# The PCF8584

## Application Note

## Using the 8584 with non specified timings and other frequently asked questions

## ANZ96003

Philips
Semiconductors

PHILIPS

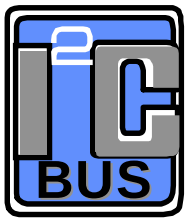
# Report No.; AN96040

**Keywords**
PCF8584
I²C Bus
Special timings
Multimaster

## Revision History

| Version | Date | Remarks |
|---|---|---|
| 0.1 | Dec 95 | first draft |
| 0.2 | Jan 96 | timing evaluation finished |
| 0.3 | April 96 | reset added |
| | | |

This application note has been written by the Telecom product support group of Philips Semiconductors in Zürich.
In case of questions and additions please contact Mr. Jürg Fries:
Fax No. +41 14 651 805
Internet J.Fries@zrh.sc.philips.com




Purchase of Philips I²C components conveys a licence under the Philips I²C patent to use the components in the I²C system, provided the system conforms to the I²C specifications defined by Philips

# 1 Table of contents

# 2 Purpose of this Application Note

This application note should answer often raised questions in the design in stage of the PCF8584. The development of the PCF8584 dates back to the time where the $I^2C$ bus was still in its definition phase. This is somehow reflected into the design of this device, and this is one of the reasons why the 8584 is quite difficult to program.
This note has the following main items:

- Behaviour of the 8584 with unspecified parallel bus sequences.
  New microcontrollers become available which have programmable bus interfaces, and which have a great freedom in the sequencing of the signals which are used to control the parallel bus side of the 8584. This gave rise to a lot of questions about parallel bus sequences.
- Multimaster systems
- Often encountered questions and problems

This document should be a live document.

- New findings will be added.
- Customer inputs are welcome and will be added if not already there

# 3 Intel Mode

This section describes the behaviour of the PCF8584 when he is initialised in the Intel mode.

## 3.1 Important Notice

- If not otherwise noted, all accesses described in this section are made to register **S0.**
- The required accesses to set up the right mode are **NOT** shown in the timing diagrams.
- For bus timings refer to the PCF8584 data sheet.
- Accesses which work only with timing restrictions are marked ***Critical.***
- Accesses which do not work at all are marked ***Illegal.***

## 3.2 Working variants

### 3.2.1 Master transmitter with typical timing

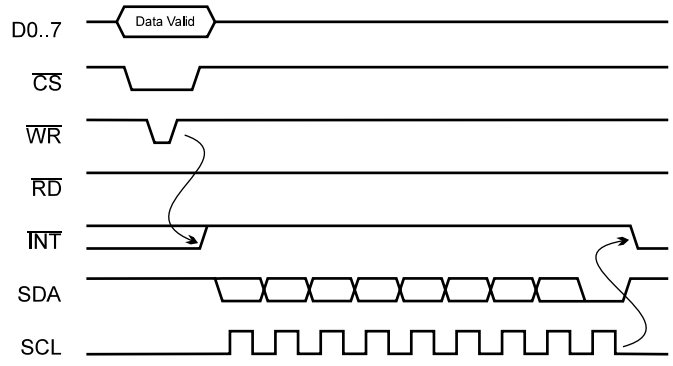



Fig. 1 Reference timing

This diagram illustrates the typical response of the INT-line (PIN bit). The PIN is reset after the raising edge of WR and negated after the 9th clock-pulse of SCL.

### 3.2.2 Master transmitter with long chip select

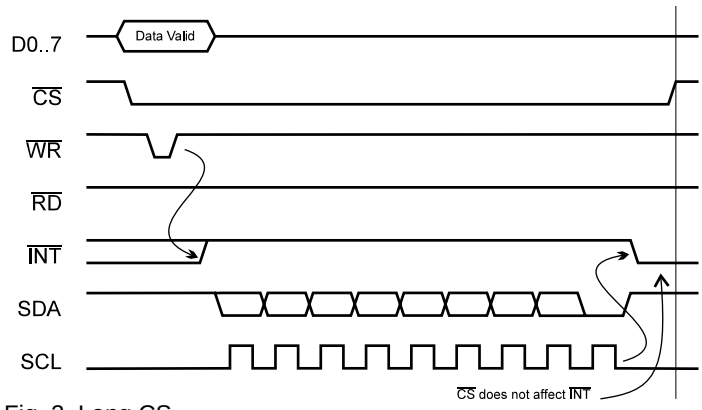



Fig. 2 Long CS

The INT is asserted after the 9th clock pulse, regardless of the CS state.

As shown in Fig. 3 the write-cycle is completed with the raising edge of WR, INT is cleared, and the next byte is sent over the $I^2C$ bus

Therefore the write timing for the master transmitter mode is uncritical.

