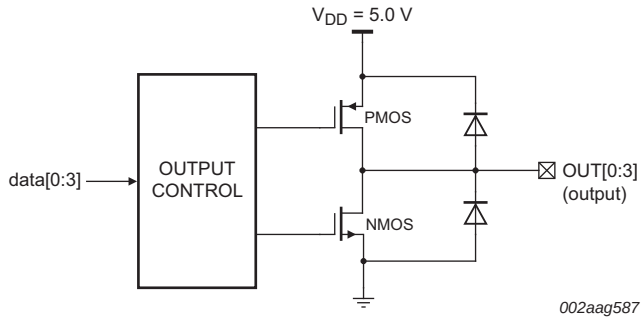


Fig 19. OUT[0:3] output equivalent circuit

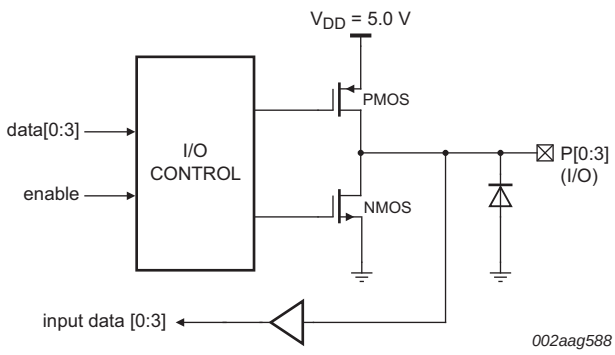


Fig 20. P[0:3] input/output equivalent circuit

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

13. Dynamic characteristics

Table 43. Dynamic characteristics

V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V; T_{amb} = −40 °C to +85 °C; unless otherwise specified.
Oscillator frequency = 1 MHz ± 5 % at 25 °C (see Figure 23).

Symbol	Parameter	Conditions		Standard-mode I²C-bus		Fast-mode I²C-bus		Fast-mode Plus I²C-bus		Unit
				Min	Max	Min	Max	Min	Max	
f_{SCL}	SCL clock frequency		[1]	0	100	0	400	0	1000	kHz
t_{BUF}	bus free time between a STOP and START condition			4.7	-	1.3	-	0.5	-	µs
$t_{HD;STA}$	hold time (repeated) START condition			4.0	-	0.6	-	0.26	-	µs
$t_{SU;STA}$	set-up time for a repeated START condition			4.7	-	0.6	-	0.26	-	µs
$t_{SU;STO}$	set-up time for STOP condition			4.0	-	0.6	-	0.26	-	µs
$t_{HD;DAT}$	data hold time			0	-	0	-	0	-	ns
$t_{VD;ACK}$	data valid acknowledge time		[2]	0.3	3.45	0.1	0.9	0.05	0.45	µs
$t_{VD;DAT}$	data valid time		[3]	0.3	3.45	0.1	0.9	0.05	0.45	µs
$t_{SU;DAT}$	data set-up time			250	-	100	-	50	-	ns
t_{LOW}	LOW period of the SCL clock			4.7	-	1.3	-	0.5	-	µs
t_{HIGH}	HIGH period of the SCL clock			4.0	-	0.6	-	0.26	-	µs
t_f	fall time of both SDA and SCL signals		[4][5]	-	300	20 + 0.1C_b	300	-	120	ns
t_r	rise time of both SDA and SCL signals			-	1000	20 + 0.1C_b	300	-	120	ns
t_{SP}	pulse width of spikes that must be suppressed by the input filter		[7]	-	50	-	50	-	50	ns
$t_{d(o)}$	output delay time	interrupt based motor control latency	[8]	5.7	7.4	5.7	7.4	5.7	7.4	µs
RESET										
$t_{w(rst)}$	reset pulse width		[9]	1	4	1	4	1	4	µs
$t_{rec(rst)}$	reset recovery time			-	1	-	1	-	1	ms

[1] Minimum SCL clock frequency is limited by the bus time-out feature, which resets the serial bus interface if either SDA or SCL is held LOW for a minimum of 25 ms. Disable bus time-out feature for DC operation.

[2] $t_{VD;ACK}$ = time for acknowledgement signal from SCL LOW to SDA (out) LOW.

[3] $t_{VD;DAT}$ = minimum time for SDA data out to be valid following SCL LOW.

[4] In order to bridge the undefined region of the SCL falling edge, a master device must internally provide a hold time of at least 300 ns for the SDA signal (refer to the V_{IL} of the SCL signal).

[5] The maximum t_f for the SDA and SCL bus lines is specified at 300 ns. The maximum fall time (t_f) for the SDA output stage is specified at 250 ns. This allows series protection resistors to be connected between the SDA and the SCL pins and the SDA/SCL bus lines without exceeding the maximum specified t_f.

