Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

PCA9665; PCA9665A

Fm+ parallel bus to I²C-bus controller

Rev. 4 — 29 September 2011 Product data sheet

1. General description

The PCA9665/PCA9665A serves as an interface between most standard parallel-bus microcontrollers/microprocessors and the serial I²C-bus and allows the parallel bus system to communicate bidirectionally with the I²C-bus. The PCA9665/PCA9665A can operate as a master or a slave and can be a transmitter or receiver. Communication with the I²C-bus is carried out on a Byte or Buffered mode using interrupt or polled handshake. The PCA9665/PCA9665A controls all the I²C-bus specific sequences, protocol, arbitration and timing with no external timing element required.

The PCA9665 and PCA9665A have the same footprint as the PCA9564 with additional features:

- 1 MHz transmission speeds
- Up to 25 mA drive capability on SCL/SDA
- 68-byte buffer
- I²C-bus General Call
- Software reset on the parallel bus

2. Features and benefits

- Parallel-bus to I²C-bus protocol converter and interface
- Both master and slave functions
- Multi-master capability
- Internal oscillator trimmed to 15 % accuracy reduces external components
- 1 Mbit/s and up to 25 mA SCL/SDA I_{OL} (Fast-mode Plus (Fm+)) capability
- I²C-bus General Call capability
- Software reset on parallel bus
- 68-byte data buffer
- Operating supply voltage: 2.3 V to 3.6 V
- 5 V tolerant I/Os
- Standard-mode and Fast-mode I²C-bus capable and compatible with SMBus
- PCA9665A ‘glitch-free’ restart is suitable for use with buffer drivers
- 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:
  - PCA9665: SO20, TSSOP20, HVQFN20
  - PCA9665A: TSSOP20

3. Applications

- Add I²C-bus port to controllers/processors that do not have one
- Add additional I²C-bus ports to controllers/processors that need multiple I²C-bus ports
- Converts 8 bits of parallel data to serial data stream to prevent having to run a large number of traces across the entire printed-circuit board

4. Ordering information

Table 1. Ordering information

T_{amb} = −40 °C to +85 °C.

Type number	Topside mark	Package		
		Name	Description	Version
PCA9665BS	9665	HVQFN20	plastic thermal enhanced very thin quad flat package; no leads; 20 terminals; body 5 × 5 × 0.85 mm	SOT662-1
PCA9665D	PCA9665D	SO20	plastic small outline package; 20 leads; body width 7.5 mm	SOT163-1
PCA9665PW	PCA9665	TSSOP20	plastic thin shrink small outline package; 20 leads; body width 4.4 mm	SOT360-1
PCA9665APW	CA9665A	TSSOP20	plastic thin shrink small outline package; 20 leads; body width 4.4 mm	SOT360-1

5. Block diagram

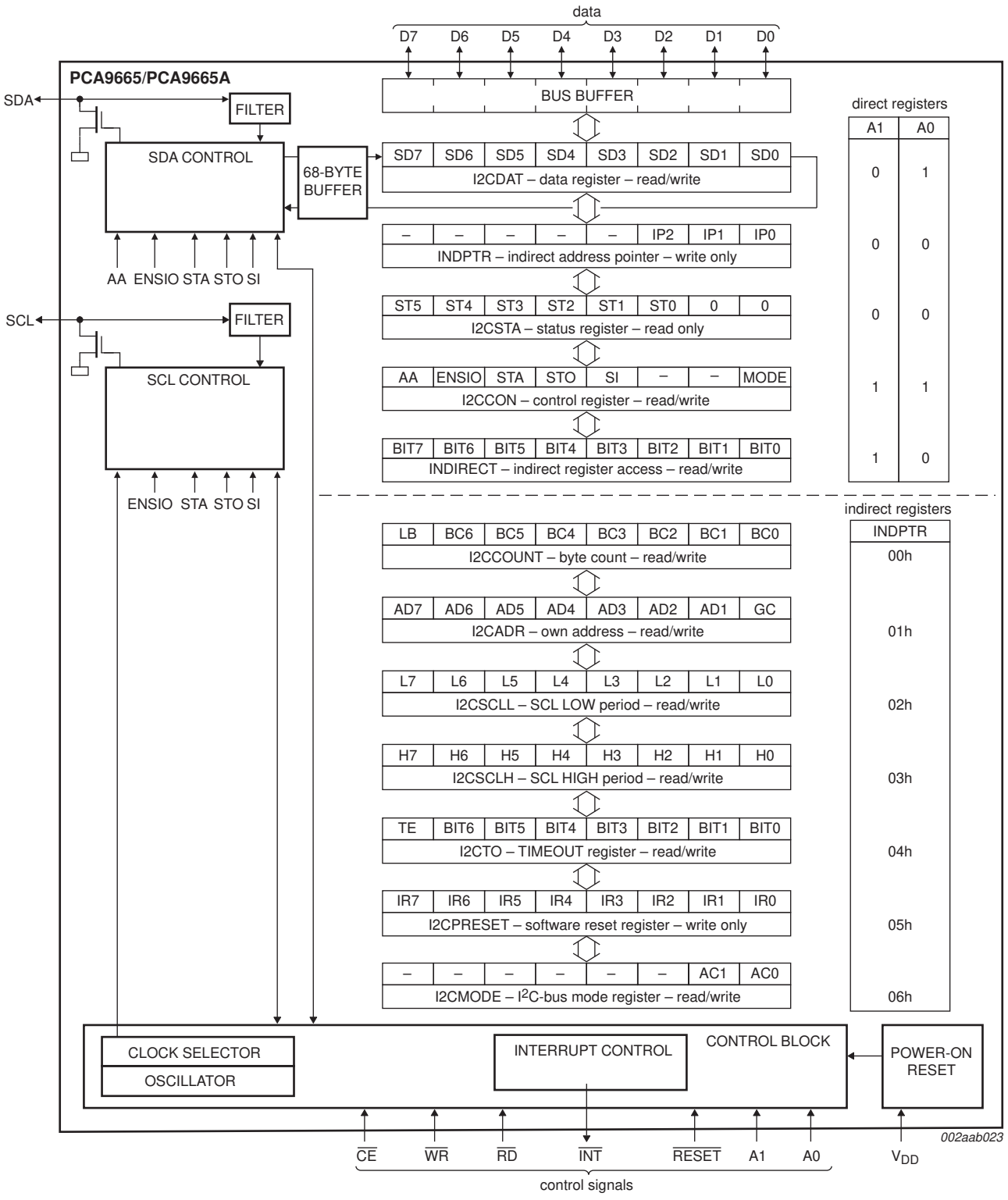


Fig 1. Block diagram of PCA9665/PCA9665A

6. Pinning information

6.1 Pinning

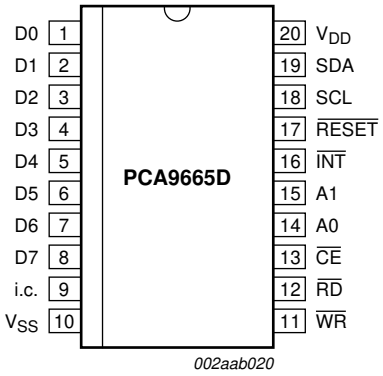


Fig 2. Pin configuration for SO20

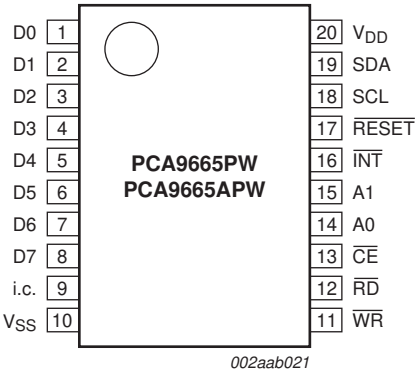
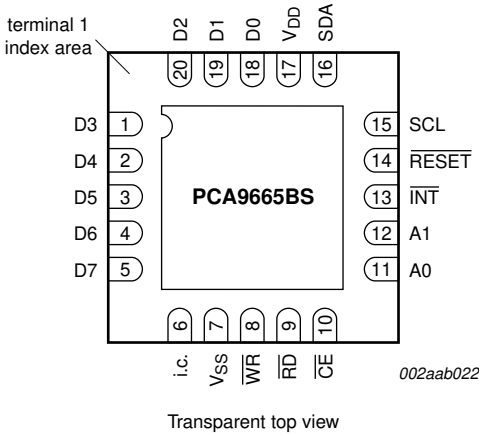