### 3.2.3 Master transmitter with long CS and long WR

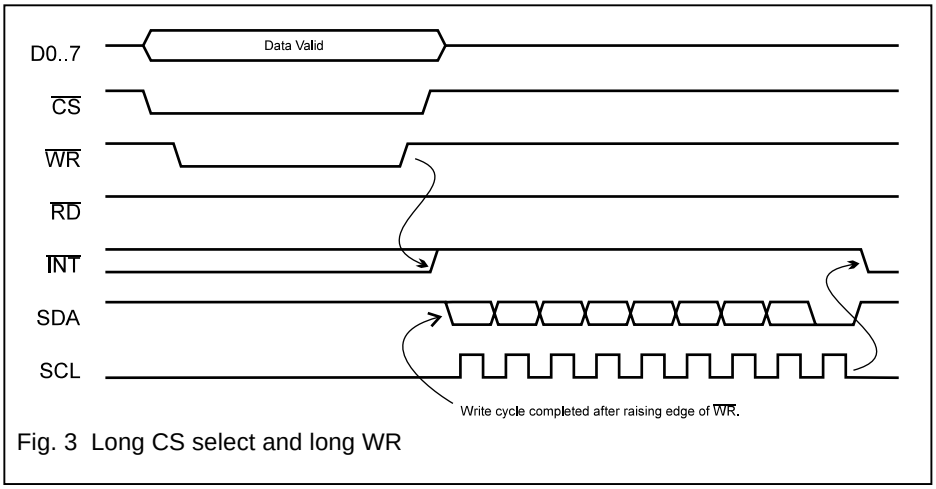



Fig. 3 Long CS select and long WR

This diagram shows the same behaviour as Fig. 2

### 3.2.4 Master receiver with typical timing

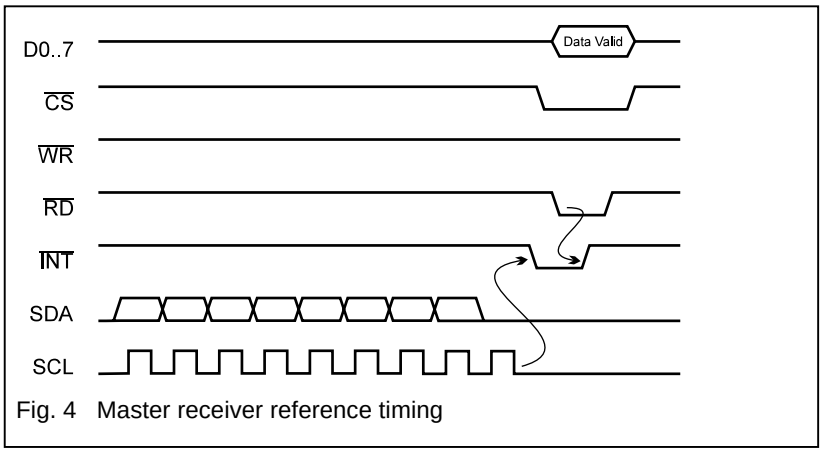



Fig. 4 Master receiver reference timing

This diagram illustrates the typical response of the INT-line (PIN bit). The PIN is reset after the falling edge of RD and negated after the 9th clock-pulse of SCL.

### 3.2.5 Master receiver with long CS and long RD Critical !

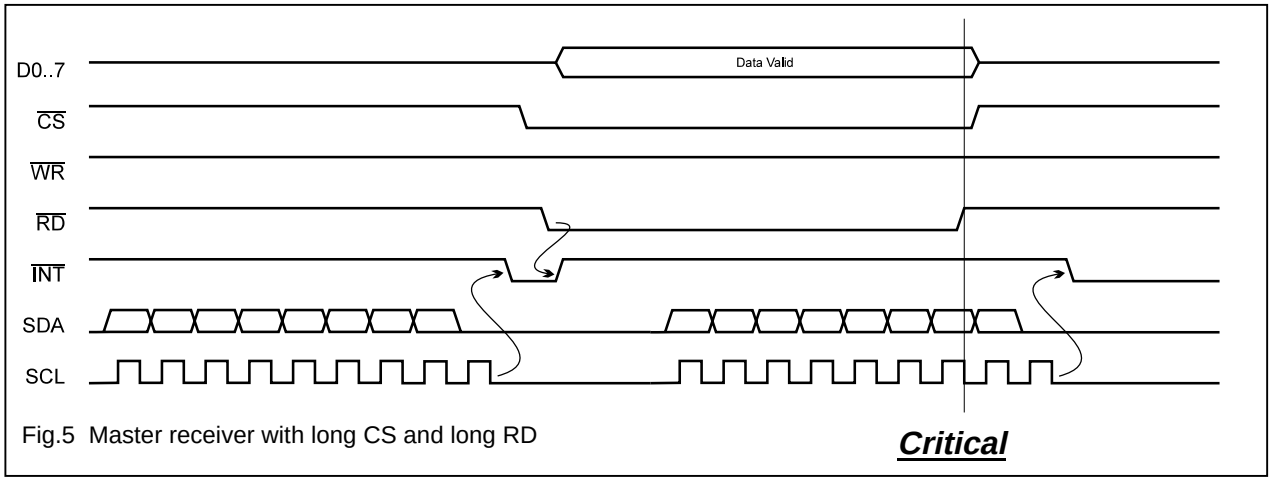



Fig.5 Master receiver with long CS and long RD

The read-cycle does not affect the behaviour of the PCF8584 as long as it is completed before the next byte is received.

