# PCA9622

## 16-bit Fm+ I²C-bus 100 mA 40 V LED driver

Rev. 5 — 2 June 2014 Product data sheet

## 1. General description

The PCA9622 is an I²C-bus controlled 16-bit LED driver optimized for voltage switch dimming and blinking 100 mA Red/Green/Blue/Amber (RGBA) LEDs. Each LED output has its own 8-bit resolution (256 steps) fixed frequency individual PWM controller that operates at 97 kHz with a duty cycle that is adjustable from 0 % to 99.6 % to allow the LED to be set to a specific brightness value. An additional 8-bit resolution (256 steps) group PWM controller has both a fixed frequency of 190 Hz and an adjustable frequency between 24 Hz to once every 10.73 seconds with a duty cycle that is adjustable from 0 % to 99.6 % that is used to either dim or blink all LEDs with the same value.

Each LED output can be off, on (no PWM control), set at its individual PWM controller value or at both individual and group PWM controller values. The PCA9622 operates with a supply voltage range of 2.3 V to 5.5 V and the 100 mA open-drain outputs allow voltages up to 40 V.

The PCA9622 is one of the first LED controller devices in a new Fast-mode Plus (Fm+) family. Fm+ devices offer higher frequency (up to 1 MHz) and more densely populated bus operation (up to 4000 pF).

The active LOW Output Enable input pin ($\overline{OE}$) blinks all the LED outputs and can be used to externally PWM the outputs, which is useful when multiple devices must be dimmed or blinked together without using software control.

Software programmable LED Group and three Sub Call I²C-bus addresses allow all or defined groups of PCA9622 devices to respond to a common I²C-bus address, allowing for example, all red LEDs to be turned on or off at the same time or marquee chasing effect, thus minimizing I²C-bus commands. Seven hardware address pins allow up to 126 devices on the same bus.

The Software Reset (SWRST) Call allows the master to perform a reset of the PCA9622 through the I²C-bus, identical to the Power-On Reset (POR) that initializes the registers to their default state causing the outputs to be set HIGH (LED off). This allows an easy and quick way to reconfigure all device registers to the same condition.

The PCA9622, PCA9625 and PCA9635 software is identical and if the PCA9622 on-chip 100 mA NAND FETs do not provide enough current or voltage to drive the LEDs, then the PCA9635 with larger current or higher voltage external drivers can be used.

## 2. Features and benefits

- 16 LED drivers. Each output programmable at:
  - Off
  - On
  - Programmable LED brightness
  - Programmable group dimming/blinking mixed with individual LED brightness
- 1 MHz Fast-mode Plus compatible I²C-bus interface with 30 mA high drive capability on SDA output for driving high capacitive buses
- 256-step (8-bit) linear programmable brightness per LED output varying from fully off (default) to maximum brightness using a 97 kHz PWM signal
- 256-step group brightness control allows general dimming (using a 190 Hz PWM signal) from fully off to maximum brightness (default)
- 256-step group blinking with frequency programmable from 24 Hz to 10.73 s and duty cycle from 0 % to 99.6 %
- Sixteen open-drain outputs can sink between 0 mA to 100 mA and are tolerant to a maximum off state voltage of 40 V. No input function.
- Output state change programmable on the Acknowledge or the STOP Command to update outputs byte-by-byte or all at the same time (default to 'Change on STOP').
- Active LOW Output Enable ($\overline{OE}$) input pin allows for hardware blinking and dimming of the LEDs
- 7 hardware address pins allow 126 PCA9622 devices to be connected to the same I²C-bus and to be individually programmed
- 4 software programmable I²C-bus addresses (one LED Group Call address and three LED Sub Call addresses) allow groups of devices to be addressed at the same time in any combination (for example, one register used for 'All Call' so that all the PCA9622s on the I²C-bus can be addressed at the same time and the second register used for three different addresses so that ⅓ of all devices on the bus can be addressed at the same time in a group). Software enable and disable for I²C-bus address.
- Software Reset feature (SWRST Call) allows the device to be reset through the I²C-bus
- 25 MHz internal oscillator requires no external components
- Internal power-on reset
- Noise filter on SDA/SCL inputs
- No glitch on power-up
- Supports hot insertion
- Low standby current
- Operating power supply voltage ($V_{DD}$) range of 2.3 V to 5.5 V
- 5.5 V tolerant inputs on non-LED pins
- –40 °C to +85 °C operation
- ESD protection exceeds 2000 V HBM per JESD22-A114 and 1000 V CDM per JESD22-C101
- Latch-up testing is done to JEDEC Standard JESD78 which exceeds 100 mA
- Packages offered: TSSOP32

## 3. Applications

- RGB or RGBA LED drivers
- LED status information
- LED displays
- LCD backlights
- Keypad backlights for cellular phones or handheld devices

## 4. Ordering information

**Table 1. Ordering information**

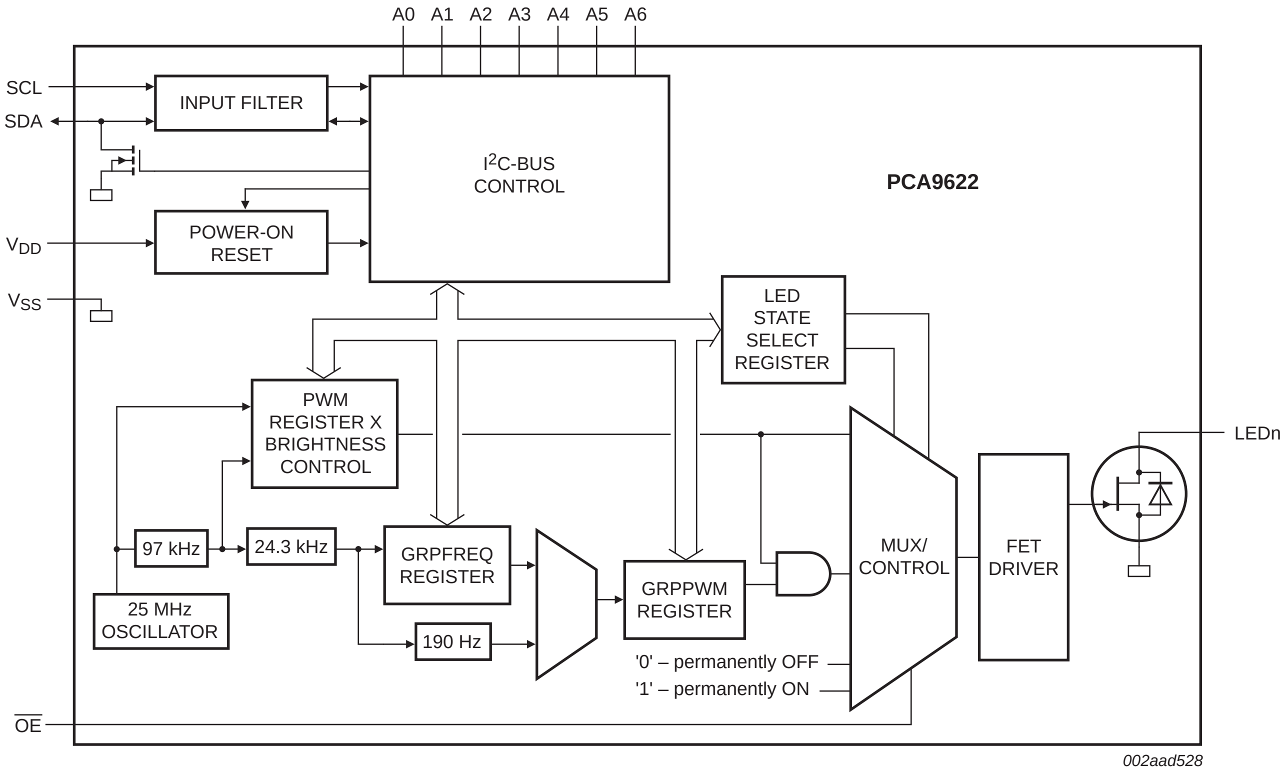
| Type number | Topside marking | Package | | |
|---|---|---|---|---|
| | | Name | Description | Version |
| PCA9622DR | PCA9622DR | TSSOP32 | plastic thin shrink small outline package; 32 leads; body width 6.1 mm; lead pitch 0.65 mm | SOT487-1 |

### 4.1 Ordering options

**Table 2. Ordering options**

| Type number | Orderable part number | Package | Packing method | Minimum order quantity | Temperature |
|---|---|---|---|---|---|
| PCA9622DR | PCA9622DR,118 | TSSOP32 | Reel 13” Q1/T1 *Standard mark SMD | 2000 | $T_{amb}$ = –40 °C to +85 °C |

## 5. Block diagram




**Remark:** Only one LED output shown for clarity.

**Fig 1. Block diagram of PCA9622**

# 6. Pinning information

## 6.1 Pinning

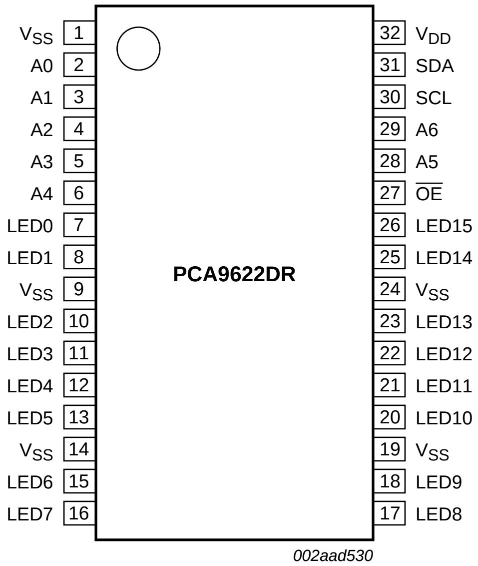



**Fig 2. Pin configuration for TSSOP32**

## 6.2 Pin description

**Table 3. Pin description**

| Symbol | Pin | Type | Description |
|---|---|---|---|
| $V_{SS}$ | 1 | power supply | supply ground |
| A0 | 2 | I | address input 0 |
| A1 | 3 | I | address input 1 |
| A2 | 4 | I | address input 2 |
| A3 | 5 | I | address input 3 |
| A4 | 6 | I | address input 4 |
| LED0 | 7 | O | LED driver 0 |
| LED1 | 8 | O | LED driver 1 |
| $V_{SS}$ | 9 | power supply | supply ground |
| LED2 | 10 | O | LED driver 2 |
| LED3 | 11 | O | LED driver 3 |
| LED4 | 12 | O | LED driver 4 |
| LED5 | 13 | O | LED driver 5 |
| $V_{SS}$ | 14 | power supply | supply ground |
| LED6 | 15 | O | LED driver 6 |
| LED7 | 16 | O | LED driver 7 |
| LED8 | 17 | O | LED driver 8 |
| LED9 | 18 | O | LED driver 9 |

Table 3. Pin description *...continued*

| Symbol | Pin | Type | Description |
|---|---|---|---|
| $V_{SS}$ | 19 | power supply | supply ground |
| LED10 | 20 | O | LED driver 10 |
| LED11 | 21 | O | LED driver 11 |
| LED12 | 22 | O | LED driver 12 |
| LED13 | 23 | O | LED driver 12 |
| $V_{SS}$ | 24 | power supply | supply ground |
| LED14 | 25 | O | LED driver 14 |
| LED15 | 26 | O | LED driver 15 |
| $\overline{OE}$ | 27 | I | active LOW output enable |
| A5 | 28 | I | address input 5 |
| A6 | 29 | I | address input 6 |
| SCL | 30 | I | serial clock line |
| SDA | 31 | I/O | serial data line |
| $V_{DD}$ | 32 | power supply | supply voltage |

# 7. Functional description

Refer to Figure 1 "Block diagram of PCA9622".

