DATA SHEET

PCF7953 Monitor and Download Interface

Functional Description

8 November 2011

Confidential

Revision 1.4

NXP
Semiconductors

CONTENT

1 FEATURES

- In-Circuit operation
- Two wire serial interface MSDA(P11)/MSCL(P10)
- Supports external or internal clock operation
- Convenient read/write access to EROM and EEPROM
- Device modes controlled by EROM/EEPROM

2 GENERAL DESCRIPTION

This manual describes the in-circuit monitor and download interface for the PCF7953 device. Other Products such as PCF7x45 uses a similar in-circuit monitor and download interface.

The device contains a 8Kbyte firmware as pre-defined and provided by Philips which is operating the in-circuit monitor interface

For development and prototyping purpose various dedicated tools are available as integral part of the development tool set.

Please also consult the corresponding specification given in PCF7953 datasheet, of the 8-Bit MICRO RISC KERNEL MRKII and observe the APPLICATION NOTES, in order to identify any variations or discrepancies applicable (see section 5).

3 FUNCTIONAL DESCRIPTION FIRMWARE

Additionally to the application program (E-ROM), the device incorporates a 8 KByte large firmware ROM. The content of this ROM is predefined by Philips and holds the device boot routine, a ROM library and controls the in-circuit monitor and download interface.

Boot Routine:

The boot routine is invoked right after a device reset and subsequently configures the device, checks and configures the supply condition, checks protection flags in E-ROM and EEPROM, reads and invokes transponder emulation modes according to the EEPROM configuration read or passes control to application in E-ROM or in-circuit monitor and download interface accordingly.

ROM Library

The ROM library features a set of functions to emulate a variety of Philips transponders, as well as a set of generic functions. The corresponding functions interrogated from the application program by a distinct instruction, called system call (SYS).

In-Circuit Monitor and Download Routine

If activated, the firmware operates the in-circuit monitor and download interface, which features communication via a two-write serial interface (MSCL = P10 / MSDA = P11), in order to provide means for E-ROM initialization and to monitor and manipulate the embedded peripherals in the context of system debugging. During debug-mode the ports P10 and P11 cannot be used for normal operation, since the corresponding pins are co-used for the MSCL and MSDA interface-lines.

For development and prototyping purposes dedicated tools are available that are part of the PCF7953-development tool set.

The Monitor routine does occupy 4 bytes Stack space, which may need to be considered during application program design.

3.1 Boot Routine

In the moment the BOOT routine commences, the on-chip RC Oscillator and other device circuitry are initialized according to the Device and TRIM Configuration (DCFG, TRIM) values stored in the EEPROM. Additional, the setting of P11 is read and stored for later use.

Next, the present device mode is evaluated. In case the device signals TAMPERED mode, device operation is halted, to render the device useless. Latter one is signalled by a MSCL low-to-high transition.

Next, the device verifies the supply condition, by testing the Supply Switch state (PMODE). Device operation commences in TRANSPONDER mode, in case PMODE signals an LF Field supply condition (PMODE = 0) or an LF Field is being detected (FLD = 1), even if a battery supply condition is present (PMODE = 1). In the latter case the device will ignore the battery supply and forces an LF Field supply condition (PMODE = 0). In this case, the system clock is derived from the Contactless Interface clock recovery circuitry (LF Field clock).

Subsequently, the device will evaluate the device configuration as stored in EEPROM (DCFG, see also PCF7953 datasheet) regarding the Transponder Emulation. In case the Transponder Emulation is disabled (TEN = 0) the device will utilize the WARM BOOT vector TRANSPONDER (0010_H) after completion of the BOOT routine. Otherwise, in case the Transponder Emulation is enabled (TEN = 1), the BOOT routine quits and passes control directly to the Monolithic Transponder Emulation according to the transponder mode configuration (TM). The Monolithic Transponder Emulation does terminate under certain conditions, meaning the emulation is configured for Sub Command handling by the Application Code (USUB = 1, see also PCF7953 datasheet) and a corresponding transponder Sub Command is received. In this case, device control is returned to the application code and program execution commences from the Sub Command Detected vector (location 0012_H, see also PCF7953 datasheet).

Anyhow, when verifying the supply condition, by testing the Supply Switch state (PMODE), the BOOT routine may also detect BATTERY mode (PMODE = 1) and no LF Field being present (FLD = 0). In case BATTERY mode is detected the internal RISC clock is set to 500 kHz and the bit BATPOR in the preprocessor status register is evaluated. This bit indicates a ‘first battery power-on sequence’. If BATPOR is set the trim and configuration registers of the active receiver will be initialized. (Bias and oscillator are trimmed). Otherwise the device will directly utilize the WARM BOOT vector BATTERY (0000_H) after completion of the BOOT routine.

Finally, the BOOT routine verifies the device mode again. This time, in order to enable the debug features, in case the device is configured for INITIAL mode. Otherwise, if the device is set to PROTECTED mode, the debug features are not available. Any other coding of the device mode configuration bits is not valid and forces the device into TAMPERED mode.

However, prior to passing device control to the corresponding WARM BOOT location in the Application Code Memory, the pin MSDA is tested. MSDA is pulled down during this sequence by an internal weak pull-down resistor (activated during the boot-sequence by setting the bit PWEAK). MSDA will be evaluated 15 times.

In case MSDA is not connected externally, the connection of the internal pull-down resistor generates a high-low-high pulse on MSDA which may be unwanted in terms of application. Thus, in case the PROTECTED mode has been detected and MSDA (P11) is detected high at the BOOT start (no button press is assumed), a WARM BOOT is performed without the possibility to enter the MONITOR mode. Thus, the high-low-high pulse on MSDA is inhibited. In case MSDA (P11) is detected low at the BOOT start (a button press is assumed), normal operation takes place.

In case MSDA is detected low, a WARM BOOT is executed after evaluating the debug configuration byte (DBG_CFG) located in EROM @1FFEh.

Since the debug lines (MSCL, MSDA) are utilized additional as general purpose ports (P10, P11, a WARM BOOT configures P10, P11 as general purpose ports in normal operation. In this case P10 and P11 are configured as input and the external break functionality (a low condition on MSDA generates a break) is disabled. Thus, a running EROM program is not breakable and no debug communication is possible.

In order to activate debug functionality even in case of a WARM BOOT, the application program should be set up as follows:

- Define a code segment in order to set EROM location 1FFEh to 00h by adding a segment definition (RIDE assembly code):

```
DBG_CONF SEGMENT CODE AT 1FFEh
ASEG DBG_CONF
db 00h,
```

or (IAR assembly code)

```
ASEGN CODE:CODE,1FFEh
db 000h
```

- Set P10 (MSCL) high and configure it as output (keep P11 as input = default after reset) by adding following code lines right after the start of the application program (assembly code):

```
setb P10     ;  set P10 (MSCL) to high
setb IO10    ;  configure P10 (MSCL) as output
```

or in C code for the IAR compiler

```
P10 = 1;  // set P10 (MSCL) to high
IO10 = 1; // configure P10 (MSCL) as
                 output
```

Note that in case a P10 (MSCL) is configured as output and the target device is force into POWER-OFF mode, the device will generate a Wake Up. Thus, the target device will be restarte after the Power Down.

In case MSDA is detected HIGH (internal pull-down overridden by an external pull-up), the MONITOR routine is entered. Thereafter a negative edge on MSCL indicates that by appropriate setting of MSDA the clock source of the device can be selected. If MSDA is detected low, the clock will be derived from the on-chip RC oscillator (internal mode). Otherwise (MSDA = high) the clock will be derived from external (external mode).

In order prevent the on-chip Watchdog to terminate the MONITOR mode, the command c_trace (see section 3.5) need to be sent before a Watchdog Timer overflow occurs. Thus, in case the MONITOR has been entered accidentally, the Watchdog terminates the MONITOR mode again.

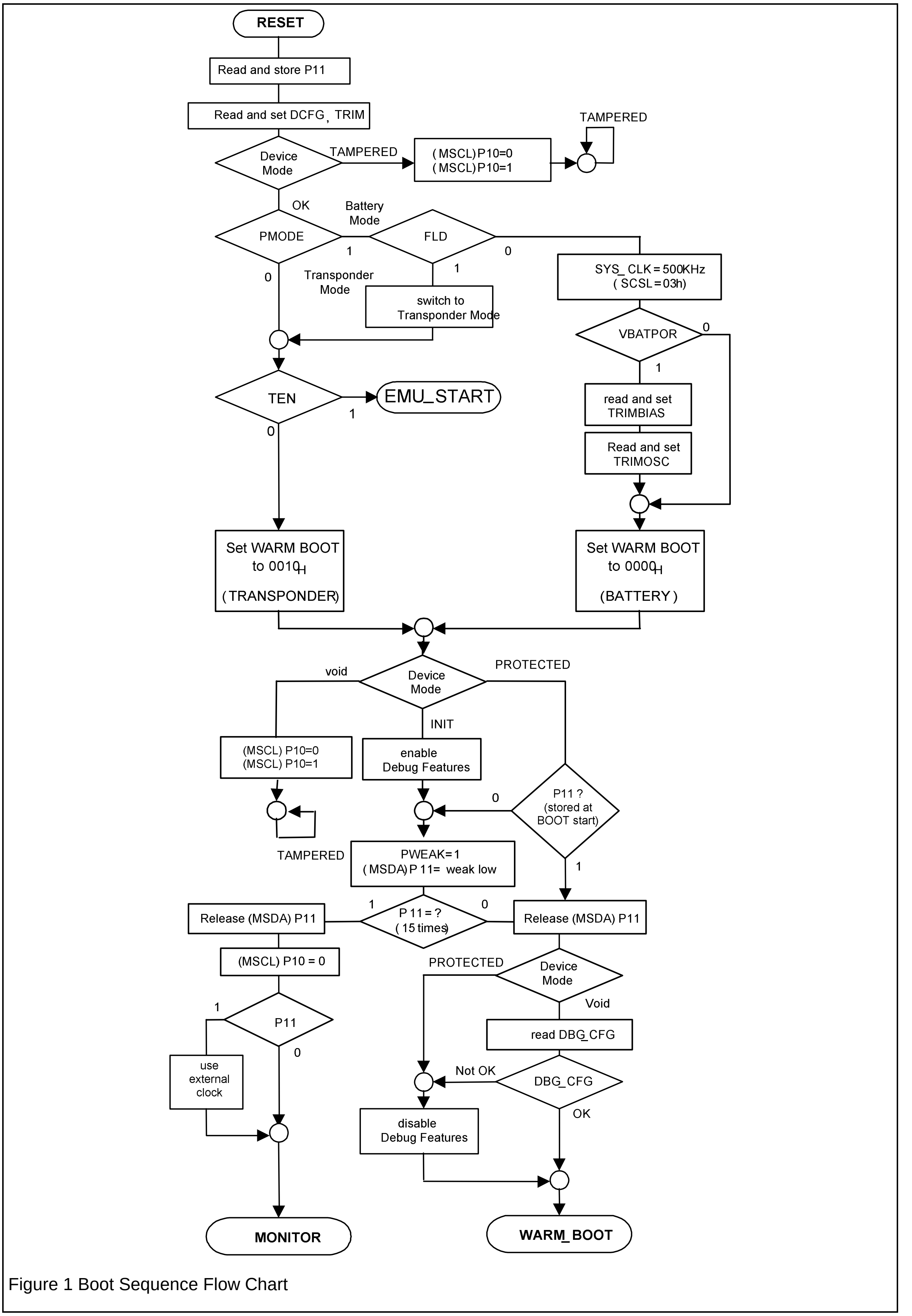


Figure 1 Boot Sequence Flow Chart

3.2 External Device Modes

The device has different modes determining the behavior of the monitor interface. The modes are controlled by the byte EDM (E-ROM device mode) in E-ROM and the byte EEDM (EEPROM device mode) in EEPROM.

EEDM (EEPROM Device Mode)
Located in EEPROM

Applicable EEPROM Device Modes:

OK, TAMPERED

EDM (E-ROM Device Mode)
Located in E-ROM

Applicable EEPROM Device Modes:

INITIAL, PROTECTED

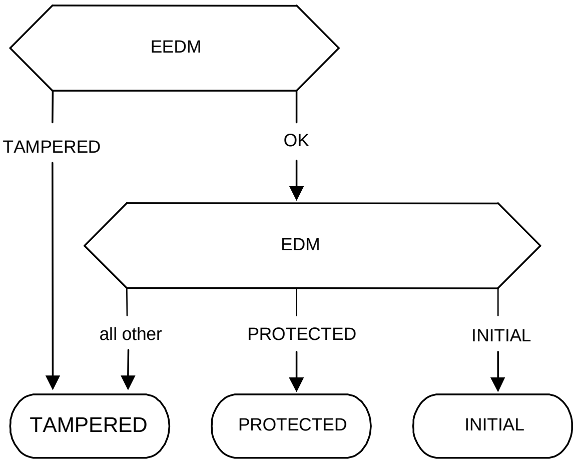


Figure 2 Switch between different external device modes

In order to change device modes the monitor commands c_er_erom and c_protect are used (see Figure 3).

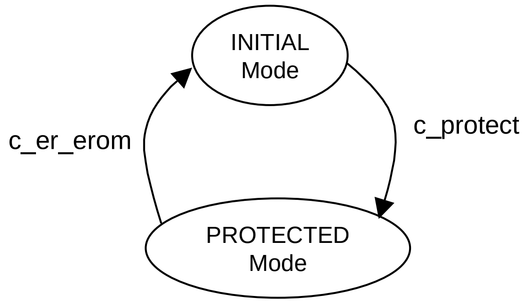


Figure 3 Change of device modes

3.2.1 INITIAL

The monitor interface is fully operational. To protect the E-ROM or EEPROM from readout the device should be set to PROTECTED mode after programming. Leaving the device in the INITIAL mode may cause the device to execute a external break, in case of a low pulse is detected at MSDA which would terminate program execution until a proper Debug Command is being issued or a device reset is forced.

3.2.2 PROTECTED

The monitor interface accepts only the c_er_erom command which switches the device from PROTECTED to INITIAL mode.

3.2.3 TAMPERED

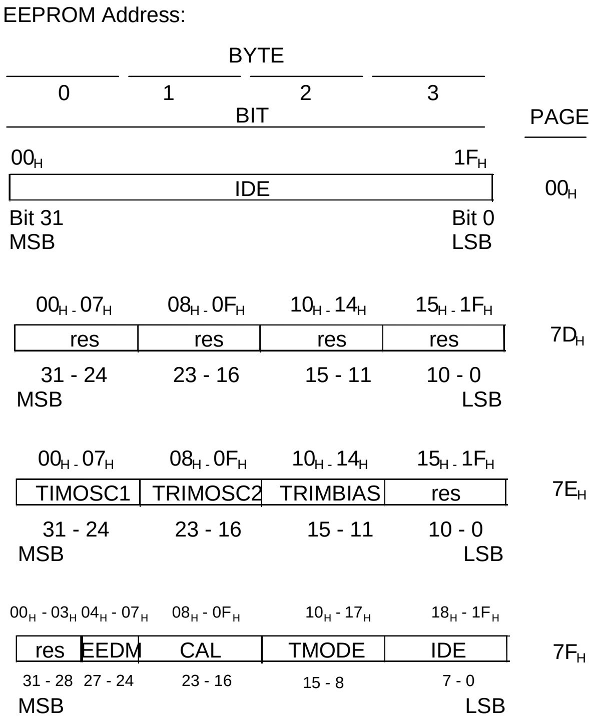
For security reasons the device is set into TAMPERED mode whenever the values of EEDM and EDM deduces an attack.

In this mode the device is irreversible locked. Neither the E-ROM application execution nor the monitor is available.

The device indicates the TAMPERED mode by a single toggle of MSCL (0, 1).

3.2.4 Protected EEPROM System Pages

The device EEPROM features protected system pages, System pages 0, 125, 126 and 127, when the device is in INITIAL mode in order to avoid an unauthorized write access.