### 3.2.6 Master receiver with very long CS

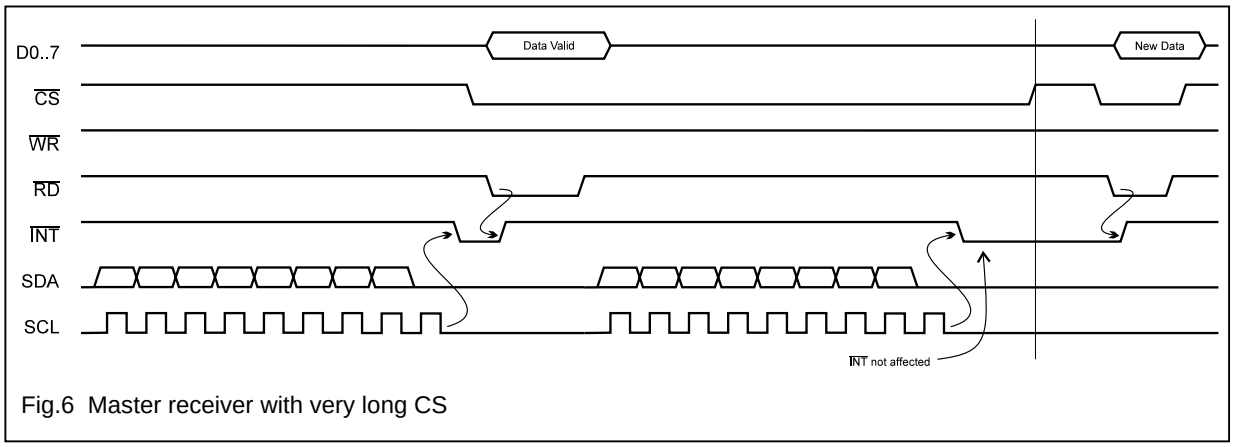



Fig.6 Master receiver with very long CS

After the 9th SCL pulse, an INT is generated regardless of the CS state

### 3.2.7 Master receiver mode with very long RD Critical !

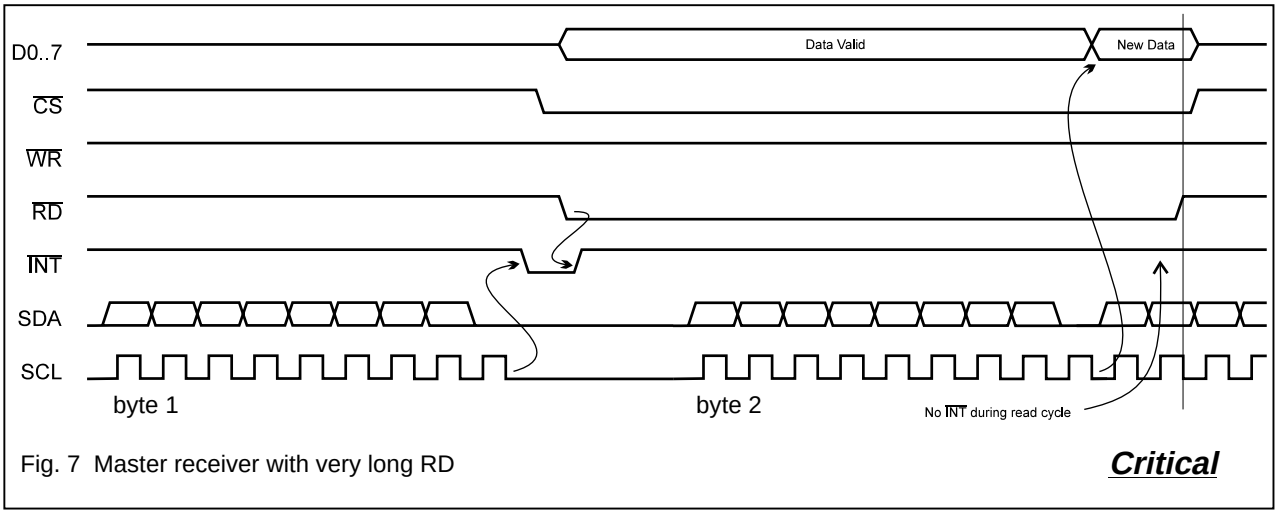



Fig. 7 Master receiver with very long RD

***Critical***

If the read-cycle is stretched beyond the 9th SCL pulse of byte 2 , no interrupt will be generated (PIN high).
After the 9th clock pulse of byte 2, the contents of byte 2 appears on the Data-Bus, byte 1 is lost.
Because the PIN bit is not set, all the transmitted bytes will be lost, because the microcontroller will not be notified by the PCF8584 of finished byte transmissions.

### 3.2.8 Master receiver mode with repeated start and write after read

The $I^2C$ bus spec. knows 4 possible combinations for a repeated start sequence. The repeated start sequence is used mainly in multimaster systems, when different devices have to be accessed without interruptions from other masters, or when memory devices have to be read. In this case very often a pointer in the memory device has to be set prior to reading the contents.