## 7.1 Device addresses

Following a START condition, the bus master must output the address of the slave it is accessing.

There are a maximum of 128 possible programmable addresses using the 7 hardware address pins. Two of these addresses, Software Reset and LED All Call, cannot be used because their default power-up state is ON, leaving a maximum of 126 addresses. Using other reserved addresses, as well as any other Sub Call address, reduces the total number of possible addresses even further.

### 7.1.1 Regular I²C-bus slave address

The I²C-bus slave address of the PCA9622 is shown in Figure 3. To conserve power, no internal pull-up resistors are incorporated on the hardware selectable address pins and they must be pulled HIGH or LOW.

**Remark:** Using reserved I²C-bus addresses will interfere with other devices, but only if the devices are on the bus and/or the bus will be open to other I²C-bus systems at some later date. In a closed system where the designer controls the address assignment these addresses can be used since the PCA9622 treats them like any other address. The LED All Call, Software Rest and PCA9564 or PCA9665 slave address (if on the bus) can never be used for individual device addresses.

- PCA9622 LED All Call address (1110 000) and Software Reset (0000 0110) which are active on start-up
- PCA9564 (0000 000) or PCA9665 (1110 000) slave address which is active on start-up

- ‘reserved for future use’ I²C-bus addresses (0000 011, 1111 1XX)
- slave devices that use the 10-bit addressing scheme (1111 0XX)
- slave devices that are designed to respond to the General Call address (0000 000)
- High-speed mode (Hs-mode) master code (0000 1XX)

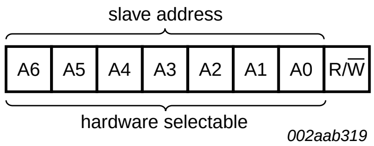



**Fig 3. Slave address**

The last bit of the address byte defines the operation to be performed. When set to logic 1 a read is selected, while a logic 0 selects a write operation.

### 7.1.2 LED All Call I²C-bus address

- Default power-up value (ALLCALLADR register): E0h or 1110 000
- Programmable through I²C-bus (volatile programming)
- At power-up, LED All Call I²C-bus address is enabled. PCA9622 sends an ACK when E0h ($R/\overline{W}$ = 0) or E1h ($R/\overline{W}$ = 1) is sent by the master.

See Section 7.3.8 “ALLCALLADR, LED All Call I²C-bus address” for more detail.

**Remark:** The default LED All Call I²C-bus address (E0h or 1110 000) must not be used as a regular I²C-bus slave address since this address is enabled at power-up. All the PCA9622s on the I²C-bus acknowledge the address if sent by the I²C-bus master.

### 7.1.3 LED Sub Call I²C-bus addresses

- 3 different I²C-bus addresses can be used
- Default power-up values:
  - SUBADR1 register: E2h or 1110 001
  - SUBADR2 register: E4h or 1110 010
  - SUBADR3 register: E8h or 1110 100
- Programmable through I²C-bus (volatile programming)
- At power-up, Sub Call I²C-bus addresses are disabled. PCA9622 does not send an ACK when E2h ($R/\overline{W}$ = 0) or E3h ($R/\overline{W}$ = 1), E4h ($R/\overline{W}$ = 0) or E5h ($R/\overline{W}$ = 1), or E8h ($R/\overline{W}$ = 0) or E9h ($R/\overline{W}$ = 1) is sent by the master.

See Section 7.3.7 “SUBADR1 to SUBADR3, I²C-bus subaddress 1 to 3” for more detail.

**Remark:** The default LED Sub Call I²C-bus addresses may be used as regular I²C-bus slave addresses as long as they are disabled.

### 7.1.4 Software Reset I²C-bus address

The address shown in Figure 4 is used when a reset of the PCA9622 must be performed by the master. The Software Reset address (SWRST Call) must be used with R/W̄ = logic 0. If R/W̄ = logic 1, the PCA9622 does not acknowledge the SWRST. See Section 7.6 "Software reset" for more detail.

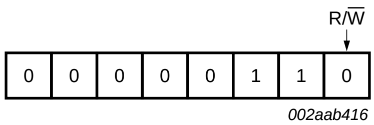


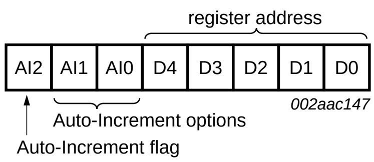

**Fig 4. Software Reset address**

**Remark:** The Software Reset I²C-bus address is a reserved address and cannot be used as a regular I²C-bus slave address or as an LED All Call or LED Sub Call address.

## 7.2 Control register

Following the successful acknowledgement of the slave address, LED All Call address or LED Sub Call address, the bus master sends a byte to the PCA9622, which is stored in the Control register.

The lowest 5 bits are used as a pointer to determine which register is accessed (D[4:0]). The highest 3 bits are used as Auto-Increment flag and Auto-Increment options (AI[2:0]).




reset state = 80h

**Remark:** The Control register does not apply to the Software Reset I²C-bus address.

**Fig 5. Control register**

When the Auto-Increment flag is set (AI2 = logic 1), the five low-order bits of the Control register are automatically incremented after a read or write. This allows the user to program the registers sequentially. Four different types of Auto-Increment are possible, depending on AI1 and AI0 values.

**Table 4. Auto-Increment options**

| AI2 | AI1 | AI0 | Function |
|---|---|---|---|
| 0 | 0 | 0 | no Auto-Increment |
| 1 | 0 | 0 | Auto-Increment for all registers. D[4:0] roll over to ‘0 0000’ after the last register (1 1011) is accessed. |
| 1 | 0 | 1 | Auto-Increment for individual brightness registers only. D[4:0] roll over to ‘0 0010’ after the last register (1 0001) is accessed. |
| 1 | 1 | 0 | Auto-Increment for global control registers only. D[4:0] roll over to ‘1 0010’ after the last register (1 0011) is accessed. |
| 1 | 1 | 1 | Auto-Increment for individual and global control registers only. D[4:0] roll over to ‘0 0010’ after the last register (1 0011) is accessed. |

**Remark:** Other combinations not shown in Table 4 (AI[2:0] = 001, 010, and 011) are reserved and must not be used for proper device operation.

AI[2:0] = 000 is used when the same register must be accessed several times during a single I²C-bus communication, for example, changes the brightness of a single LED. Data is overwritten each time the register is accessed during a write operation.

AI[2:0] = 100 is used when all the registers must be sequentially accessed, for example, power-up programming.

AI[2:0] = 101 is used when the 16 LED drivers must be individually programmed with different values during the same I²C-bus communication, for example, changing color setting to another color setting.

AI[2:0] = 110 is used when the LED drivers must be globally programmed with different settings during the same I²C-bus communication, for example, global brightness or blinking change.

AI[2:0] = 111 is used when individual and global changes must be performed during the same I²C-bus communication, for example, changing a color and global brightness at the same time.

Only the 5 least significant bits D[4:0] are affected by the AI[2:0] bits.

When the Control register is written, the register entry point determined by D[4:0] is the first register that is addressed (read or write operation), and can be anywhere between 0 0000 and 1 1011 (as defined in Table 5). When AI[2] = 1, the Auto-Increment flag is set and the rollover value at which the register increment stops and goes to the next one is determined by AI[2:0]. See Table 4 for rollover values. For example, if the Control register = 1111 0100 (F4h), then the register addressing sequence is (in hexadecimal): 14 → … → 1B → 00 → … → 13 → 02 → … → 13 → 02 → … → 13 → 02 → … as long as the master keeps sending or reading data.

## 7.3 Register definitions

Table 5. Register summary[1]

| Register number | D4 | D3 | D2 | D1 | D0 | Name | Type | Function |
|---|---|---|---|---|---|---|---|---|
| 00h | 0 | 0 | 0 | 0 | 0 | MODE1 | read/write | Mode register 1 |
| 01h | 0 | 0 | 0 | 0 | 1 | MODE2 | read/write | Mode register 2 |
| 02h | 0 | 0 | 0 | 1 | 0 | PWM0 | read/write | brightness control LED0 |
| 03h | 0 | 0 | 0 | 1 | 1 | PWM1 | read/write | brightness control LED1 |
| 04h | 0 | 0 | 1 | 0 | 0 | PWM2 | read/write | brightness control LED2 |
| 05h | 0 | 0 | 1 | 0 | 1 | PWM3 | read/write | brightness control LED3 |
| 06h | 0 | 0 | 1 | 1 | 0 | PWM4 | read/write | brightness control LED4 |
| 07h | 0 | 0 | 1 | 1 | 1 | PWM5 | read/write | brightness control LED5 |
| 08h | 0 | 1 | 0 | 0 | 0 | PWM6 | read/write | brightness control LED6 |
| 09h | 0 | 1 | 0 | 0 | 1 | PWM7 | read/write | brightness control LED7 |
| 0Ah | 0 | 1 | 0 | 1 | 0 | PWM8 | read/write | brightness control LED8 |
| 0Bh | 0 | 1 | 0 | 1 | 1 | PWM9 | read/write | brightness control LED9 |
| 0Ch | 0 | 1 | 1 | 0 | 0 | PWM10 | read/write | brightness control LED10 |
| 0Dh | 0 | 1 | 1 | 0 | 1 | PWM11 | read/write | brightness control LED11 |
| 0Eh | 0 | 1 | 1 | 1 | 0 | PWM12 | read/write | brightness control LED12 |
| 0Fh | 0 | 1 | 1 | 1 | 1 | PWM13 | read/write | brightness control LED13 |
| 10h | 1 | 0 | 0 | 0 | 0 | PWM14 | read/write | brightness control LED14 |
| 11h | 1 | 0 | 0 | 0 | 1 | PWM15 | read/write | brightness control LED15 |
| 12h | 1 | 0 | 0 | 1 | 0 | GRPPWM | read/write | group duty cycle control |
| 13h | 1 | 0 | 0 | 1 | 1 | GRPFREQ | read/write | group frequency |
| 14h | 1 | 0 | 1 | 0 | 0 | LEDOUT0 | read/write | LED output state 0 |
| 15h | 1 | 0 | 1 | 0 | 1 | LEDOUT1 | read/write | LED output state 1 |
| 16h | 1 | 0 | 1 | 1 | 0 | LEDOUT2 | read/write | LED output state 2 |
| 17h | 1 | 0 | 1 | 1 | 1 | LEDOUT3 | read/write | LED output state 3 |
| 18h | 1 | 1 | 0 | 0 | 0 | SUBADR1 | read/write | I²C-bus subaddress 1 |
| 19h | 1 | 1 | 0 | 0 | 1 | SUBADR2 | read/write | I²C-bus subaddress 2 |
| 1Ah | 1 | 1 | 0 | 1 | 0 | SUBADR3 | read/write | I²C-bus subaddress 3 |
| 1Bh | 1 | 1 | 0 | 1 | 1 | ALLCALLADR | read/write | LED All Call I²C-bus address |

[1] Only D[4:0] = 0 0000 to 1 1011 are allowed and are acknowledged. D[4:0] = 1 1100 to 1 1111 are reserved and are not acknowledged.

