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

PCA8565A

Real-time clock/calendar

Rev. 3 — 1 September 2014 Product data sheet

1. General description

The PCA8565A is a CMOS[1] Real-Time Clock (RTC) and calendar optimized for low power consumption. A programmable clock output, interrupt output, and voltage-low detector are also provided. All addresses and data are transferred serially via a two-line bidirectional I^2C-bus. Maximum bus speed is 400 kbit/s. The built-in word address register is incremented automatically after each written or read data byte.

For a selection of NXP Real-Time Clocks, see Table 37 on page 39

2. Features and benefits

- AEC-Q100 compliant for automotive applications
- Provides year, month, day, weekday, hours, minutes, and seconds based on 32.768 kHz quartz crystal
- Clock operating voltage: 1.8 V to 5.5 V
- Extended operating temperature range: −40 °C to +125 °C
- Low backup current: typical 0.65 μA at V_{DD} = 3.0 V and T_{amb} = 25 °C
- 400 kHz two-line I^2C-bus interface (at V_{DD} = 1.8 V to 5.5 V)
- Programmable clock output for peripheral devices (32.768 kHz, 1.024 kHz, 32 Hz, and 1 Hz)
- Alarm and timer functions
- Two integrated oscillator capacitors
- Internal Power-On Reset (POR)
- I^2C-bus slave address: read A3h; write A2h
- Open-drain interrupt pin

3. Applications

- Automotive
- Industrial
- Applications that require a wide operating temperature range

1. The definition of the abbreviations and acronyms used in this data sheet can be found in Section 21.

4. Ordering information

Table 1. Ordering information

Type number	Package		
	Name	Description	Version
PCA8565AU	wire bond die	9 bonding pads	PCA8565AU

4.1 Ordering options

Table 2. Ordering options

Product type number	Orderable part number	Sales item (12NC)	Delivery form	IC revision
PCA8565AU/5BB/1	PCA8565AU/5BB/1,01	935289264015	unsawn wafer; thickness 280 μm	1

5. Marking

Table 3. Marking codes

Type number	Marking code
PCA8565AU/5BB/1	PC8565A-1

6. Block diagram

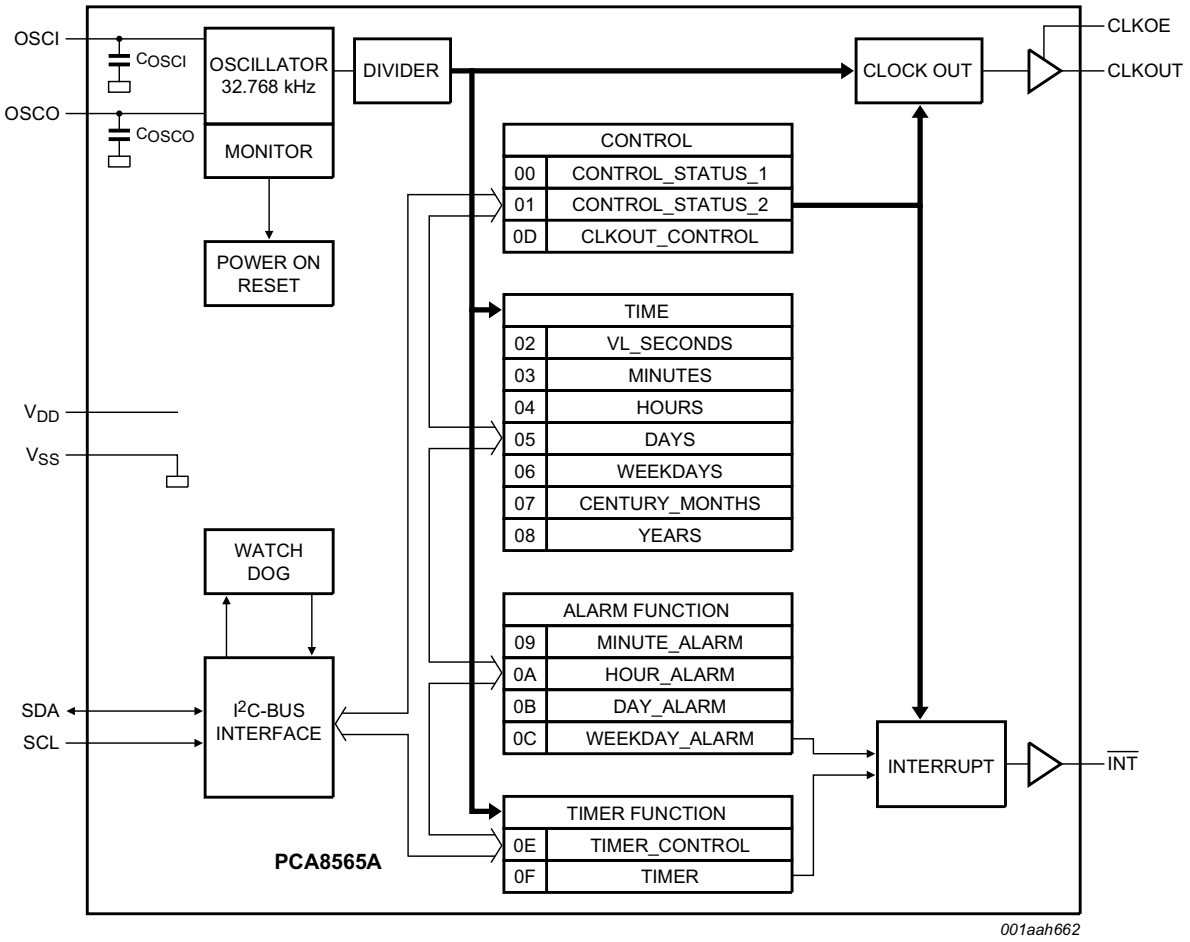
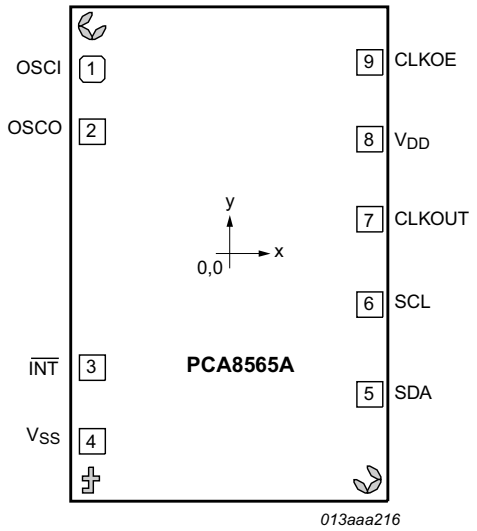


Fig 1. Block diagram of PCA8565A

7. Pinning information

7.1 Pinning



Viewed from active side. For mechanical details, see Figure 24.