Fig 3. Pin configuration for TSSOP20



Transparent top view

Fig 4. Pin configuration for HVQFN20

6.2 Pin description

Table 2. Pin description

Symbol	Pin		Type	Description
	SO20, TSSOP20	HVQFN20		
D0	1	18	I/O	**Data bus:** Bidirectional 3-state data bus used to transfer commands, data and status between the bus controller and the CPU. D0 is the least significant bit.
D1	2	19	I/O	
D2	3	20	I/O	
D3	4	1	I/O	
D4	5	2	I/O	
D5	6	3	I/O	
D6	7	4	I/O	
D7	8	5	I/O	
i.c.	9	6	-	**internally connected:** must be left floating (pulled LOW internally)
V_{SS}	10	7[1]	power	**Supply ground**
$\overline{WR}$	11	8	I	**Write strobe:** When LOW and $\overline{CE}$ is also LOW, the content of the data bus is loaded into the addressed register. Data are latched on the rising edge of either $\overline{WR}$ or $\overline{CE}$.
$\overline{RD}$	12	9	I	**Read strobe:** When LOW and $\overline{CE}$ is also LOW, causes the contents of the addressed register to be presented on the data bus. The read cycle begins on the falling edge of $\overline{RD}$.
$\overline{CE}$	13	10	I	**Chip Enable:** Active LOW input signal. When LOW, data transfers between the CPU and the bus controller are enabled on D0 to D7 as controlled by the $\overline{WR}$, $\overline{RD}$ and A0 to A1 inputs. When HIGH, places the D0 to D7 lines in the 3-state condition. Data are written into the addressed register on rising edge of either $\overline{CE}$ or $\overline{WR}$.
A0	14	11	I	**Address inputs:** Selects the bus controller's internal registers and ports for read/write operations.
A1	15	12	I	
$\overline{INT}$	16	13	O	**Interrupt request:** Active LOW, open-drain, output. This pin requires a pull-up device.
$\overline{RESET}$	17	14	I	**Reset:** Active LOW input. A LOW level clears internal registers and resets the I²C-bus state machine.
SCL	18	15	I/O	I²C-bus serial clock input/output (open-drain). This pin requires a pull-up device.
SDA	19	16	I/O	I²C-bus serial data input/output (open-drain). This pin requires a pull-up device.
V_{DD}	20	17	power	**Power supply:** 2.3 V to 3.6 V

[1] HVQFN20 package die supply ground is connected to both the V_{SS} pin and the exposed center pad. The V_{SS} pin must be connected to supply ground for proper device operation. For enhanced thermal, electrical, and board-level performance, the exposed pad needs to be soldered to the board using a corresponding thermal pad on the board, and for proper heat conduction through the board thermal vias need to be incorporated in the PCB in the thermal pad region.

7. Functional description

7.1 General

The PCA9665/PCA9665A acts as an interface device between standard high-speed parallel buses and the serial I²C-bus. On the I²C-bus, it can act either as a master or slave. Bidirectional data transfer between the I²C-bus and the parallel-bus microcontroller is carried out on a byte or buffered basis, using either an interrupt or polled handshake.

7.2 Internal oscillator

The PCA9665/PCA9665A contains an internal oscillator which is used for all I²C-bus timing. Typical oscillator frequency is 28.5 MHz for the PCA9665 and 32 MHz for the PCA9665A. The oscillator requires up to 550 μs to start-up after ENSIO bit is set to '1'.

7.3 Registers

The PCA9665/PCA9665A contains eleven registers which are used to configure the operation of the device as well as to send and receive serial data. There are four registers that can be accessed directly and seven registers that are accessed indirectly by setting a register pointer.

The four direct registers are selected by setting pins A0 and A1 to the appropriate logic levels before a read or write operation is executed on the parallel bus.

The seven indirect registers require that the INDPTR (indirect register pointer, one of the four direct registers described above) is initially loaded with the address of the register in the indirect address space before a read or write is performed to the INDIRECT data field.

For example, in order to write to the indirectly addressed I2CSCLL register, the INDPTR register should be loaded with 02h by performing a write to the direct INDPTR register (A1 = 0, A0 = 0). Then the I2CSCLL register can be programmed by writing to the INDIRECT data field (A1 = 1, A0 = 0) in the direct address space. Register mapping is described in Table 3, Table 4 and Figure 5.

Remark: Do not write to any I²C-bus registers while the I²C-bus is busy and the PCA9665/PCA9665A is in master or addressed slave mode.

Table 3. Direct register selection by setting A0 and A1

Register name	Register function	A1	A0	Read/Write	Default
I2CSTA	status	0	0	R	F8h
INDPTR	indirect register pointer	0	0	W	00h
I2CDAT	data	0	1	R/W	00h
I2CCON	control	1	1	R/W	00h[1]
INDIRECT	indirect data field access	1	0	R/W	00h

[1] See Section 8.10 "Power-on reset" for more detail.

Table 4. Indirect register selection by setting A1 = 1 and A0 = 0

Register name	Register function	INDPTR	Read/Write	Default
I2CCOUNT	byte count	00h	R/W	01h
I2CADR	own address	01h	R/W	E0h
I2CSCLL	SCL LOW period	02h	R/W	9Dh
I2CSCLH	SCL HIGH period	03h	R/W	86h
I2CTO	time-out	04h	R/W	FFh
I2CPRESET	parallel software reset	05h	W	00h
I2CMODE	I²C-bus mode	06h	R/W	00h