### 7.3.1 Mode register 1, MODE1

**Table 6. MODE1 - Mode register 1 (address 00h) bit description**

*Legend: * default value.*

| Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|
| 7 | AI2 | read only | 0 | Register Auto-Increment disabled. |
| | | | 1* | Register Auto-Increment enabled. |
| 6 | AI1 | read only | 0* | Auto-Increment bit 1 = 0. |
| | | | 1 | Auto-Increment bit 1 = 1. |
| 5 | AI0 | read only | 0* | Auto-Increment bit 0 = 0. |
| | | | 1 | Auto-Increment bit 0 = 1. |
| 4 | SLEEP[1] | R/W | 0 | Normal mode[2]. |
| | | | 1* | Low-power mode. Oscillator off[3]. |
| 3 | SUB1 | R/W | 0* | PCA9622 does not respond to I²C-bus subaddress 1. |
| | | | 1 | PCA9622 responds to I²C-bus subaddress 1. |
| 2 | SUB2 | R/W | 0* | PCA9622 does not respond to I²C-bus subaddress 2. |
| | | | 1 | PCA9622 responds to I²C-bus subaddress 2. |
| 1 | SUB3 | R/W | 0* | PCA9622 does not respond to I²C-bus subaddress 3. |
| | | | 1 | PCA9622 responds to I²C-bus subaddress 3. |
| 0 | ALLCALL | R/W | 0 | PCA9622 does not respond to LED All Call I²C-bus address. |
| | | | 1* | PCA9622 responds to LED All Call I²C-bus address. |

[1] Bit 4 must be programmed with logic 0 for proper device operation.

[2] It takes 500 μs max. for the oscillator to be up and running once SLEEP bit has been set to logic 0. Timings on LEDn outputs are not guaranteed if PWMx, GRPPWM or GRPFREQ registers are accessed within the 500 μs window.

[3] No blinking or dimming is possible when the oscillator is off.

### 7.3.2 Mode register 2, MODE2

**Table 7. MODE2 - Mode register 2 (address 01h) bit description**

*Legend: * default value.*

| Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|
| 7 | - | read only | 0* | reserved |
| 6 | - | read only | 0* | reserved |
| 5 | DMBLNK | R/W | 0* | group control = dimming |
| | | | 1 | group control = blinking |
| 4 | INVRT | R/W | 0* | reserved; write must always be a logic 0 |
| 3 | OCH | R/W | 0* | outputs change on STOP command[1] |
| | | | 1 | outputs change on ACK |
| 2 | - | R/W | 1* | reserved; write must always be a logic 1[2] |
| 1 | - | R/W | 0* | reserved; write must always be a logic 0[2] |
| 0 | - | R/W | 1* | reserved; write must always be a logic 1[2] |

[1] Change of the outputs at the STOP command allows synchronizing outputs of more than one PCA9622. Applicable to registers from 02h (PWM0) to 17h (LEDOUT) only.

[2] **Remark:** If you change these bits from their default values, the device will not perform as expected.

### 7.3.3 PWM0 to PWM15, individual brightness control

**Table 8. PWM0 to PWM15 - PWM registers 0 to 15 (address 02h to 11h) bit description**
*Legend: * default value.*

| Address | Register | Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|---|---|
| 02h | PWM0 | 7:0 | IDC0[7:0] | R/W | 0000 0000* | PWM0 Individual Duty Cycle |
| 03h | PWM1 | 7:0 | IDC1[7:0] | R/W | 0000 0000* | PWM1 Individual Duty Cycle |
| 04h | PWM2 | 7:0 | IDC2[7:0] | R/W | 0000 0000* | PWM2 Individual Duty Cycle |
| 05h | PWM3 | 7:0 | IDC3[7:0] | R/W | 0000 0000* | PWM3 Individual Duty Cycle |
| 06h | PWM4 | 7:0 | IDC4[7:0] | R/W | 0000 0000* | PWM4 Individual Duty Cycle |
| 07h | PWM5 | 7:0 | IDC5[7:0] | R/W | 0000 0000* | PWM5 Individual Duty Cycle |
| 08h | PWM6 | 7:0 | IDC6[7:0] | R/W | 0000 0000* | PWM6 Individual Duty Cycle |
| 09h | PWM7 | 7:0 | IDC7[7:0] | R/W | 0000 0000* | PWM7 Individual Duty Cycle |
| 0Ah | PWM8 | 7:0 | IDC8[7:0] | R/W | 0000 0000* | PWM8 Individual Duty Cycle |
| 0Bh | PWM9 | 7:0 | IDC9[7:0] | R/W | 0000 0000* | PWM9 Individual Duty Cycle |
| 0Ch | PWM10 | 7:0 | IDC10[7:0] | R/W | 0000 0000* | PWM10 Individual Duty Cycle |
| 0Dh | PWM11 | 7:0 | IDC11[7:0] | R/W | 0000 0000* | PWM11 Individual Duty Cycle |
| 0Eh | PWM12 | 7:0 | IDC12[7:0] | R/W | 0000 0000* | PWM12 Individual Duty Cycle |
| 0Fh | PWM13 | 7:0 | IDC13[7:0] | R/W | 0000 0000* | PWM13 Individual Duty Cycle |
| 10h | PWM14 | 7:0 | IDC14[7:0] | R/W | 0000 0000* | PWM14 Individual Duty Cycle |
| 11h | PWM15 | 7:0 | IDC15[7:0] | R/W | 0000 0000* | PWM15 Individual Duty Cycle |

A 97 kHz fixed frequency signal is used for each output. Duty cycle is controlled through 256 linear steps from 00h (0 % duty cycle = LED output off) to FFh (99.6 % duty cycle = LED output at maximum brightness). Applicable to LED outputs programmed with LDRx = 10 or 11 (LEDOUT0 to LEDOUT3 registers).

$$duty\ cycle = \frac{IDCx[7:0]}{256} \quad (1)$$

### 7.3.4 GRPPWM, group duty cycle control

**Table 9. GRPPWM - Group brightness control register (address 12h) bit description**
*Legend: * default value*

| Address | Register | Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|---|---|
| 12h | GRPPWM | 7:0 | GDC[7:0] | R/W | 1111 1111 | GRPPWM register |

When DMBLNK bit (MODE2 register) is programmed with logic 0, a 190 Hz fixed frequency signal is superimposed with the 97 kHz individual brightness control signal. GRPPWM is then used as a global brightness control allowing the LED outputs to be dimmed with the same value. The value in GRPFREQ is then a 'Don't care'.

General brightness for the 16 outputs is controlled through 256 linear steps from 00h (0 % duty cycle = LED output off) to FFh (99.6 % duty cycle = maximum brightness). Applicable to LED outputs programmed with LDRx = 11 (LEDOUT0 to LEDOUT3 registers).

When DMBLNK bit is programmed with logic 1, GRPPWM and GRPFREQ registers define a global blinking pattern, where GRPFREQ contains the blinking period (from 24 Hz to 10.73 s) and GRPPWM the duty cycle (ON/OFF ratio in %).

$$duty\ cycle = \frac{GDC[7:0]}{256} \tag{2}$$

### 7.3.5 GRPFREQ, group frequency

**Table 10. GRPFREQ - Group Frequency register (address 13h) bit description**
*Legend: * default value.*

| Address | Register | Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|---|---|
| 13h | GRPFREQ | 7:0 | GFRQ[7:0] | R/W | 0000 0000* | GRPFREQ register |

GRPFREQ is used to program the global blinking period when DMBLNK bit (MODE2 register) is equal to 1. Value in this register is a 'Don't care' when DMBLNK = 0. Applicable to LED outputs programmed with LDRx = 11 (LEDOUT0 to LEDOUT3 registers).

Blinking period is controlled through 256 linear steps from 00h (41 ms, frequency 24 Hz) to FFh (10.73 s).

$$global\ blinking\ period = \frac{GFRQ[7:0] + 1}{24}(s) \tag{3}$$

### 7.3.6 LEDOUT0 to LEDOUT3, LED driver output state

**Table 11. LEDOUT0 to LEDOUT3 - LED driver output state register (address 14h to 17h) bit description**

*Legend: * default value.*

| Address | Register | Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|---|---|
| 14h | LEDOUT0 | 7:6 | LDR3 | R/W | 00* | LED3 output state control |
| | | 5:4 | LDR2 | R/W | 00* | LED2 output state control |
| | | 3:2 | LDR1 | R/W | 00* | LED1 output state control |
| | | 1:0 | LDR0 | R/W | 00* | LED0 output state control |
| 15h | LEDOUT1 | 7:6 | LDR7 | R/W | 00* | LED7 output state control |
| | | 5:4 | LDR6 | R/W | 00* | LED6 output state control |
| | | 3:2 | LDR5 | R/W | 00* | LED5 output state control |
| | | 1:0 | LDR4 | R/W | 00* | LED4 output state control |
| 16h | LEDOUT2 | 7:6 | LDR11 | R/W | 00* | LED11 output state control |
| | | 5:4 | LDR10 | R/W | 00* | LED10 output state control |
| | | 3:2 | LDR9 | R/W | 00* | LED9 output state control |
| | | 1:0 | LDR8 | R/W | 00* | LED8 output state control |
| 17h | LEDOUT3 | 7:6 | LDR15 | R/W | 00* | LED15 output state control |
| | | 5:4 | LDR14 | R/W | 00* | LED14 output state control |
| | | 3:2 | LDR13 | R/W | 00* | LED13 output state control |
| | | 1:0 | LDR12 | R/W | 00* | LED12 output state control |

**LDRx = 00 —** LED driver x is off (default power-up state).

**LDRx = 01 —** LED driver x is fully on (individual brightness and group dimming/blinking not controlled).

**LDRx = 10 —** LED driver x individual brightness can be controlled through its PWMx register.

**LDRx = 11 —** LED driver x individual brightness and group dimming/blinking can be controlled through its PWMx register and the GRPPWM registers.

### 7.3.7 SUBADR1 to SUBADR3, I²C-bus subaddress 1 to 3

**Table 12. SUBADR1 to SUBADR3 - I²C-bus subaddress registers 0 to 3 (address 18h to 1Ah) bit description**

*Legend: * default value.*

| Address | Register | Bit | Symbol | Access | Value | Description |
|---|---|---|---|---|---|---|
| 18h | SUBADR1 | 7:1 | A1[7:1] | R/W | 1110 001* | I²C-bus subaddress 1 |
| | | 0 | A1[0] | R only | 0* | reserved |
| 19h | SUBADR2 | 7:1 | A2[7:1] | R/W | 1110 010* | I²C-bus subaddress 2 |
| | | 0 | A2[0] | R only | 0* | reserved |
| 1Ah | 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 SUBx bit in MODE1 register is equal to 0).

Once subaddresses have been programmed to their right values, SUBx bits must be set to logic 1 in order to have the device acknowledging these addresses (MODE1 register).

Only the 7 MSBs representing the I2C-bus subaddress are valid. The LSB in SUBADRx register is a read-only bit (0).

When SUBx is set to logic 1, the corresponding I2C-bus subaddress can be used during either an I2C-bus read or write sequence.

#### 7.3.8 ALLCALLADR, LED All Call I2C-bus address

**Table 13. ALLCALLADR - LED All Call I2C-bus address register (address 1Bh) bit description**

*Legend: * default value.*

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

The LED All Call I2C-bus address allows all the PCA9622s on the bus to be programmed at the same time (ALLCALL bit in register MODE1 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. The register address can also be programmed as a Sub Call.

Only the 7 MSBs representing the All Call I2C-bus address are valid. The LSB in ALLCALLADR register is a read-only bit (0).

If ALLCALL bit = 0, the device does not acknowledge the address programmed in register ALLCALLADR.

### 7.4 Active LOW output enable input

The active LOW output enable ($\overline{OE}$) pin, allows enabling or disabling all the LED outputs at the same time.

- When a LOW level is applied to $\overline{OE}$ pin, all the LED outputs are enabled.
- When a HIGH level is applied to $\overline{OE}$ pin, all the LED outputs are high-impedance.

The $\overline{OE}$ pin can be used as a synchronization signal to switch on/off several PCA9622 devices at the same time. This requires an external clock reference that provides blinking period and the duty cycle.

The $\overline{OE}$ pin can also be used as an external dimming control signal. The frequency of the external clock must be high enough not to be seen by the human eye, and the duty cycle value determines the brightness of the LEDs.