The table below shows the 4 possible combinations.

| No | **$I^2C$ bus mode before repeated start** | **$I^2C$ bus mode after repeated start** |
|---|---|---|
| 1 | write to slave | read from slave |
| 2 | write to slave | write to slave |
| 3 | read from slave | read from slave |
| 4 | read from slave | write to slave |
| | | |

The figure below shows an example of combination No. 4

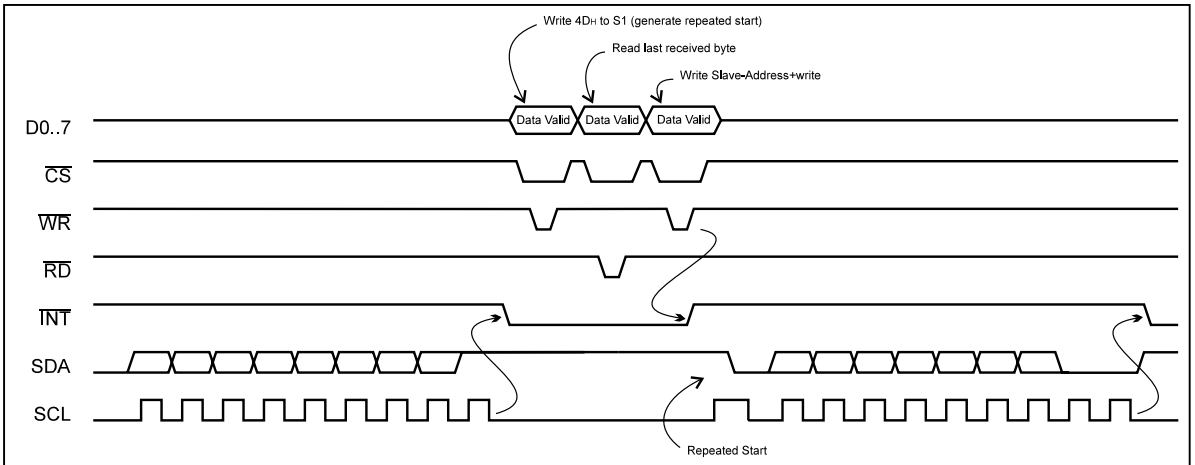



Fig. 8 Master receiver with repeated start and write after read

Fig. 8 shows how to do a repeated start-sequence.

Below the necessary steps to do this are shown:

1. Write 4Dh to S1 (generate repeated start)
2. Read the last received byte from S0
3. Write the Slave-Address + write to S0, now the repeated start will be generated.

The INT (PIN bit) is reset after the write to S0.

## 3.3 Motorola access sequence in Intel mode

### 3.3.1 WR low before CS low

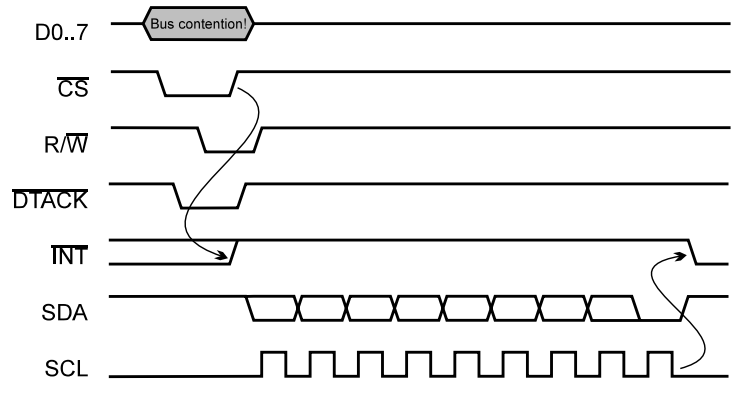



Fig. 9 WR low before CS low

A falling WR before CS does not cause any problems and is recognised as well by the PCF8584.

### 3.3.2 WR low after CS low Illegal !

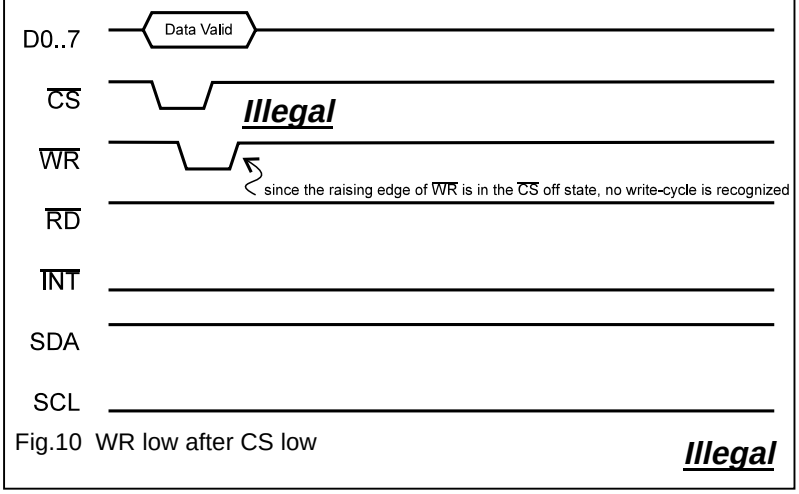



Fig.10 WR low after CS low

The write-cycle is not recognised by the PCF8584 because the raising edge of WR is not in the CS active (low) state.