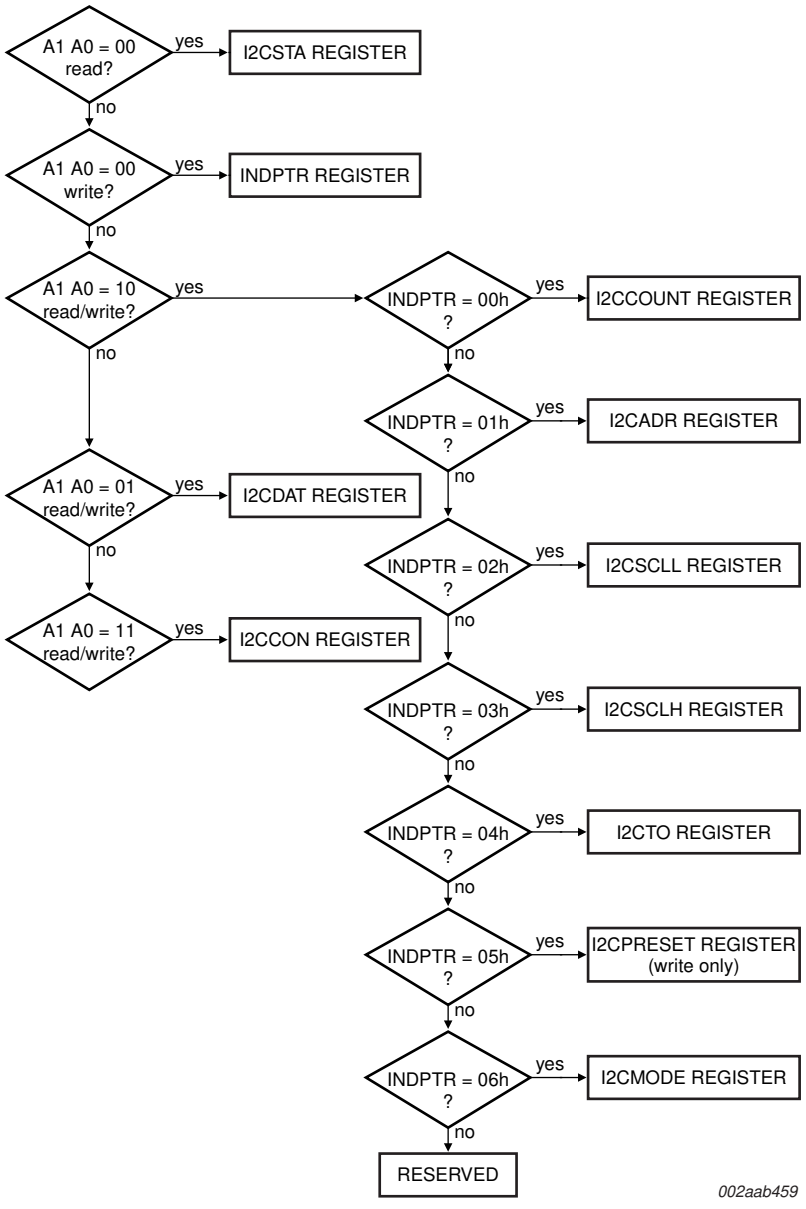


Fig 5. Register mapping flowchart

7.3.1 Direct registers

7.3.1.1 The Status register, I2CSTA (A1 = 0, A0 = 0)

I2CSTA is an 8-bit read-only register. The two least significant bits are always zero. The six most significant bits contain the status code. There are 30 possible status codes. When I2CSTA contains F8h, it indicates the idle state and therefore no serial interrupt is requested. All other I2CSTA values correspond to defined states. When each of these states is entered, a serial interrupt is requested (SI = 1 and $\overline{INT}$ asserted LOW).

Remark: Data in I2CSTA is valid only when a serial interrupt occurs (SI = 1 and $\overline{INT}$ asserted LOW). Reading the register when SI = 0 and $\overline{INT}$ is HIGH may cause wrong values to be read.

Table 5. I2CSTA - Status register (A1 = 0, A0 = 0) bit allocation

7	6	5	4	3	2	1	0
ST5	ST4	ST3	ST2	ST1	ST0	0	0

Table 6. I2CSTA - Status register (A1 = 0, A0 = 0) bit description

Bit	Symbol	Description
7:2	ST[5:0]	status code corresponding to the different I²C-bus states
1:0	-	always at zero

7.3.1.2 The Indirect Pointer register, INDPTR (A1 = 0, A0 = 0)

Table 7. INDPTR - Indirect Register Pointer (A1 = 0, A0 = 0) bit allocation

7	6	5	4	3	2	1	0
-	-	-	-	-	IP2	IP1	IP0

Table 8. INDPTR - Indirect Pointer register (A1 = 0, A0 = 0) bit description

Bit	Symbol	Description
7:3	-	reserved; must be written with zeroes
2:0	IP2 to IP0	address of the indirect register

INDPTR is an 8-bit write-only register. It contains a pointer to a register in the indirect address space (IP[2:0]). The value in the register will determine what indirect register will be accessed when the INDIRECT register is read or written, as defined in Table 4.

7.3.1.3 The I²C-bus Data register, I2CDAT (A1 = 0, A0 = 1)

I2CDAT is an 8-bit read/write register. It contains a byte of serial data to be transmitted or a byte which has just been received. In master mode, this includes the slave address that the master wants to send out on the I²C-bus, with the most significant bit of the slave address in the SD7 bit position and the Read/Write bit in the SD0 bit position. The CPU can read from and write to this 8-bit register while the PCA9665/PCA9665A is not in the process of shifting a byte. This occurs when PCA9665/PCA9665A is in a defined state and the serial interrupt flag is set. Data in I2CDAT remains stable as long as SI is set. Whenever the PCA9665/PCA9665A generates an interrupt, the I2CDAT register contains the data byte that was just transferred on the I²C-bus.

In Byte mode, the CPU can read or write a single byte at a time. In Buffered mode, the CPU can read or write up to 68 bytes at a time. See Section 8.1 “Configuration modes” for more detail.

Remark: The I2CDAT register will capture the serial address as data when addressed via the serial bus.

Remark: In Byte mode only, the data register will capture data from the serial bus during 38h (arbitration lost in slave address + R/$\overline{W}$ or data bytes causing this data in I2CDAT to be changed), so the I2CDAT register will need to be reloaded when the bus becomes free.

In Buffered mode, the data is not written in the data register when arbitration is lost, which keeps the buffer intact.

Table 9. I2CDAT - Data register (A1 = 0, A0 = 1) bit allocation

7	6	5	4	3	2	1	0
SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0

Table 10. I2CDAT - Data register (A1 = 0, A0 = 1) bit description

Bit	Symbol	Description
7:0	SD[7:0]	Eight bits to be transmitted or just received. A logic 1 in I2CDAT corresponds to a HIGH level on the I²C-bus. A logic 0 corresponds to a LOW level on the bus.

7.3.1.4 The Control register, I2CCON (A1 = 1, A0 = 1)

I2CCON is an 8-bit read/write register. Two bits are affected by the bus controller hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the I²C-bus. A Write to the I2CCON register via the parallel interface automatically clears the SI bit, which causes the Serial Interrupt line to be de-asserted and the next clock pulse on the SCL line to be generated.

Remark: Since none of the registers should be written to via the parallel interface once the Serial Interrupt line has been de-asserted, all the other registers that need to be modified should be written to before the content of the I2CCON register is modified.