**Remark:** Do not use $\overline{OE}$ as an external blinking control signal when internal global blinking is selected (DMBLNK = 1, MODE2 register) since it results in an undefined blinking pattern. Do not use $\overline{OE}$ as an external dimming control signal when internal global dimming is selected (DMBLNK = 0, MODE2 register) since it results in an undefined dimming pattern.

**Remark:** During power-down, slow decay of voltage supplies may keep LEDs illuminated. Consider disabling LED outputs using HIGH level applied to $\overline{OE}$ pin.

## 7.5 Power-on reset

When power is applied to $V_{DD}$, an internal power-on reset holds the PCA9622 in a reset condition until $V_{DD}$ has reached $V_{POR}$. At this point, the reset condition is released and the PCA9622 registers and $I^2C$-bus state machine are initialized to their default states (all zeroes) causing all the channels to be deselected. Thereafter, $V_{DD}$ must be lowered below 0.2 V to reset the device.

## 7.6 Software reset

The Software Reset Call (SWRST Call) allows all the devices in the $I^2C$-bus to be reset to the power-up state value through a specific formatted $I^2C$-bus command. To be performed correctly, it implies that the $I^2C$-bus is functional and that there is no device hanging the bus.

The SWRST Call function is defined as the following:

1. A START command is sent by the $I^2C$-bus master.
2. The reserved SWRST $I^2C$-bus address '0000 011' with the R/$\overline{W}$ bit set to '0' (write) is sent by the $I^2C$-bus master.
3. The PCA9622 device(s) acknowledge(s) after seeing the SWRST Call address '0000 0110' (06h) only. If the R/$\overline{W}$ bit is set to '1' (read), no acknowledge is returned to the $I^2C$-bus master.
4. Once the SWRST Call address has been sent and acknowledged, the master sends 2 bytes with 2 specific values (SWRST data byte 1 and byte 2):
    a. Byte 1 = A5h: the PCA9622 acknowledges this value only. If byte 1 is not equal to A5h, the PCA9622 does not acknowledge it.
    b. Byte 2 = 5Ah: the PCA9622 acknowledges this value only. If byte 2 is not equal to 5Ah, then the PCA9622 does not acknowledge it.

    If more than 2 bytes of data are sent, the PCA9622 does not acknowledge any more.
5. Once the right 2 bytes (SWRST data byte 1 and byte 2 only) have been sent and correctly acknowledged, the master sends a STOP command to end the SWRST Call: the PCA9622 then resets to the default value (power-up value) and is ready to be addressed again within the specified bus free time ($t_{BUF}$).

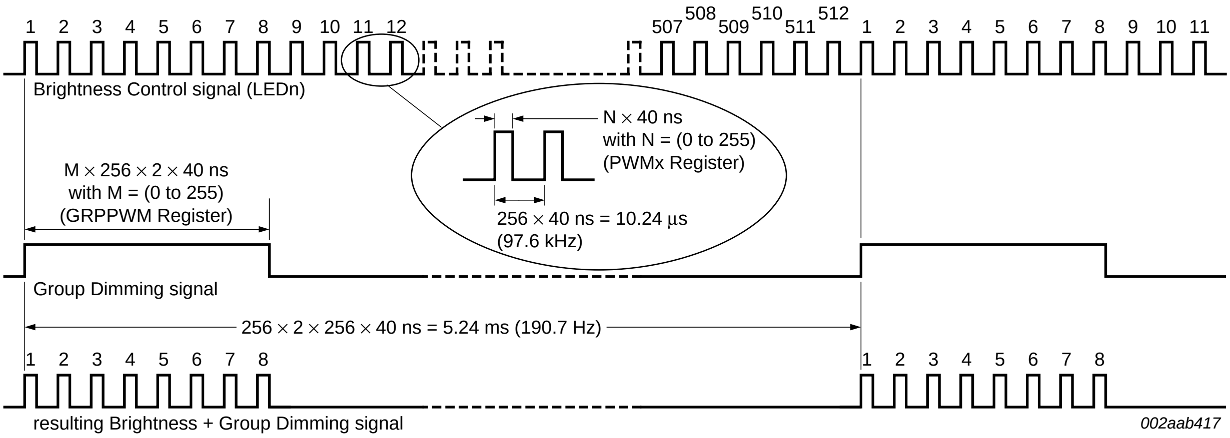
The $I^2C$-bus master must interpret a non-acknowledge from the PCA9622 (at any time) as a 'SWRST Call Abort'. The PCA9622 does not initiate a reset of its registers. This happens only when the format of the SWRST Call sequence is not correct.

## 7.7 Individual brightness control with group dimming/blinking

A 97 kHz fixed frequency signal with programmable duty cycle (8 bits, 256 steps) is used to control individually the brightness for each LED.

On top of this signal, one of the following signals can be superimposed (this signal can be applied to the 4 LED outputs):

- A lower 190 Hz fixed frequency signal with programmable duty cycle (8 bits, 256 steps) is used to provide a global brightness control.
- A programmable frequency signal from 24 Hz to $\frac{1}{10.73}$ Hz (8 bits, 256 steps) with programmable duty cycle (8 bits, 256 steps) is used to provide a global blinking control.




Minimum pulse width for LEDn Brightness Control is 40 ns.

Minimum pulse width for Group Dimming is 20.48 µs.

When M = 1 (GRPPWM register value), the resulting LEDn Brightness Control + Group Dimming signal has 2 pulses of the LED Brightness Control signal (pulse width = N × 40 ns, with 'N' defined in PWMx register).

This resulting Brightness + Group Dimming signal above shows a resulting Control signal with M = 4 (8 pulses).

**Fig 6. Brightness + Group Dimming signals**

# 8. Characteristics of the I²C-bus

The I²C-bus is for 2-way, 2-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 are interpreted as control signals (see Figure 7).

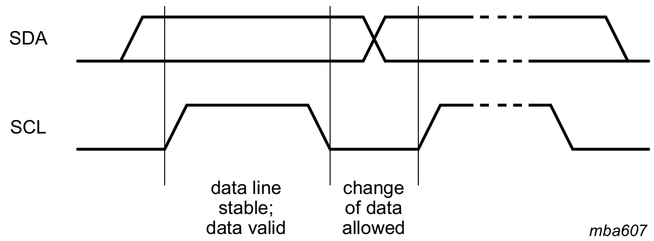



**Fig 7. Bit transfer**

### 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 8).

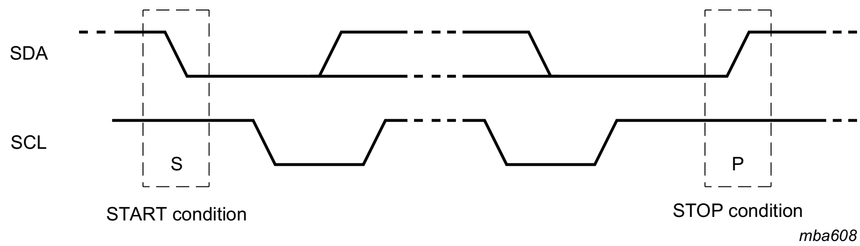



**Fig 8. Definition of START and STOP conditions**

## 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 9).

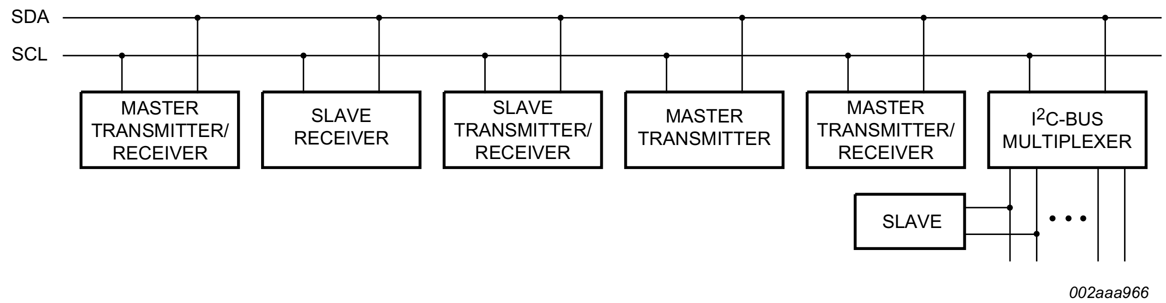



**Fig 9. System configuration**

## 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 8 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.

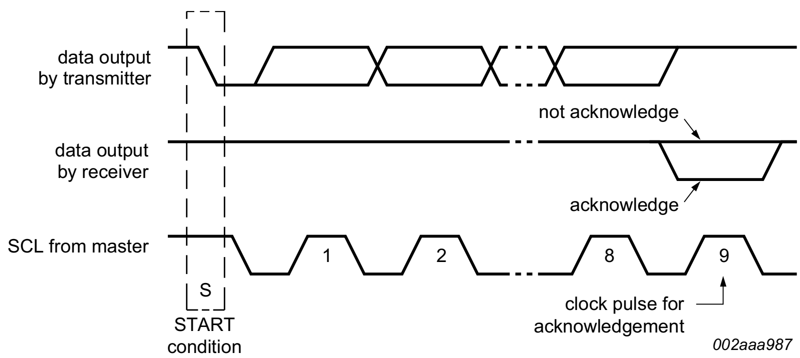
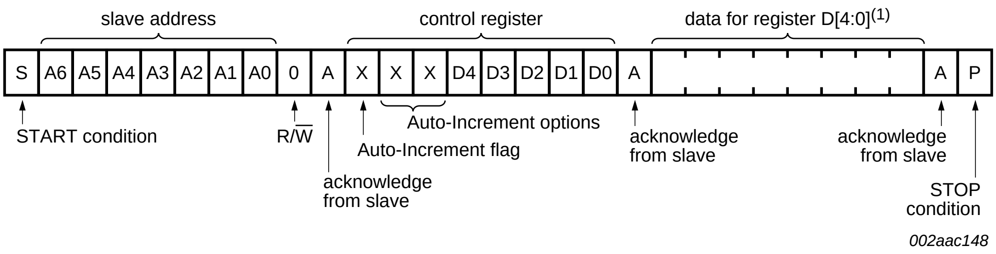



**Fig 10. Acknowledgement on the I2C-bus**

# 9. Bus transactions




(1) See Table 5 for register definition.