***This is a non working illegal variant***

### 3.3.3 CS shorter than, and within WR Not working !

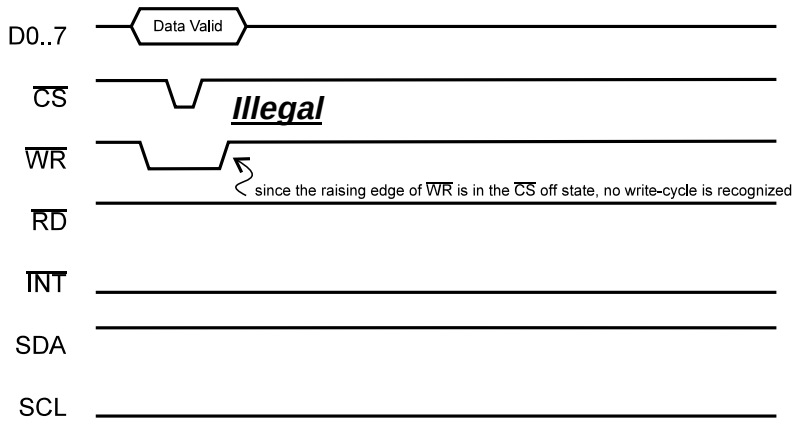



Fig.11 CS shorter than, and within WR

***Illegal***

This write-cycle fails. (see description of Fig. 10)

***This is a non working illegal variant***

# 4 Motorola Mode

This section describes the behaviour when the PCF8584 is initialised in the Motorola mode.

## 4.1 Access sequences which work under limited circumstances

### 4.1.1 Write access timing Critical !

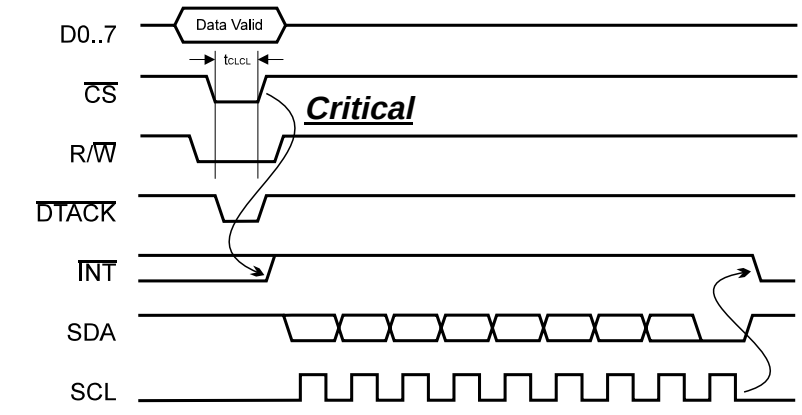



Fig.12 write access timing

***The write-cycle ($t_{CLCL}$) should not exceed 2us. This is a critical timing***

This restriction is valid for the whole frequency range.

Provided standard clock input frequencies are used.

### 4.1.2 Master receiver, very long read cycle Critical !

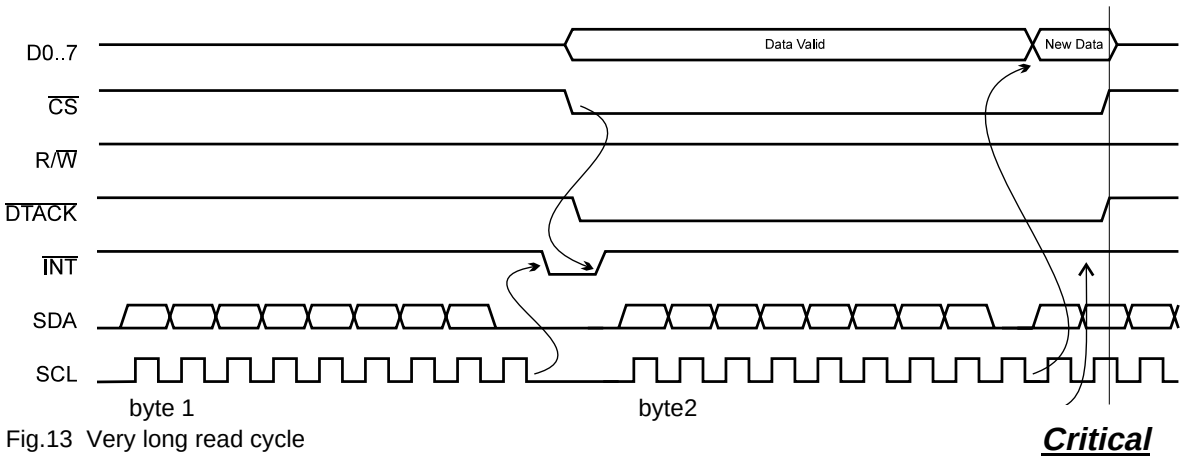



Fig.13 Very long read cycle

If the read-cycle is stretched beyond the 9th SCL pulse of byte 2 , no interrupt will be generated (PIN high).
After the 9th clock pulse of byte 2, the contents of byte 2 appears on the Data-Bus, byte 1 is lost.
Because the PIN bit is not set, all the transmitted bytes will be lost, because the microcontroller will not be notified by the PCF8584 of finished byte transmissions.