EEPROM Address:

BYTE: 0, 1, 2, 3 — BIT — PAGE

00H - 1FH	PAGE
IDE	00H
Bit 31 MSB … Bit 0 LSB	

00_H - 07_H	08_H - $0F_H$	10_H - 14_H	15_H - $1F_H$	PAGE
res	res	res	res	$7D_H$
31 - 24 MSB	23 - 16	15 - 11	10 - 0 LSB	

00_H - 07_H	08_H - $0F_H$	10_H - 14_H	15_H - $1F_H$	PAGE
TIMOSC1	TRIMOSC2	TRIMBIAS	res	$7E_H$
31 - 24 MSB	23 - 16	15 - 11	10 - 0 LSB	

00_H - 03_H	04_H - 07_H	08_H - $0F_H$	10_H - 17_H	18_H - $1F_H$	PAGE
res	EEDM	CAL	TMODE	IDE	$7F_H$
31 - 28 MSB	27 - 24	23 - 16	15 - 8	7 - 0 LSB	

Figure 4 EEPROM System Pages (bank 0)

3.3 Monitor and Download Interface

The PCF7953 supports in-circuit programming of the Program Memory (E-ROM) as well as the configuration EEPROM via the Monitor and Debug Interface. The Monitor and Debug Interface is implemented as a serial interface that utilize the MSDA (monitor serial data = P11) and MSCL (monitor serial clock = P10) pin. The interface may be operated in external and internal manner.

During any communication MSCL (P10) is defined as output. MSDA (P11) alternates between output and input mode as shown in Figure 4, 5, 6 and 7 below. The port lines MSDA and MSCL are configured in “push-pull” fashion when used in output mode. In general the communication is performed LSB first.

During E-ROM download, 32 byte at a time are subject to ERASE/WRITE, which take 4ms and yield a programming time of approx. 700ms for 4 k byte. Download and ERASE/WRITE may be performed with a number of devices connected in parallel, if external operation is selected.

3.3.1 Enter the MONITOR Mode

The MONITOR mode is controlled by the bi-directional pin MSDA and pin MSCL provided by the device. The MONITOR mode is entered by forcing MSDA high during the boot sequence. The RISC is clocked from an on-chip RC oscillator with 125kHz at the beginning of the boot sequence. While entering the MONITOR mode it is possible to switch to external clock (external mode) or to stay with the initial clock source (internal mode). If the debug mode is successfully entered the RISC responds with 55h. In this state the watchdog WT is running. To stop the watchdog a single step command (c_trace) has to be issued within a period of 500 ms. If the device resides in the PROTECTED mode the command c_er_erom must be sent in order to set the device into INITIAL mode. After any switching of the device mode the MONITOR mode must be entered again. In the MONITOR mode a variety of commands are available e.g. for programming the device or for calculation of memory signatures etc. The following figures are illustrating the basic timing sequences.

Each timing symbol has five characters. The first character is always a “t”(=time), the remaining four characters of the symbol (typed in subscript) indicate the signal or the logical status of a signal as follows:

- A = address
- C = clock
- D = input data
- E = external clock
- H = logic level HIGH
- I = Instruction (program memory contents)
- L = logic level LOW
- W = Wake Up
- Q = output data
- t = time
- V = valid
- X= no longer valid logic level
- Z= float

Furthermore

- t_{CLCL} = clock period
- t_{DV} = data valid after the falling edge of MSCL
- t_{wup}= Wake Up time

MSCL (P10) and MSDA (P11) provide internal pull-up resistors which define static HIGH states at these ports unless an external or internal condition pulls them LOW.

3.3.1.1 Enter MONITOR Mode Internal

1. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA (P11) must be set to low externally during the start of the BOOT sequence.
2. In any case the power up of VBat causes a Wake Up and starts device operation.
3. The BOOT sequence starts after a certain Wake Up time (t_{wup}). For more information refer to the corresponding data sheet.
4. In order to start the MONITOR mode MSDA must be set to high externally after t_{WDH}. Otherwise a WARM BOOT is performed.
5. MSDA is pulled down during the BOOT sequence by an internal weak pull-down resistor. MSDA will be evaluated 15 times. In case MSDA is detected high, the MONITOR interface is started. Otherwise a WARM BOOT is performed.
6. Due to the reason that MSCL (P10) and MSDA (P11) may be used as general purpose ports by the application program, MSCL is configured as input (high impedance internally, see dotted line). Since the MSCL port provides an internal pull-up resistor, MSCL is driven high in case of external connected high impedance.

   The target device signals the decision of external/internal mode by a falling edge on MSCL. The Master has to react on this edge. In case MSDA is set to low the internal mode is entered.
7. The target device signals a successful enter into the MONITOR mode by transmitting 55h. For more information refer to chapter 3.4.

Note: After the MONITOR has been started, the Watchdog is running and should be switched off by sending the c_trace command (02h) within approximately 500ms. For more information refer to chapter 3.4 and 4.

The timing of the Enter Debug Mode Internal sequence is depicted in Figure 5. For detailed information of the timings refer to Table 1. For examples refer to chapter 4.

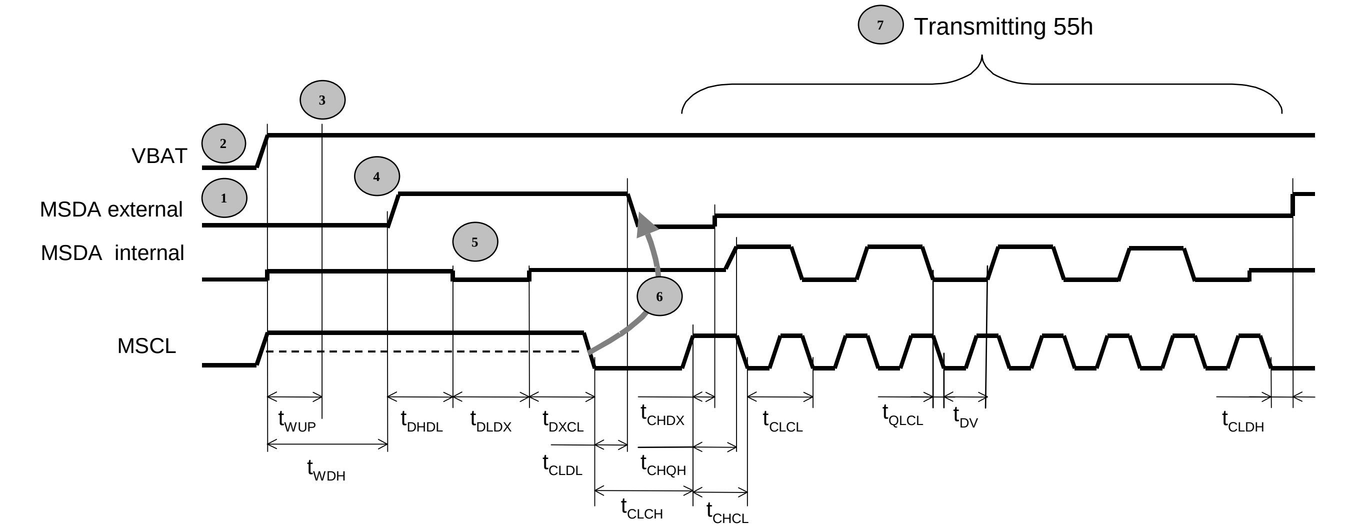


Figure 5 Timing Sequence Enter Monitor Mode Internal

3.3.1.2 Enter MONITOR Mode External

1. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA (P11) must be set to low externally during the start of the BOOT sequence.
2. In any case the power up of VBatt causes a Wake Up and starts device operation.
3. The BOOT sequence starts after a certain Wake Up time (t_{wup}). For more information refer to the corresponding data sheet.
4. In order to start the MONITOR mode MSDA must be set to high externally after t_{WDH}. Otherwise a WARM BOOT is performed.
5. MSDA is pulled down during the BOOT sequence by an internal weak pull-down resistor. MSDA will be evaluated 15 times. In case MSDA is detected high, the MONITOR interface is started. Otherwise a WARM BOOT is performed.
6. Due to the reason that MSCL (P10) and MSDA (P11) may be used as general purpose ports by the application program, MSCL is configured as input (high impedance internally, see dotted line). Since the MSCL port provides an internal pull-up resistor, MSCL is driven high in case of external connected high impedance.

   The target device signals the decision of external/internal mode by a falling edge on MSCL. The Master has to react on this edge. In case MSDA is kept high the external mode is entered. Subsequently, the system clock is derived from an external source connected to port P15. The frequency of the external clock should not exceed 3.5 MHz.
7. The target device signals a successful enter into the MONITOR mode by transmitting 55h. For more information refer to chapter 3.4.

Note: After the MONITOR has been started, the Watchdog is running and should be switched off by sending the c_trace command (02h) within approximately 500ms. For more information refer to chapter 3.4 and 4.

The timing of the Enter Debug Mode Internal sequence is depicted in Figure 5. For detailed information of the timings refer to Table 1.

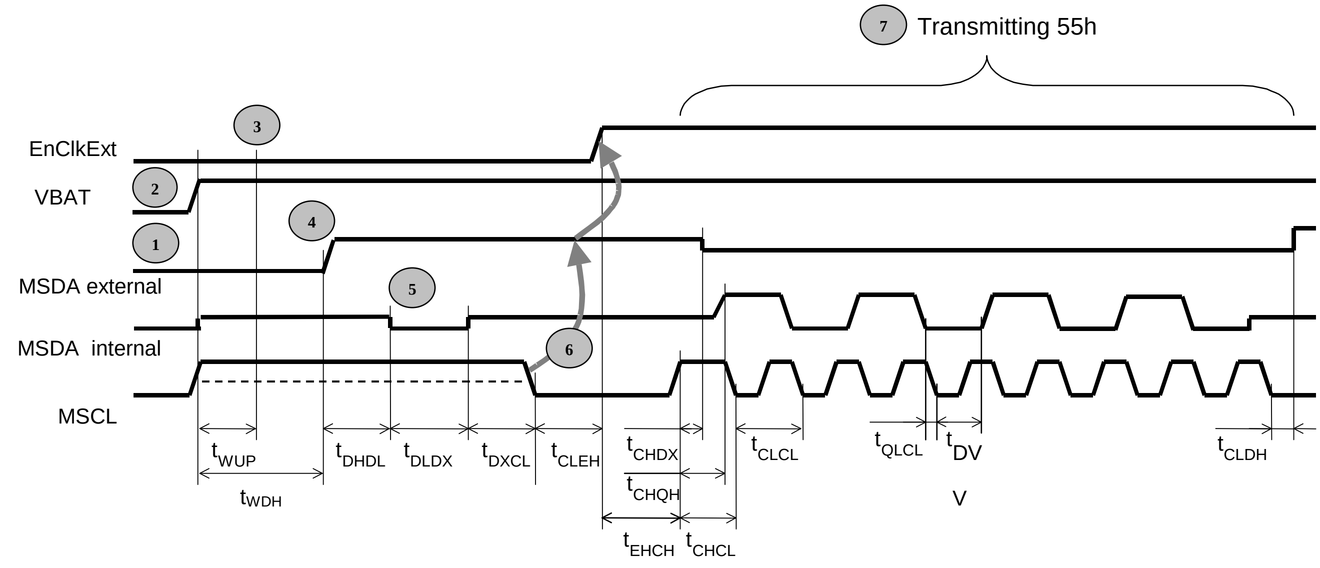


Figure 6 Timing Sequence Enter Monitor Mode External

3.3.1.3 Enter MONITOR Mode with Permanent Power Supply (Example for Internal Mode)

1. In case of a permanent power supply (e. g. battery) the debug operating mode (see chapter 3.4) must be detected before any operation. In case the MONITOR mode is not already entered, the target device may reside in two possible modes:

   - POWER-OFF Mode: Is signalled by MSCL = 0 (see Figure 7). In this case an Enter Debug Mode sequence like it is depicted in Figure 7 must be performed.

   - RUN Mode: Is signalled by MSCL = 1. In order to enter the MONITOR mode perform an external break (low condition on MSDA). For more information refer to chapter 3.4 and 4.

2. A falling edge on a port with Wake Up sense (in this case MSDA (P11)) generates a Wake Up and starts device operation.

   In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA (P11) must be set to low externally during the start of the BOOT sequence.

3. The BOOT sequence starts after a certain Wake Up time (t_{wup}). For more information refer to the corresponding data sheet.

4. In order to start the MONITOR mode MSDA must be set to high externally after t_{WDH}. Otherwise a WARM BOOT is performed.

5. MSDA is pulled down during the BOOT sequence by an internal weak pull-down resistor. MSDA will be evaluated 15 times. In case MSDA is detected high, the MONITOR interface is started. Otherwise a WARM BOOT is performed.

6. Due to the reason that MSCL (P10) and MSDA (P11) may be used as general purpose ports by the application program, MSCL is configured as input (high impedance internally, see dotted line). Since the MSCL port provides an internal pull-up resistor, MSCL is driven high in case of external connected high impedance.

   The target device signals the decision of external/internal mode by a falling edge on MSCL. The Master has to react on this edge. In case MSDA is set to low the internal mode is entered.

7. The target device signals a successful enter into the MONITOR mode by transmitting 55h. For more information refer to chapter 3.4.

Note: After the MONITOR has been started, the Watchdog is running and should be switched off by sending the c_trace command (02h) within approximately 500ms. For more information refer to chapter 3.4 and 4.

The timing of the Enter Debug Mode Internal sequence is depicted in Figure 5.

For detailed information of the timings refer to Table 1.