**Fig 11. Write to a specific register**

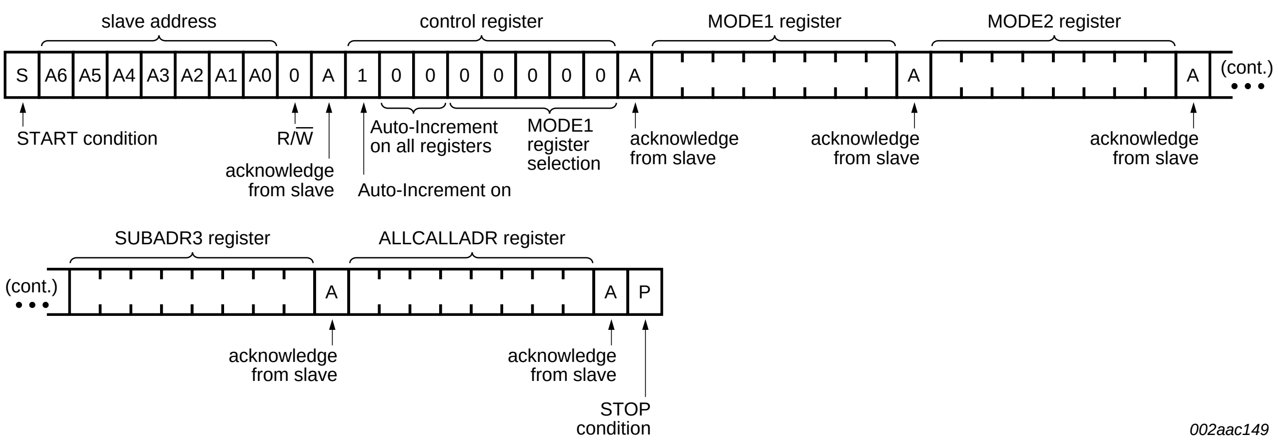



**Fig 12. Write to all registers using the Auto-Increment feature**

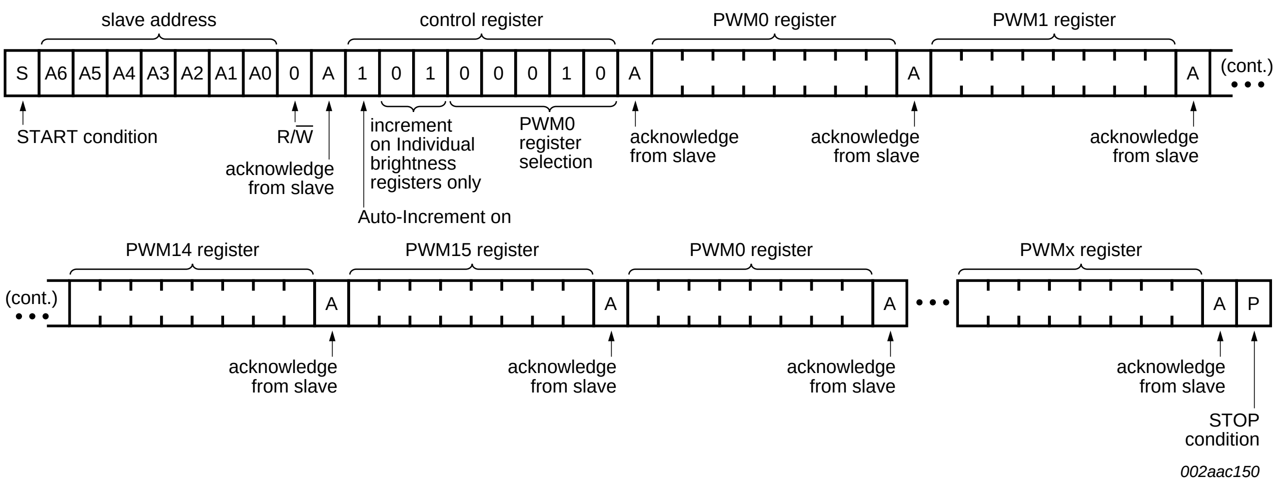



**Fig 13. Multiple writes to Individual Brightness registers only using the Auto-Increment feature**

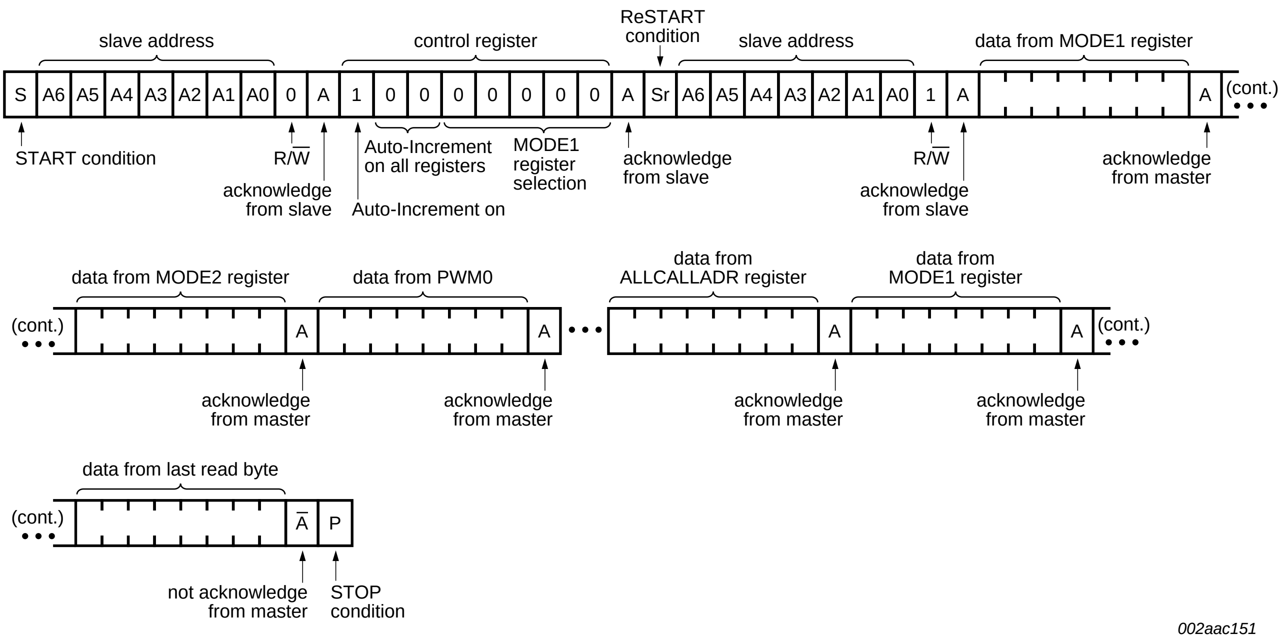
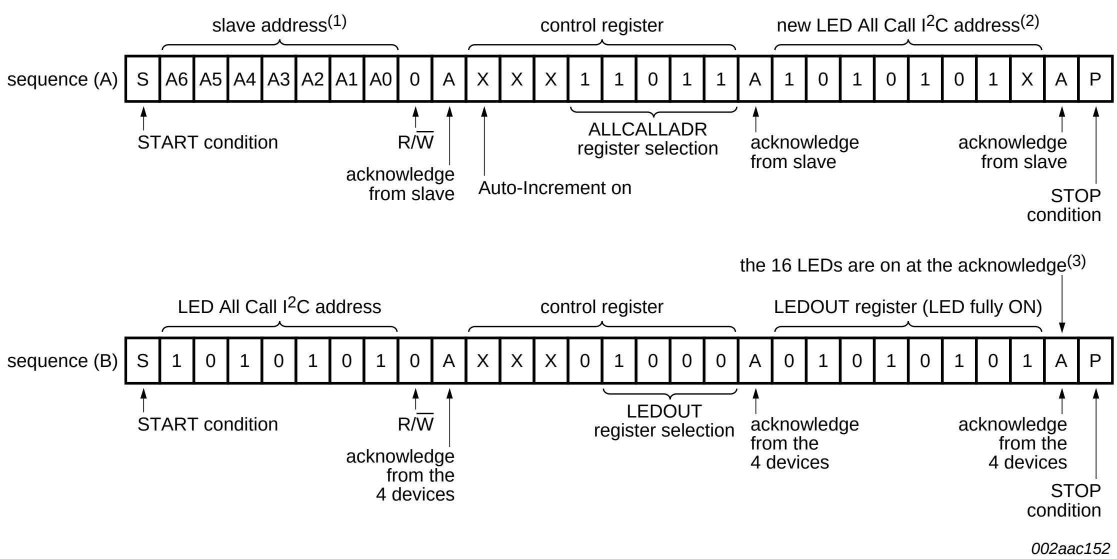



**Fig 14. Read all registers using the Auto-Increment feature**



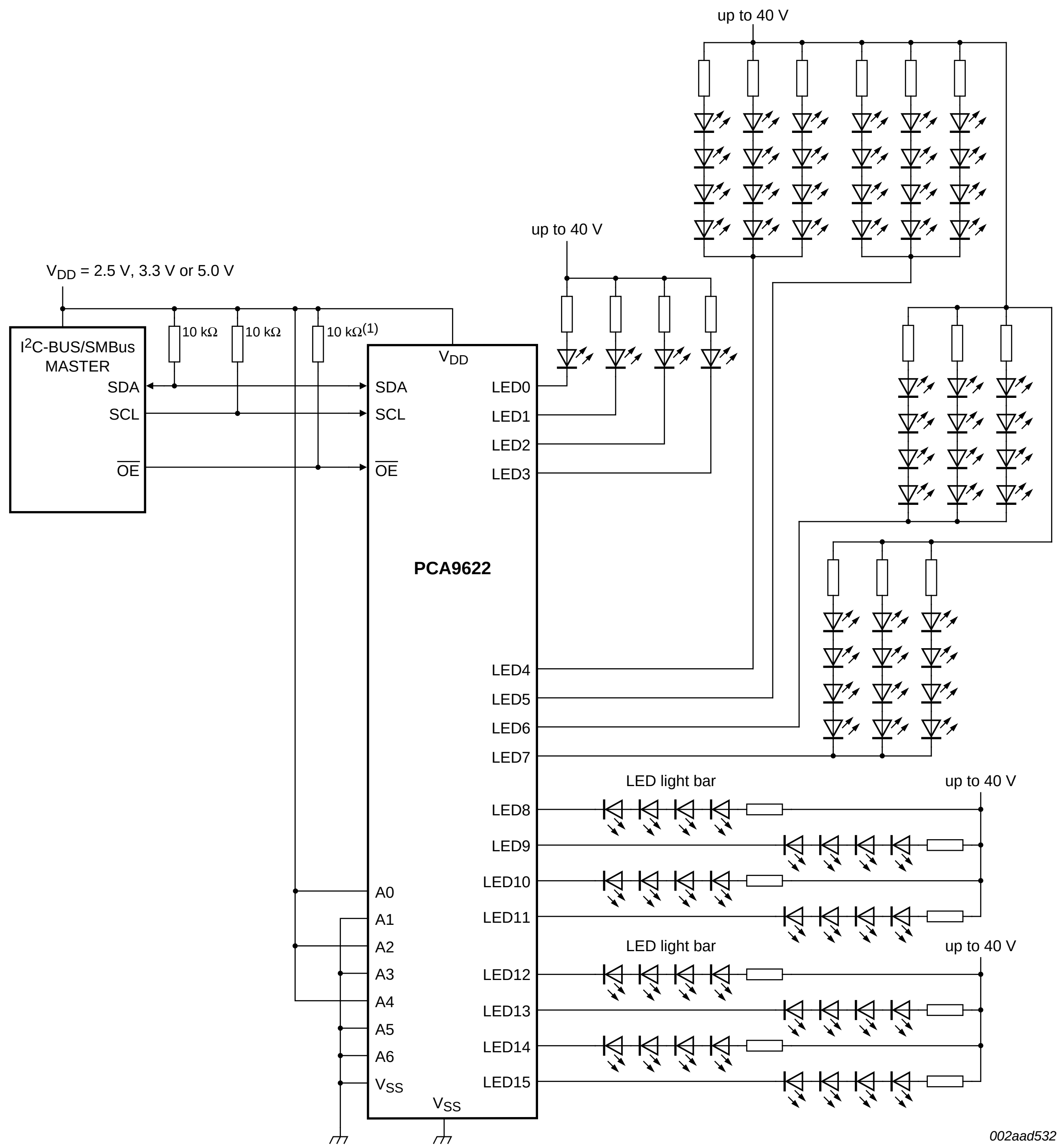

(1) In this example, several PCA9622s are used and the same sequence (A) (above) is sent to each of them.

(2) ALLCALL bit in MODE1 register is equal to 1 for this example.

(3) OCH bit in MODE2 register is equal to 1 for this example.

**Fig 15. LED All Call I2C-bus address programming and LED All Call sequence example**

## 10. Application design-in information




(1) OE requires pull-up resistor if control signal from the master is open-drain.

I²C-bus address = 0010 101x.

**Remark:** During power-down, slow decay of voltage supplies may keep LEDs illuminated. Consider disabling LED outputs using HIGH level applied to OE pin.

**Fig 16. Typical application**

## 10.1 Junction temperature calculation

A device junction temperature can be calculated when the ambient temperature or the case temperature is known.