***This sequence is critical***

### 4.1.3 Falling CS before R/W Illegal !

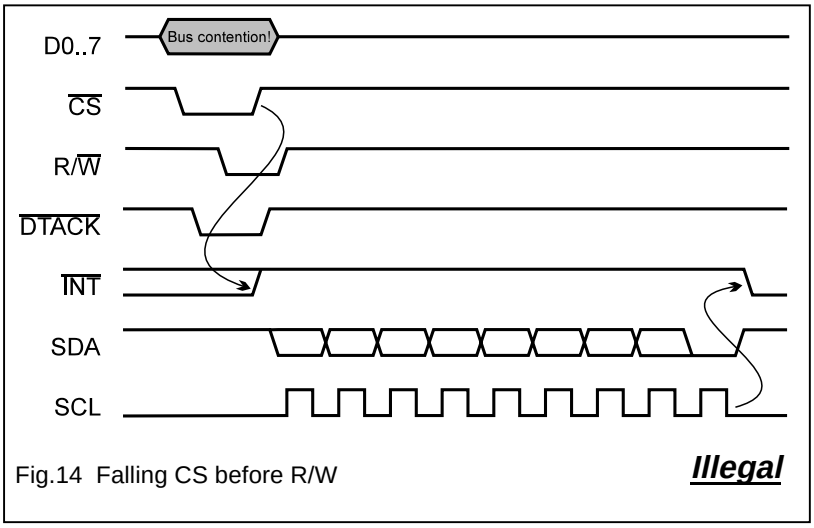



Fig.14 Falling CS before R/W

A falling CS before R/W will cause a bus contention, because before the write is done, a device read will be performed. If this happens, the written data to the PCF8584 is scrambled.

***This is a non working illegal variant***

### 4.1.4 R/W high before CS high

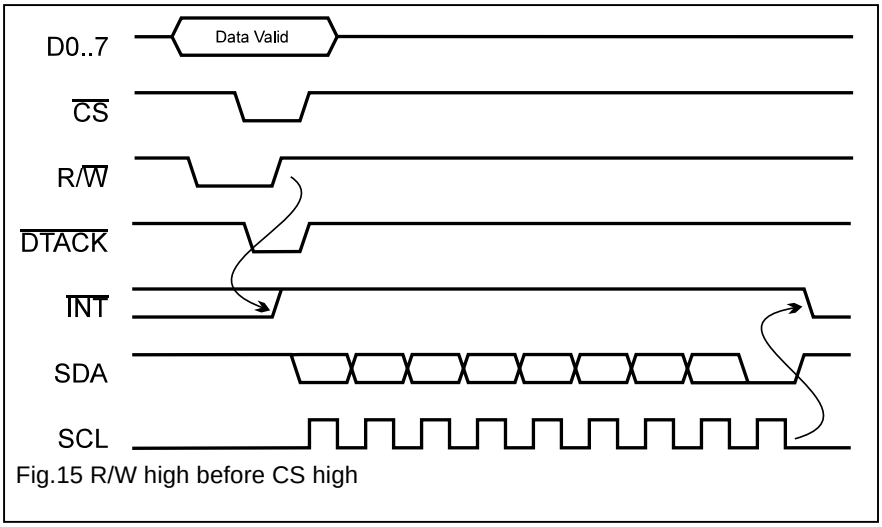



Fig.15 R/W high before CS high

This write cycle does not cause any problems.

### 4.1.5 R/W shorter than CS and within CS low period Illegal !

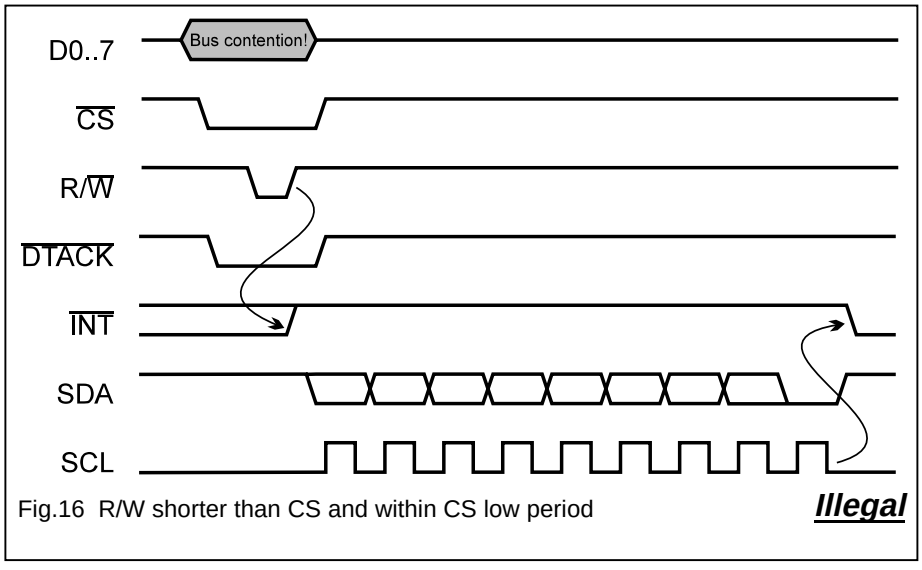



Fig.16 R/W shorter than CS and within CS low period

This write cycle fails.
(see description of Fig. 14).