Fig 2. Pin configuration for PCA8565A

7.2 Pin description

Table 4. Pin description

Input or input/output pins must always be at a defined level (V_{SS} or V_{DD}) unless otherwise specified.

Symbol	Pin	Description
OSCI	1	oscillator input
OSCO	2	oscillator output
$\overline{INT}$	3	interrupt output (open-drain; active LOW)
V_{SS}	4	ground supply voltage[1]
SDA	5	serial data input and output
SCL	6	serial clock input
CLKOUT	7	clock output (open-drain)
V_{DD}	8	supply voltage
CLKOE	9	CLKOUT output enable input

[1] The substrate (rear side of the die) is wired to V_{SS} but should not be electrically contacted.

8. Functional description

The PCA8565A contains 16 8-bit registers with an auto-incrementing address register, an on-chip 32.768 kHz oscillator with two integrated capacitors, a frequency divider which provides the source clock for the RTC, a programmable clock output, a timer, an alarm, a voltage low detector, and a 400 kHz I^2C-bus interface.

All 16 registers (see Table 5) are designed as addressable 8-bit parallel registers although not all bits are implemented. The first two registers (memory address 00h and 01h) are used as control and/or status registers. The memory addresses 02h through 08h are used as counters for the clock function (seconds up to years counters). Address locations 09h through 0Ch contain alarm registers which define the conditions for an alarm. Address 0Dh controls the CLKOUT output frequency. 0Eh and 0Fh are the timer control and timer registers, respectively.

The seconds, minutes, hours, days, weekdays, months, years, as well as the minute alarm, hour alarm, day alarm, and weekday alarm registers are all in Binary Coded Decimal (BCD) format.

When one of the RTC registers is written or read, the contents of all time counters are frozen. Therefore, faulty writing or reading of the clock and calendar during a carry condition is prevented.

8.1 Register organization

Table 5. Register overview

Bit positions labelled as - are not implemented. Bit positions labelled as N should always be written with logic 0. After reset, all registers are set according to Table 8.

Address	Register name	Bit							
		7	6	5	4	3	2	1	0
Control registers									
00h	Control_status_1	TEST1	N	STOP	N	TESTC	N	N	N
01h	Control_status_2	N	N	N	TI_TP	AF	TF	AIE	TIE
Time and date registers									
02h	VL_seconds	VL	SECONDS (0 to 59)						
03h	Minutes	-	MINUTES (0 to 59)						
04h	Hours	-	-	HOURS (0 to 23)					
05h	Days	-	-	DAYS (1 to 31)					
06h	Weekdays	-	-	-	-	-	WEEKDAYS (0 to 6)		
07h	Century_months	C	-	-	MONTH (1 to 12)				
08h	Years	YEARS (0 to 99)							
Alarm registers									
09h	Minute_alarm	AE_M	MINUTE_ALARM (0 to 59)						
0Ah	Hour_alarm	AE_H	-	HOUR_ALARM (0 to 23)					
0Bh	Day_alarm	AE_D	-	DAY_ALARM (1 to 31)					
0Ch	Weekday_alarm	AE_W	-	-	-	-	WEEKDAY_ALARM (0 to 6)		
CLKOUT control register									
0Dh	CLKOUT_control	-	-	-	-	-	-	FD	
Timer registers									
0Eh	Timer_control	TE	-	-	-	-	-	TD	
0Fh	Timer	TIMER							

8.2 Control registers

8.2.1 Register Control_status_1

Table 6. Control_status_1 - control and status register 1 (address 00h) bit description

Bit	Symbol	Value	Description	Reference
7	TEST1	0[1]	normal mode; must be set logic 0 during normal operations	Section 8.10
		1	EXT_CLK test mode	
6	N	0[2]	default value	-
5	STOP	0[1]	RTC clock runs	Section 8.11
		1	RTC clock is stopped; all RTC divider chain flip-flops are asynchronously set logic 0; the RTC clock is stopped (CLKOUT at 32.768 kHz is still available); I^2C-bus watchdog doesn't work	
4	N	0[2]	default value	
3	TESTC	0	Power-On Reset (POR) override facility is disabled; set logic 0 for normal operation	Section 8.3.1
		1[1]	Power-On Reset (POR) override is enabled	
2 to 0	N	000[2]	default value	-

[1] Default value.

[2] Bits labeled as N should always be written with logic 0.

8.2.2 Register Control_status_2

Table 7. Control_status_2 - control and status register 2 (address 01h) bit description

Bit	Symbol	Value	Description	Reference
7 to 5	N	000[1]	default value	
4	TI_TP	0[2]	$\overline{INT}$ active when TF or AF is active (subject to the status of TIE and AIE)	Section 8.9 and Section 8.7
		1	$\overline{INT}$ pulses active according to Table 29 (subject to the status of TIE); **Remark:** if AF and AIE are active then $\overline{INT}$ will be permanently active	
3	AF	0[2]	read: alarm flag inactive	Section 8.9
			write: alarm flag is cleared	
		1	read: alarm flag active	
			write: alarm flag remains unchanged	
2	TF	0[2]	read: timer flag inactive	Section 8.9
			write: timer flag is cleared	
		1	read: timer flag active	
			write: timer flag remains unchanged	
1	AIE	0[2]	alarm interrupt disabled	Section 8.9
		1	alarm interrupt enabled	
0	TIE	0[2]	timer interrupt disabled	Section 8.9
		1	timer interrupt enabled	

[1] Bits labeled as N should always be written with logic 0.