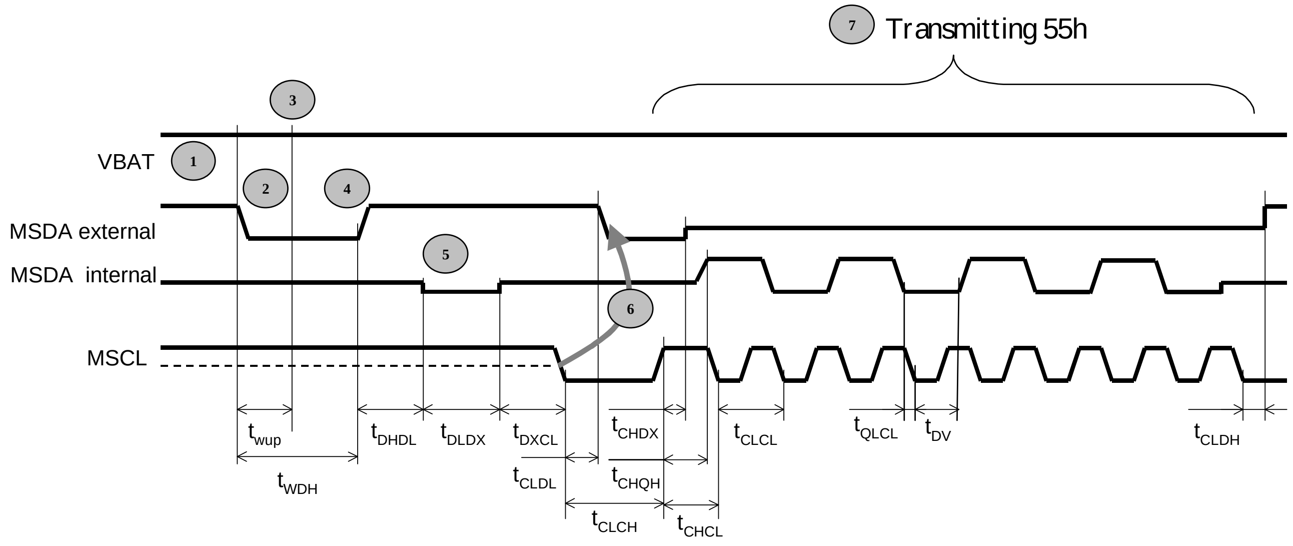


Figure 7 Timing Sequence Enter Monitor Mode with permanent power supply (example for internal mode)

Table 1 Timing values for Enter MONITOR Mode Sequence

Symbol	Min	Typ	Max	Comments	Note
t_{wup}		0,1 ms		Time between Wake Up and BOOT sequence start	
t_{WDH}		0,3 ms		Time between Wake Up and setting of MSDA external high	
t_{DHDL}	31 T_{125kHz} + 98 T_{500kHz}	31 T_{125kHz} + 99 T_{500kHz}	73 T_{125kHz}	Time between setting of MSDA external high and setting MSDA internal weak low	Note 1, 7,8
t_{DLDX}	62 T_{500kHz}	62 T_{500kHz}	62 T_{125kHz}	Time between setting of MSDA internal to weak low and release of MSDA internal	Note 1,2
t_{DXCL}	1 T_{500kHz}	1 T_{500kHz}	1 T_{125kHz}	Time between release of MSDA internal and setting of MSCL to low	Note2,3
t_{CLDL}	- 1 T_{500kHz}	0 T_{500kHz}	2 T_{500kHz}	Time between falling edge of MSCL and setting of MSDA external to low	Note 3, 4
t_{CLCH}	10 T_{500kHz}	10 T_{500kHz}	10 T_{125kHz}	Time between falling edge of MSCL and rising edge of MSCL	Note 5,6
t_{CLEH}	3 T_{500kHz}	3 T_{500kHz}	3 T_{125kHz}	Time between falling edge of MSCL and choose of external clock	Note 3, 9
t_{EHCH}	7 T_{500kHz}	7 T_{500kHz}	7 T_{125kHz}	Time between choose of external clock start of transmission 55h	Note 9
t_{CLCH}	10 T_{500kHz}	10 T_{500kHz}	10 T_{125kHz}	Time between falling edge of MSCL and rising edge of MSCL	Note 5
t_{CHQH}	5 T_{500kHz}	5 T_{500kHz}	5 T_{125kHz}	Time between first rising edge of MSCL and a valid data on MSDA internal	Note 5
t_{CHDX}	- 1 T_{500kHz}	0 T_{500kHz}	2 T_{500kHz}	Time between first rising edge of MSCL and setting of MSDA external to tristate	Note 5
t_{CHCL}	6 T_{500kHz}	6 T_{500kHz}	6 T_{125kHz}	Time of the first high pulse of MSCL	
t_{CLCL}	6 T_{500kHz}	6 T_{500kHz}	6 T_{125kHz}	Time of one MSCL cycle during communication	
t_{QLCL}	1 T_{500kHz}	1 T_{500kHz}	1 T_{125kHz}	Time between a valid data on MSDA internal and falling edge of MSCL	
t_{DV}	4 T_{500kHz}	4 T_{500kHz}	4 T_{125kHz}	Data valid after a falling edge of MSCL	
t_{CLDH}	-2 T_{500kHz}	0 T_{500kHz}	2 T_{500kHz}	Time between last falling edge of MSCL and release of MSDA external to tristate	Note 10,11

Note

1. The setting of MSDA internal to weak low indicates the start of the decision whether the device starts the monitor or the EROM execution
2. The release of MSDA internal indicates the end of the decision whether the device starts the monitor or the EROM execution
3. The falling edge of MSCL indicates the start of the decision whether the device operates in external or in internal mode
4. MSDA external set to low causes the choose of the internal clock
5. The rising edge of MSCL is occuring with the start of transmission 55h
6. The transmission starts not before MSDA external has been set to low.
7. Different device modes cause different runtimes.
8. 74 T_{125kHz} = 592 µs

   31 T_{125kHz} + 98 T_{500kHz} = 444 µs
9. only in external mode
10. The target device signals a successful enter into the MONITOR mode by transmitting 55h
11. Since each data transfer using the monitor interface starts not before MSDA external has been set to low, MSDA should be set to high right after the end of transmission 55h

3.3.2 Transmit and Receive bytes (Device to Master)

3.3.2.1 Transmitted Byte Sequence

In general the communication is performed LSB first. Following steps explain the transmission of one byte from the target device to the external communication device (Master). The communication speed depends on the stetting of the system clock,T_{SYS}.

The MONITOR mode is assumed to be started. During data transmission MSDA is configured as output. The MSCL line is always configured as output.

1. The transmission starts not before MSDA external has been set to low. Thus, the Master starts the communication by setting MSDA external to low.
2. The target device signals the start of the data transfer by setting MSCL to high. To avoid bus conflicts the master has to set MSDA external to tristate right after this first rising edge of MSCL.
3. Thereafter the target device signals valid data on MSDA internal by a falling edge on MSCL. Consequently the master reads the data on MSDA internal right after the falling edge of MSCL (8 falling edges → 8 bits).
4. The target device signals the end of transmission by the 9th falling edge of MSCL. Hence, the master is allowed to release the tristate on MSDA external right after the 9th falling edge. Since the next transmission/reception starts not before MSDA external has been set to low, MSDA should be set to high until the next communication.

The timing of the transmitted byte sequence is depicted in Figure 8. For detailed information about the timings refer to Table 2. For examples refer to chapter 4.

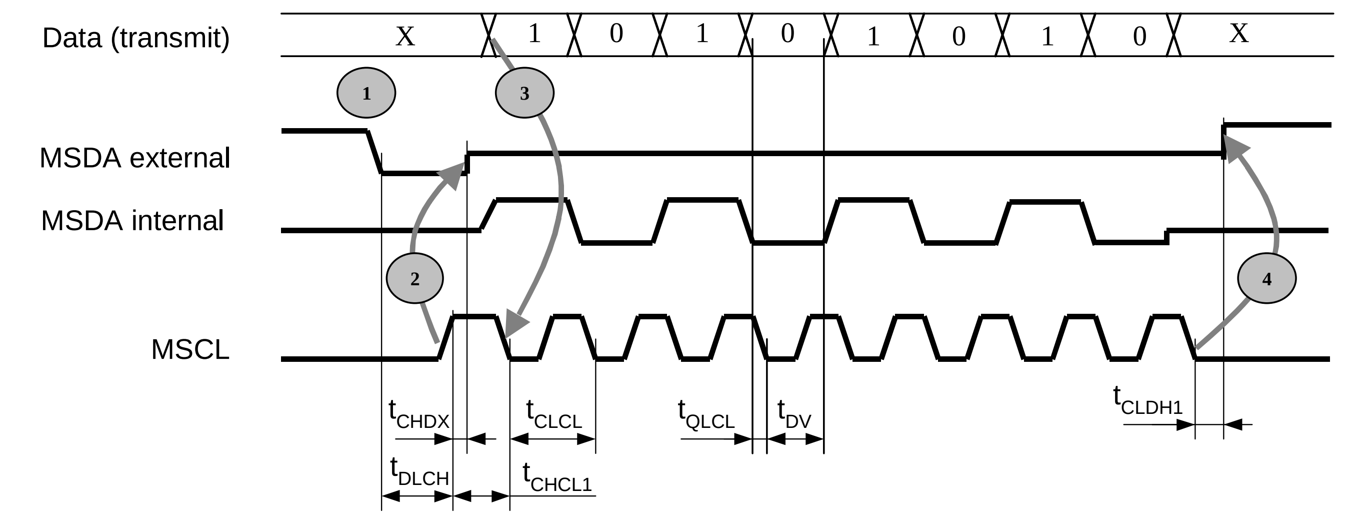


Figure 8 Timing Diagram for Transmitted Byte Sequence (55h)

3.3.2.2 Received Byte Sequence

In general the communication is performed LSB first. Following steps explain the reception of one byte by the target device. The communication speed depends on the stetting of the system clock,T_{SYS}.

The MONITOR mode is assumed to be started. During data reception MSDA s configured as input. The MSCL line is always configured as output.

1. The reception starts not before MSDA external has been set to low. Thus, the Master starts the communication by setting MSDA external to low.
2. The target device signals the start of the data transfer by setting MSCL to high. The master sets MSDA external to the value of the first data bit (LSB) right after the first rising edge of MSCL.
3. Thereafter the target device signals a successful received data bit by a falling edge on MSCL (8 falling edges → 8 bits). Consequently the master is allowed to change the data on MSDA external right after the falling edge of MSCL.
4. The 8th falling edge of MSCL signals the reception of the last bit (MSB) and subsequently the end of the data transfer. Since the next transmission/reception starts not before MSDA external has been set to low, MSDA should be set to high until the next communication (after the 8th falling edge of MSCL).

The timing of the received byte sequence is depicted in Figure 9. For detailed information about the timings refer to Table 2. For examples refer to chapter 4.

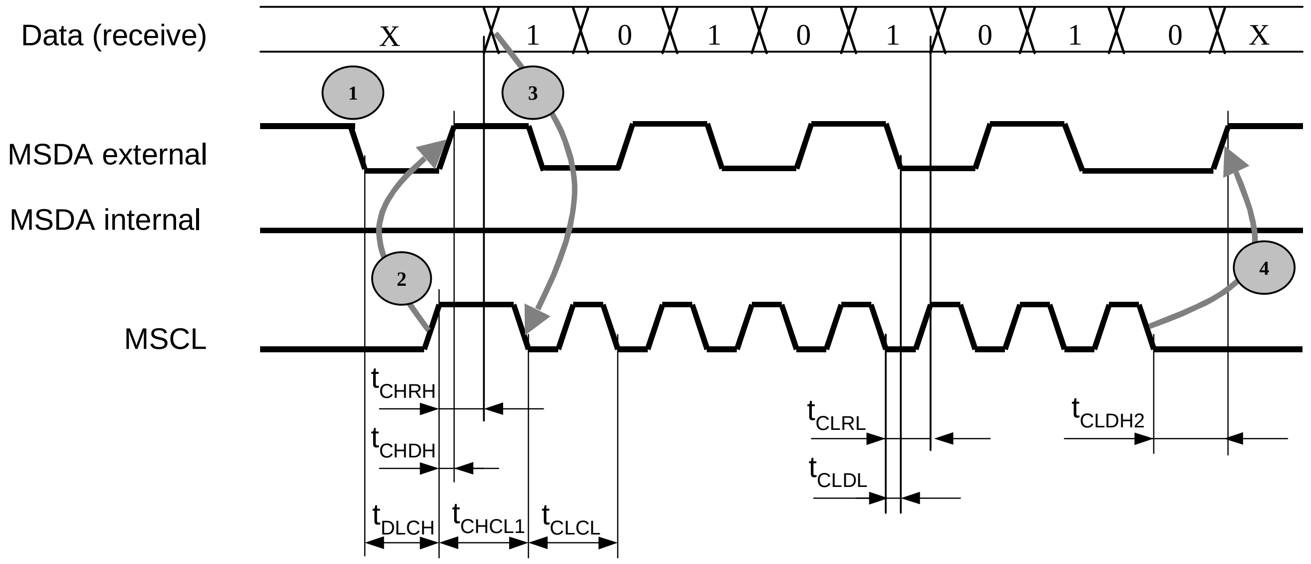


Figure 9 Timing Diagram of the Received Byte Sequence (55h)

Table 2: Timing values for Transmitted and Received byte Sequence

Symbol	Min	Type	max	Comment	Note
t_{DLCH}	2 T_{SYS}		4 T_{SYS}	Time between first falling edge of MSDA external and the first rising edge of MSCL	Note 4
t_{CHDX}	- 1 T_{SYS}		2 T_{SYS}	Time between first rising edge of MSCL and setting of MSDA internal to tristate	Note 2
t_{CHDH}	0		4 T_{SYS}	Time between first rising edge of MSCL and a valid extern data (=MSDA external)	Note 3
t_{CHRH}		4 T_{SYS}		Time between first rising edge of MSCL and a successfully received bit	Note 3
t_{CHCL1}		X T_{SYS}		Time of the first high pulse of MSCL	Note 1
t_{CLCL}		6 T_{SYS}		Time of one MSCL cycle during communication	
t_{QLCL}		1 T_{SYS}		Time between valid data on MSDA internal and falling edge of MSCL	
t_{DV}		4 T_{SYS}		Data valid after falling edge of MSCL	
t_{CLDL}	-1 T_{SYS}		3 T_{SYS}	Time between falling edge of MSCL and setting of new data (=MSDA external)	
t_{CLRL}		3 T_{SYS}		Time between a falling edge of MSCL and a successfully received bit	Note 3
t_{CLDH1}	-2 T_{SYS}		2 T_{SYS}	Time between last falling edge of MSCL and end of transmission	Note 2, 5
t_{CLDH2}	0		4 T_{SYS}	Time between last falling edge of MSCL and end of reception	Note 3, 5

Note

1. X=6 in Transmitted Byte Sequence, X=7 for Received Byte Sequence
2. time refers to Transmitted Byte Sequence
3. time refers to Received Byte Sequence
4. The transmission/reception starts not before MSDA external has been set to low
5. Since the next transmission/reception starts not before MSDA external has been set to low, MSDA should be set to high right after the end of transmission/reception

3.4 Debug Operating Modes

The device resides in one of three debug operating modes, POWER-OFF, MONITOR or RUN mode. Each mode corresponds to a certain set of applicable debug operations. MONITOR and RUN mode can be distinguished externally by tracking MSCL, see Figure 10.

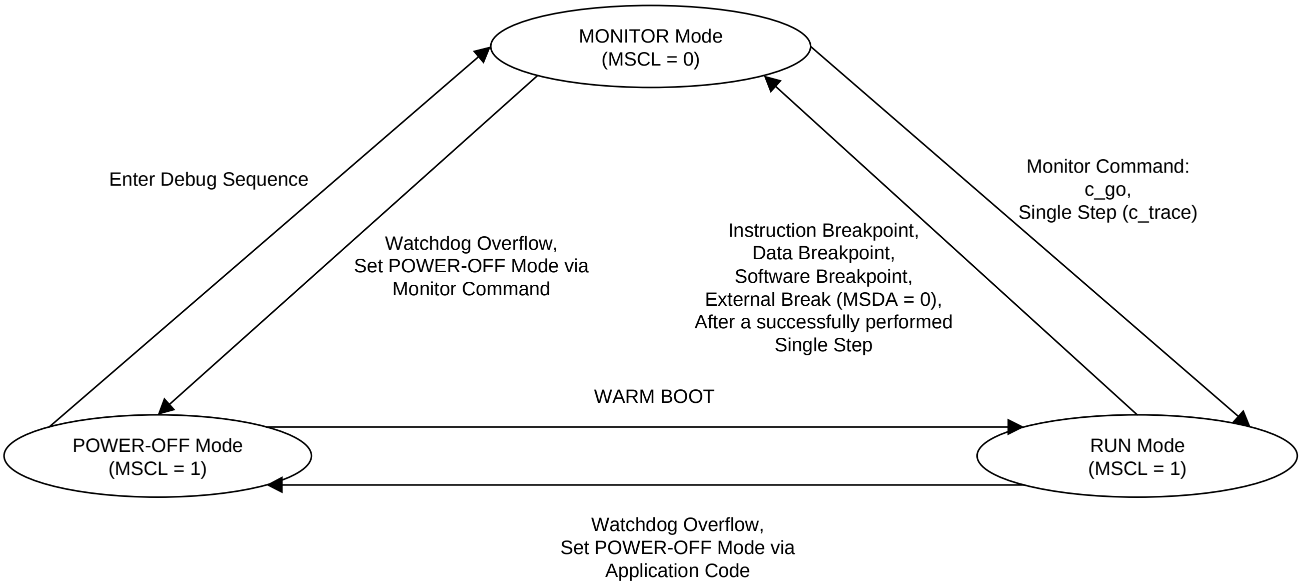


Figure 10 Debug Operating Modes

3.4.1 POWER-OFF Mode

In POWER-OFF mode, the internal supply switch disconnects the device from the battery. The internal device supply voltage (V_{DD}) stays below the power on reset threshold voltage ($V_{POR,FLD}$) and device operation is halted.