[6] C_b = total capacitance of one bus line in pF.

[7] Input filters on the SDA and SCL inputs suppress noise spikes less than 50 ns.

[8] The time delay from one of the P[1:0] inputs edge changes to the motor control outputs OUT[3:0] change. Typical value = 6.5 µs.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

[9] The internal glitch filter rejects any LOW pulse less than 1 μs. The system level reset pulse should be > 4 μs for the chip to guarantee reset condition.

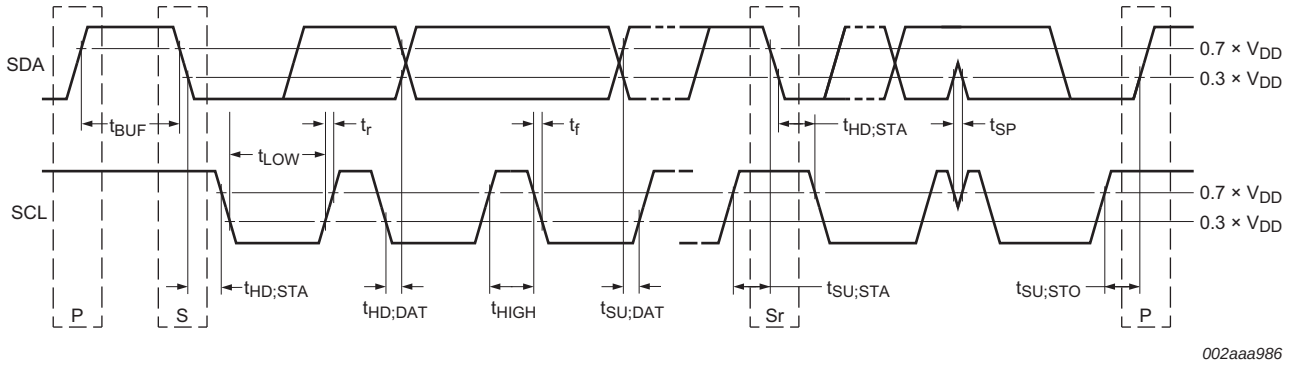


Fig 21. Definition of timing

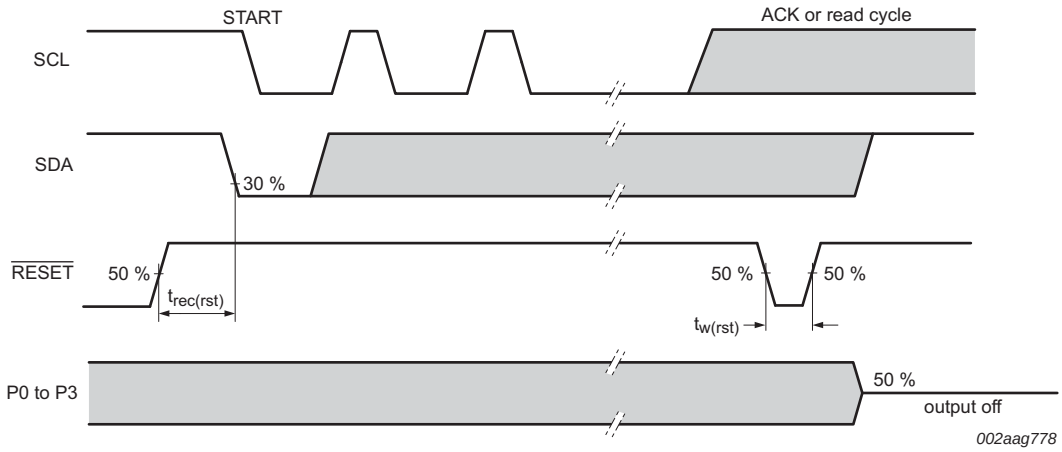


Fig 22. Reset timing

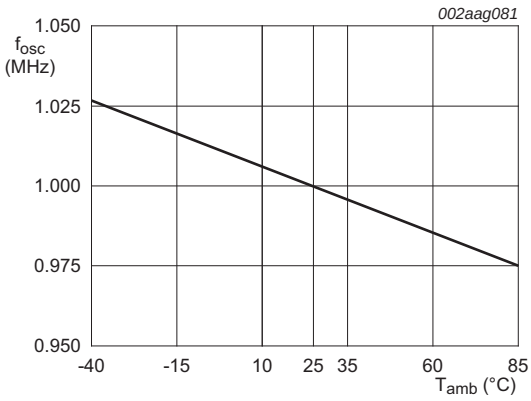
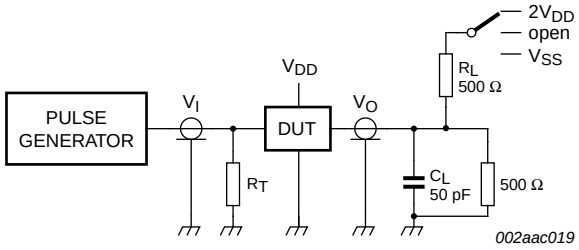