[2] Default value.

8.3 Reset

The PCA8565A includes an internal reset circuit which is active whenever the oscillator is stopped. In the reset state the I²C-bus logic is initialized including the address pointer and all registers are set according to Table 8. I²C-bus communication is not possible during reset.

Table 8. Register reset values[1]

Address	Register name	Bit							
		7	6	5	4	3	2	1	0
00h	Control_status_1	0	0	0	0	1	0	0	0
01h	Control_status_2	0	0	0	0	0	0	0	0
02h	VL_seconds	1	x	x	x	x	x	x	x
03h	Minutes	x	x	x	x	x	x	x	x
04h	Hours	x	x	x	x	x	x	x	x
05h	Days	x	x	x	x	x	x	x	x
06h	Weekdays	x	x	x	x	x	x	x	x
07h	Century_months	x	x	x	x	x	x	x	x
08h	Years	x	x	x	x	x	x	x	x
09h	Minute_alarm	1	x	x	x	x	x	x	x
0Ah	Hour_alarm	1	x	x	x	x	x	x	x
0Bh	Day_alarm	1	x	x	x	x	x	x	x
0Ch	Weekday_alarm	1	x	x	x	x	x	x	x
0Dh	CLKOUT_control	x	x	x	x	x	x	0	0
0Eh	Timer_control	0	x	x	x	x	x	1	1
0Fh	Timer	x	x	x	x	x	x	x	x

[1] Registers marked ‘x’ are undefined at power-up and unchanged by subsequent resets.

8.3.1 Power-On Reset (POR) override

The POR duration is directly related to the crystal oscillator start-up time. Due to the long start-up times experienced by these types of circuits, a mechanism has been built in to disable the POR and hence speed up on-board test of the device. The setting of this mode requires that the I²C-bus pins, SDA and SCL, be toggled in a specific order as shown in Figure 3. All timings are required minimums.

Once the override mode has been entered, the device immediately stops, being reset, and normal operation may commence, i.e., entry into the EXT_CLK test mode via I²C-bus access. The override mode may be cleared by writing logic 0 to TESTC. TESTC must be set logic 1 before re-entry into the override mode is possible. Setting TESTC logic 0 during normal operation has no effect, except to prevent entry into the POR override mode.

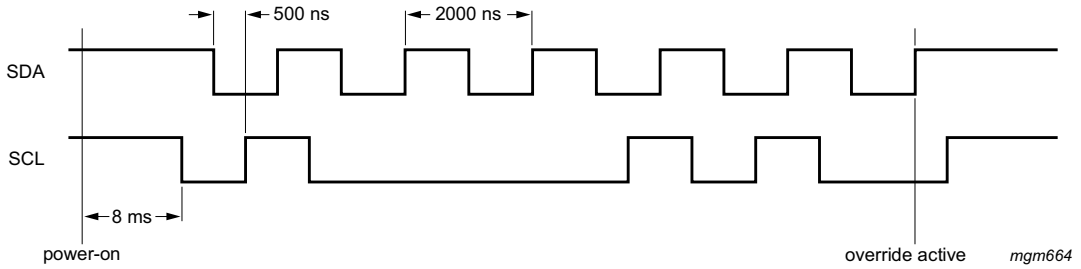


Fig 3. POR override sequence

8.4 Time and date registers

The majority of the registers are coded in the BCD format to simplify application use.

8.4.1 Register VL_seconds

Table 9. VL_seconds - seconds and clock integrity status register (address 02h) bit description

Bit	Symbol	Value	Place value	Description
7	VL	0	-	clock integrity is guaranteed
		1[1]	-	integrity of the clock information is not guaranteed
6 to 4	SECONDS	0 to 5	ten's place	actual seconds coded in BCD format, see Table 10
3 to 0		0 to 9	unit place	

[1] Start-up value.

Table 10. Seconds coded in BCD format

Seconds value in decimal	Upper-digit (ten's place)			Digit (unit place)			
	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00	0	0	0	0	0	0	0
01	0	0	0	0	0	0	1
02	0	0	0	0	0	1	0
:	:	:	:	:	:	:	:
09	0	0	0	1	0	0	1
10	0	0	1	0	0	0	0
:	:	:	:	:	:	:	:
58	1	0	1	1	0	0	0
59	1	0	1	1	0	0	1

8.4.1.1 Voltage-low detector and clock monitor

The PCA8565A has an on-chip voltage-low detector (see Figure 4). When V_{DD} drops below V_{low}, bit VL in the VL_seconds register is set to indicate that the integrity of the clock information is no longer guaranteed. The VL flag can only be cleared by command.

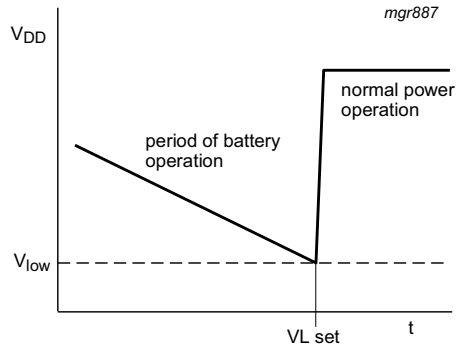


Fig 4. Voltage-low detection

The VL flag is intended to detect the situation when V_{DD} is decreasing slowly, for example under battery operation. Should the oscillator stop or V_{DD} reach V_{low} before power is re-asserted, then the VL flag is set. This will indicate that the time may be corrupted.

8.4.2 Register Minutes

Table 11. Minutes - minutes register (address 03h) bit description

Bit	Symbol	Value	Place value	Description
7	-	-	-	unused
6 to 4	MINUTES	0 to 5	ten's place	actual minutes coded in BCD format
3 to 0		0 to 9	unit place	

8.4.3 Register Hours

Table 12. Hours - hours register (address 04h) bit description

Bit	Symbol	Value	Place value	Description
7 to 6	-	-	-	unused
5 to 4	HOURS	0 to 2	ten's place	actual hours coded in BCD format
3 to 0		0 to 9	unit place	

8.4.4 Register Days

Table 13. Days - days register (address 05h) bit description

Bit	Symbol	Value	Place value	Description
7 to 6	-	-	-	unused
5 to 4	DAYS[1]	0 to 3	ten's place	actual day coded in BCD format
3 to 0		0 to 9	unit place	

[1] The PCA8565A compensates for leap years by adding a 29th day to February if the year counter contains a value which is exactly divisible by 4, including the year 00.