Table 11. I2CCON - Control register (A1 = 1, A0 = 1) bit allocation

7	6	5	4	3	2	1	0
AA	ENSIO	STA	STO	SI	-	-	MODE

Table 12. I2CCON - Control register (A1 = 1, A0 = 1) bit description

Bit	Symbol	Description
7	AA	The Assert Acknowledge flag. AA = 1: If the AA flag is set, an acknowledge (LOW level on SDA) will be returned during the acknowledge clock pulse on the SCL line when: • 'Own slave address' has been received (as defined in I2CADR register). • A data byte has been received while the bus controller is in the Master Receiver mode. • A data byte has been received while the bus controller is in the addressed Slave Receiver mode. AA = 0: if the AA flag is reset, a not acknowledge (HIGH level on SDA) will be returned during the acknowledge clock pulse on SCL when: • 'Own slave address' has been received (as defined in I2CADR register). • A data byte has been received while the PCA9665/PCA9665A is in the Master Receiver mode. • A data byte has been received while the PCA9665/PCA9665A is in the addressed Slave Receiver mode. When the bus controller is in the addressed Slave Transmitter mode, state C8h will be entered after the last data byte is transmitted and an ACK is received from the Master Receiver (see Figure 9 and Figure 13). When SI is cleared, the PCA9665/PCA9665A enters the not addressed Slave Receiver mode, and the SDA line remains at a HIGH level. In state C8h, the AA flag can be set again for future address recognition. When the PCA9665/PCA9665A is in the not addressed slave mode, its own slave address is ignored. Consequently, no acknowledge is returned, and a serial interrupt is not requested. Thus, the bus controller can be temporarily released from the I²C-bus while the bus status is monitored. While the bus controller is released from the bus, START and STOP conditions are detected, and serial data is shifted in. Address recognition can be resumed at any time by setting the AA flag.
6	ENSIO	The bus controller enable bit. ENSIO = 0: When ENSIO is '0', the SDA and SCL outputs are in a high-impedance state. SDA and SCL input signals are ignored, the PCA9665/PCA9665A is in the 'not addressed' slave state. Internal oscillator is off. ENSIO = 1: When ENSIO is '1', the PCA9665/PCA9665A is enabled. After the ENSIO bit is set to '1', it takes 550 μs enable time for the internal oscillator to start up and the serial interface to initialize. The PCA9665/PCA9665A will enter either the master or the slave mode after this time. ENSIO should not be used to temporarily release the PCA9665/PCA9665A from the I²C-bus since, when ENSIO is reset, the I²C-bus status is lost. The AA flag should be used instead (see description of the AA flag above). In the following text, it is assumed that ENSIO = '1' for Normal mode operation. For power-up behavior, please refer to Section 8.10 "Power-on reset".

Table 12. I2CCON - Control register (A1 = 1, A0 = 1) bit description *...continued*

Bit	Symbol	Description
5	STA	The START flag. STA = 1: When the STA bit is set to enter a master mode, the bus controller hardware checks the status of the I²C-bus and generates a START condition if the bus is free. If the bus is not free, then the bus controller waits for a STOP condition (which will free the bus) and generates a START condition after the minimum buffer time (t_{BUF}) has elapsed. If STA is set while the bus controller is already in a master mode and one or more bytes are transmitted or received, the bus controller transmits a repeated START condition. STA may be set at any time. STA may also be set when the bus controller is an addressed slave. A START condition will then be generated after a STOP condition and the minimum buffer time (t_{BUF}) has elapsed. STA = 0: When the STA bit is reset, no START condition or repeated START condition will be generated.
4	STO	The STOP flag. STO = 1: When the STO bit is set while the bus controller is in a master mode, a STOP condition is transmitted on the I²C-bus. When a STOP condition is detected on the bus, the hardware clears the STO flag. If the STA and STO bits are both set and the PCA9665/PCA9665A is in master mode, then a STOP condition is transmitted on the I²C-bus. The bus controller then transmits a START condition after the minimum buffer time (t_{BUF}) has elapsed. STO = 0 : When the STO bit is reset, no STOP condition will be generated.
3	SI	The Serial Interrupt flag. SI = 1: When the SI flag is set, and, if the ENSIO bit is also set, a serial interrupt is requested. SI is set by hardware when one of 29 of the 30 possible states of the bus controller states is entered. The only state that does not cause SI to be set is state F8h, which indicates that no relevant state information is available. While SI is set, the LOW period of the serial clock on the SCL line is stretched, and the serial transfer is suspended. A HIGH level on the SCL line is unaffected by the serial interrupt flag. SI is automatically cleared when the I2CCON register is written. The SI bit cannot be set by the user. SI = 0: When the SI flag is reset, no serial interrupt is requested, and there is no stretching of the serial clock on the SCL line.
2:1	-	Reserved. When I2CCON is read, zeroes are read. Must be written with zeroes.
0	MODE	The Mode flag. MODE = 0; Byte mode. See Section 8.1.1 “Byte mode” for more detail. MODE = 1; buffered mode. See Section 8.1.2 “Buffered mode” for more detail.

Remark: ENSIO bit value must be changed only when the I²C-bus is idle.

7.3.1.5 The indirect data field access register, INDIRECT (A1 = 1, A0 = 0)

The registers in the indirect address space can be accessed using the INDIRECT data field. Before writing or reading such a register, the INDPTR register should be written with the address of the indirect register that needs to be accessed. Once the INDPTR register contains the appropriate value, reads and writes to the INDIRECT data field will actually read and write the selected indirect register.

7.3.2 Indirect registers

7.3.2.1 The Byte Count register, I2CCOUNT (indirect address 00h)

The I2CCOUNT register is an 8-bit read/write register. It contains the number of bytes that have been stored in Master/Slave Buffered Receiver mode, and the number of bytes to be sent in Master/Slave Buffered Transmitter mode. Bit 7 is the last byte control bit and applies to the Master/Slave Buffered Receiver mode only. The data in the I2CCOUNT register is relevant only in Buffered mode (MODE = 1) and should not be used (read or written) in Byte mode (MODE = 0).

Table 13. I2CCOUNT - Byte Count register (indirect address 00h) bit allocation

7	6	5	4	3	2	1	0
LB	BC6	BC5	BC4	BC3	BC2	BC1	BC0

Table 14. I2CCOUNT - Byte Count register (indirect address 00h) bit description

Bit	Symbol	Description
7	LB	Last Byte control bit. Master/Slave Buffered Receiver mode only. LB = 1: PCA9665/PCA9665A does not acknowledge the last received byte. LB = 0: PCA9665/PCA9665A acknowledges the last received byte. A future bus transaction must complete the read sequence by not acknowledging the last byte.
6:0	BC[6:0]	Number of bytes to be read or written (up to 68 bytes). If BC[6:0] is equal to 0 or greater than 68 (44h), no bytes will be read or written and an interrupt is immediately generated after writing to the I2CCON register (in Buffered mode only).

7.3.2.2 The Own Address register, I2CADR (indirect address 01h)

I2CADR is an 8-bit read/write register. It is not affected by the bus controller hardware. The content of this register is unused when the controller is in a master mode. A master should never transmit its own slave address. In the slave modes, the seven most significant bits must be loaded with the microcontroller's own slave address and the least significant bit determines if the General Call address will be recognized or not.

Remark: AD[7:1] must be different from the General Call address (000 0000) for proper device operation.

Remark: The I2CADR default value is E0h.

Table 15. I2CADR - Address register (indirect address 01h) bit allocation

7	6	5	4	3	2	1	0
AD7	AD6	AD5	AD4	AD3	AD2	AD1	GC

Table 16. I2CADR - Address register (indirect address 01h) bit description

Bit	Symbol	Description
7:1	AD[7:1]	Own slave address. The most significant bit corresponds to the first bit received from the I²C-bus after a START condition. A logic 1 in I2CADR corresponds to a HIGH level on the I²C-bus, and a logic 0 corresponds to a LOW level on the bus.
0	GC	General Call. GC = 1: General Call address (00h) is recognized. GC = 0: General Call address (00h) is ignored.