Fig 23. Typical oscillator frequency versus temperature

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

14. Test information

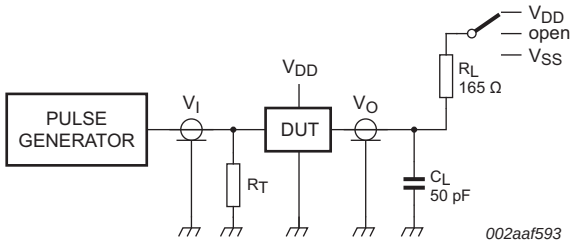


R_L = load resistance.

C_L = load capacitance includes jig and probe capacitance.

R_T = termination resistance should be equal to the output impedance Z_o of the pulse generators.

Fig 24. Test circuitry for switching times for GPIO pins P0 to P3



R_L for SDA and SCL = 165 Ω (30 mA or less current).

C_L = load capacitance includes jig and probe capacitance.

R_T = termination resistance should be equal to the output impedance Z_o of the pulse generators.

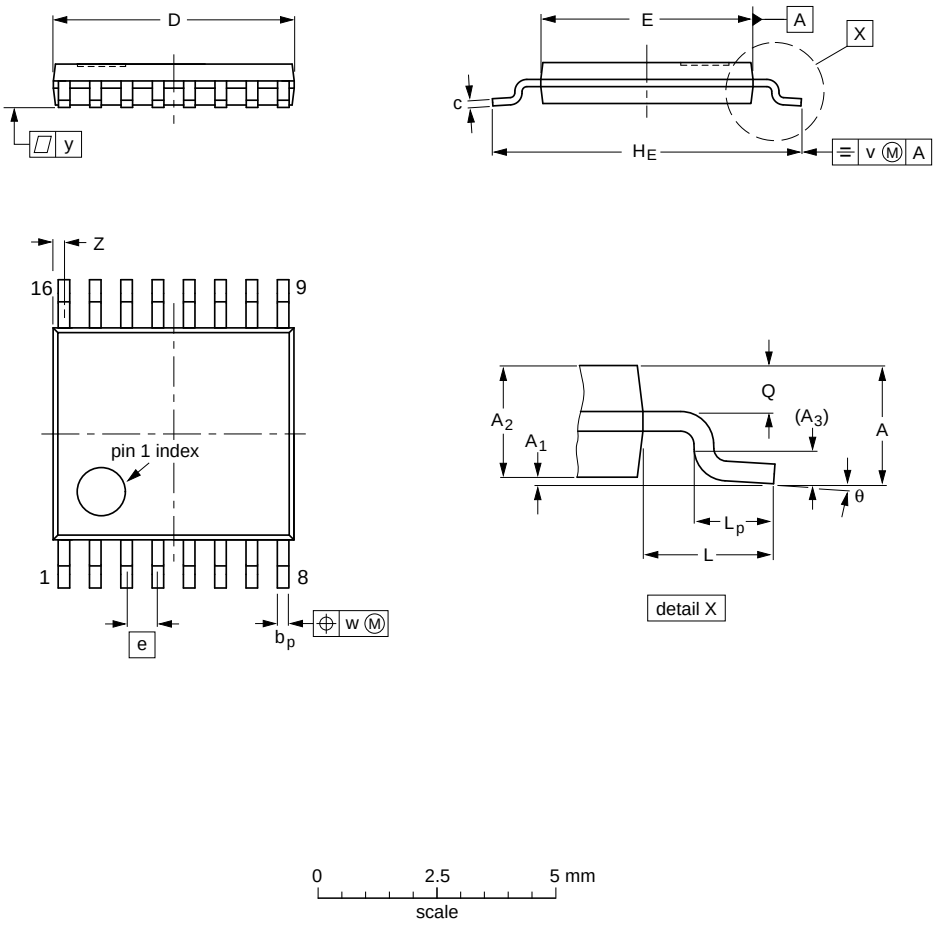
Fig 25. Test circuitry for switching times for SDA and SCL