The internal pull-up resistor drives MSCL high in POWER-OFF mode.

The POWER-OFF mode is terminated upon either a WARM BOOT or an Enter Debug Sequence is performed. In case a WARM BOOT enables the application code to be executed, the RUN mode takes place. In case of an Enter Debug Sequence, the MONITOR mode is entered.

3.4.2 MONITOR Mode

In MONITOR mode, the device is able to communicate via the debug interface (MSDA, MSCL) with an external communication device. In VIRGIN and INITIAL mode the full set of monitor commands is available. In case the device resides in PROTECTED mode the command c_er_erom is possible only.

The MONITOR mode is signalled by MSCL =0.

The MONITOR mode is terminated and the device is forced into POWER-OFF mode by following events:

- After entering the MONITOR mode, the watchdog is running. To stop the watchdog a single step command (c_trace) has to be issued within a period of 500ms. Otherwise the device is forced into POWER-OFF mode.
- The POWER-OFF mode is entered after a successful execution of the monitor command c_er_erom or c_protect.
- If the device resides in PROTECTED mode all monitor commands except c_er_erom will force the device directly into POWER-OFF mode without executing the command.
- The Special Function Register PCON may be modified by the monitor command c_setdat in a way that the POWER-OFF mode is enforced (for more information refer the PCF7952 datasheet).

The MONITOR mode is terminated and the device is forced into RUN mode by the monitor commands c_go and c_trace (Single Step).

3.4.3 RUN Mode

In RUN mode the application code is executed.

The RUN mode is signalled by MSCL =1.

The RUN mode is terminated and the MONITOR mode is entered by following events:

- Instruction breakpoint
- Data breakpoint
- Software breakpoint (TRAP instruction)
- External break (MSDA is forced to low externally)
- After a successful performed Single Step

Thereafter the program counter (PC) is transmitted. During program execution from ROM (Syscall) the RUN mode cannot be interrupted.

The RUN mode is terminated and the device is forced into POWER-OFF mode by either a watchdog overflow or the Special Function Register PCON is modified by the application code in a way that the POWER-OFF mode is enforced (for more information refer the PCF7953 datasheet).

3.5 Command Set Description

Device operation is controlled by a set of commands. Table 3 gives a comprehensive summary of applicable commands. Command operation and acceptance depends on the actual device mode in which the command is being issued or executed (see chapter 3.2). The general form of a control sequence consists of a command sequence send to the target device and a response sequence received from the target device.

A not valid function code causes a device in:

- INITIAL mode to ignore the current monitor command and read the next command.
- PROTECTED mode to enter the POWER-OFF mode.

Table 3. Monitor Command Set Summary

NAME	FUNCTION CODE	DESCRIPTION	APPLICABLE DEVICE STATE		
			Initial	Protected Tampered	Note
c_go	01h	Set device in to RUN mode, during program execution cannot be interrupted	•	-	
c_trace	02h	Set device in to RUN mode for a single step	•	-	
c_getdat	03h	Read registers, SFR’s and RAM	•	-	
c_setdat	04h	Write registers, SFR’s and RAM	•	-	
c_setpc	05h	Set PC to specified location	•	-	
c_reset	06h	Execute device reset	•	-	
c_setbrk	07h	Set breakpoint	•	-	
c_er_erom	08h	Erase E-ROM and set device into INITIAL mode	•	•	
c_wr_erom	09h	Write E-ROM page (32 bytes)	•	-	
c_wr_eeprom	0Ah	Write EEPROM	•	-	
c_wr_erom_b	0Bh	Write E-ROM byte	•	-	
c_sig_rom	0Ch	Calculate ROM signature (system ROM only)	•	-	
c_sig_erom	0Dh	Calculate E-ROM signature	•	•	
c_ee_dump	0Eh	EEPROM dump	•	-	
c_er_dump	0Fh	E-ROM dump	•	-	
c_sig_ee	11h	Calculate EEPROM signature	•	-	
c_protect	12h	Set device into PROTECTED mode	•	-	
c_prog_config	14h	Program byte 2 and 3 of page 127 (for details refer to the PCF7953 datasheet)	•	-	
c_wr_erom64	18h	Write E-ROM page (64 bytes)	•	-	
c_sig_xrom	1Eh – 00h	Calculate ROM signature (USER ROM only)	•	•	
c_sig_ee_norm	1Eh – 20h	Calculate normalised EEPROM signature	•	• / –	1

Note

1. c_sig_ee_norm in protected mode is only valid for PCF7953 products based on mask version V0B (e.g. PCF7953ATT/M**0B**C05, PCF7953PTT/C**0B**C1100, PCF7953PTJ/C**0B**C1200..)

C_GO

Function code: 01h

Description: The command c_go sets the device into RUN mode which is signalled by MSCL = 1. After an instruction breakpoint, a data breakpoint, a software breakpoint (TRAP instruction) or an external break (MSDA is forced to low externally) the program counter (PC) is transmitted.

Remark: The Run mode is signalled by MSCL = 1.

During program execution from ROM (Syscall) the RUN mode cannot be interrupted.

After an instruction breakpoint, a data breakpoint, a software breakpoint (TRAP instruction) or an external break (MSDA is forced to low externally) PC_{LOW} and PC_{HIGH} are transmitted.

If no external break is wanted during the RUN mode, MSDA has to be set to high externally while MSCL=1 (RUM mode).

Parameter: none

Response: PC_{LOW} (1 byte)

PC_{HIGH} (1 byte)

Applicable

Device State: INITIAL

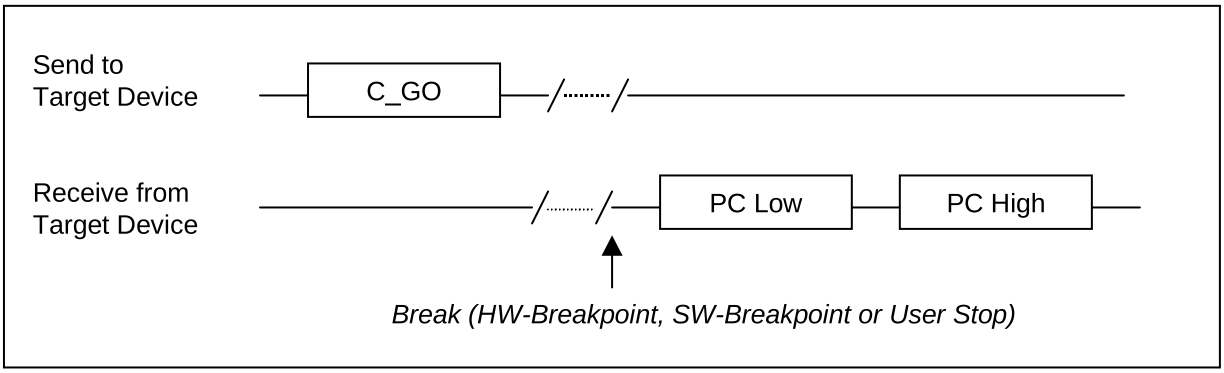


Figure 11 Command C_GO

C_TRACE

Function code: 02h

Description: The command c_trace sets the device into RUN mode for a single instruction which is signalled by MSCL = 1.

In case c_trace is executed within 500ms period after the MONITOR mode has been entered, the watchdog is stopped instead of performing a Single Step.

Remark: The Run mode is signalled by MSCL = 1.

During program execution from ROM (Syscall) the RUN mode cannot be interrupted.

After the execution of one instruction PC_{LOW} and PC_{HIGH} are transmitted.

In case c_trace is executed within 500ms period after the monitor has been entered, the watchdog is stopped.

Since an external break (MSDA is forced to low externally) during a Single Step can cause unwanted behaviour, MSDA must be set to high externally while MSCL=1 (RUN mode).

Due to processor design specific reasons a Single Step over a 'setb IDLE' instruction is not working in case the wake up source is the timer. In this case the target device will not be waked up from IDLE mode anymore.

Parameter: none

Response: PC_{LOW} (1byte)

PC_{HIGH} (1 byte)

Applicable

Device State: INITIAL

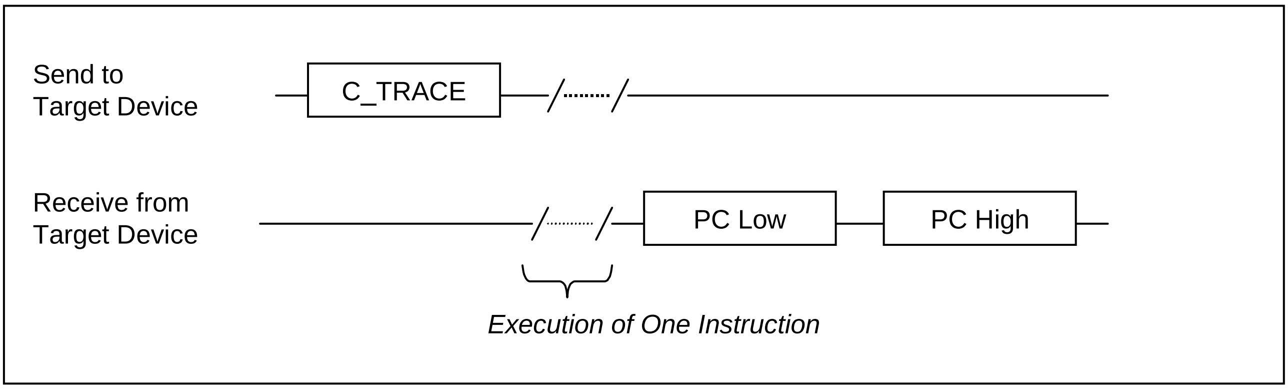


Figure 12 Command C_TRACE

C_GETDAT

Function code: 03h

Description: By executing the command c_getdat all registers including the Special Function Registers SFR and the RAM can be read.

After issuing this command each address will send a one data byte response (data[address1], data[address2], ...). By sending 0Ch (@R0+) the communication is stopped. Since the address 0Ch (@R0+) provides an auto-increment functionality, a read operation provided on this location would cause debug inconsistencies. Thus, this address is chosen to stop the communication.

Remark: Register addresses used for indirect addressing with auto - in/decrement are modified by any access. The same is valid for Special Function Register EEDAT.

Parameter: address (1 byte)

Response: data (1 byte)

Applicable

Device State: INITIAL

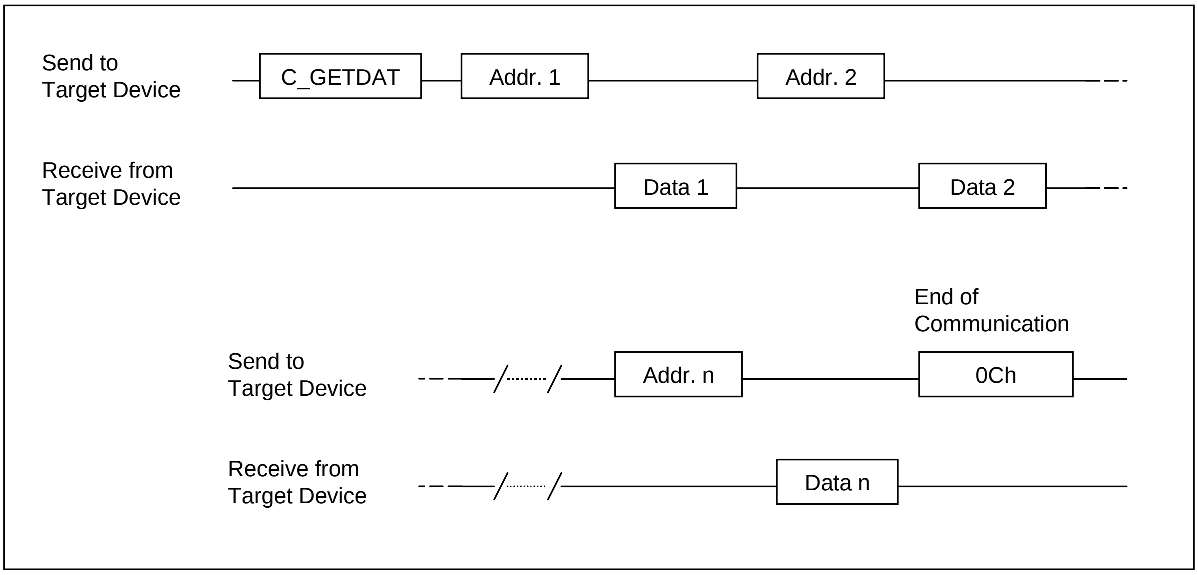


Figure 13 Command C_GETDAT

C_SETDAT

Function code: 04h

Description: By executing the command c_setdat all registers including the Special Function Registers SFR and the RAM can be written. Write Access is controlled by the Device mode.

Remark: In VIRGIN mode write access is unrestricted.
In INITIAL mode write access is restricted for addresses between 37_{hex} - $3F_{hex}$ (System SFRs).
Register addresses used for indirect addressing with auto - in/decrement are modified by any access. The same is valid for Special Function Register EEDAT.

Parameter: address (1 byte)

data (1 byte)

Response: none

Applicable

Device State: INITIAL

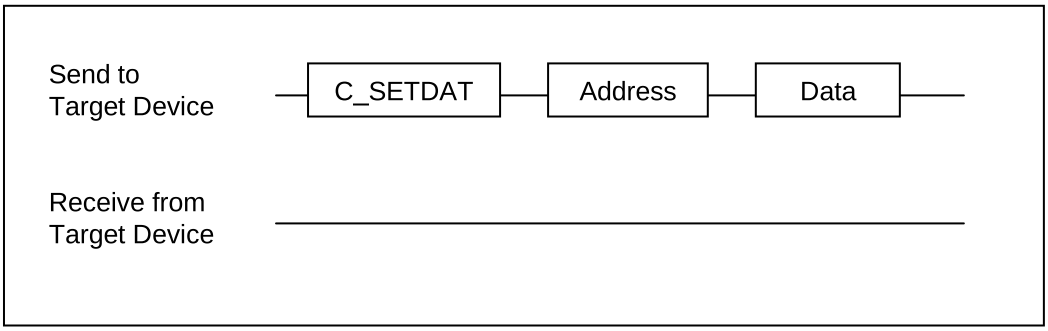


Figure 14 Command C_SETDAT

C_SETPC

Function code: 05h

Description: Sets the program counter (PC) to a specified location.

Remark: none

Parameter: PC_{LOW} (1byte)

PC_{HIGH} (1 byte)

Response: none

Applicable

Device State: INITIAL

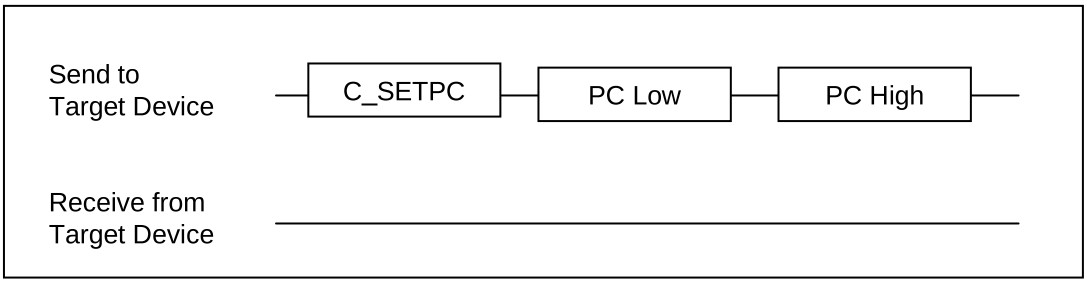


Figure 15 Command C_SETPC

C_RESET

Function code: 06h

Description: Performs a device reset.