7.3.2.3 The Clock Rate registers, I2CSCLL and I2CSCLH (indirect addresses 02h and 03h)

I2CSCLL and I2CSCLH are 8-bit read/write registers. They define the data rate for the PCA9665/PCA9665A when used as a bus master. The actual frequency is determined by t_{HIGH} (time where SCL is HIGH), t_{LOW} (time where SCL is LOW), t_r (rise time), t_f (fall time) and t_d (delay time) values.

t_{HIGH} and t_{LOW} are calculated based on the values that are programmed into I2CSCLH and I2CSCLL registers and the internal oscillator frequency. t_r and t_f are system/application dependent. t_d for the PCA9665 is approximately 175 ns and the PCA9665A is approximately 300 ns.

$$f_{SCL} = \frac{1}{T_{osc}(I2CSCLL + I2CSCLH) + t_r + t_f + t_d} \quad (1)$$

with T_{osc} = internal oscillator period = 35 ns ± 5 ns for the PCA9665 and 33 ns ± 5 ns for the PCA9665A

The delay time 't_d' is the sum of the time between the oscillator edge of the SCLL terminal count until the SCL is up to $0.3V_{DD}$ and the oscillator edge of the SCLH terminal count until the SCL is down to $0.7V_{DD}$.

Remark: The I2CMODE register needs to be programmed before programming the I2CSCLL and I2CSCLH registers in order to know which I²C-bus mode is selected. See Section 7.3.2.6 "The I²C-bus mode register, I2CMODE (indirect address 06h)" for more detail.

Standard-mode is the default selected mode at power-up or after reset.

Table 17. I2CSCLL - Clock Rate Low register (indirect address 02h) bit allocation

7	6	5	4	3	2	1	0
L7	L6	L5	L4	L3	L2	L1	L0

Table 18. I2CSCLL - Clock Rate Low register (indirect address 02h) bit description

Bit	Symbol	Description
7:0	L[7:0]	Eight bits defining the LOW state of SCL.

Table 19. I2CSCLH - Clock Rate High register (indirect address 03h) bit allocation

7	6	5	4	3	2	1	0
H7	H6	H5	H4	H3	H2	H1	H0

Table 20. I2CSCLH - Clock Rate High register (indirect address 03h) bit description

Bit	Symbol	Description
7:0	H[7:0]	Eight bits defining the HIGH state of SCL.

7.3.2.4 The Time-out register, I2CTO (indirect address 04h)

I2CTO is an 8-bit read/write register. It is used to determine the maximum time that SCL is allowed to be in a LOW logic state before the I²C-bus state machine is reset or the PCA9665/PCA9665A initiates a forced action on the I²C-bus.

When the I²C-bus interface is operating, I2CTO is loaded in the time-out counter at every LOW SCL transition.

Table 21. I2CTO - Time-out register (indirect register 04h) bit allocation

7	6	5	4	3	2	1	0
TE	TO6	TO5	TO4	TO3	TO2	TO1	TO0

Table 22. I2CTO - Time-out register (indirect register 04h) bit description

Bit	Symbol	Description
7	TE	Time-out enable/disable TE = 1: Time-out function enabled TE = 0: Time-out function disabled
6:0	TO[6:0]	Time-out value. The time-out value may vary some, and is an approximate value. PCA9665 typical time-out period = (I2CTO[6:0] + 1) × 143 μs. PCA9665A typical time-out period = (I2CTO[6:0] + 1) × 134 μs.

The Time-out register can be used in the following cases:

- When the bus controller, in the master mode, wants to send a START condition and the SCL line is held LOW by some other device. Then the bus controller waits a time period equivalent to the time-out value for the SCL to be released. In case it is not released, the bus controller concludes that there is a bus error, loads 78h in the I2CSTA register, generates an interrupt signal and releases the SCL and SDA lines. After the microcontroller reads the status register, it needs to send a reset in order to reset the bus controller.
- In the master mode, the time-out feature starts every time the SCL goes LOW. If SCL stays LOW for a time period equal to or greater than the time-out value, the bus controller concludes there is a bus error and behaves in the manner described above. When the I²C-bus interface is operating, I2CTO is loaded in the time-out counter at every SCL transition. See Section 8.11 “Reset” for more information.
- In case of a forced access to the I²C-bus. (See more details in Section 8.9.3 “Forced access to the I²C-bus”.)

7.3.2.5 The Parallel Software Reset register, I2CPRESET (indirect address 05h)

I2CPRESET is an 8-bit write-only register. Programming the I2CPRESET register allows the user to reset the PCA9665/PCA9665A under software control. The software reset is achieved by writing two consecutive bytes to this register. The first byte must be A5h while the second byte must be 5Ah. The writes must be consecutive and the values must match A5h and 5Ah. If this sequence is not followed as described, the reset is aborted.

7.3.2.6 The I²C-bus mode register, I2CMODE (indirect address 06h)

I2CMODE is an 8-bit read/write register. It contains the control bits that select the correct timing parameters when the device is used in master mode (AC[1:0]). Timing parameters involved with AC[1:0] are t_{BUF}, $t_{HD;STA}$, $t_{SU;STA}$, $t_{SU;STO}$, t_{HIGH}, t_{LOW}.

Table 23. I2CMODE - I²C-bus Mode register (indirect address 06h) bit allocation

7	6	5	4	3	2	1	0
-	-	-	-	-	-	AC1	AC0

Table 24. I2CMODE - I²C-bus Mode register (indirect address 06h) bit description

Bit	Symbol	Description
7:2	-	Reserved. When I2CMODE is read, zeroes are read. Must be written with zeroes.
1:0	AC[1:0]	I²C-bus mode selection to ensure proper timing parameters (see Table 25 and Table 51). AC[1:0] = 00: Standard-mode AC parameters selected. AC[1:0] = 01: Fast-mode AC parameters selected. AC[1:0] = 10: Fast-mode Plus AC parameters selected. AC[1:0] = 11: Turbo mode. In this mode, the user is not limited to a maximum frequency of 1 MHz.

Remark: Change from an I²C-bus mode to a slower one (Fast-mode to Standard-mode, for example) will cause the HIGH and LOW timings of SCL to be violated. It is then required to program the I2CSCLL and I2CSCLH registers with values in accordance with the selected mode.

Table 25. I²C-bus mode selection example[1]

I2CSCLL (hexadecimal)	I2CSCLH (hexadecimal)	I²C-bus frequency (kHz)		AC[1:0]	Mode
		PCA9665[2]	PCA9665A[3]		
9D	86	98.0	103.3	00	Standard
2C	14	371.1	371.4	01	Fast
11	09	836.8	788.6	10	Fast-mode Plus
0E	05	1015	932.8	11	Turbo mode

[1] I2CSCLL and I2CSCLH values in the table also represents the minimum values that can be used for the corresponding I²C-bus mode. Use of lower values will cause the minimum values to be loaded.

[2] Using the formula $f_{SCL} = \frac{1}{T_{osc}(I2CSCLL + I2CSCLH) + t_r + t_f + t_d}$ with T_{osc} at 30 ns (minimum pulse width), t_d = 175 ns, and t_r and t_f at maximum data sheet values for the mode.

[3] Using the formula $f_{SCL} = \frac{1}{T_{osc}(I2CSCLL + I2CSCLH) + t_r + t_f + t_d}$ with T_{osc} at 28 ns (minimum pulse width), t_d = 300 ns, and t_r and t_f at maximum data sheet values for the mode.

8. PCA9665/PCA9665A modes

8.1 Configuration modes

Byte mode and Buffered mode are selected using the MODE bit in I2CCON register:

MODE = 0: Byte mode

MODE = 1: Buffered mode

8.1.1 Byte mode

The Byte mode allows communication on a single command basis. Only one specific command is executed at a time and the Status Register is updated once this single command has been performed. A command can be a START, a STOP, a Byte Write, a Byte Read, and so on.