8.4.5 Register Weekdays

Table 14. Weekdays - weekdays register (address 06h) bit description

Bit	Symbol	Value	Description
7 to 3	-	-	unused
2 to 0	WEEKDAYS	0 to 6	actual weekday values, see Table 15.

Table 15. Weekday assignments

Day[1]	Bit		
	2	1	0
Sunday	0	0	0
Monday	0	0	1
Tuesday	0	1	0
Wednesday	0	1	1
Thursday	1	0	0
Friday	1	0	1
Saturday	1	1	0

[1] Definition may be re-assigned by the user.

8.4.6 Register Century_months

Table 16. Century_months - century flag and months register (address 07h) bit description

Bit	Symbol	Value	Place value	Description
7	C[1]	0[2]	-	indicates the century is x
		1	-	indicates the century is x + 1
6 to 5	-	-	-	unused
4	MONTHS	0 to 1	ten's place	actual month coded in BCD format, see Table 17
3 to 0		0 to 9	unit place	

[1] This bit may be re-assigned by the user.

[2] This bit is toggled when the register Years overflows from 99 to 00.

Table 17. Month assignments in BCD format

Month	Upper-digit (ten's place)	Digit (unit place)			
	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
January	0	0	0	0	1
February	0	0	0	1	0
March	0	0	0	1	1
April	0	0	1	0	0
May	0	0	1	0	1
June	0	0	1	1	0
July	0	0	1	1	1
August	0	1	0	0	0
September	0	1	0	0	1
October	1	0	0	0	0
November	1	0	0	0	1
December	1	0	0	1	0

8.4.7 Register Years

Table 18. Years - years register (08h) bit description

Bit	Symbol	Value	Place value	Description
7 to 4	YEARS	0 to 9	ten's place	actual year coded in BCD format
3 to 0		0 to 9	unit place	

8.5 Setting and reading the time

Figure 5 shows the data flow and data dependencies starting from the 1 Hz clock tick.

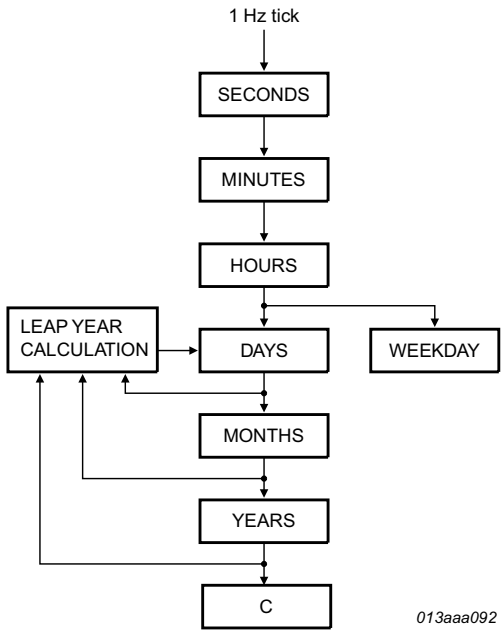


Fig 5. Data flow for the time function

During read/write operations, the time counting circuits (memory locations 02h through 08h) are blocked.

This prevents

- Faulty reading of the clock and calendar during a carry condition
- Incrementing the time registers, during the read cycle

After this read/write access is completed, the time circuit is released again and any pending request to increment the time counters that occurred during the read access is serviced. A maximum of 1 request can be stored; therefore, all accesses must be completed within 1 second (see Figure 6).

As a consequence of this method, it is very important to make a read or write access in one go, that is, setting or reading seconds through to years should be made in one single access. Failing to comply with this method could result in the time becoming corrupted.

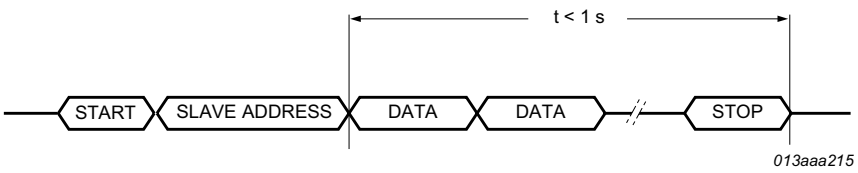


Fig 6. Access time for read/write operations

As an example, if the time (seconds through to hours) is set in one access and then in a second access the date is set, it is possible that the time may increment between the two accesses. A similar problem exists when reading. A roll over may occur between reads thus giving the minutes from one moment and the hours from the next.

Recommended method for reading the time:

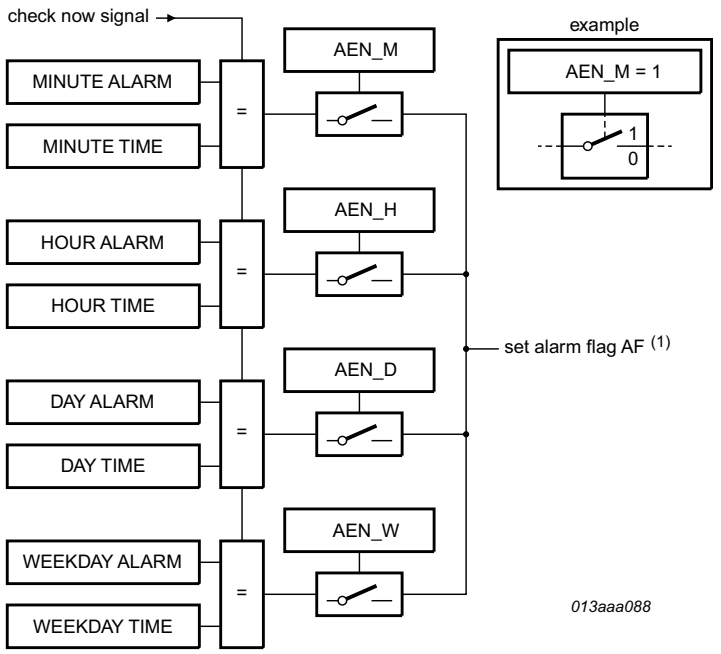
1. Send a START condition and the slave address for write (A2h).
2. Set the address pointer to 2 (VL_seconds) by sending 02h.
3. Send a RESTART condition or STOP followed by START.
4. Send the slave address for read (A3h).
5. Read VL_seconds.
6. Read Minutes.
7. Read Hours.
8. Read Days.
9. Read Weekdays.
10. Read Century_months.
11. Read Years.
12. Send a STOP condition.

