Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	80C51
Core Size	8-Bit
Speed	48MHz
Connectivity	SmartCard, SPI, UART/USART, USB
Peripherals	LED, POR, WDT
Number of I/O	46
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-VQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at89c5122d-alrum

Pin Description

Table 2. Pin Description

Port	VQFP64	VQFP32	PLCC68	PLCC28	QFN64	QFN32	Internal Power Supply	ESD	I/O	Reset Level	Alt	Reset Config	Conf 1	Conf 2	Conf 3	Led
P0.0	30	-	41	-	30	-	VCC	2KV	I/O	Float	AD0	P0		KB_OUT	Push-pull	
P0.1	29	-	40	-	29	-	VCC	2KV	I/O	Float	AD1	P0		KB_OUT	Push-pull	
P0.2	28	-	39	-	28	-	VCC	2KV	I/O	Float	AD2	P0		KB_OUT	Push-pull	
P0.3	27	-	38	-	27	-	VCC	2KV	I/O	Float	AD3	P0		KB_OUT	Push-pull	
P0.4	25	-	36	-	25	-	VCC	2KV	I/O	Float	AD4	P0		KB_OUT	Push-pull	
P0.5	24	-	35	-	24	-	VCC	2KV	I/O	Float	AD5	P0		KB_OUT	Push-pull	
P0.6	23	-	34	-	23	-	VCC	2KV	I/O	Float	AD6	P0		KB_OUT	Push-pull	
P0.7	22	-	33	-	22	-	VCC	2KV	I/O	Float	AD7	P0		KB_OUT	Push-pull	
CIO	64	32	9	4	64	32	CVCC	6KV	I/O	0		Port51	CVCC inactive at reset. ESD tested with a 10µF on CVCC An external pull-up of 10K is recommended to support ICC's with too weak internal pull-ups.			
CC4	3	3	12	7	3	3	CVCC	6KV	I/O	0		Port51	CVCC inactive at reset ESD tested with a 10µF on CVCC			
P1.2	2	2	11	6	2	2	VCC	2KV	I/O	1	CPRES	Port51	Weak & medium pull-up can be disconnected			
CC4	9	5	18	9	9	5	CVCC	6KV	I/O	0		Port51	CVCC inactive at reset ESD tested with a 10µF on CVCC			
CCLK	12	6	21	10	12	6	CVCC	6KV	O	0		Push-pull	CVCC inactive at reset ESD tested with a 10µF on CVCC			
CRST	6	4	15	8	6	4	CVCC	6KV	O	0		Push-pull	CVCC inactive at reset ESD tested with a 10µF on CVCC			
P1.6	47	23	58	-	47	23	VCC	2KV	I/O	1	SS	Port51				
P1.7	62	31	7	-	62	31	VCC	2KV	I/O	1	CCLK1	Port51				
P2.0	58	-	3	-	58	-	VCC	2KV	I/O	1	A8	Port51	Push-pull	KB_OUT	Input WPU	
P2.1	57	-	2	-	57	-	VCC	2KV	I/O	1	A9	Port51	Push-pull	KB_OUT	Input WPU	
P2.2	56	-	1	-	56	-	VCC	2KV	I/O	1	A10	Port51	Push-pull	KB_OUT	Input WPU	
P2.3	52	-	65	-	52	-	VCC	2KV	I/O	1	A11	Port51	Push-pull	KB_OUT	Input WPU	
P2.4	51	-	64	-	51	-	VCC	2KV	I/O	1	A12	Port51	Push-pull	KB_OUT	Input WPU	
P2.5	50	-	63	-	50	-	VCC	2KV	I/O	1	A13	Port51	Push-pull	KB_OUT	Input WPU	