8.1.2 Buffered mode

The Buffered mode allows several instructions to be executed before an Interrupt is generated and before the I2CSTA register is updated. This allows the microcontroller to request a sequence, up to 68 bytes in a single transmission and lets the PCA9665/PCA9665A perform it without having to access the Status Register and the Control Register each time a single command is performed. The microcontroller can then perform other tasks while the PCA9665/PCA9665A performs the requested sequence.

The number of bytes that needs to be sent from the internal buffer (Transmitter mode) or received into the internal buffer (Receiver mode) is defined in the indirectly addressed I2CCOUNT Register (BC[6:0]). Up to 68 bytes can be sent or received.

8.2 Operating modes

The four operating modes are:

- Master Transmitter
- Master Receiver
- Slave Receiver
- Slave Transmitter

Each mode can be used on a byte basis (Byte mode) or in an up to 68-byte buffer basis (Buffered mode).

Data transfers in each mode of operation are shown in Figure 6 through Figure 9. These figures contain the following abbreviations:

S — START condition

SLA — 7-bit slave address

R — Read bit (HIGH level at SDA)

W — Write bit (LOW level at SDA)

A — Acknowledge bit (LOW level at SDA)

$\overline{\textbf{A}}$ — Not acknowledge bit (HIGH level at SDA)

Data — 8-bit data byte

P — STOP condition

In Figure 6, Figure 7, Figure 8, Figure 9, Figure 10, Figure 11, Figure 12 and Figure 13, circles are used to indicate when the serial interrupt flag is set. A serial interrupt is not generated when I2CSTA = F8h. This happens on a STOP condition or when an external reset is generated (at power-up, when $\overline{\text{RESET}}$ pin is going LOW or during a software reset on the parallel bus). The numbers in the circles show the status code held in the I2CSTA register. At these points, a service routine must be executed to continue or complete the serial transfer. These service routines are not critical since the serial transfer is suspended until the serial interrupt flag is cleared by software.

When a serial interrupt routine is entered, the status code in I2CSTA is used to branch to the appropriate service routine. For each status code, the required software action and details of the following serial transfer are given in Table 27, Table 28, Table 31, Table 32, Table 35, Table 36, Table 40, and Table 41.

8.3 Byte mode

8.3.1 Master Transmitter Byte mode

In the Master Transmitter Byte mode, a number of data bytes are transmitted to a slave receiver (see Figure 6). Before the Master Transmitter Byte mode can be entered, I2CCON must be initialized as shown in Table 26.

Table 26. I2CCON initialization (Byte mode)

Bit	7	6	5	4	3	2	1	0
Symbol	AA	ENSIO	STA	STO	SI	reserved	reserved	MODE
Value	X	1	0	0	0	X	X	0

ENSIO must be set to logic 1 to enable the PCA9665/PCA9665A. If the AA bit is reset, the PCA9665/PCA9665A will not acknowledge its own slave address in the event of another device becoming master of the bus. (In other words, if AA is reset, PCA9665/PCA9665A cannot enter a slave mode.) STA, STO, and SI must be reset. Once ENSIO has been set to 1, it takes about 550 μs for the oscillator to start up.

The Master Transmitter Byte mode may now be entered by setting the STA bit. The I²C-bus state machine will first test the I²C-bus and generate a START condition as soon as the bus becomes free. When a START condition is transmitted, the serial interrupt flag (SI) is set, the Interrupt line ($\overline{\text{INT}}$) goes LOW and the status code in the status register (I2CSTA) will be 08h. This status code must be used to vector to an interrupt service routine that loads I2CDAT with the slave address and the data direction bit (SLA+W). A write to I2CCON resets the SI bit, clears the Interrupt ($\overline{\text{INT}}$ goes HIGH) and allows the serial transfer to continue.

When the slave address with the direction bit have been transmitted, the Serial Interrupt flag (SI) is set again, the Interrupt line ($\overline{\text{INT}}$) goes LOW again and I2CSTA is loaded with the following possible codes:

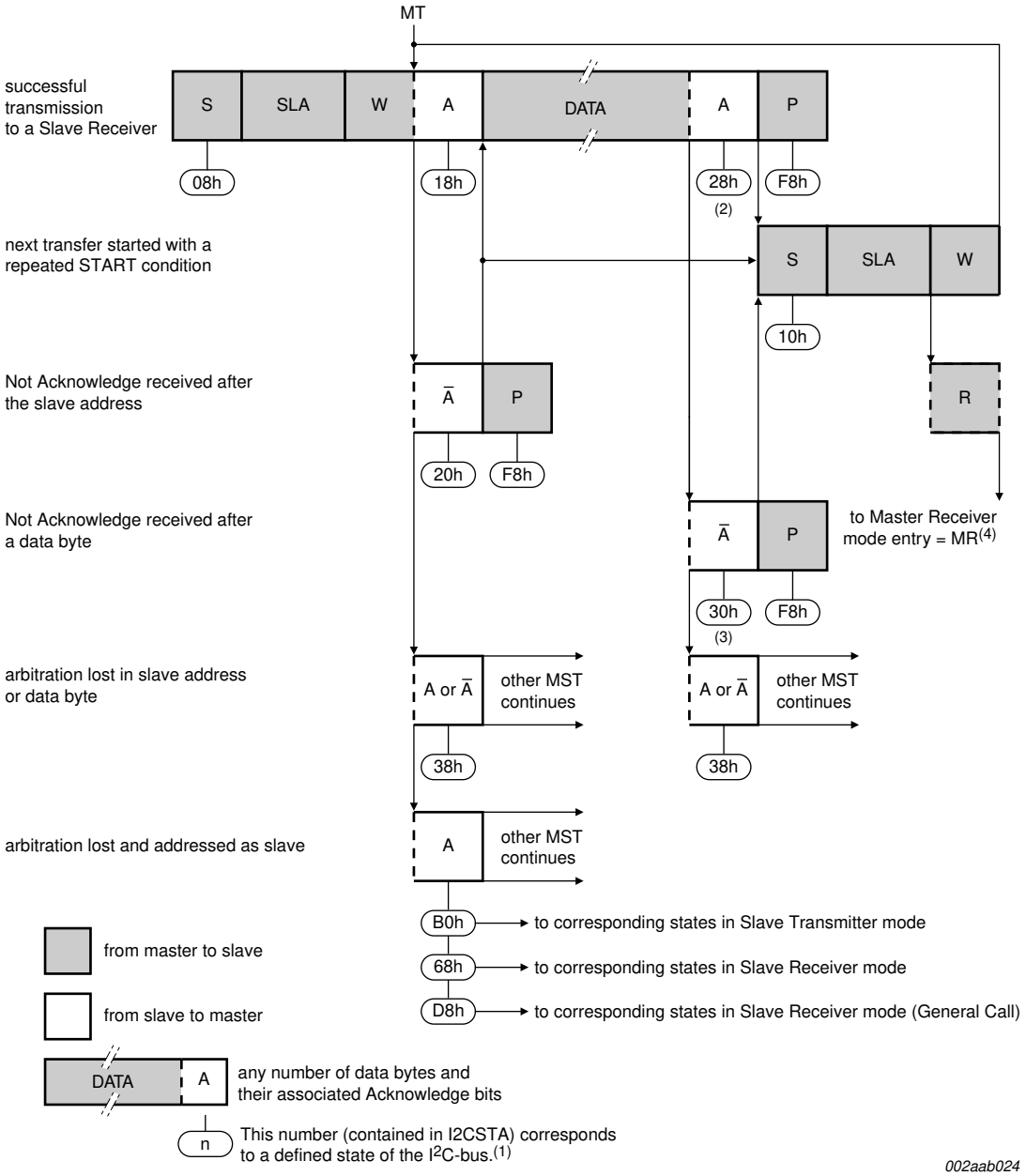
- 18h if an acknowledgment bit (ACK) has been received
- 20h if an no acknowledgment bit (NACK) has been received
- 38h if the PCA9665/PCA9665A lost the arbitration

- B0h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave transmitter (slave mode enabled with AA = 1)
- 68h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave receiver (slave mode enabled with AA = 1)
- D8h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave receiver during a General Call sequence (slave mode enabled with AA = 1 and General Call address enabled with GC = 1 in I2CADR register)

The appropriate action to be taken for each of these status codes is detailed in Table 27. ENSIO is not affected by the serial transfer and is not referred to in Table 27.