Remark: none

Parameter: none

Response: none

Applicable Device State: INITIAL

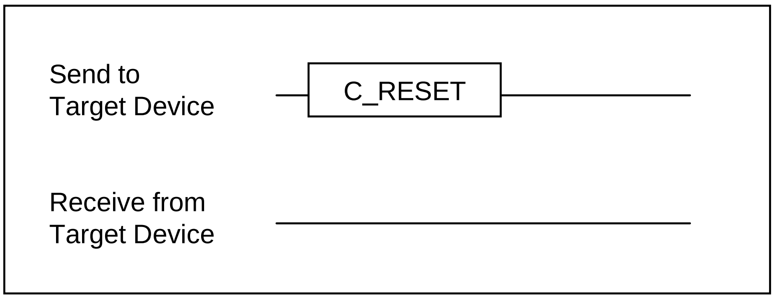


Figure 16 Command C_RESET

C_SETBRK

Function code: 07h

Description: Sets an instruction breakpoint, a data breakpoint or a trigger.

Remark: The address is specified by a 13-bit address.

The first parameter (BRKPL) specifies the lower 8 bits, the second parameter (BRKPH) the higher 5 bits of the address.

The type of the breakpoint is chosen by the higher 3 bits of BRKPH.

BRK: If set a instruction breakpoint is enabled.

TRG: If set a trigger point is enabled.

DBP: If set a data breakpoint is enabled.

bit 7							bit 0	
BP7	BP6	BP5	BP4	BP3	BP2	BP1	BP0	BRKPL

bit 7							bit 0	
BRK	TRG	DBP	BP4	BP3	BP2	BP1	BP0	BRKPH

Since the trigger pulses are transmitted via the two wire serial debug interface of the target device, a debug communication while transmitting trigger pulses may cause unexpected behaviour. Thus, trigger functionality should not be performed in combination with debug operations (instruction breakpoint, data breakpoint, SW breakpoint, …).

Parameter: BRKPL (1 byte)

BRKPH (1 byte)

Response: none

Applicable Device State: INITIAL

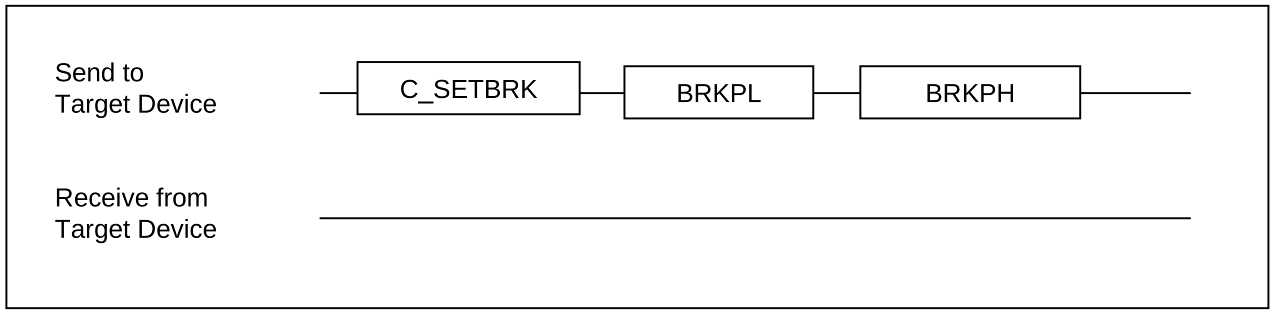


Figure 17 Command C_SETBRK

C_ER_EROM

Function code: 08h

Description: Erases the whole E-ROM/EEPROM, except the system pages, and sets the device into INITIAL mode.

Remark: After a successful pass the device is forced into POWER-OFF mode.

For security reasons the device will be set into TAMPERED mode if this command is interrupted.

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: Byte 0-15:

55h, 45h, E8h, 92h, D6h, B1h, 62h, 59h, FCh, 8Ah, C8h, F2h, D6h, E1h, 4Ah, 35h

Response: 88h

Special Function Register EECON

Applicable

Device State: INITIAL, PROTECTED, TAMPERED

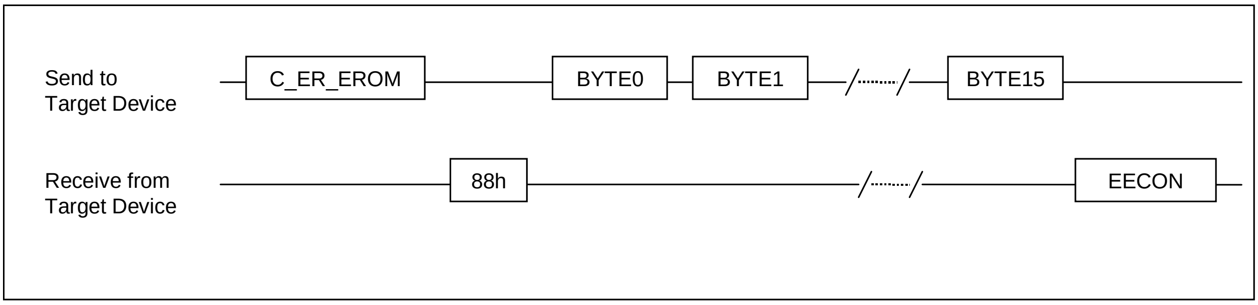


Figure 18 Command ER_EROM

C_WR_EROM

Function code: 09h

Description: Programs one page of E-ROM (32 byte).

Remark: The content of Special Function Register BSCON, EEADR and EECON is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.
Most significant address bit of the page address is ignored.
The last byte of E-ROM (1FFFh) is write restricted.

bit 7							bit 0	
PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	Page Address

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: Page Address (1 byte)

Data [Addr.+0] (1 byte)

Data [Addr.+1] (1 byte)

…

Data [Addr.+31] (1 byte)

Response: Special Function Register EECON

Applicable

Device State: INITIAL

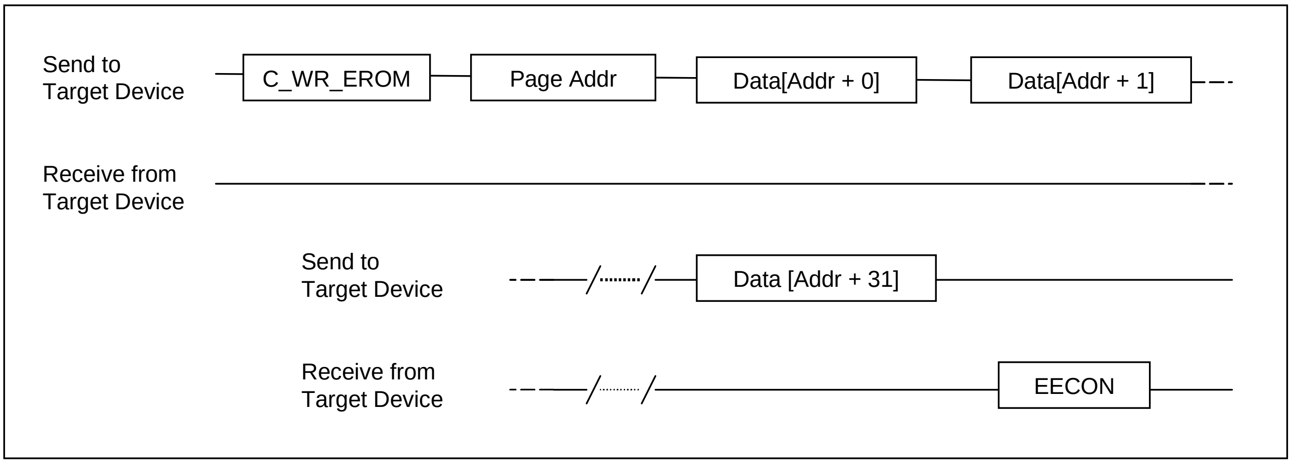


Figure 19 Command C_WR_EROM

C_WR_EEPROM

Function code: 0Ah

Description: Programs one page of EEPROM.

Remark: The content of Special Function Register BSCON, EEADR and EECON is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.
Most significant address bit is ignored.
In VIRGIN mode write access is unrestricted.
In INITIAL mode write access is restricted for page 0, 126 and 127.
BS (bank select) distinguishes between EEPROM bank 0 and 1.

bit 7							bit 0	
BS	PG6	PG5	PG4	PG3	PG2	PG1	PG0	Page Address

Due to historical reasons the data bytes must be sent bit reversed order (1011000b → 0001101b).

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: Page Address (1 byte)
Data [Addr.+0] (1 byte)
Data [Addr.+1] (1 byte)
Data [Addr.+2] (1 byte)
Data [Addr.+3] (1 byte)

Response: Special Function Register EECON if the command is successful.
FFh if the command fails (page is write protected).

Applicable

Device State: INITIAL

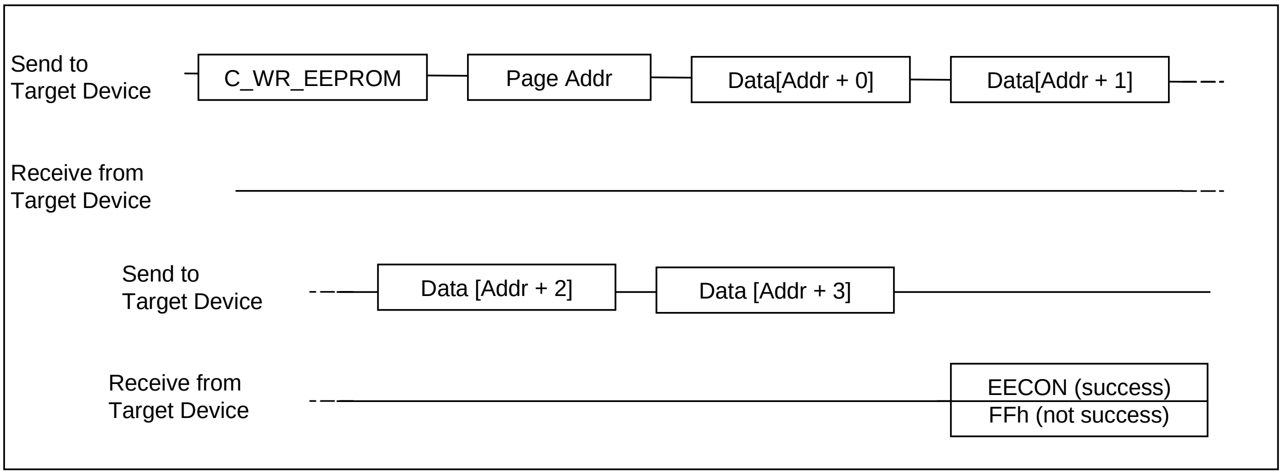


Figure 20 Command C_WR_EEPROM

C_WR_EROM_B

Function code: 0Bh

Description: Programs one byte of E-ROM.

Remark: The content of BSCON, EEADR and EECON is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.

The last byte of E-ROM (1FFFh) is write restricted.

Bits marked ‘X’ are ignored.

bit 7							bit 0	
Addr 7	Addr 6	Addr 5	Addr 4	Addr 3	Addr 2	Addr 1	Addr 0	$\text{Address}_{\text{LOW}}$

bit 7							bit 0	
X	X	X	Addr 12	Addr 11	Addr 10	Addr 9	Addr 8	$\text{Address}_{\text{HIGH}}$

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: $\text{Address}_{\text{LOW}}$ (1 byte)

$\text{Address}_{\text{HIGH}}$ (1 byte)

Data (1 byte)

Response: Special Function Register EECON

Applicable

Device State: INITIAL

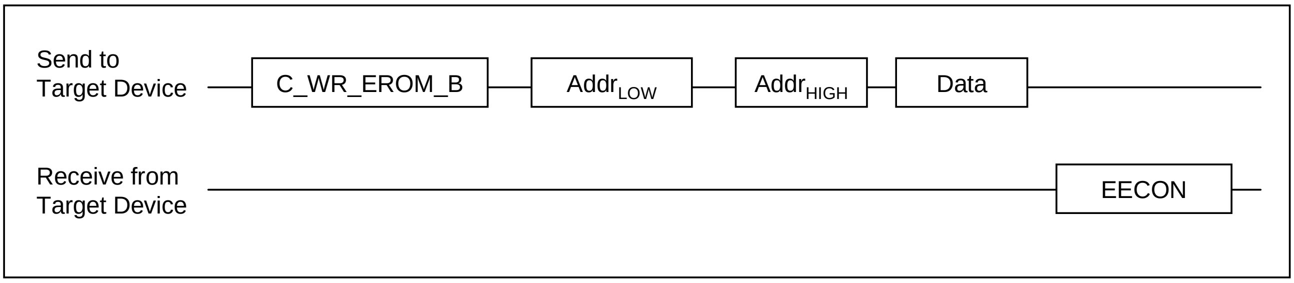


Figure 21 Command C_WR_EROM_B

C_SIG_ROM

Function code: 0Ch

Description: Calculates the ROM signature of SYSTEM ROM

Remark: The content of all registers and RAM location 40h is destroyed.

Parameter: none

Response: Signature 0 (1 byte)

Signature 1 (1 byte)

Signature 2 (1 byte)

Applicable

Device State: INITIAL

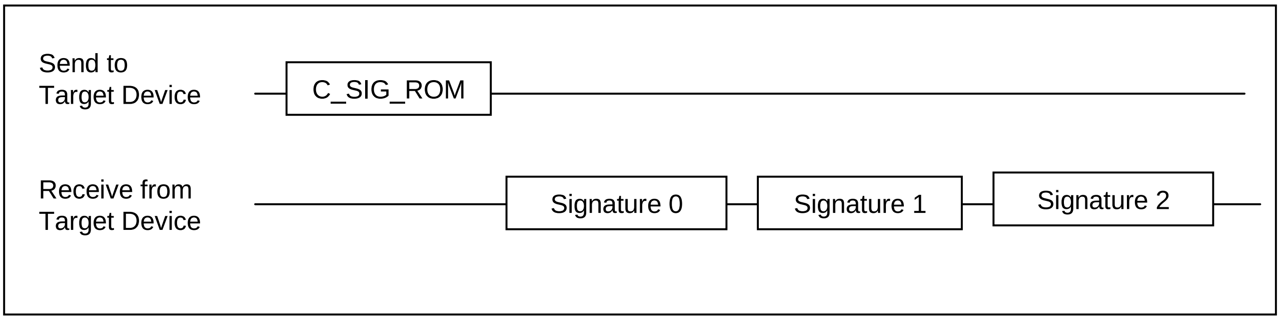


Figure 22 Command C_SIG_ROM

C_SIG_EROM

Function code: 0Dh

Description: Calculates the E-ROM signature

Remark: The content of all registers, BSCON, EECON, EEADR and RAM location 40h is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.

Parameter: none

Response: Signature 0 (1 byte)

Signature 1 (1 byte)

Signature 2 (1 byte)

Applicable

Device State: INITIAL, PROTECED, TAMPERED

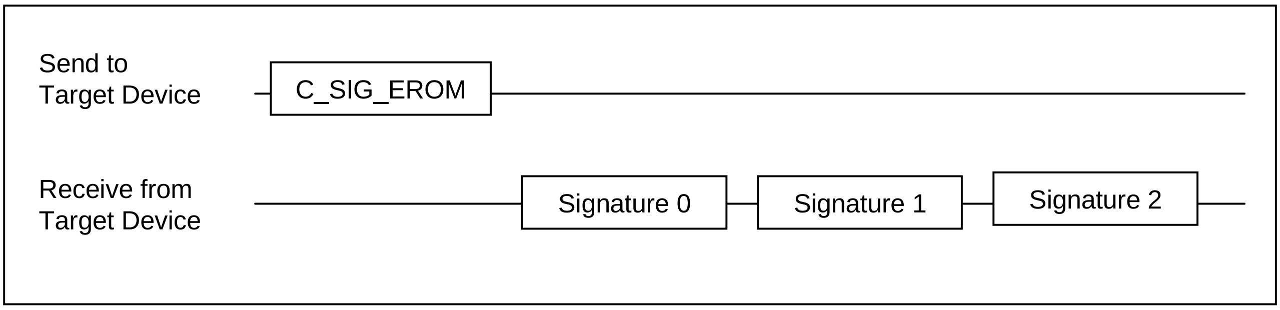


Figure 23 Command C_SIG_EROM

C_EE_DUMP

Function code: 0Fh

Description: Reads the whole EEPROM content (1024 bytes, 1st 512 byte from EEPROM bank 0, 2nd 512 byte from EEPROM bank 1)

Remark: The content of all registers, BSCON, EECON and EEADR is destroyed. To avoid context inconsistency, these registers should be stored and restored.