When the ambient temperature is known, the junction temperature is calculated using Equation 4 and the ambient temperature, junction to ambient thermal resistance and power dissipation.

$$T_j = T_{amb} + R_{th(j\text{-}a)} \times P_{tot} \quad (4)$$

where:

$T_j$ = junction temperature

$T_{amb}$ = ambient temperature

$R_{th(j-a)}$ = junction to ambient thermal resistance

$P_{tot}$ = (device) total power dissipation

When the case temperature is known, the junction temperature is calculated using Equation 5 and the case temperature, junction to case thermal resistance and power dissipation.

$$T_j = T_{case} + R_{th(j\text{-}c)} \times P_{tot} \quad (5)$$

where:

$T_j$ = junction temperature

$T_{case}$ = case temperature

$R_{th(j-c)}$ = junction to case thermal resistance

$P_{tot}$ = (device) total power dissipation

Here are two examples regarding how to calculate the junction temperature using junction to case and junction to ambient thermal resistance. In the first example (Section 10.1.1), given the operating condition and the junction to ambient thermal resistance, the junction temperature of PCA9622DR, in the TSSOP32 package, is calculated for a system operating condition in 50 °C[1] ambient temperature. In the second example (Section 10.1.2), based on a specific customer application requirement where only the case temperature is known, applying the junction to case thermal resistance equation, the junction temperature of the PCA9622DR, in the TSSOP32 package, is calculated.

1. 50 °C is a typical temperature inside an enclosed system. The designers should feel free, as needed, to perform their own calculation using the examples.

### 10.1.1 Example 1: $T_j$ calculation of PCA9622DR, in TSSOP32 package, when $T_{amb}$ is known

$R_{th(j-a)}$ = 83 °C/W

$T_{amb}$ = 50 °C

LED output low voltage (LED $V_{OL}$) = 0.5 V

LED output current per channel = 80 mA

Number of outputs = 16

$I_{DD(max)}$ = 12 mA

$V_{DD(max)}$ = 5.5 V

$I^2C$-bus clock (SCL) maximum sink current = 25 mA

$I^2C$-bus data (SDA) maximum sink current = 25 mA

1. Find $P_{tot}$ (device total power dissipation):
   - output total power = 80 mA × 16 × 0.5 V = 640 mW
   - chip core power consumption = 12 mA × 5.5 V = 66 mW
   - SCL power dissipation = 25 mA × 0.4 V = 10 mW
   - SDA power dissipation = 25 mA × 0.4 V = 10 mW

   $P_{tot}$ = (640 + 66 + 10 + 10) mW = **726 mW**
2. Find $T_j$ (junction temperature):

   $T_j = (T_{amb} + R_{th(j-a)} \times P_{tot})$ = (50 °C + 83 °C/W × 726 mW) = **110.26 °C**

### 10.1.2 Example 2: $T_j$ calculation where only $T_{case}$ is known

This example uses a customer's specific application of the PCA9622DR, 16-channel LED controller in the TSSOP32 package, where only the case temperature ($T_{case}$) is known.

$T_j = T_{case} + R_{th(j-c)} \times P_{tot}$, where:

$R_{th(j-c)}$ = 23 °C/W

$T_{case}$ (measured) = 94.6 °C

$V_{OL}$ of LED ~ 0.5 V

$I_{DD(max)}$ = 12 mA

$V_{DD(max)}$ = 5.5 V

LED output voltage LOW = 0.5 V

LED output current:

60 mA on 1 port = (60 mA × 1)

50 mA on 6 ports = (50 mA × 6)

40 mA on 2 ports = (40 mA × 2)

20 mA on 7 ports = (20 mA × 7)

$I^2C$-bus maximum sink current on clock line = 25 mA

$I^2C$-bus maximum sink current on data line = 25 mA

1. Find $P_{tot}$ (device total power dissipation)
   - output current (60 mA × 1 port); output power (60 mA × 1 × 0.5 V) = 30 mW
   - output current (50 mA × 6 ports); output power (50 mA × 6 × 0.5 V) = 150 mW
   - output current (40 mA × 2 ports); output power (40 mA × 2 × 0.5 V) = 40 mW
   - output current (20 mA × 7 ports); output power (20 mA × 7 × 0.5 V) = 70 mW

   Output total power = **290 mW**
   - chip core power consumption = 12 mA × 5.5 V = 66 mW
   - SCL power dissipation = 25 mA × 0.4 V = 10 mW
   - SDA power dissipation = 25 mA × 0.4 V = 10 mW

   $P_{tot}$ (device total power dissipation) = 376 mW
2. Find $T_j$ (junction temperature):

   $T_j = T_{case} + R_{th(j-a)} \times P_{tot}$ = 94.6 °C + 23 °C/W × 376 mW = **103.25 °C**

# 11. Limiting values

**Table 14. 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 |
| $V_{drv(LED)}$ | LED driver voltage | | $V_{SS}$ – 0.5 | 40 | V |
| $I_{O(LEDn)}$ | output current on pin LEDn | | - | 100 | mA |
| $I_{OL(tot)}$ | total LOW-level output current | $V_{OL}$ = 0.5 V [1] | 1600 | - | mA |
| $I_{SS}$ | ground supply current | per $V_{SS}$ pin | - | 800 | mA |
| $P_{tot}$ | total power dissipation | $T_{amb}$ = 25 °C | - | 1.8 | W |
| | | $T_{amb}$ = 85 °C | - | 0.72 | W |
| P/ch | power dissipation per channel | $T_{amb}$ = 25 °C | - | 100 | mW |
| | | $T_{amb}$ = 85 °C | - | 45 | mW |
| $T_j$ | junction temperature | [2] | - | 125 | °C |
| $T_{stg}$ | storage temperature | | –65 | +150 | °C |
| $T_{amb}$ | ambient temperature | operating | –40 | +85 | °C |

[1] Each bit must be limited to a maximum of 100 mA and the total package limited to 1600 mA due to internal busing limits.

[2] Refer to Section 10.1 for junction temperature calculation.

**Table 15. TSSOP32 power dissipation and output current capability**

| Measurement | TSSOP32 |
|---|---|
| $T_{amb}$ = 25 °C | |
| maximum power dissipation (chip + output drivers) | 1200 mW |
| maximum power dissipation (output drivers only) | 1110 mW |
| maximum drive current per channel | $< \frac{1110\ mW}{16\text{-}bit \times 0.5\ V} = 138.8\ mA$ [1] |
| $T_{amb}$ = 60 °C | |
| maximum power dissipation (chip + output drivers) | 723 mW |
| maximum power dissipation (output drivers only) | 637 mW |
| maximum drive current per channel | $< \frac{637\ mW}{16\text{-}bit \times 0.5\ V} = 79.6\ mA$ |
| $T_{amb}$ = 80 °C | |
| maximum power dissipation (chip + output drivers) | 542 mW |
| maximum power dissipation (output drivers only) | 456 mW |
| maximum drive current per channel | $< \frac{456\ mW}{16\text{-}bit \times 0.5\ V} = 57\ mA$ |

[1] This value signifies package's ability to handle more than 100 mA per output driver. The device's maximum current rating per output is 100 mA.

## 12. Thermal characteristics

**Table 16. Thermal characteristics**

| Symbol | Parameter | Conditions | Typ | Unit |
|---|---|---|---|---|
| $R_{th(j-a)}$ | thermal resistance from junction to ambient | TSSOP32 [1] | 83 | °C/W |
| $R_{th(j-c)}$ | thermal resistance from junction to case | TSSOP32 [1] | 23 | °C/W |

[1] Calculated in accordance with JESD 51-7.

## 13. Static characteristics

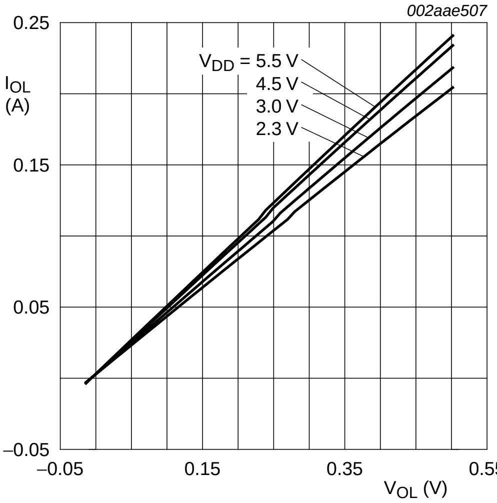
**Table 17. Static characteristics**

*$V_{DD}$ = 2.3 V to 5.5 V; $V_{SS}$ = 0 V; $T_{amb}$ = –40 °C to +85 °C; unless otherwise specified.*

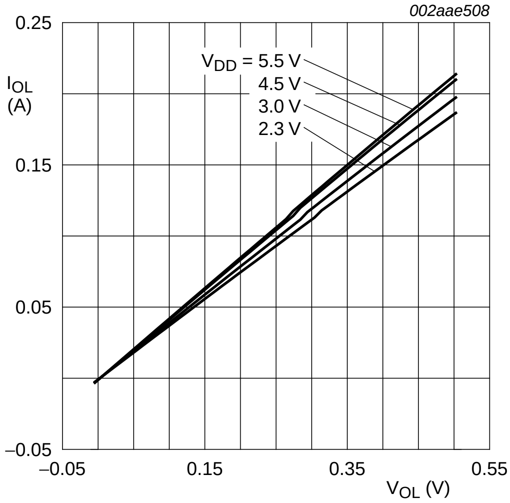
| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Supply** | | | | | | |
| $V_{DD}$ | supply voltage | | 2.3 | - | 5.5 | V |
| $I_{DD}$ | supply current | on pin $V_{DD}$; operating mode; no load; $f_{SCL}$ = 1 MHz | | | | |
| | | $V_{DD}$ = 2.7 V | - | 0.2 | 4 | mA |
| | | $V_{DD}$ = 3.6 V | - | 2 | 6 | mA |
| | | $V_{DD}$ = 5.5 V | - | 8.5 | 12 | mA |
| $I_{stb}$ | standby current | on pin $V_{DD}$; no load; $f_{SCL}$ = 0 Hz; I/O = inputs; $V_I$ = $V_{DD}$ | | | | |
| | | $V_{DD}$ = 2.7 V | - | 1.3 | 5 | μA |
| | | $V_{DD}$ = 3.6 V | - | 1.8 | 6 | μA |
| | | $V_{DD}$ = 5.5 V | - | 3.2 | 7 | μA |
| $V_{POR}$ | power-on reset voltage | no load; $V_I$ = $V_{DD}$ or $V_{SS}$ [1] | - | 1.70 | 2.0 | V |
| **Input SCL; input/output SDA** | | | | | | |
| $V_{IL}$ | LOW-level input voltage | | –0.5 | - | +0.3$V_{DD}$ | V |
| $V_{IH}$ | HIGH-level input voltage | | 0.7$V_{DD}$ | - | 5.5 | V |
| $I_{OL}$ | LOW-level output current | $V_{OL}$ = 0.4 V; $V_{DD}$ = 2.3 V | 20 | - | - | mA |
| | | $V_{OL}$ = 0.4 V; $V_{DD}$ = 5.0 V | 30 | - | - | 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 |
| **LED driver outputs** | | | | | | |
| $V_{drv(LED)}$ | LED driver voltage | | 0 | - | 40 | V |
| $I_{OL}$ | LOW-level output current | $V_{OL}$ = 0.5 V [2] | 100 | - | - | mA |
| $I_{LOH}$ | HIGH-level output leakage current | $V_{drv(LED)}$ = 5 V | - | - | ±1 | μA |
| | | $V_{drv(LED)}$ = 40 V | - | ±1 | 15 | μA |
| $R_{on}$ | ON-state resistance | $V_{drv(LED)}$ = 40 V; $V_{DD}$ = 2.3 V | - | 2 | 5 | Ω |
| $C_o$ | output capacitance | | - | 2.5 | 5 | pF |
| **OE input** | | | | | | |
| $V_{IL}$ | LOW-level input voltage | | –0.5 | - | +0.3$V_{DD}$ | V |
| $V_{IH}$ | HIGH-level input voltage | | 0.7$V_{DD}$ | - | 5.5 | V |
| $I_{LI}$ | input leakage current | | –1 | - | +1 | μA |
| $C_i$ | input capacitance | | - | 15 | 40 | pF |
| **Address inputs** | | | | | | |
| $V_{IL}$ | LOW-level input voltage | | –0.5 | - | +0.3$V_{DD}$ | V |
| $V_{IH}$ | HIGH-level input voltage | | 0.7$V_{DD}$ | - | 5.5 | V |
| $I_{LI}$ | input leakage current | | –1 | - | +1 | μA |
| $C_i$ | input capacitance | | - | 3.7 | 5 | pF |

[1] $V_{DD}$ must be lowered to 0.2 V in order to reset part.