***This is an non working illegal variant***

# 5 Reset

This is an addition to chapter 6.10 of the data sheet from August 95. In later revisions of the data sheet the following text will be added.

The PCF8584 only resets when the clock is running. This is also true at power up condition.

# 6 Good to know

## 6.1 Detection of $I^2C$ bus traffic

The PCF8584 must be initialised as recommended in the initialisation flowchart of the data sheet . After that the chip is in the slave receiver mode listening to the bus. It can now detect a start-, repeated start - and a stop sequence and thus determine if the bus is busy or free.

- The PCF8584 can only detect a bus busy condition, when it was able to listen to the bus and ready to track a start condition
- *It can not detect a bus busy if the initialisation is finished during an ongoing $I^2C$ transmission*
- *It will not flag a bus busy if the SDA or SCL lines are toggled without a preceding start sequence*

## 6.2 Not detected start in slave mode

In the slave mode there is a time slot where the PCF8584 is not able to detect a start sequence.

- In slave transmitter mode, this is between the time that elapses between negative acknowledge received (LRB=1) and write last byte to S0 register
- In slave receiver mode this is the time that elapses between Stop detected (STS=1) read last byte from S0 register. See the flowchart for slave receiver/transmitter mode in the data sheet

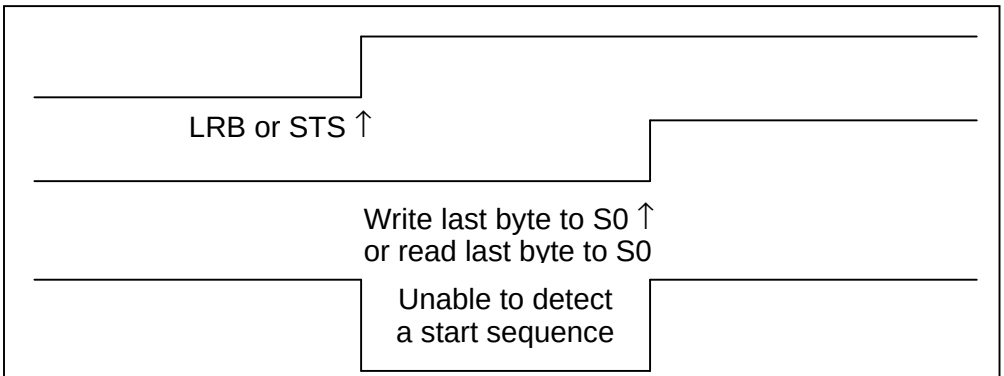



Fig. 17 Non detected start sequence

- This is no problem in a single master system
- It is usually no problem in Multimaster systems using only PCF8584s, as long as the microcontrollers which serve the 8584 have about the same performance
- It can be a problem in Multimaster systems where PCF8584s are mixed with microcontrollers which have an integrated $I^2C$ bus controller on board, as a lot of Philips 8051 derivatives do. The $I^2C$ bus controller of these micros can be set up in such a way, that they wait until the bus is free without interaction of the microcontroller. They will then start im-

mediately the transmission. In this case, the PCF8584 will not detect the start sequence, because his microcontoller is still in the loop of detecting the end of the transmission and then set the chip free, which is only the case when PIN = 1

## 6.3 Live insertion of hardware

Be aware, that a PCF8584 can only detect a bus busy condition when he was able to track the bus and see the start condition.

## 6.4 Statusregister S1

The statusregister always reflects the status of the $I^2C$ bus. During an $I^2C$ transmission the bits in the status register toggle depending on the bus status. For most cases the status should only be updated when the PIN bit is asserted.

The PIN bit is asserted in the following cased:
- 1 Byte transmitted or received
- Misplaced start
- Misplaced stop
- Repeated start

***Exception***: The PIN bit is not set when arbitration is lost. In this case the Pin bit is set after the byte of the winning master has been fully transmitted, which is at the falling edge of the acknowledge clock.

## 6.5 PCF8584 in systems with micros with on board $I^2C$

Read chapter Good to know section 6.2

# 7 Multimaster systems

Multimaster systems require some system considerations to avoid problems which can occur if the system is not designed properly.
First some general remarks:
- It is good practice to have if possible a supervisor in Multimaster systems which can handle exceptions
- Every device which can be bus master should have a time-out in the $I^2C$ bus access routines which can abort its own bus access if it is not able to finish it within a certain time limit.
- It is good practice to check PIN, BB, BER always, because they determine if a data is valid
- LAB should be added in master transmitter / receiver mode
- STS must be checked in slave receiver mode
- AAS must be checked at the beginning of the slave receiver / transmitter mode
- Check the preceding chapter Good to know
- Do not try to minimise the software. Handle every possible situation !