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

15. Package outline

TSSOP16: plastic thin shrink small outline package; 16 leads; body width 4.4 mm **SOT403-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	D (1)	E (2)	e	H_E	L	L_p	Q	v	w	y	Z (1)	θ
mm	1.1	0.15 0.05	0.95 0.80	0.25	0.30 0.19	0.2 0.1	5.1 4.9	4.5 4.3	0.65	6.6 6.2	1	0.75 0.50	0.4 0.3	0.2	0.13	0.1	0.40 0.06	8° 0°

Notes

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT403-1		MO-153				~~99-12-27~~ 03-02-18

Fig 26. Package outline SOT403-1 (TSSOP16)

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

16. Handling information

All input and output pins are protected against ElectroStatic Discharge (ESD) under normal handling. When handling ensure that the appropriate precautions are taken as described in *JESD625-A* or equivalent standards.

17. Soldering of SMD packages

This text provides a very brief insight into a complex technology. A more in-depth account of soldering ICs can be found in Application Note *AN10365 "Surface mount reflow soldering description"*.

17.1 Introduction to soldering

Soldering is one of the most common methods through which packages are attached to Printed Circuit Boards (PCBs), to form electrical circuits. The soldered joint provides both the mechanical and the electrical connection. There is no single soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and Surface Mount Devices (SMDs) are mixed on one printed wiring board; however, it is not suitable for fine pitch SMDs. Reflow soldering is ideal for the small pitches and high densities that come with increased miniaturization.

17.2 Wave and reflow soldering

Wave soldering is a joining technology in which the joints are made by solder coming from a standing wave of liquid solder. The wave soldering process is suitable for the following:

- Through-hole components
- Leaded or leadless SMDs, which are glued to the surface of the printed circuit board

Not all SMDs can be wave soldered. Packages with solder balls, and some leadless packages which have solder lands underneath the body, cannot be wave soldered. Also, leaded SMDs with leads having a pitch smaller than ~0.6 mm cannot be wave soldered, due to an increased probability of bridging.

The reflow soldering process involves applying solder paste to a board, followed by component placement and exposure to a temperature profile. Leaded packages, packages with solder balls, and leadless packages are all reflow solderable.

Key characteristics in both wave and reflow soldering are:

- Board specifications, including the board finish, solder masks and vias
- Package footprints, including solder thieves and orientation
- The moisture sensitivity level of the packages
- Package placement
- Inspection and repair
- Lead-free soldering versus SnPb soldering

17.3 Wave soldering

Key characteristics in wave soldering are:

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

- Process issues, such as application of adhesive and flux, clinching of leads, board transport, the solder wave parameters, and the time during which components are exposed to the wave
- Solder bath specifications, including temperature and impurities

17.4 Reflow soldering

Key characteristics in reflow soldering are:

- Lead-free versus SnPb soldering; note that a lead-free reflow process usually leads to higher minimum peak temperatures (see Figure 27) than a SnPb process, thus reducing the process window
- Solder paste printing issues including smearing, release, and adjusting the process window for a mix of large and small components on one board
- Reflow temperature profile; this profile includes preheat, reflow (in which the board is heated to the peak temperature) and cooling down. It is imperative that the peak temperature is high enough for the solder to make reliable solder joints (a solder paste characteristic). In addition, the peak temperature must be low enough that the packages and/or boards are not damaged. The peak temperature of the package depends on package thickness and volume and is classified in accordance with Table 44 and 45

Table 44. SnPb eutectic process (from J-STD-020C)

Package thickness (mm)	Package reflow temperature (°C)	
	Volume (mm³)	
	< 350	≥ 350
< 2.5	235	220
≥ 2.5	220	220

Table 45. Lead-free process (from J-STD-020C)

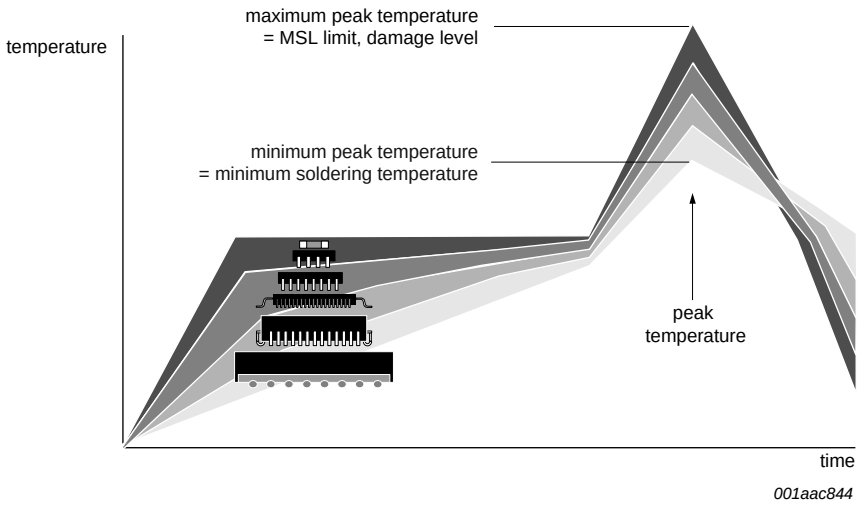
Package thickness (mm)	Package reflow temperature (°C)		
	Volume (mm³)		
	< 350	350 to 2000	> 2000
< 1.6	260	260	260
1.6 to 2.5	260	250	245
> 2.5	250	245	245