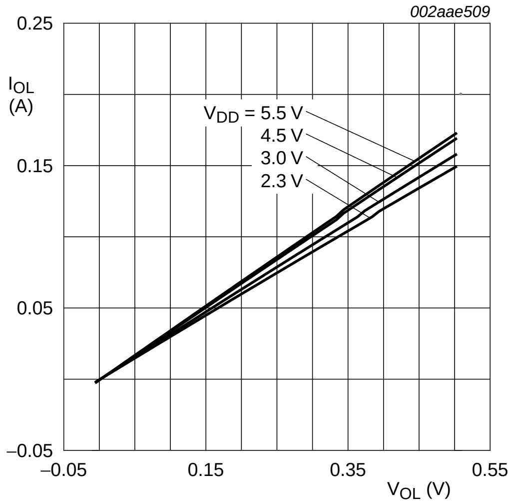
[2] Each bit must be limited to a maximum of 100 mA and the total package limited to 1600 mA due to internal busing limits.




a. $T_{amb}$ = –40 °C




b. $T_{amb}$ = 25 °C




c. $T_{amb}$ = 85 °C

**Fig 17. $V_{OL}$ versus $I_{OL}$**

# 14. Dynamic characteristics

**Table 18. Dynamic characteristics**

| 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 | | 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 | [1] | 0.3 | 3.45 | 0.1 | 0.9 | 0.05 | 0.45 | µs |
| $t_{VD;DAT}$ | data valid time | [2] | 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 | [3][4] | - | 300 | $20 + 0.1C_b$[5] | 300 | - | 120 | ns |
| $t_r$ | rise time of both SDA and SCL signals | | - | 1000 | $20 + 0.1C_b$[5] | 300 | - | 120 | ns |
| $t_{SP}$ | pulse width of spikes that must be suppressed by the input filter | [6] | - | 50 | - | 50 | - | 50 | ns |
| **Output propagation delay** | | | | | | | | | |
| $t_{PLH}$ | LOW to HIGH propagation delay | $\overline{OE}$ to LEDn; MODE2[1:0] = 01 | - | - | - | - | - | 150 | ns |
| $t_{PHL}$ | HIGH to LOW propagation delay | $\overline{OE}$ to LEDn; MODE2[1:0] = 01 | - | - | - | - | - | 150 | ns |
| **Output port timing** | | | | | | | | | |
| $t_{d(SCL-Q)}$ | delay time from SCL to data output | SCL to LEDn; MODE2[3] = 1; outputs change on ACK | - | - | - | - | - | 450 | ns |
| $t_{d(SDA-Q)}$ | delay time from SDA to data output | SDA to LEDn; MODE2[3] = 0; outputs change on STOP condition | - | - | - | - | - | 450 | ns |

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

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

[3] 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) in order to bridge the undefined region of SCL's falling edge.

[4] 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 $t_f$.

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

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

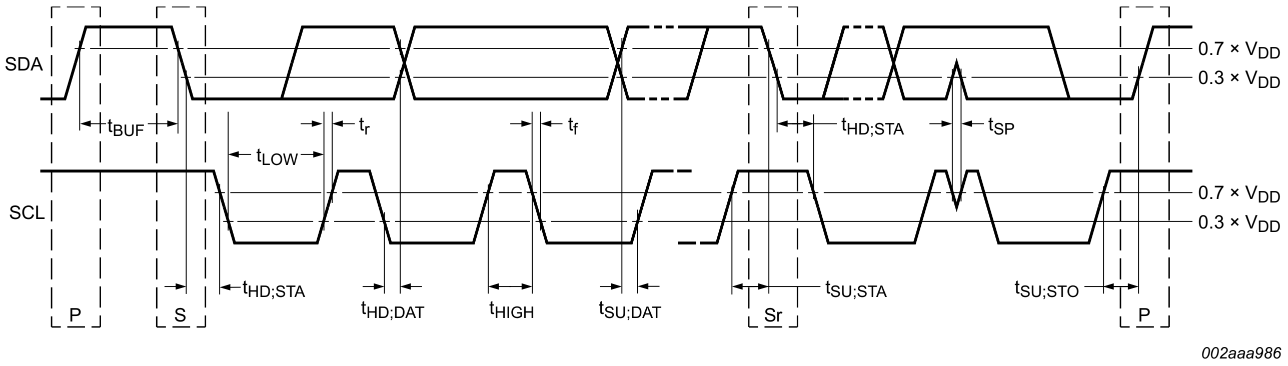
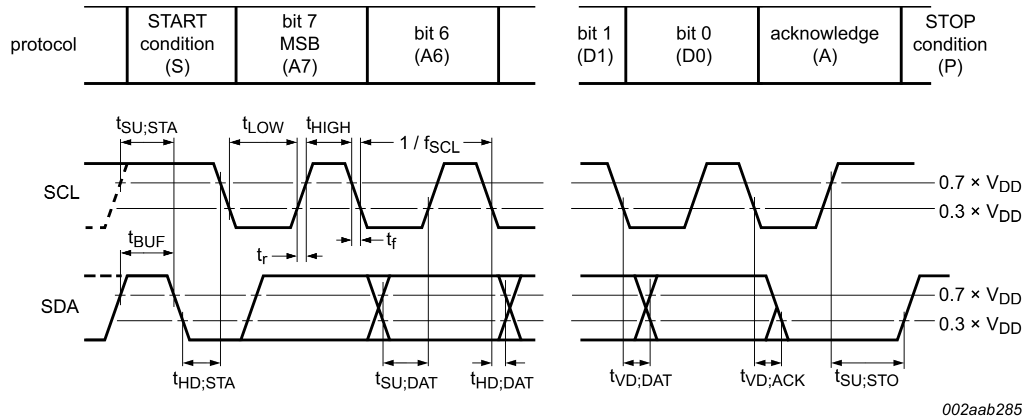


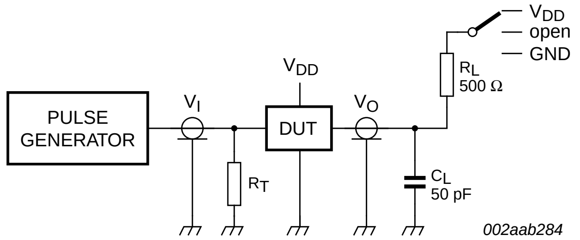

**Fig 18. Definition of timing**




Rise and fall times refer to $V_{IL}$ and $V_{IH}$.

**Fig 19. I²C-bus timing diagram**

## 15. Test information




$R_L$ = Load resistor for LEDn. $R_L$ for SDA and SCL > 1 kΩ (3 mA or less current).

$C_L$ = Load capacitance includes jig and probe capacitance.

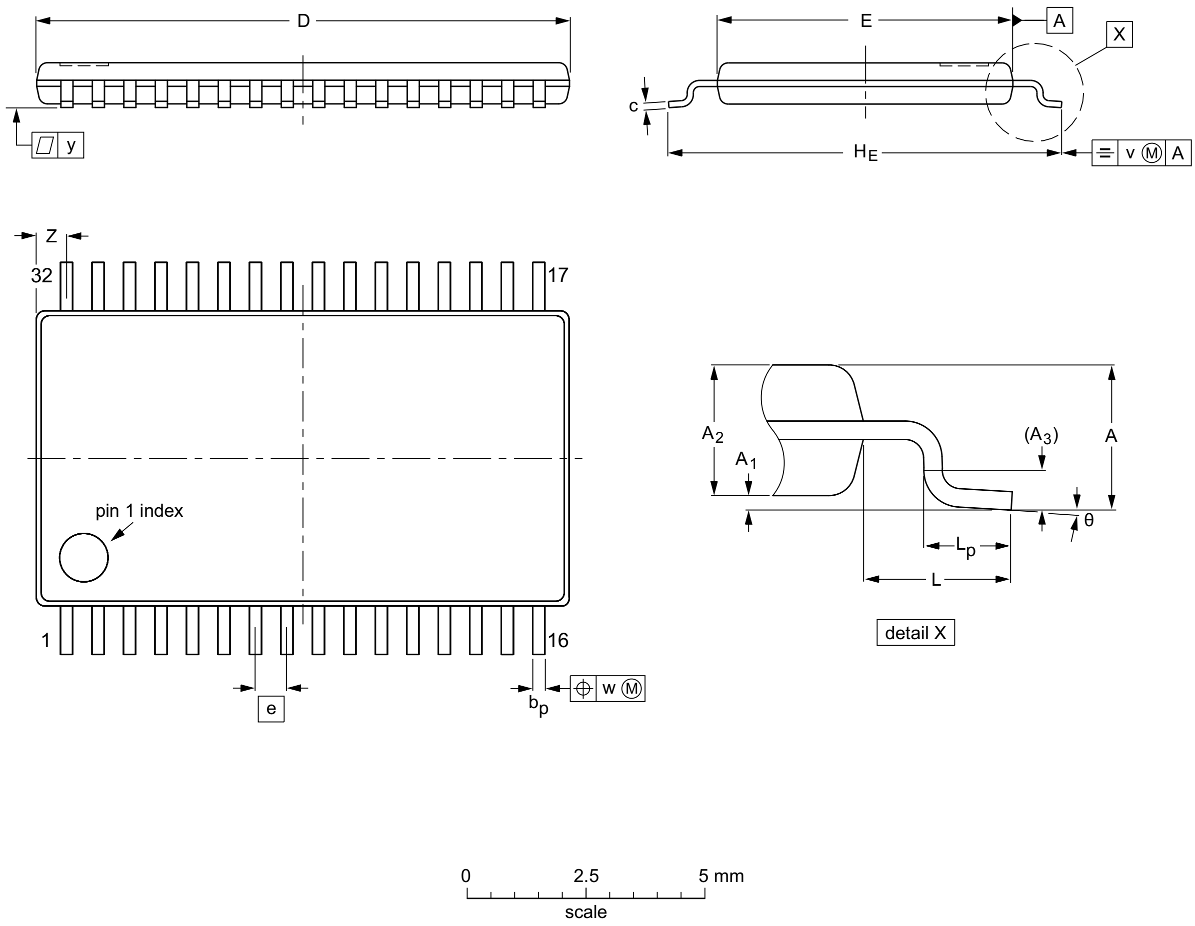
$R_T$ = Termination resistance should be equal to the output impedance $Z_o$ of the pulse generators.