USB Smart Card Reader Using the AT83C5123 Version

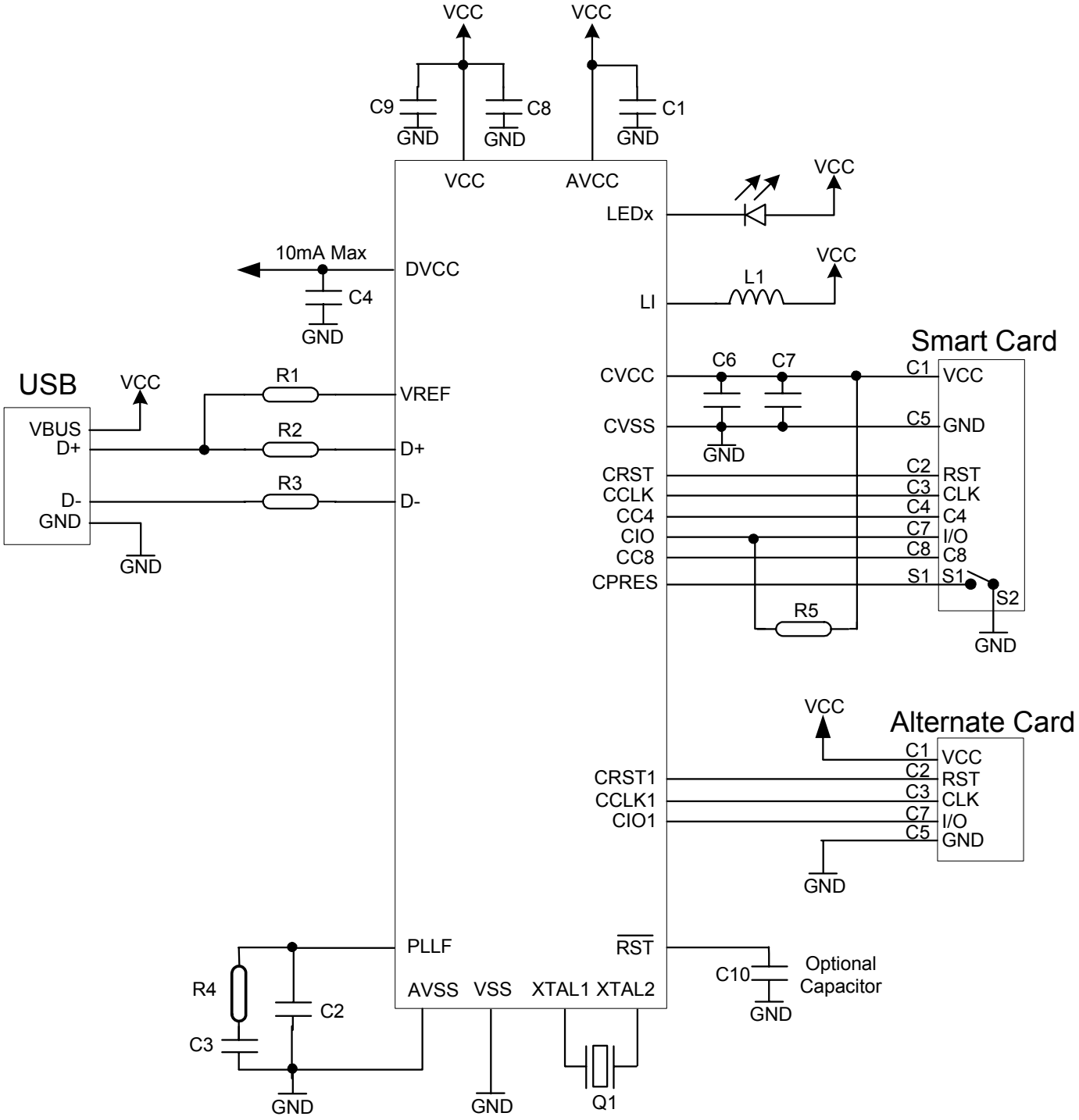


Table 8. CRAM Configuration Register - RCON (D1h)

7	6	5	4	3	2	1	0
-	-	-	-	RPS	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 4	-	**Reserved** The value read from this bit is indeterminate. Do not change these bits.
3	RPS	**CRAM Memory Mapping Bit** Set to map the CRAM memory during MOVX instructions Clear to map the XRAM memory during MOVX. This bit has priority over the EXTRAM bit.
2-0	-	**Reserved** The value read from this bit is indeterminate. Do not change these bits.

Reset Value = XXXX 0XXXb

AT8xC5122's CRAM and E2PROM Versions

The AT8xC5122's CRAM and E2PROM versions implements :

- 32 KB of ROM mapped from 8000 to FFFF in which is embedded a bootloader for In-System Programming feature

- 32 KB of CRAM (Code RAM) , a volatile program memory mapped from 0000 to 7FFF

In CRAM versions only :

- 512 bytes of E2PROM can be optionally implemented to store permanent data

In E2PROM version :

- 32KB of E2PROM are implemented to store permanent code

Warnings :

– some bytes of user program memory space are reserved for bootloader configuration. Depending on the configuration, up to 256 bytes of code may be not available for the user code from 7F00h location. Refer to bootloader datasheet for further details.