Moisture sensitivity precautions, as indicated on the packing, must be respected at all times.

Studies have shown that small packages reach higher temperatures during reflow soldering, see Figure 27.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.



MSL: Moisture Sensitivity Level

Fig 27. Temperature profiles for large and small components

For further information on temperature profiles, refer to Application Note *AN10365 "Surface mount reflow soldering description"*.

18. Abbreviations

Table 46. Abbreviations

Acronym	Description
AI	Auto-Increment
CCW	Counter-ClockWise
CDM	Charged-Device Model
CMOS	Complementary Metal-Oxide Semiconductor
CPU	Central Processing Unit
CW	ClockWise
DMOS	double-Diffused Metal-Oxide Semiconductor
DUT	Device Under Test
ESD	ElectroStatic Discharge
FET	Field-Effect Transistor
Fm+	Fast-mode Plus
GPIO	General Purpose Input/Output
HBM	Human Body Model
HVAC	Heating, Venting and Air Conditioning
I/O	Input/Output
I²C-bus	Inter-Integrated Circuit bus
IC	Integrated Circuit
LED	Light Emitting Diode
LSB	Least Significant Bit

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Table 46. Abbreviations *...continued*

Acronym	Description
NMOS	Negative-channel Metal-Oxide Semiconductor
MSB	Most Significant Bit
PCB	Printed-Circuit Board
pps	pulses per second
PWM	Pulse Width Modulator
POR	Power-On Reset

19. Revision history

Table 47. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
PCA9629 v.1	20120229	Product data sheet	-	-

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

20. Legal information

20.1 Data sheet status

Document status[1][2]	Product status[3]	Definition
Objective [short] data sheet	Development	This document contains data from the objective specification for product development.
Preliminary [short] data sheet	Qualification	This document contains data from the preliminary specification.
Product [short] data sheet	Production	This document contains the product specification.

[1] Please consult the most recently issued document before initiating or completing a design.

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL http://www.nxp.com.

20.2 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Short data sheet — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

Product specification — The information and data provided in a Product data sheet shall define the specification of the product as agreed between NXP Semiconductors and its customer, unless NXP Semiconductors and customer have explicitly agreed otherwise in writing. In no event however, shall an agreement be valid in which the NXP Semiconductors product is deemed to offer functions and qualities beyond those described in the Product data sheet.

20.3 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

20.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

I²C-bus — logo is a trademark of NXP B.V.

21. Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

22. Contents

continued >>

All information provided in this document is subject to legal disclaimers.

© NXP B.V. 2012. All rights reserved.

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© NXP B.V. 2012. **All rights reserved.**

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 29 February 2012
Document identifier: PCA9629

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Motor/Motion/Ignition Controllers & Drivers ***category:***

Click to view products by NXP ***manufacturer:***

Other Similar products are found below :

FSB50550TB2 FSBF15CH60BTH MSVCPM2-63-12 MSVGW45-14-2 MSVGW54-14-3 MSVGW54-14-5 NTE7043 LA6565VR-TLM-E LB11650-E LB1837M-TLM-E LB1845DAZ-XE LC898300XA-MH SS30-TE-L-E 26700 LV8281VR-TLM-H BA5839FP-E2 IRAM236-1067A LA6584JA-AH LB11847L-E NCV70501DW002R2G AH293-PL-B STK672-630CN-E TND315S-TL-2H FNA23060 FSB50250AB FNA41060 MSVB54 MSVBTC50E MSVCPM3-54-12 MSVCPM3-63-12 MSVCPM4-63-12 MSVTA120 FSB50550AB NCV70501DW002G LC898301XA-MH LV8413GP-TE-L-E MSVGW45-14-3 MSVGW45-14-4 MSVGW45-14-5 MSVGW54-14-4 STK984-091A-E MP6519GQ-Z LB11651-E IRSM515-025DA4 LV8127T-TLM-H MC33812EKR2 NCP81382MNTXG TDA21801 LB11851FA-BH NCV70627DQ001R2G