**Fig 20. Test circuitry for switching times**

# 16. Package outline

**TSSOP32: plastic thin shrink small outline package; 32 leads; body width 6.1 mm; lead pitch 0.65 mm**

**SOT487-1**




**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 | $L_p$ | v | w | y | Z | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1.1 | 0.15<br>0.05 | 0.95<br>0.85 | 0.25 | 0.30<br>0.19 | 0.20<br>0.09 | 11.1<br>10.9 | 6.2<br>6.0 | 0.65 | 8.3<br>7.9 | 1 | 0.75<br>0.50 | 0.2 | 0.1 | 0.1 | 0.78<br>0.48 | 8°<br>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 | | | |
| SOT487-1 | | MO-153 | | | | ~~99-12-27~~<br>03-02-18 |

**Fig 21. Package outline SOT487-1 (TSSOP32)**

## 17. 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.

## 18. 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”*.

### 18.1 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.

### 18.2 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

### 18.3 Wave soldering

Key characteristics in wave soldering are:

- 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

### 18.4 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 22) 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 19 and 20

**Table 19. SnPb eutectic process (from J-STD-020D)**

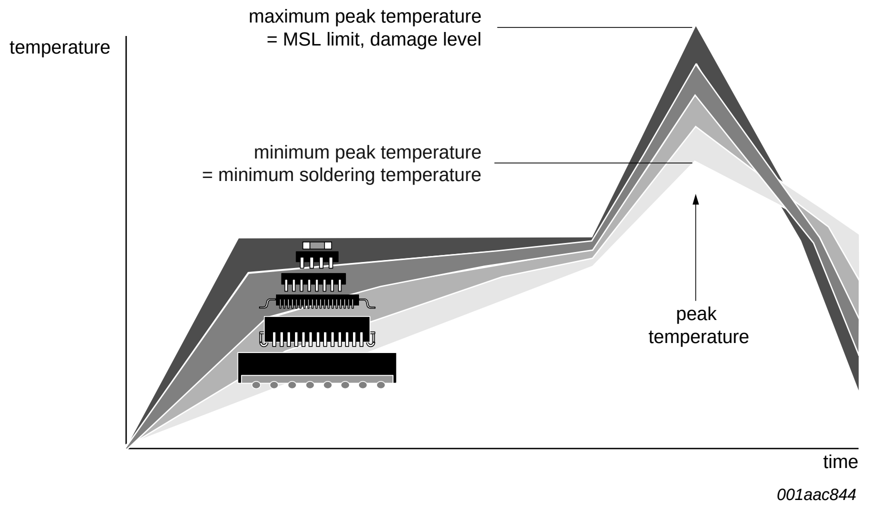
| Package thickness (mm) | Package reflow temperature (°C) | |
|---|---|---|
| | Volume (mm$^3$) | |
| | < 350 | ≥ 350 |
| < 2.5 | 235 | 220 |
| ≥ 2.5 | 220 | 220 |

**Table 20. Lead-free process (from J-STD-020D)**

| Package thickness (mm) | Package reflow temperature (°C) | | |
|---|---|---|---|
| | Volume (mm$^3$) | | |
| | < 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 22.



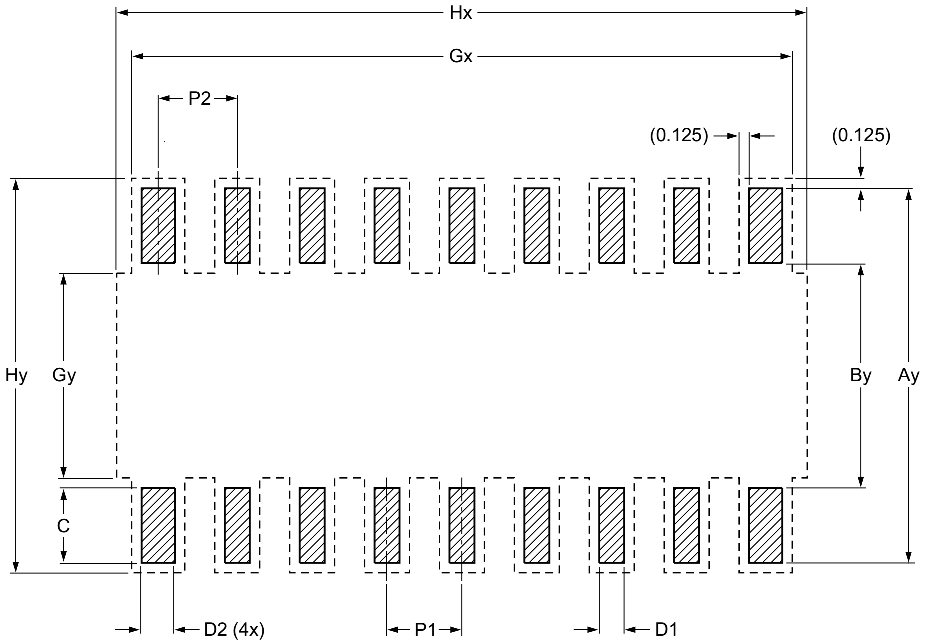

MSL: Moisture Sensitivity Level

**Fig 22. Temperature profiles for large and small components**

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

## 19. Soldering: PCB footprints

**Footprint information for reflow soldering of TSSOP32 package** **SOT487-1**




Generic footprint pattern

Refer to the package outline drawing for actual layout

solder land

occupied area

DIMENSIONS in mm

| P1 | P2 | Ay | By | C | D1 | D2 | Gx | Gy | Hx | Hy |
|---|---|---|---|---|---|---|---|---|---|---|
| 0.650 | 0.750 | 8.900 | 6.100 | 1.400 | 0.400 | 0.600 | 10.800 | 7.000 | 11.800 | 9.150 |

*sot487-1_fr*

**Fig 23. PCB footprint for SOT487-1 (TSSOP32); reflow soldering**

## 20. Abbreviations

**Table 21. Abbreviations**

| Acronym | Description |
|---|---|
| CDM | Charged-Device Model |
| DUT | Device Under Test |
| ESD | ElectroStatic Discharge |
| FET | Field-Effect Transistor |
| HBM | Human Body Model |
| $I^2C$-bus | Inter-Integrated Circuit bus |
| LED | Light Emitting Diode |
| LCD | Liquid Crystal Display |
| LSB | Least Significant Bit |
| MSB | Most Significant Bit |
| NMOS | Negative-channel Metal-Oxide Semiconductor |
| PCB | Printed-Circuit Board |
| PMOS | Positive-channel Metal-Oxide Semiconductor |
| PWM | Pulse Width Modulation |
| RGB | Red/Green/Blue |
| RGBA | Red/Green/Blue/Amber |
| SMBus | System Management Bus |

## 21. Revision history

**Table 22. Revision history**

| Document ID | Release date | Data sheet status | Change notice | Supersedes |
|---|---|---|---|---|
| PCA9622 v.5 | 20140602 | Product data sheet | - | PCA9622 v.4 |
| Modifications: | • Added Section 4.1 “Ordering options”<br>• Table 5 “Register summary[1]”: deleted old table note [2]<br>• Table 6 “MODE1 - Mode register 1 (address 00h) bit description”: added (new) Table note [1] and its cross-reference at SLEEP bit (bit 4)<br>• Added Section 19 “Soldering: PCB footprints” | | | |
| PCA9622 v.4 | 20120906 | Product data sheet | - | PCA9622 v.3 |
| PCA9622 v.3 | 20090831 | Product data sheet | - | PCA9622 v.2 |
| PCA9622 v.2 | 20090611 | Product data sheet | - | PCA9622 v.1 |
| PCA9622 v.1 | 20090327 | Product data sheet | - | - |

# 22. Legal information

## 22.1 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.

## 22.2 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.

## 22.3 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.

**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.

### 22.4 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 Semiconductors N.V.

## 23. Contact information

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

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

# 24. Contents

1 General description . . . 1
2 Features and benefits . . . 2
3 Applications . . . 3
4 Ordering information . . . 3
4.1 Ordering options . . . 3
5 Block diagram . . . 4
6 Pinning information . . . 5
6.1 Pinning . . . 5
6.2 Pin description . . . 5
7 Functional description . . . 6
7.1 Device addresses . . . 6
7.1.1 Regular I²C-bus slave address . . . 6
7.1.2 LED All Call I²C-bus address . . . 7
7.1.3 LED Sub Call I²C-bus addresses . . . 7
7.1.4 Software Reset I²C-bus address . . . 8
7.2 Control register . . . 8
7.3 Register definitions . . . 10
7.3.1 Mode register 1, MODE1 . . . 11
7.3.2 Mode register 2, MODE2 . . . 11
7.3.3 PWM0 to PWM15, individual brightness control . . . 12
7.3.4 GRPPWM, group duty cycle control . . . 13
7.3.5 GRPFREQ, group frequency . . . 13
7.3.6 LEDOUT0 to LEDOUT3, LED driver output state . . . 14
7.3.7 SUBADR1 to SUBADR3, I²C-bus subaddress 1 to 3 . . . 14
7.3.8 ALLCALLADR, LED All Call I²C-bus address . . . 15
7.4 Active LOW output enable input . . . 15
7.5 Power-on reset . . . 16
7.6 Software reset . . . 16
7.7 Individual brightness control with group dimming/blinking . . . 17
8 Characteristics of the I²C-bus . . . 18
8.1 Bit transfer . . . 18
8.1.1 START and STOP conditions . . . 18
8.2 System configuration . . . 18
8.3 Acknowledge . . . 19
9 Bus transactions . . . 20
10 Application design-in information . . . 22
10.1 Junction temperature calculation . . . 23
10.1.1 Example 1: $T_j$ calculation of PCA9622DR, in TSSOP32 package, when $T_{amb}$ is known . . . 24
10.1.2 Example 2: $T_j$ calculation where only $T_{case}$ is known . . . 24
11 Limiting values . . . 25
12 Thermal characteristics . . . 26
13 Static characteristics . . . 27
14 Dynamic characteristics . . . 29
15 Test information . . . 31
16 Package outline . . . 32
17 Handling information . . . 33
18 Soldering of SMD packages . . . 33
18.1 Introduction to soldering . . . 33
18.2 Wave and reflow soldering . . . 33
18.3 Wave soldering . . . 33
18.4 Reflow soldering . . . 34
19 Soldering: PCB footprints . . . 36
20 Abbreviations . . . 37
21 Revision history . . . 37
22 Legal information . . . 38
22.1 Data sheet status . . . 38
22.2 Definitions . . . 38
22.3 Disclaimers . . . 38
22.4 Trademarks . . . 39
23 Contact information . . . 39
24 Contents . . . 40

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



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

Date of release: 2 June 2014
Document identifier: PCA9622

# X-ON Electronics

Largest Supplier of Electrical and Electronic Components

***Click to view similar products for*** LED Lighting Drivers ***category:***

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

Other Similar products are found below :

LV5235V-MPB-H MB39C602PNF-G-JNEFE1 MIC2871YMK-T5 AL1676-10BS7-13 AL1676-20AS7-13 AP5726WUG-7 ICL8201 IS31BL3228B-UTLS2-TR IS31BL3506B-TTLS2-TR AL3157F-7 AP5725FDCG-7 AP5726FDCG-7 LV52204MTTBG AP5725WUG-7 STP4CMPQTR NCL30086BDR2G CAT4004BHU2-GT3 LV52207AXA-VH AP1694AS-13 TLE4242EJ AS3688 IS31LT3172-GRLS4-TR TLD2311EL KTD2694EDQ-TR KTZ8864EJAA-TR IS32LT3174-GRLA3-TR MP2488DN-LF-Z NLM0010XTSA1 AL1676-20BS7-13 ZXLD1370QESTTC MPQ7220GF-AEC1-P MPQ7220GR-AEC1-P MPQ4425BGJ-AEC1-P MPQ7220GF-AEC1-Z MPQ7220GR-AEC1-Z MPQ4425BGJ-AEC1-Z NCL30486A2DR2G IS31FL3737B-QFLS4-TR IS31FL3239-QFLS4-TR KTD2058EUAC-TR KTD2037EWE-TR DIO5662ST6 IS31BL3508A-TTLS2-TR KTD2026BEWE-TR MAX20052CATC/V+ MAX25606AUP/V+ BD6586MUV-E2 BD9206EFV-E2 BD9416FS-E2 LYT4227E