Due to historical reasons the data bytes are sent in bit reversed order (1011000b → 0001101b).

Parameter: none

Response: EEPROM byte 0 (1 byte)

EEPROM byte 1 (1 byte)

…

EEPROM byte 1023 (1 byte)

Applicable Device State: INITIAL

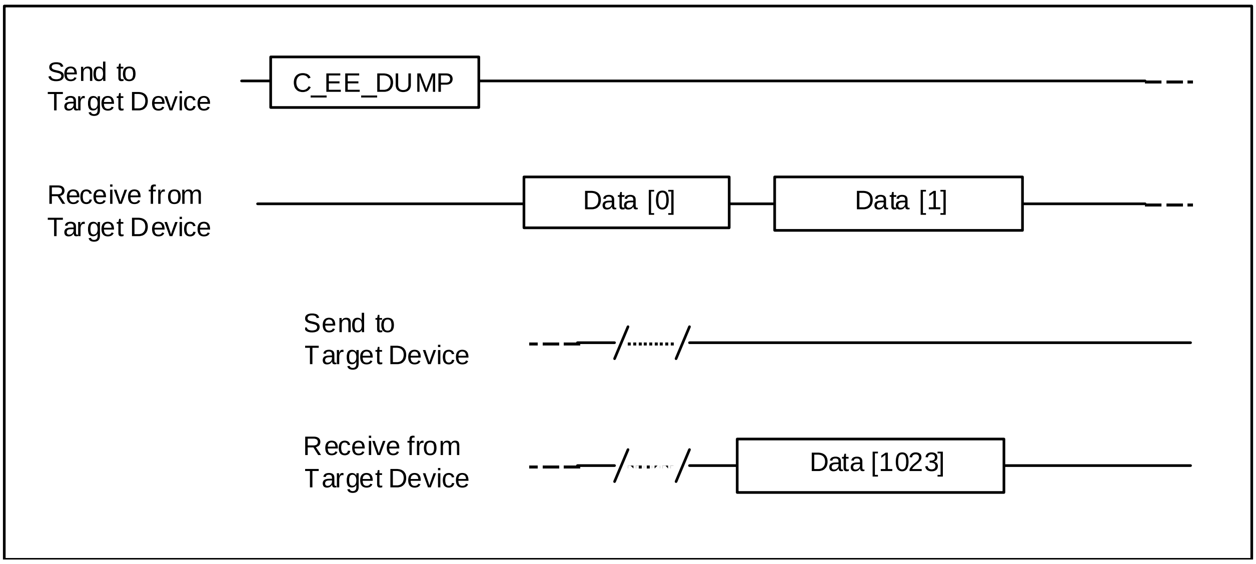


Figure 24 Command C_EE_DUMP

C_ER_DUMP

Function code: 0Fh

Description: Reads the whole E-ROM content (8192 bytes)

Remark: The content of all registers, BSCON, EECON and EEADR is destroyed. To avoid context inconsistency, these registers should be stored and restored.

Parameter: none

Response: E-ROM byte 0 (1 byte)

E-ROM byte 1 (1 byte)

…

E-ROM byte 8191 (1 byte)

Applicable

Device State: INITIAL

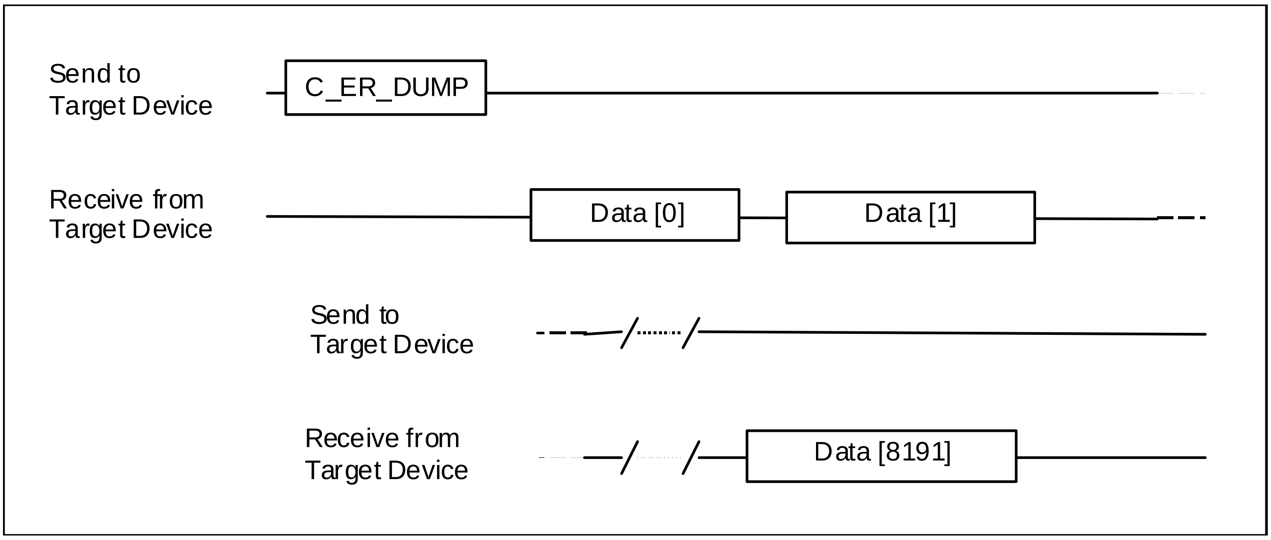


Figure 25 Command C_ER_DUMP

C_SIG_EE

Function code: 11h

Description: Calculates the EEPROM signature (3 bytes)

Remark: The content of all registers, BSCON, EECON, EEADR and RAM location 40h is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.

Parameter: none

Response: Signature 0 (1 byte)

Signature 1 (1 byte)

Signature 2 (1 byte)

Applicable

Device State: INITIAL

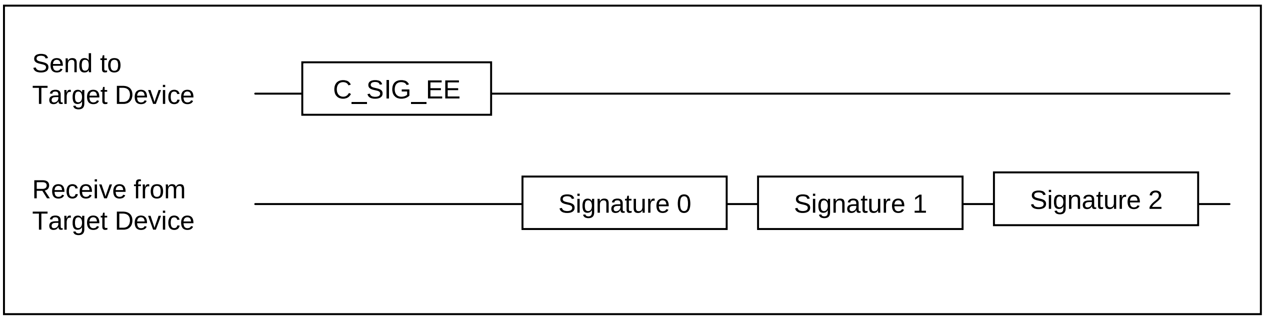


Figure 26 Command C_SIG_EE

C_PROTECT

Function code: 12h

Description: Sets the device into PROTECTED mode.

Remark: After a successful pass the device is forced into POWER-OFF mode.

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: none

Response: EECON

Applicable Device State: INITIAL

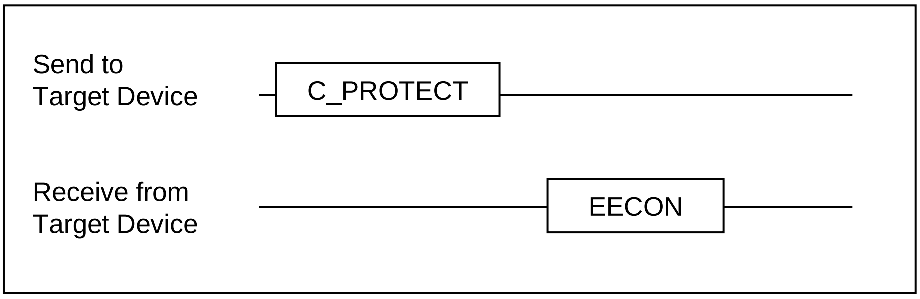


Figure 27 Command C_PROTECT

C_PROG_CONFIG

Function code: 14h

Description: Programs byte 2 and byte 3 of page 127 (for details refer to the PCF7953 datasheet).

Remark: In general page 127 is write protected in INITIAL mode. The command gives the possibility to change the content of byte 2 and byte 3 even in INITIAL mode.

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: EEPROM page 127 byte 2 (1byte)

EEPROM page 127 byte 3 (1byte)

Response: EECON

Applicable

Device State: INITIAL

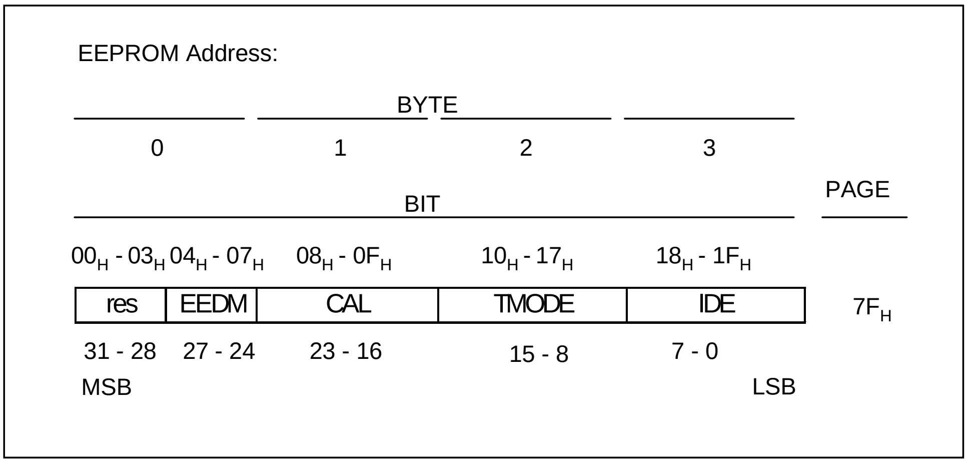


Figure 28 EEPROM address organization

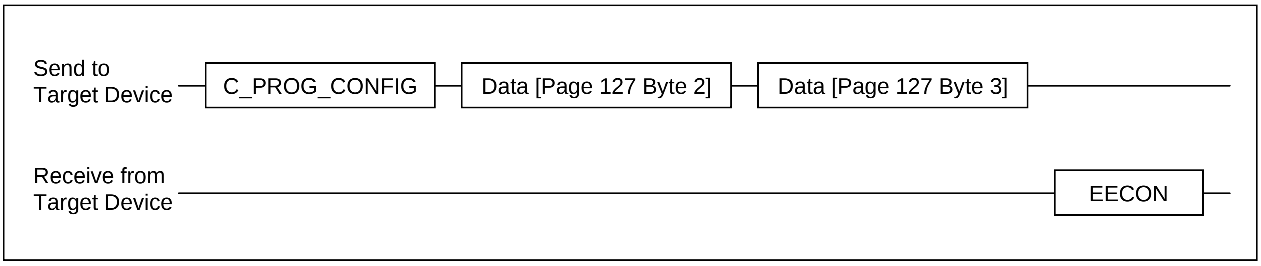


Figure 29 Command C_PROTECT_EE

C_WR_EROM64

Function code: 18h

Description: Programs two pages of E-ROM (64 byte).

Remark: The content of Special Function Register EEADR and EECON is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.
Most significant address bit of the page address is ignored.
The last byte of E-ROM (1FFFh) is write restricted.

bit 7							bit 0	
PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	Page Address

The target device responses the Special Function Register EECON. This register contains the PERR bit which is set, if an EROM/EEPROM write access was not successful (desired high voltage level was not reached). In order to ensure proper command execution the PERR bit should be checked.

Parameter: Page Address (1 byte)

Data [Addr.+0] (1 byte)

Data [Addr.+1] (1 byte)

…

Data [Addr.+63] (1 byte)

Response: Special Function Register EECON

Applicable

Device State: INITIAL

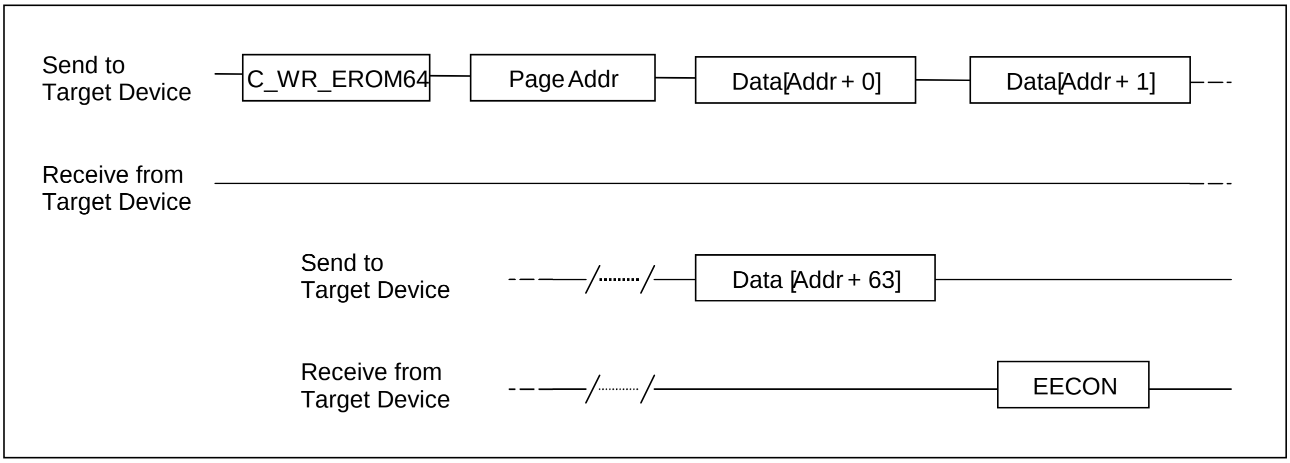


Figure 30 Command C_WR_EROM64

C_SIG_XROM

Function code:1E - 00

Description: Calculates the ROM signature of shared USER/SYSTEM ROM

Remark: The content of all registers and RAM location 40h is destroyed.

Parameter: none

Response: Signature 0 (1 byte)

Signature 1 (1 byte)

Signature 2 (1 byte)

Applicable

Device State: INITIAL, PROTECED, TAMPERED

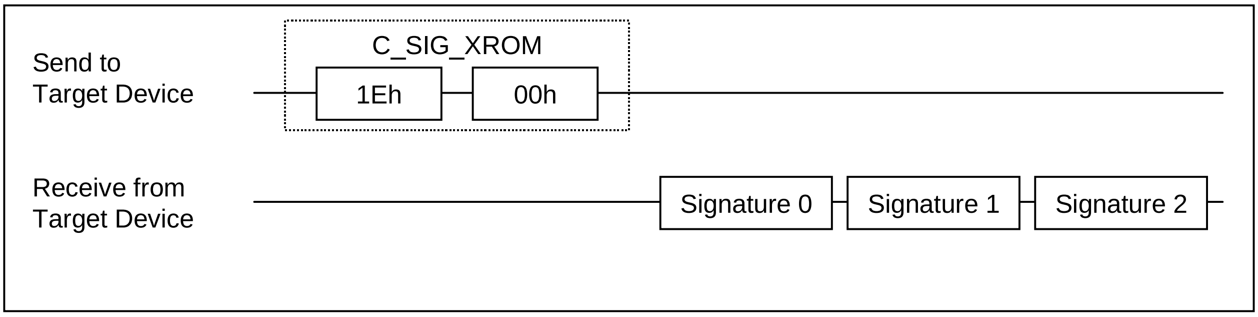


Figure 31 Command C_SIG_XROM

C_SIG_EE_NORM

Function code:1E - 20

Description: Calculates the normalised EEPROM signature (3 bytes)

Remark: The content of EEPROM pages 0 (serial number), 125, 126 and the lower two bytes of page 127 (trimming information) of EEPROM bank 0 is replaced by zeros.