– Port P3.7 may be used by the bootloader as a hardware condition at reset to select the In-System Programming mode. Once the bootloader has started, the P3.7 Port is no more used.

Figure 13. AT83R5122, AT8xC5122's ROM Version

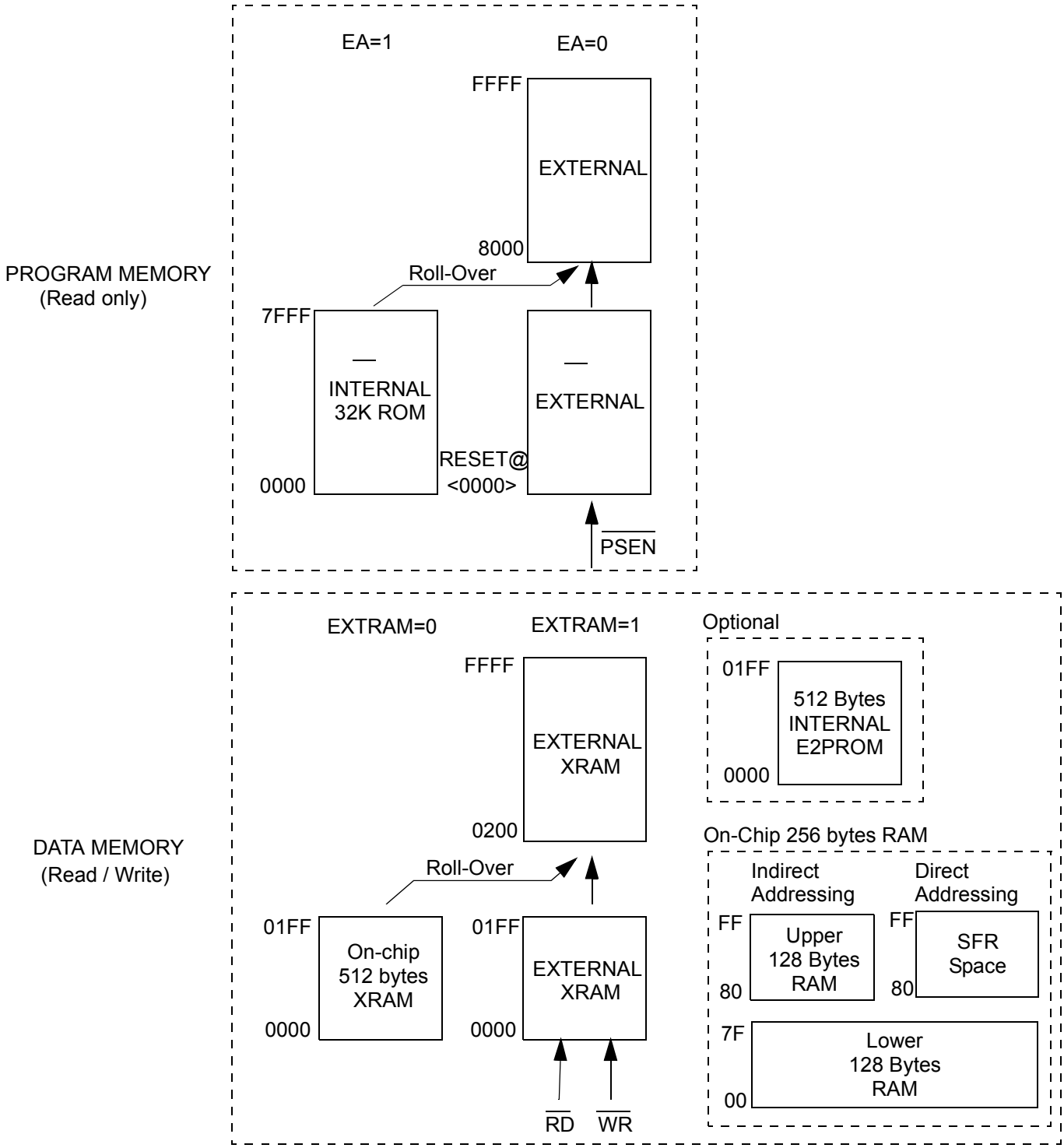
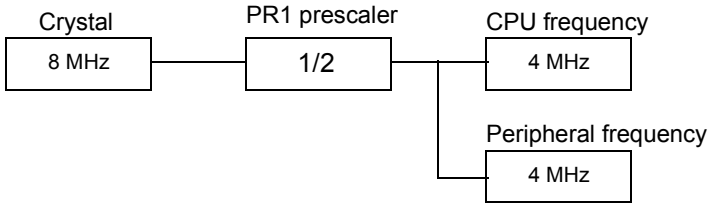
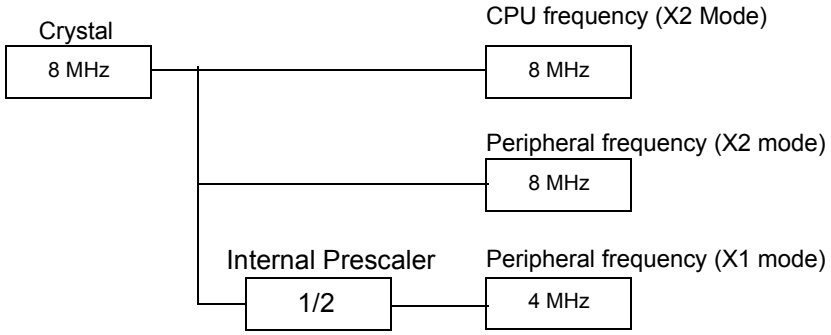