8.6 Alarm registers

When one or more of the alarm registers are loaded with a valid minute, hour, day or weekday and its corresponding bit alarm enable (AE_x) is logic 0, then that information is compared with the actual minute, hour, day and weekday.

When all enabled comparisons first match, the Alarm Flag (AF) is set. AF will remain set until cleared by command. **Once AF has been cleared it is only set again when the time increments to match the alarm condition once more.** (For clearing the AF, see Section 8.9.1.1 on page 19.)

Alarm registers which have their bit AE_x at logic 1 are ignored.



(1) Only when all enabled alarm settings are matching.
It's only on increment to a matched case that the alarm is set, see Section 8.9.1.1.

Fig 7. Alarm function block diagram

8.6.1 Register Minute_alarm

Table 19. Minute_alarm - minute alarm register (address 09h) bit description

Bit	Symbol	Value	Place value	Description
7	AE_M	0	-	minute alarm is enabled
		1[1]	-	minute alarm is disabled
6 to 4	MINUTE_ALARM	0 to 5	ten's place	minute alarm information coded in BCD format
3 to 0		0 to 9	unit place	

[1] Default value.

8.6.2 Register Hour_alarm

Table 20. Hour_alarm - hour alarm register (address 0Ah) bit description

Bit	Symbol	Value	Place value	Description
7	AE_H	0	-	hour alarm is enabled
		1[1]	-	hour alarm is disabled
6	-	-	-	unused
5 to 4	HOUR_ALARM	0 to 2	ten's place	hour alarm information coded in BCD format
3 to 0		0 to 9	unit place	

[1] Default value.

8.6.3 Register Day_alarm

Table 21. Day_alarm - day alarm register (address 0Bh) bit description

Bit	Symbol	Value	Place value	Description
7	AE_D	0	-	day alarm is enabled
		1[1]	-	day alarm is disabled
6	-	-	-	unused
5 to 4	DAY_ALARM	0 to 3	ten's place	day alarm information coded in BCD format
3 to 0		0 to 9	unit place	

[1] Default value.

8.6.4 Register Weekday_alarm

Table 22. Weekday_alarm - weekday alarm register (address 0Ch) bit description

Bit	Symbol	Value	Description
7	AE_W	0	weekday alarm is enabled
		1[1]	weekday alarm is disabled
6 to 3	-	-	unused
2 to 0	WEEKDAY_ALARM	0 to 6	weekday alarm information coded in BCD format

[1] Default value.

8.7 Timer function

The 8-bit countdown timer at address 0Fh is controlled by the timer control register at address 0Eh. The timer control register determines one of 4 source clock frequencies for the timer (4.096 kHz, 64 Hz, 1 Hz, or $^1/_{60}$ Hz) and enables or disables the timer. The timer counts down from a software-loaded 8-bit binary value. At the end of every countdown, the timer sets the Timer Flag (TF) in the register Control_status_2. The TF may only be cleared by command. The asserted TF can be used to generate an interrupt (on pin $\overline{\text{INT}}$). The interrupt may be generated as a pulsed signal every countdown period or as a permanently active signal which follows the state of TF. Bit TI_TP is used to control this mode selection. When reading the timer, the current countdown value is returned.

8.7.1 Register Timer_control

The timer register is an 8-bit binary countdown timer. It is enabled and disabled via the bit TE in register Timer_control. The source clock for the timer is also selected by the TD[1:0] in register Timer_control. Other timer properties such as interrupt generation are controlled via register Control_2.

Table 23. Timer_control - timer control register (address 0Eh) bit description

Bit	Symbol	Value	Description
7	TE	0[1]	timer is disabled
		1	timer is enabled
6 to 2	-	-	unused
1 to 0	TD[1:0]		timer source clock frequency select[2]
		00	4.096 kHz
		01	64 Hz
		10	1 Hz
		11[2]	$^1/_{60}$ Hz

[1] Default value.

[2] These bits determine the source clock for the countdown timer; when not in use, TD[1:0] should be set to $^1/_{60}$ Hz for power saving.

8.7.2 Register Timer

Table 24. Timer - timer register (address 0Fh) bit description

Bit	Symbol	Value	Description
7 to 0	TIMER[7:0]	00h to FFh	countdown period in seconds: $CountdownPeriod = \frac{n}{SourceClockFrequency}$ where n is the countdown value

Table 25. Timer register bits value range

Bit							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
128	64	32	16	8	4	2	1

The timer register is an 8-bit binary countdown timer. It is enabled or disabled via the Timer_control register. The source clock for the timer is also selected by the Timer_control register. Other timer properties such as single or periodic interrupt generation are controlled via the register Control_status_2 (address 01h).

For accurate read back of the count down value, it is recommended to read the register twice and check for consistent results, since it is not possible to freeze the countdown timer counter during read back.

8.8 Register CLKOUT_control and clock output

A programmable square wave is available at the CLKOUT pin. Frequencies of 32.768 kHz, 1.024 kHz, 32 Hz and 1 Hz can be generated for use as a system clock, microcontroller clock, input to a charge pump, or for calibration of the oscillator. CLKOUT is an open-drain output, and if disabled it becomes high-impedance.