The content of all registers, BSCON, EECON, EEADR and RAM location 40h is destroyed. To avoid context inconsistency, these SFRs should be stored and restored.

Parameter: none

Response: Signature 0 (1 byte)

Signature 1 (1 byte)

Signature 2 (1 byte)

Applicable

Device State: INITIAL, PROTECED, TAMPERED

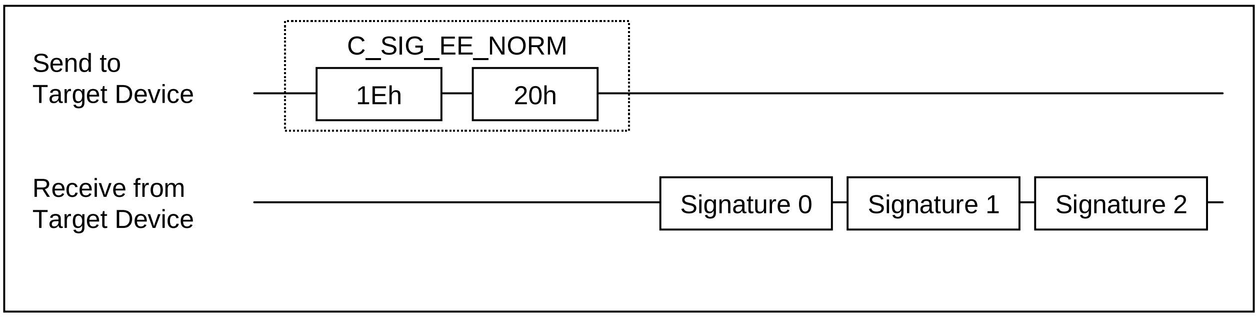


Figure 32 Command C_SIG_EE_NORM

4 APPLICATION NOTES

This section provides additional information concerning device application and highlights differences in the device operation, when compared with the existing product family.

4.1.1 Enter MONITOR Mode Internal Mode and Disable Watchdog

Following example shows how to enter the MONITOR mode in internal mode. Additional the Watchdog is disabled by sending the c_trace command right after the MONITOR mode has been entered. For detailed information about the EnterDebug sequence and the Transmitted/Received Byte sequence refer to chapter 3.3.1 and 3.3.2.

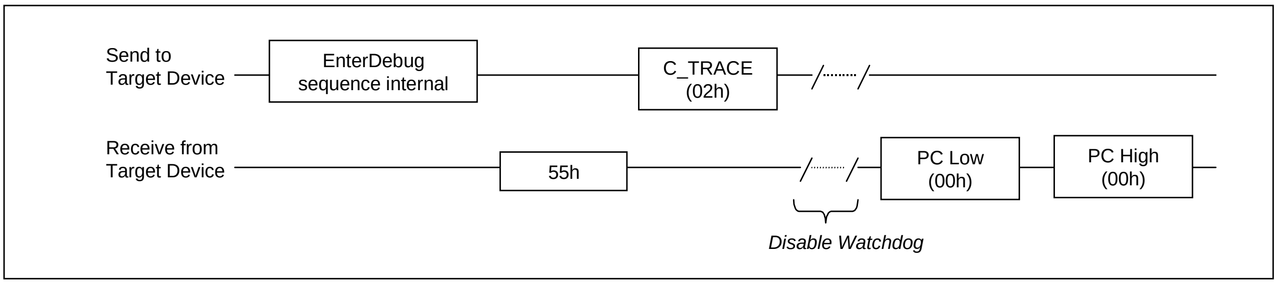


Figure 33 Enter Monitor Internal mode + disable Watchdog

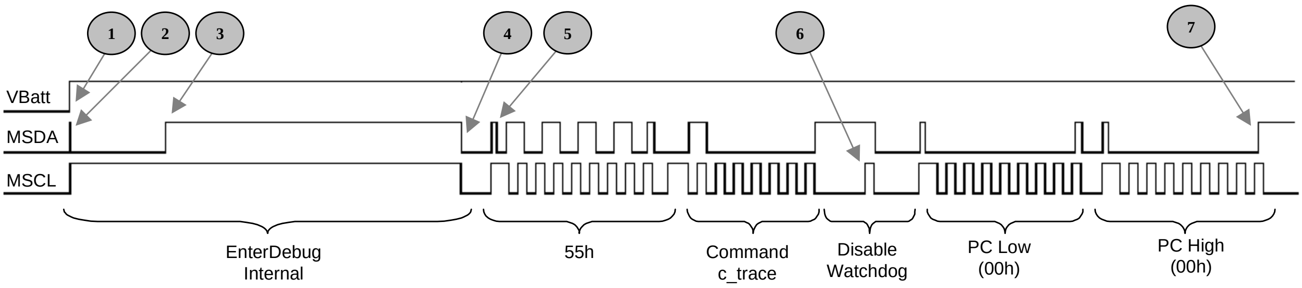


Figure 34 Oscillograph of Enter Monitor Internal mode + disable Watchdog

1. In any case the power up of VBat causes a Wake Up and starts device operation.
2. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA must be set to low externally during the start of the BOOT sequence.
3. MSDA external has to be set high in order to start the MONITOR mode, otherwise a WARM BOOT is performed.
4. The target device signals the decision of external/internal mode by a falling edge on MSCL. In order to enter the internal mode MSDA has to be set to low right after the falling edge of MSCL.
5. The target device signals the start of the data transfer by setting MSCL to high. In order to read data the master sets MSDA to tristate externally right after the first rising edge of MSCL. At this time MSDA is set to tristate internally either. Thus, the external pull-up resistor drives MSDA to high. Thereafter the target device configures MSDA as output. This causes a low condition on MSDA. Subsequently the target device sets MSDA to the desired value and signals valid data by a falling edge on MSCL.
6. After the c_trace command has been sent the target device is set to RUN mode during the time the Watchdog is switched off. Since an external break (low condition on MSDA) during the c_trace command can cause unwanted behaviour, MSDA must be set to high externally while MSCL=1 (RUN mode).
7. Since the next transmission/reception starts not before MSDA external has been set to low, MSDA should be kept to high until the next communication.

4.1.2 Enter MONITOR Mode Internal Mode with Permanent Power Supply and Disable Watchdoc

Following example shows how to enter the MONITOR mode in internal mode in case of a permanent power supply. Additional the Watchdog is disabled by sending the c_trace command right after the MONITOR mode has been entered. For detailed information about the EnterDebug sequence and the Transmitted/Received Byte sequence refer to chapter 3.3.1 and 3.3.2.

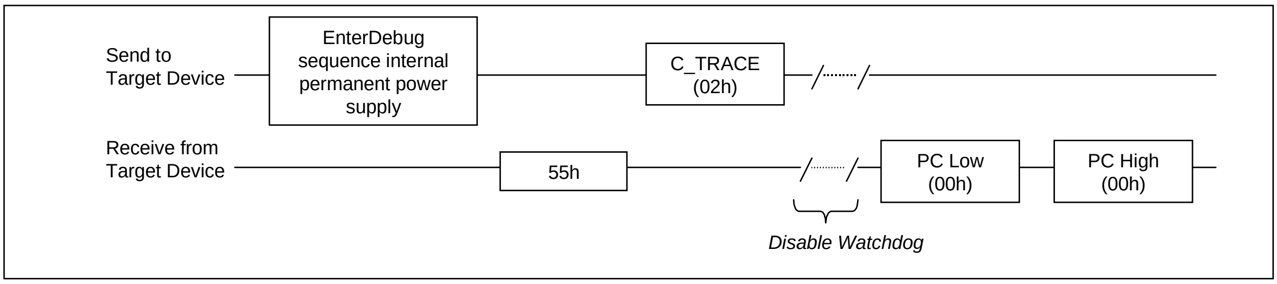


Figure 35 Enter Monitor Internal mode with permanent power supply + disable Watchdog

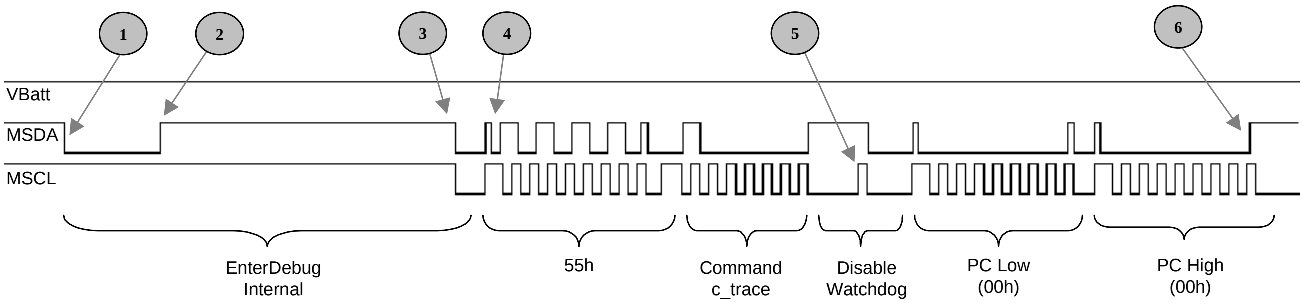


Figure 36 Oscillograph of Enter Monitor Internal mode with permanent power supply + disable Watchdog

1. A falling edge on MSDA generates a Wake Up and starts device operation. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA must be set to low externally during the start of the BOOT sequence.
2. MSDA external has to be set high in order to start the MONITOR mode otherwise a WARM BOOT is performed.
3. The target device signals the decision of external/internal mode by a falling edge on MSCL. In order to enter the internal mode MSDA has to be set to low right after the falling edge of MSCL.
4. The target device signals the start of the data transfer by setting MSCL to high In order to read data the master sets MSDA to tristate externally right after the first rising edge of MSCL. At this time MSDA is set to tristate internally either. Thus, the external pull-up resistor drives MSDA to high. Thereafter the target device configures MSDA as output. This causes a low condition on MSDA. Subsequently the target device sets MSDA to the desired value and signals valid data by a falling edge on MSCL.
5. After the c_trace command has been sent the target device is set to RUN mode during the time the Watchdog is switched off. Since an external break (low condition on MSDA) during the c_trace command can cause unwanted behaviour, MSDA must be set to high externally while MSCL=1 (RUN mode).
6. Since the next transmission/reception starts not before MSDA external has been set to low, MSDA should be kept to high until the next communication.

4.1.3 Detect PROTECTED Mode

In case the target device resides in PROTECTED mode a c_trace command is not applicable and forces the device into POWER-OFF mode. Thus, if the target device is not set into RUN mode (MSCL = 1) after the c_trace command has been sent successfully, the device is assumed to be in PROTECTED mode.

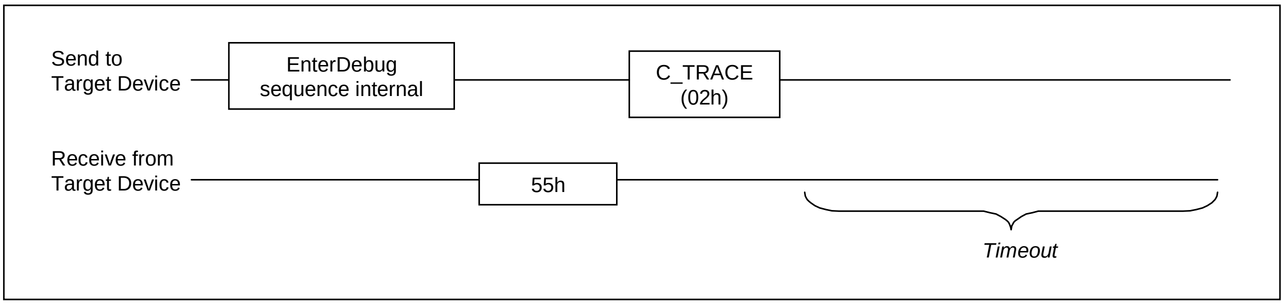


Figure 37 Detect PROTECTED mode

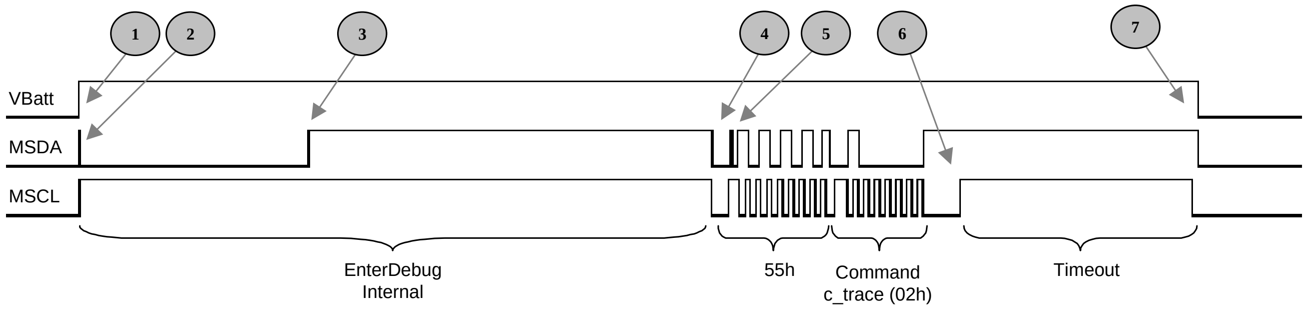


Figure 38 Oscillograph of detecting PROTECTED mode

1. In any case the power up of VBatt causes a Wake Up and starts device operation.
2. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA must be set to low externally during the start of the BOOT sequence.
3. MSDA external has to be set high in order to start the MONITOR mode, otherwise a WARM BOOT is performed.
4. The target device signals the decision of external/internal mode by a falling edge on MSCL. In order to enter the internal mode MSDA has to be set to low right after the falling edge of MSCL.
5. The target device signals the start of the data transfer by setting MSCL to high. In order to read data the master sets MSDA to tristate externally right after the first rising edge of MSCL. At this time MSDA is set to tristate internally either. Thus, the external pull-up resistor drives MSDA to high. Thereafter the target device configures MSDA as output. This causes a low condition on MSDA. Subsequently the target device sets MSDA to the desired value and signals valid data by a falling edge on MSCL.
6. In case the target device resides in INITIAL mode, the device is supposed to be set into RUN mode (MSCL = 1) temporarily after the c_trace command has been sent successfully. Thus, the master is waiting until a MSCL pulse.

   In case the target device resides in PROTECTED mode, the device is forced into POWER-OFF mode. Subsequently the MSCL pulse is not occuring.
7. In case the MSCL pulse does not occur within a time period of 300 μs, the device is assumed to be in PROTECTED mode and the power supply is switched off.

4.1.4 Remove PROTECTED Mode

In order to set a target device which resides in PROTECTED mode back to INITIAL mode the c_er_erom command has to be executed right after the EnterDebug sequence.