After a repeated START condition (state 10h), the PCA9665/PCA9665A may switch to the Master Receiver mode by loading I2CDAT with SLA+R.

Remark: A master should not transmit its own slave address.



(1) See Table 27

(2) Defined state when a single byte is sent and an ACK is received.

(3) Defined state when a single byte is sent and a NACK is received.

(4) Master Receiver Byte mode is entered when MODE = 0. Master Receiver Buffered mode is entered when MODE = 1.

Fig 6. Format and states in the Master Transmitter Byte mode (MODE = 0)

Table 27. Master Transmitter Byte mode (MODE = 0)

Status code (I2CSTA)	Status of the I²C-bus and the PCA9665/65A	Application software response						Next action taken by the PCA9665/PCA9665A
		To/from I2CDAT	To I2CCON					
			STA	STO	SI	AA	MODE	
08h	A START condition has been transmitted	Load SLA+W	X	X	0	X	0	SLA+W will be transmitted; ACK/NACK will be received
10h	A repeated START condition has been transmitted	Load SLA+W or	X	X	0	X	0	SLA+W will be transmitted; ACK/NACK will be received
		Load SLA+R	X	X	0	X	0	SLA+R will be transmitted; PCA9665/PCA9665A will be switched to Master Receiver Byte mode
18h	SLA+W has been transmitted; ACK has been received	Load data byte or	0	0	0	X	0	Data byte will be transmitted; ACK/NACK will be received
		no I2CDAT action or	1	0	0	X	0	Repeated START will be transmitted;
		no I2CDAT action or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		no I2CDAT action	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset
20h	SLA+W has been transmitted; NACK has been received	Load data byte or	0	0	0	X	0	Data byte will be transmitted; ACK/NACK will be received
		no I2CDAT action or	1	0	0	X	0	Repeated START will be transmitted;
		no I2CDAT action or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		no I2CDAT action	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset
28h	Data byte in I2CDAT has been transmitted; ACK has been received	Load data byte or	0	0	0	X	0	Data byte will be transmitted; ACK/NACK will be received
		no I2CDAT action or	1	0	0	X	0	Repeated START will be transmitted;
		no I2CDAT action or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		no I2CDAT action	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset

Table 27. Master Transmitter Byte mode (MODE = 0) *…continued*

Status code (I2CSTA)	Status of the I²C-bus and the PCA9665/65A	Application software response						Next action taken by the PCA9665/PCA9665A
		To/from I2CDAT	To I2CCON					
			STA	STO	SI	AA	MODE	
30h	Data byte in I2CDAT has been transmitted; NACK has been received	Load data byte or	0	0	0	X	0	Data byte will be transmitted; ACK/NACK will be received
		no I2CDAT action or	1	0	0	X	0	Repeated START will be transmitted;
		no I2CDAT action or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		no I2CDAT action	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset
38h	Arbitration lost in SLA+W or Data bytes	No I2CDAT action or	0	0	0	0	0	I²C-bus will be released; PCA9665/PCA9665A will enter Slave mode.
		No I2CDAT action or	0	0	0	1	0	I²C-bus will be released; PCA9665/PCA9665A will enter the Slave mode.
		No I2CDAT action	1	0	0	X	0	A START condition will be transmitted when the bus becomes free

8.3.2 Master Receiver Byte mode

In the Master Receiver Byte mode, a number of data bytes are received from a slave transmitter one byte at a time (see Figure 7). The transfer is initialized as in the Master Transmitter Byte mode.

The Master Receiver Byte mode may now be entered by setting the STA bit. The I²C-bus state machine will first test the I²C-bus and generate a START condition as soon as the bus becomes free. When a START condition is transmitted, the Serial Interrupt flag (SI) is set, the Interrupt line ($\overline{\text{INT}}$) goes LOW and the status code in the status register (I2CSTA) will be 08h. This status code must be used to vector to an interrupt service routine that loads I2CDAT with the slave address and the data direction bit (SLA+R). A write to I2CCON resets the SI bit, clears the Interrupt ($\overline{\text{INT}}$ goes HIGH) and allows the serial transfer to continue.

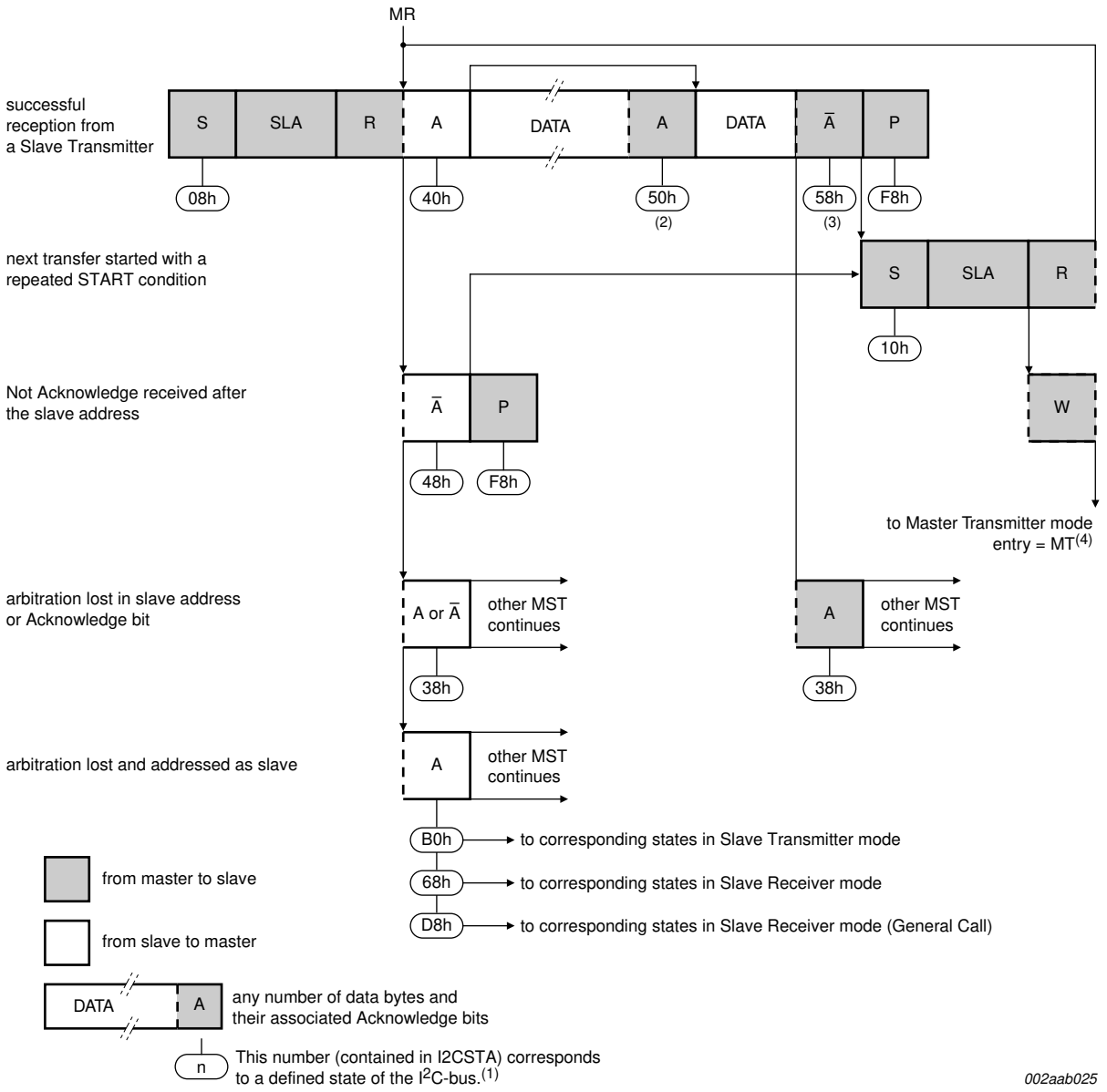
When the slave address and the data direction bit have been transmitted, the serial interrupt flag (SI) is set again, the Interrupt line ($\overline{\text{INT}}$) goes LOW again and I2CSTA is loaded with the following possible codes:

- 40h if an acknowledgment bit (ACK) has been received for the slave address with direction bit
- 48h if a no acknowledgment bit (NACK) has been received for the slave address with direction bit
- 38h if the PCA9665/PCA9665A lost the arbitration
- B0h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave transmitter (slave mode enabled with AA = 1)
- 68h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave receiver (slave mode enabled with AA = 1)
- D8h if the PCA9665/PCA9665A lost the arbitration and is addressed as a slave receiver during a General Call sequence (slave mode enabled with AA = 1 and General Call address enabled with GC = 1 in I2CADR register).

The appropriate action to be taken for each of these status codes is detailed in Table 28. ENSIO is not affected by the serial transfer and is not referred to in Table 28.

After a repeated START condition (state 10h), the PCA9665/PCA9665A may switch to the Master Transmitter mode by loading I2CDAT with SLA+W.

Remark: A master should not transmit its own slave address.



(1) See Table 28.

(2) Defined state when a single byte is received and an ACK is sent (AA = 1).

(3) Defined state when a single byte is received and a NACK is sent (AA = 0).

(4) Master Transmitter Byte mode is entered when MODE = 0. Master Transmitter Buffered mode is entered when MODE = 1.

Fig 7. Format and states in the Master Receiver Byte mode (MODE = 0)

Table 28. Master Receiver Byte mode (MODE = 0)

Status code (I2CSTA)	Status of the I²C-bus and the PCA9665/65A	Application software response						Next action taken by the PCA9665/PCA9665A
		To/from I2CDAT	To I2CCON					
			STA	STO	SI	AA	MODE	
08h	A START condition has been transmitted	Load SLA+R	X	X	0	X	0	SLA+R will be transmitted; ACK/NACK bit will be received
10h	A repeated START condition has been transmitted	Load SLA+R or	X	X	0	X	0	SLA+R will be transmitted; ACK/NACK bit will be received
		Load SLA+W	X	X	0	X	0	SLA+W will be transmitted; PCA9665/PCA9665A will be switched to Master Transmitter Byte mode
38h	Arbitration lost in NACK bit	No I2CDAT action or	0	0	0	X	0	I²C-bus will be released; PCA9665/PCA9665A will enter a slave mode
		no I2CDAT action	1	0	0	X	0	A START condition will be transmitted when the bus becomes free
40h	SLA+R has been transmitted; ACK has been received	No I2CDAT action or	0	0	0	0	0	Data byte will be received; NACK bit will be returned
		no I2CDAT action	0	0	0	1	0	Data byte will be received; ACK bit will be returned
48h	SLA+R has been transmitted; NACK has been received	No I2CDAT action or	1	0	0	X	0	Repeated START condition will be transmitted
		no I2CDAT action or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		no I2CDAT action	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset
50h	Data byte has been received; ACK has been returned	Read data byte or	0	0	0	0	0	Data byte will be received; NACK bit will be returned
		read data byte	0	0	0	1	0	Data byte will be received; ACK bit will be returned
58h	Data byte has been received; NACK has been returned	Read data byte or	1	0	0	X	0	Repeated START condition will be transmitted
		read data byte or	0	1	0	X	0	STOP condition will be transmitted; STO flag will be reset
		read data byte	1	1	0	X	0	STOP condition followed by a START condition will be transmitted; STO flag will be reset

8.3.3 Slave Receiver Byte mode

In the Slave Receiver Byte mode, a number of data bytes are received from a master transmitter one byte at a time (see Figure 8). To initiate the Slave Receiver mode, I2CADR and I2CCON must be loaded as shown in Table 29 and Table 30.

Table 29. I2CADR initialization

Bit	7	6	5	4	3	2	1	0
Symbol	AD7	AD6	AD5	AD4	AD3	AD2	AD1	GC
Value	own slave address							X

The upper 7 bits are the I²C-bus address to which PCA9665/PCA9665A will respond when addressed by a master. GC is the control bit that allows the PCA9665/PCA9665A to respond or not to the General Call address (00h).

When programmed to logic 1, the PCA9665/PCA9665A will acknowledge the General Call address.

When programmed to logic 0, the PCA9665/PCA9665A will not acknowledge the General Call address.

Table 30. I2CCON initialization

Bit	7	6	5	4	3	2	1	0
Symbol	AA	ENSIO	STA	STO	SI	-	-	MODE
Value	1	1	0	0	0	X	X	0

ENSIO must be set to logic 1 to enable the I²C-bus interface. The AA bit must be set to enable PCA9665/PCA9665A to acknowledge its own slave address, STA, STO, and SI must be reset.

When I2CADR and I2CCON have been initialized, the PCA9665/PCA9665A waits until it is addressed by its own slave address followed by the data direction bit which must be '0' (W) to operate in the Slave Receiver mode. After its own slave address and the W bit have been received, the Serial Interrupt flag (SI) is set, the Interrupt line ($\overline{INT}$) goes LOW, and I2CSTA is loaded with 60h. This status code is used to vector to an interrupt service routine, and the appropriate action to be taken is detailed in Table 31.

The Slave Receiver Buffered mode may also be entered when:

- The arbitration is lost while the PCA9665/PCA9665A is in the master mode. See status 68h and D8h.
- The General Call Address (00h) has been received (General Call address enabled with GC = 1). See status D0h.

If the AA bit is reset during a transfer, the PCA9665/PCA9665A will return a not acknowledge (logic 1) on SDA after the next received data byte. While AA is reset, the I²C-bus state machine does not respond to its own slave address. However, the I²C-bus is still monitored and address recognition may be resumed at any time by setting AA. This means that the AA bit may be used to temporarily isolate PCA9665/PCA9665A from the I²C-bus.