Table 26. CLKOUT_control - CLKOUT control register (address 0Dh) bit description

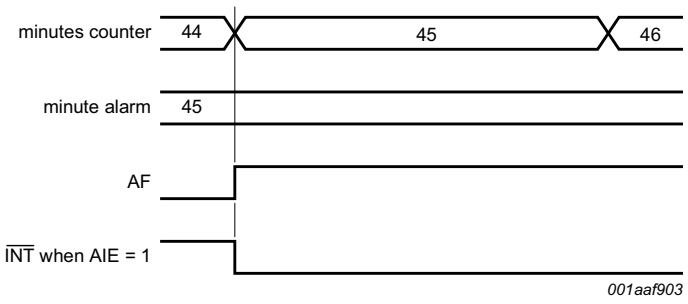
Bit	Symbol	Value	Description
7 to 2	-	-	unused
1 to 0	FD[1:0]		frequency output at pin CLKOUT
		00[1]	32.768 kHz
		01	1.024 kHz
		10	32 Hz
		11	1 Hz

[1] Default value.

8.9 Interrupt output

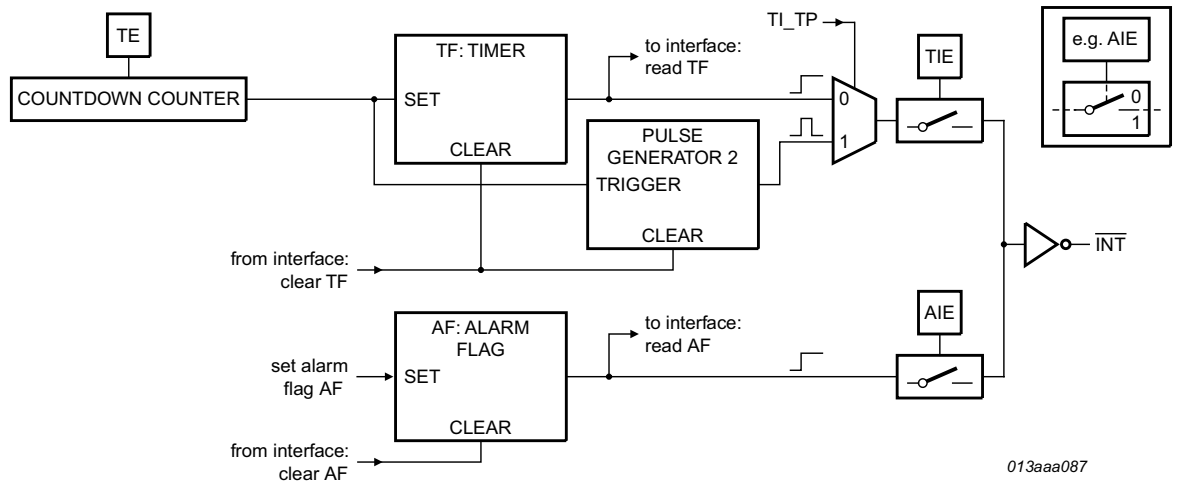
8.9.1 Bits TF and AF

When an alarm occurs, AF is set to 1. Similarly, at the end of a timer countdown, TF is set to 1. These bits maintain their value until overwritten by command. If both timer and alarm interrupts are required in the application, the source of the interrupt is determined by reading these bits.



Example where only the minute alarm is used and no other interrupts are enabled.

Fig 8. AF timing



When bits TIE and AIE are disabled, pin $\overline{INT}$ will remain high-impedance.

Fig 9. Interrupt scheme

8.9.1.1 Clearing the alarm flag (AF)

Table 28 shows an example for clearing bit AF but leaving bit TF unaffected. Clearing the flags is made by a write command; therefore bits 7, 6, 4, 1 and 0 must be written with their previous values. Repeatedly re-writing these bits has no influence on the functional behavior.

To prevent the timer flags being overwritten while clearing AF, a logical AND is performed during a write access. Writing a logic 1 will cause the flag to maintain its value, whereas writing a logic 0 will cause the flag to be reset.

Table 27. Flag location in register Control_2

Register	Bit							
	7	6	5	4	3	2	1	0
Control_2	-	-	-	-	AF	TF	-	-

The following table shows what instruction must be sent to clear bit AF. In this example bit TF is unaffected.

Table 28. Example to clear only AF (bit 3) in register Control_2

Register	Bit							
	7	6	5	4	3	2	1	0
Control_2	-	-	-	-	0	1	-	-

8.9.2 Bits TIE and AIE

These bits activate or deactivate the generation of an interrupt when TF or AF is asserted, respectively. The interrupt is the logical OR of these two conditions when both AIE and TIE are set.

8.9.3 Countdown timer interrupts

The pulse generator for the countdown timer interrupt uses an internal clock and is dependent on the selected source clock for the countdown timer and on the countdown value n. As a consequence, the width of the interrupt pulse varies (see Table 29).

Table 29. $\overline{INT}$ operation (bit TI_TP = 1)[1]

Source clock (Hz)	$\overline{INT}$ period (s)	
	n = 1[2]	n > 1[2]
4096	$1/8192$	$1/4096$
64	$1/128$	$1/64$
1	$1/64$	$1/64$
$1/60$	$1/64$	$1/64$

[1] TF and $\overline{INT}$ become active simultaneously.

[2] n = loaded countdown value. Timer stops when n = 0.

8.10 External clock (EXT_CLK) test mode

A test mode is available which allows for on-board testing. In such a mode it is possible to set up test conditions and control the operation of the RTC.

The test mode is entered by setting bit TEST1 in register Control_status_1. Then pin CLKOUT becomes an input. The test mode replaces the internal 64 Hz signal with the signal applied to pin CLKOUT. Every 64 positive edges applied to pin CLKOUT will then generate an increment of one second.

The signal applied to pin CLKOUT should have a minimum pulse width of 300 ns and a maximum period of 1000 ns. The internal 64 Hz clock, now sourced from CLKOUT, is divided down to 1 Hz by a 2^6 divide chain called a prescaler. The prescaler can be set into a known state by using the bit STOP. When the STOP bit is set, the prescaler is reset to logic 0 (STOP must be cleared before the prescaler can operate again).

From a STOP condition, the first 1 second increment will take place after 32 positive edges on pin CLKOUT. Thereafter, every 64 positive edges will cause a one-second increment.