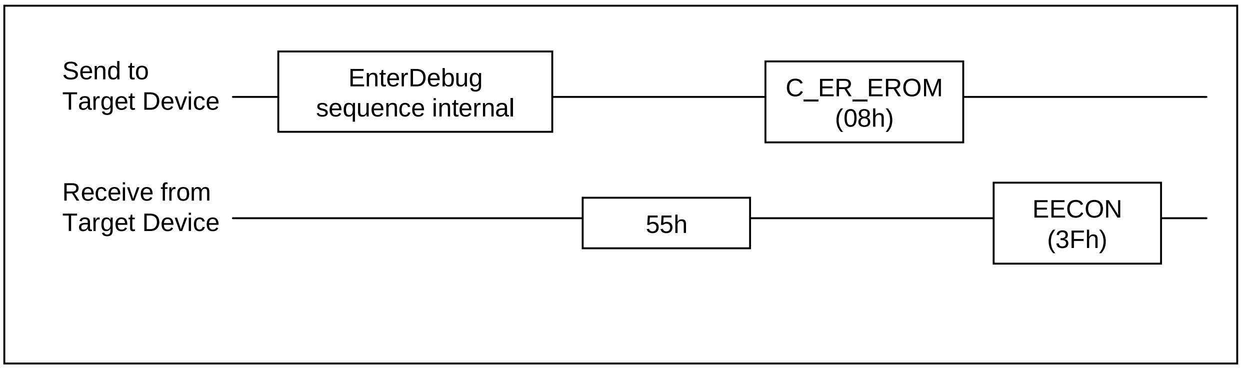


Figure 39 Remove PROTECTED mode

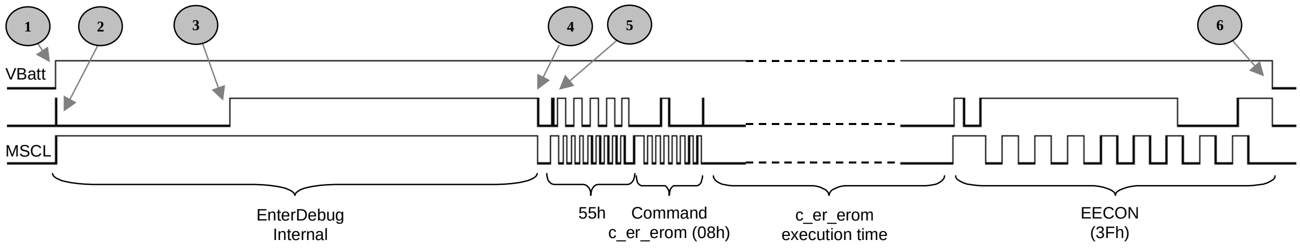


Figure 40 Oscillograph of removing PROTECTED mode

1. In any case the power up of VBat causes a Wake Up and starts device operation.
2. In order to enter the MONITOR mode even in case the target device resides in PROTECTED mode (for more information refer to chapter 3.1) MSDA must be set to low externally during the start of the BOOT sequence.
3. MSDA external has to be set high in order to start the MONITOR mode otherwise a WARM BOOT is performed.
4. The target device signals the decision of external/internal mode by a falling edge on MSCL. In order to enter the internal mode MSDA has to be set to low right after the falling edge of MSCL.
5. The target device signals the start of the data transfer by setting MSCL to high. In order to read data the master sets MSDA to tristate externally right after the first rising edge of MSCL. At this time MSDA is set to tristate internally either. Thus, the external pull-up resistor drives MSDA to high. Thereafter the target device configures MSDA as output. This causes a low condition on MSDA. Subsequently the target device sets MSDA to the desired value and signals valid data by a falling edge on MSCL.
6. During the execution of the command c_er_erom the system clock is switched to 125kHz. Subsequently the Special Function Register EECON is transmitted by using a system clock of 125kHz.

   After the command c_er_erom has been executed successfully, the target device is forced into POWER-OFF mode. Thus, the power supply can be switched off.

4.1.5 Switch System Clock

In order to switch the system clock, the Special Function register SCSL has to be modified. For detailed information about system clock generation refer to the corresponding datasheet. Following example shows how to switch the system clock to 2MHz by setting SCSL to 05h. The MONITOR mode is assumed to be already started.

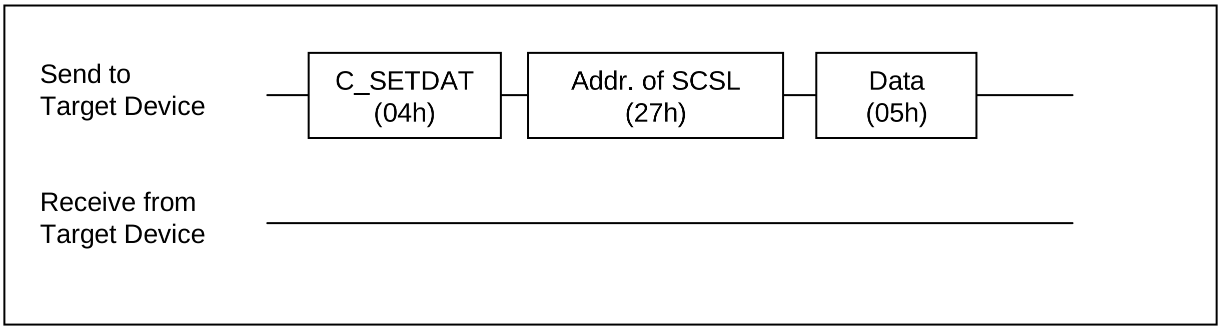


Figure 41 Switch system clock

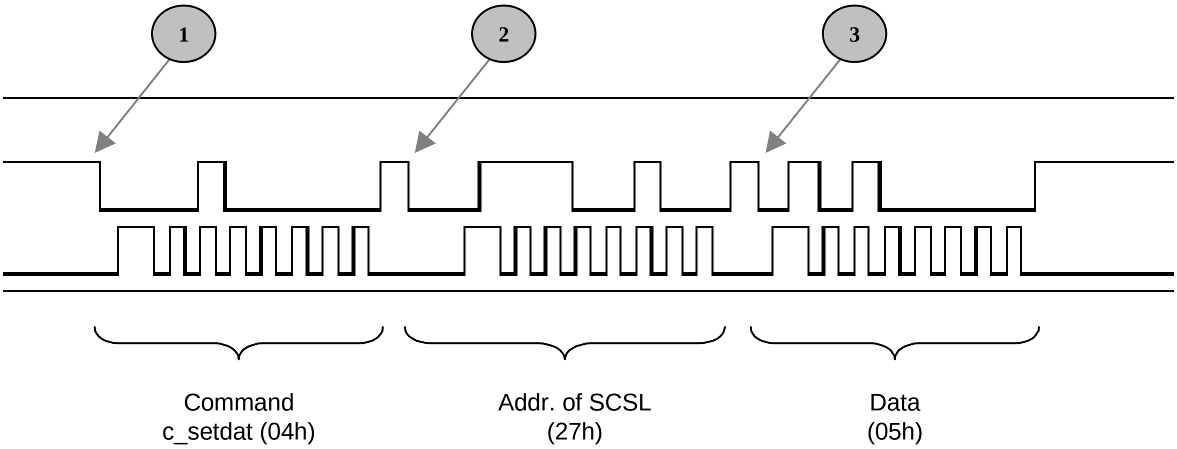


Figure 42 Oscillograph of switching system clock

1. The falling edge of MSDA starts the transmission of command c_setdat (04h)
2. The falling edge of MSDA starts the transmission of the address (27h)
3. The falling edge of MSDA starts the transmission of the content of SCSL (05h)

4.1.6 Read content of Special Function Register SCSL

Following example shows how to retrieve the content (in this case 05h) of the Special Function Register SCSL (located at address 27h). The MONITOR mode is assumed to be already started.

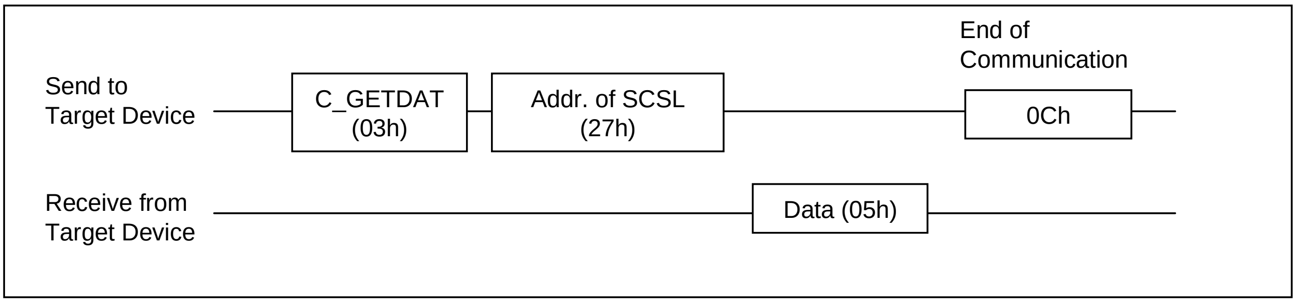


Figure 43 Read content of Special Function Register SCSL

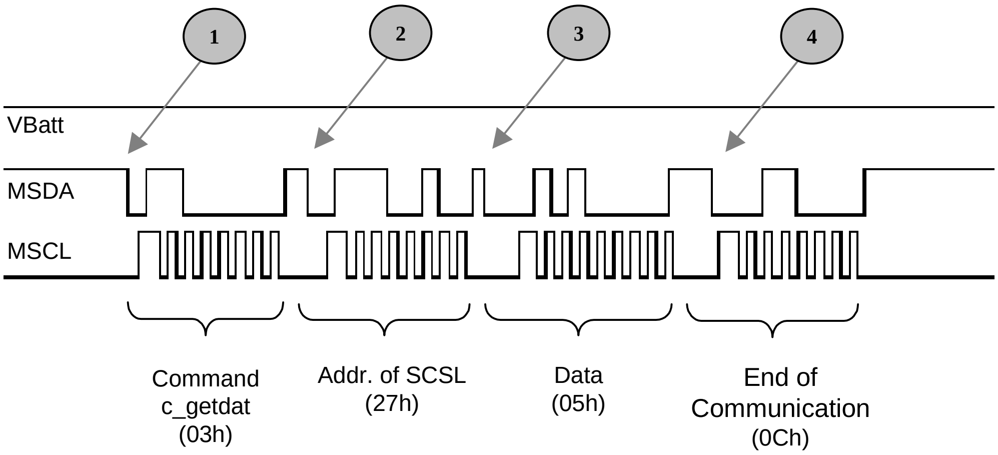


Figure 44 Oscillograph of reading content of Special Function Register SCSL

1. The falling edge of MSDA starts the transmission of command c_getdat (03h)
2. The falling edge of MSDA starts the transmission of the address (27h)
3. The falling edge of MSDA starts the transmission of the content of SCSL (05h)
4. The falling edge of MSDA starts the transmission of 0Ch

4.1.7 External Break

In order to perform an external break during RUN mode (execution of the application program) a low condition on MSDA must be generated. Thereafter the program counter (in this case PC_{Low} = 13h, PC_{High}=00h) is transmitted and the target device enters the MONITOR mode. During program execution from ROM (Syscall) the RUN mode cannot be interrupted.

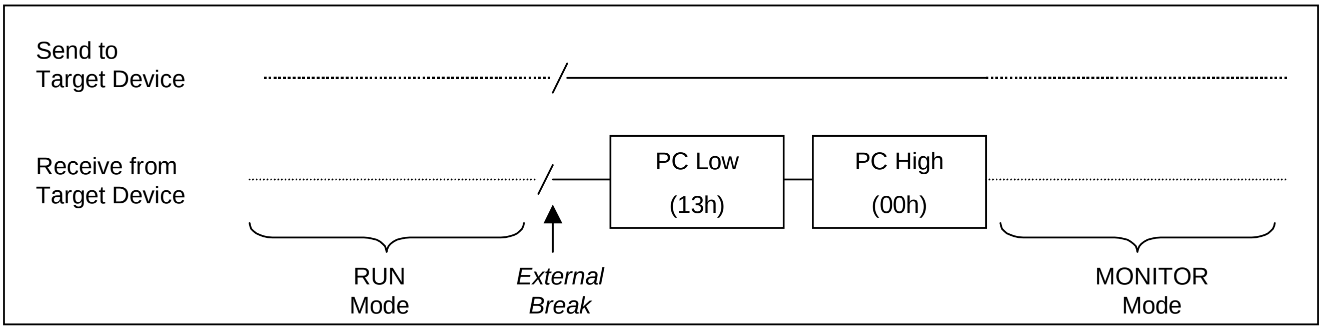


Figure 45 External Break

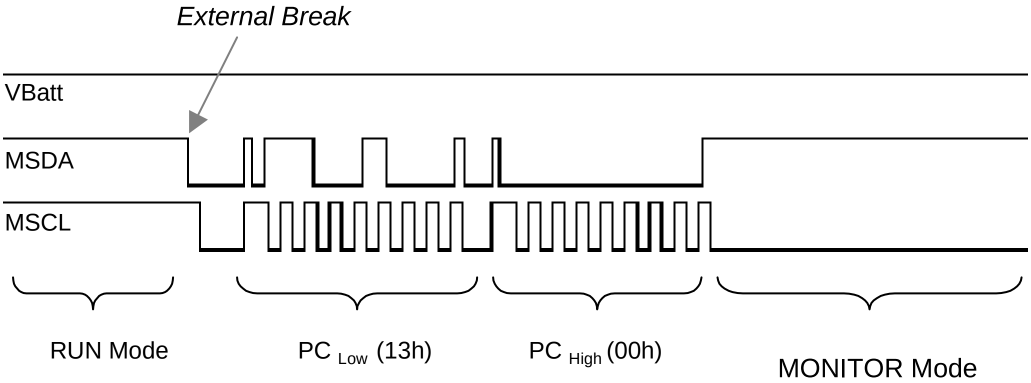


Figure 46 Oscillograph of External Break

5 RELATED DOCUMENTS

Type	Name	Description
Data Sheet	PCF7953 datasheet	Passive Entry Risc Controller
Data Sheet	MRK II	Architecture and Instruction Set
Data Sheet	MRK II Family	Architecture and Instruction Set
Data Sheet	PCF7x41 ROM Library	Implementation and Description
Application Note	AN01034	Software Development for PCF7X41ATS (STARC 2XLite)

6 HISTORY

Date	Revision	Description
2004 Sep 17	1.0	Creation of Document based on “PCH7952 Monitor and Download Interface” (2003 Dec 04)
2004 Nov 26	1.1	Table 3 updated Functions C_ER_EROM, C_WR_EROM, C_WR_EEPORM, W_WR_EROM_B, C_SIG_EROM, C_EE_DUMP, C_ER_DUMP, and C_SIG_XROM updated Function C_SIG_EE_NORM added
2005 May 19		Editorial updates and corrections
2011 Oct 04	1.2	Creation of single PCF7953 Document based on “PCF7953/PCF7945 Monitor and Download Interface” (2005 May 19) - Update Figure 1 Boot Sequence Flow Chart - Correction in the activating of debug functionality in case of a WARM BOOT, chapter 3.1 - Restriction in usage of c_sig_ee_norm MDI command in protected mode, chapter 3.5
2011 Oct 14	1.3	Updated after review by S&A - Add C code for the IAR compiler of debug functionality in case of a WARM BOOT, chapter 3.1 - Legal INFORMATION
2011 Nov 8	1.4	Second Update after review by S&A - Complete Boot Routine flow chart and correct activate debug functionality in case of a WARM BOOT in chapter 3.1

7 LEGAL INFORMATION

7.1 Data sheet status

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

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

[2] The term ‘short data sheet’ is explained in section “Definitions”.

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

7.2 Definitions

Draft

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

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

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

7.3 Disclaimers

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

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

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

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

Suitability for use in automotive applications — This NXP Semiconductors product has been qualified for use in automotive applications. Unless otherwise agreed in writing, the product is not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customers own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer’s sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer’s applications and products planned, as well as for the planned application and use of customer’s third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

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

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

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

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

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

Quick reference data — The Quick reference data is an extract of the product data given in the Limiting values and Characteristics sections of this document, and as such is not complete, exhaustive or legally binding.

8 Contact information

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

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

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