## 7.1 Power up in Multimaster systems

Start $I^2C$ transmissions only after all slaves have started up. Allow enough time for the initialisation, before starting any transmissions. A non initialised PCF8584 will miss ongoing transmissions, and disturb them by trying to start an own transmission.

## 7.2 Lost arbitration

If in a Multimaster system arbitration is lost, the LAB bit (lost arbitration) is set. This can happen anywhere in a transmission.

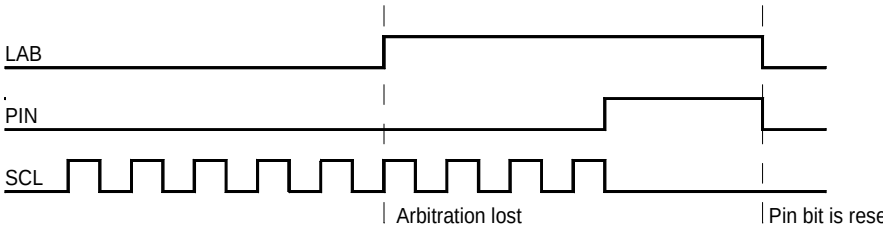



Fig. 18 Lost arbitration

If the PCF8584 is loosing the arbitration, he will release the bus. The pin bit is set, LAB is set. The PCF8584 is now in slave receiver mode. There are two possibilities to end the transmission for the loosing master.

1. Read S1, then S0
2. Read S1, then send reset to S1 (80h), then switch the ESO on again.

***Attention*** ! In both cases, the PCF8584 is not able to recognise that there is still an ongoing transmission. In order to prevent another arbitration problem the controlling software should wait for a time, longer than the longest message which can be sent over the bus, before trying to access the bus again.

## 7.3 Lost arbitration with general call- or own- address

In case the winning master has sent out or the address of the loosing master, the AAS bit is set, if the general call address has been sent, the AAS and AD0/LRB bits will be set. Basically the transmission can then go on, as the chip is now in slave receiver mode. In many cases the decision is taken to finish the transmission because there is always a doubt if during the arbitration process all the data has been processed properly. In this case read the preceding chapter.

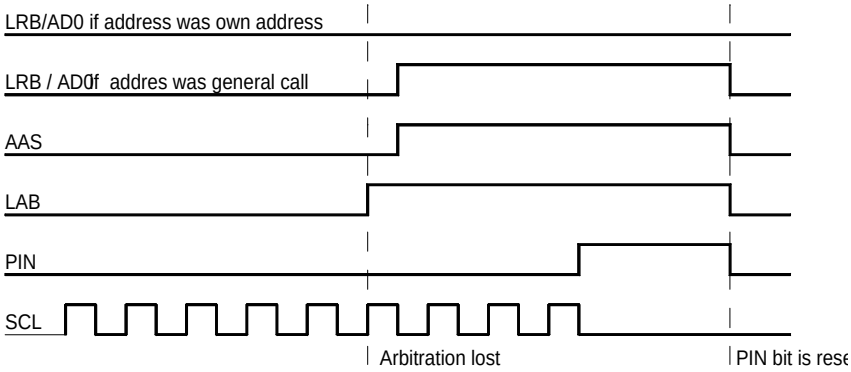



Fig. 19 Lost arbitration with general call or own address

# 8 Appendix

## 8.1 Recommended literature

- $I^2C$ Peripherals Philips Data Handbook IC 12 form 1995 or newer
- The I2C bus and how to use it 1995 update (12 NC 9398 393 400119) Included in IC 12

## 8.2 Distribution

This manual is available on the Philips Semiconductor BBS in Eindhoven and Sunnyvale in a compressed self extracting postscript file.
For access information see below.

**BBS Eindhoven, The Netherlands**
The Bulletin Board system in Eindhoven has available application and demonstration programs for downloading.

| | |
|---|---|
| Modemspeed: | V32bis/HST/Vterbo/Vfast/V34 |
| Transfer protocols: | X/Y/Z-modem, Sealink, Hydra, Telink, 1K-Xmodem |
| Line parameters: | 8 data bits, 1 stop bit, no parity |

The telephone numbers for the BBS in Eindhoven are:

| | | |
|---|---|---|
| BBS: | +31 - 40 72 11 02 | (24 hours a day, 7 days a week) |
| BBS Helpdesk : | +31 - 40 72 27 49 | (Mon - Fri, 9:00 to 16:00) |

**BBS Sunnyvale, USA**
The MCO Bulletin Board system in Sunnyvale has available newsletters, application and demonstration programs for downloading. The access is done by modem at 300 to 14'400 Baud.
The telephone numbers are:

| | |
|---|---|
| In the US: | (800) 451 66 44 |
| or | +1 - 408 991 24 |

This application note has been written by the Telecom product support group of Philips Semiconductors in Zürich.
In case of questions and additions please contact Mr. Jürg Fries:

| | |
|---|---|
| Fax No. | +41 14 651 805 |
| Internet | J.Fries@zrh.sc.philips.com |