Remark: Entry into EXT_CLK test mode is not synchronized to the internal 64 Hz clock. When entering the test mode, no assumption as to the state of the prescaler can be made.

Operation example:

1. Set EXT_CLK test mode (Control_status_1, bit TEST1 = 1).
2. Set bit STOP (Control_status_1, bit STOP = 1).
3. Clear bit STOP (Control_status_1, bit STOP = 0).
4. Set time registers to desired value.
5. Apply 32 clock pulses to pin CLKOUT.
6. Read time registers to see the first change.
7. Apply 64 clock pulses to pin CLKOUT.
8. Read time registers to see the second change.

Repeat steps 7 and 8 for additional increments.

8.11 STOP bit function

The function of the STOP bit is to allow for accurate starting of the time circuits. The STOP bit function will cause the upper part of the prescaler (F_2 to F_{14}) to be held in reset and thus no 1 Hz ticks will be generated (see Figure 10). The time circuits can then be set and will not increment until the STOP bit is released (see Figure 11 and Table 30).

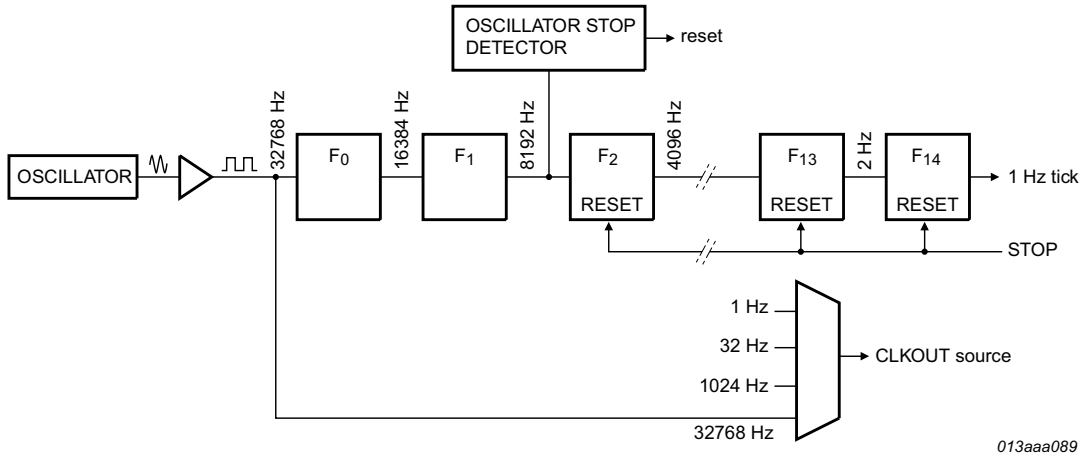


Fig 10. STOP bit functional diagram

The STOP bit function will not affect the output of 32.768 kHz on CLKOUT, but will stop the generation of 1.024 kHz, 32 Hz, and 1 Hz.

The lower two stages of the prescaler (F_0 and F_1) are not reset; and because the I^2C-bus is asynchronous to the crystal oscillator, the accuracy of re-starting the time circuits will be between zero and one 8.192 kHz cycle (see Figure 11).

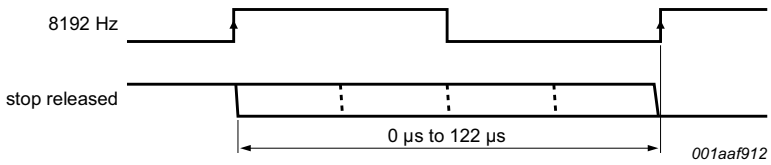
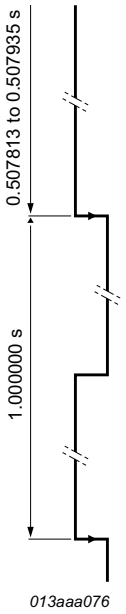


Fig 11. STOP bit release timing

Table 30. First increment of time circuits after STOP bit release

Bit	Prescaler bits	[1] 1 Hz tick	Time	Comment
STOP	F_0F_1-F_2 to F_{14}		hh:mm:ss	
Clock is running normally				
0	01-0 0001 1101 0100		12:45:12	prescaler counting normally
STOP bit is activated by user. F_0F_1 are not reset and values cannot be predicted externally				
1	XX-0 0000 0000 0000		12:45:12	prescaler is reset; time circuits are frozen
New time is set by user				
1	XX-0 0000 0000 0000		08:00:00	prescaler is reset; time circuits are frozen
STOP bit is released by user				
0	XX-0 0000 0000 0000		08:00:00	prescaler is now running
	XX-1 0000 0000 0000		08:00:00	-
	XX-0 1000 0000 0000		08:00:00	-
	XX-1 1000 0000 0000		08:00:00	-
	:		:	:
	11-1 1111 1111 1110		08:00:00	-
	00-0 0000 0000 0001		08:00:01	0 to 1 transition of F_{14} increments the time circuits
	10-0 0000 0000 0001		08:00:01	-
	:		:	:
	11-1 1111 1111 1111		08:00:01	-
	00-0 0000 0000 0000		08:00:01	-
	10-0 0000 0000 0000		08:00:01	-
	:		:	-
	11-1 1111 1111 1110		08:00:01	-
	00-0 0000 0000 0001		08:00:02	0 to 1 transition of F_{14} increments the time circuits

[1] F_0 is clocked at 32.768 kHz.

The first increment of the time circuits is between 0.507813 s and 0.507935 s after STOP bit is released. The uncertainty is caused by the prescaler bits F_0 and F_1 not being reset (see Table 30) and the unknown state of the 32 kHz clock.

9. Characteristics of the I²C-bus

The I²C-bus is for bidirectional, two-line communication between different ICs or modules. The two lines are a Serial Data Line (SDA) and a Serial CLock line (SCL). Both lines must be connected to a positive supply via a pull-up resistor. Data transfer may be initiated only when the bus is not busy.

9.1 Bit transfer

One data bit is transferred during each clock pulse. The data on the SDA line must remain stable during the HIGH period of the clock pulse, as changes in the data line at this time will be interpreted as a control signal (see Figure 12).

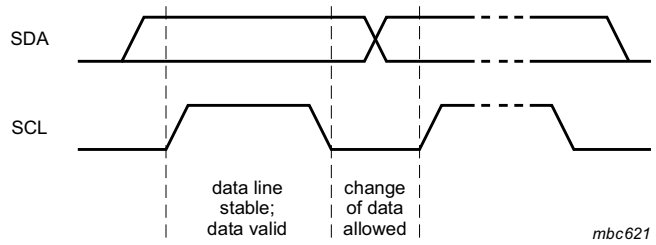


Fig 12. Bit transfer

9.2 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 13.

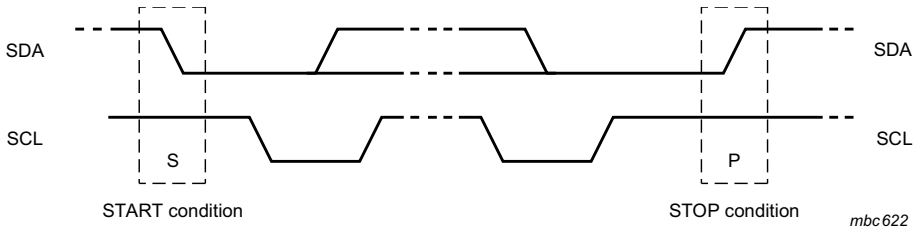


Fig 13. Definition of START and STOP conditions

9.3 System configuration

A device generating a message is a transmitter; a device receiving a message is a receiver. The device that controls the message is the master; and the devices which are controlled by the master are the slaves (see Figure 14).

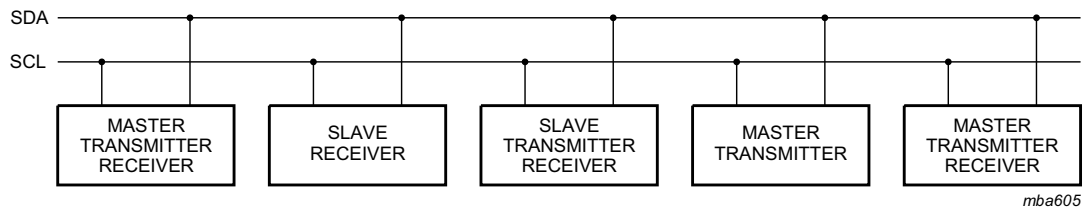


Fig 14. System configuration

9.4 Acknowledge

The number of data bytes transferred between the START and STOP conditions from transmitter to receiver is unlimited. Each byte of eight bits is followed by an acknowledge cycle.

- A slave receiver, which is addressed, must generate an acknowledge after the reception of each byte.
- Also a master receiver must generate an acknowledge after the reception of each byte that has been clocked out of the slave transmitter.
- The device that acknowledges must 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 and hold times must be taken into consideration).
- 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.

Acknowledgement on the I^2C-bus is shown in Figure 15.

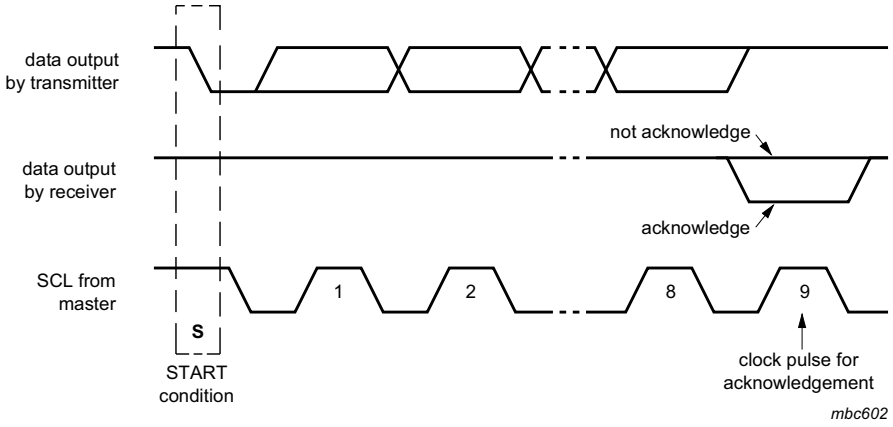


Fig 15. Acknowledgement on the I^2C-bus

9.5 I²C-bus protocol

9.5.1 Addressing

Before any data is transmitted on the I²C-bus, the device which should respond is addressed first. The addressing is always carried out with the first byte transmitted after the start procedure.

The PCA8565A acts as a slave receiver or slave transmitter. Therefore, the clock signal SCL is only an input signal, but the data signal SDA is a bidirectional line.

Two slave addresses are reserved for the PCA8565A:

Read: A3h (10100011)

Write: A2h (10100010)

The PCA8565A slave address is shown in Figure 16.

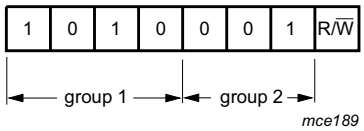


Fig 16. Slave address

9.5.2 Clock and calendar READ or WRITE cycles

The I²C-bus configuration for the different PCA8565A READ and WRITE cycles is shown in Figure 17, Figure 18, and Figure 19. The word address is a 4-bit value that defines which register is to be accessed next. The upper four bits of the word address are not used.

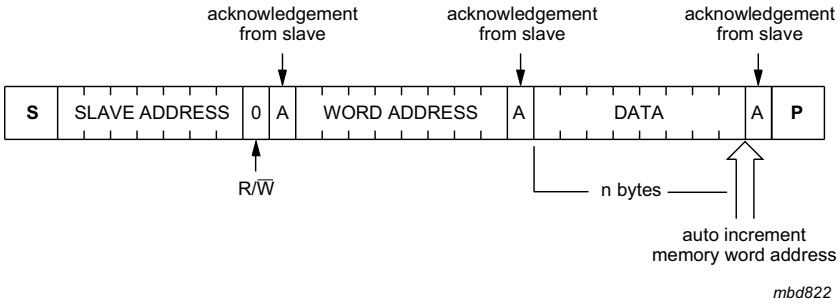


Fig 17. Master transmits to slave receiver (WRITE mode)