Figure 20. X1 mode



When the X1 mode is selected, the CPU and Peripherals work at 8Mhz / X1

Figure 21. X2 mode



When the X2 mode is selected, the CPU works at 8 MHz / X2. The Peripherals can work at 8 MHz / X2 or 8 MHz / X1.

When the PR1 prescaler is different from 1/2, the usual X1 mode can not be defined. In this case, it is necessary to define a X1 or X2 equivalent mode from equivalent clock circuits.

Example : PR1=1/8, X2=0.

In this configuration, the CPU works at 1 MHz. This frequency could also be obtained by an equivalent clock circuit where the on-chip oscillator would run at 2 MHz in X1 mode or at 1 MHz in X2 mode. So we can say that the CPU works at 2 MHz / X1 or 1MHz / X2.

As the X2 bit is cleared in CKCON0 register, we have $F_{CK\_IDLE} = F_{CK\_PERIPH}$.

Registers

Table 36. Port Mode Register 0 - PMOD0 (91h) for AT8xC5122

7	6	5	4	3	2	1	0
P3C1	P3C0	P2C1	P2C0	CPRESRES	-	P0C1	P0C0

Bit Number	Bit Mnemonic	Description
7 - 6	P3C1-P3C0	**Port 3 Configuration bits (Applicable to P3.0, P3.1, P3.3, P3.4 only)** **00** Quasi bi-directional **01** Push-pull **10** Output Low Speed **11** Input with weak pull-up
5-4	P2C1-P2C0	**Port 2 Configuration bits** **00** Quasi bi-directional **01** Push-pull **10** Output Low Speed **11** Input with weak pull-down
3	CPRESRES	**Card Presence Pull-up resistor** Cleared to connect the internal 100K pull-up Set to disconnect the internal pull-up
2	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
1-0	P0C1-P0C0	**Port 0 Configuration bits** **00** C51 Standard P0 **01** Reserved **10** Output Low Speed **11** Push-pull

Reset Value = 0000 0x00b

Table 37. Port Mode Register 0 - PMOD0 (91h) for AT83C5123

7	6	5	4	3	2	1	0
P3C1	P3C0	-	-	CPRESRES	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 6	P3C1-P3C0	**Port 3 Configuration bits (Applicable to P3.0, P3.1, P3.3, P3.4 only)** **00** Quasi bi-directional **01** Push-pull **10** Output Low Speed **11** Input with weak pull-up
5-4		**Reserved** The value read from these bits are indeterminate. Do not set these bit.
3	CPRESRES	**Card Presence Pull-up resistor** Cleared to connect the internal 100K pull-up Set to disconnect the internal pull-up
2-0	-	**Reserved** The value read from these bits are indeterminate. Do not set these bit.

Reset Value = 00xx 0xxxb

Table 40. LED Port Control Register 0 - LEDCON0 (F1h)

7	6	5	4	3	2	1	0
LED3.1	LED3.0	LED2.1	LED2.0	LED1.1	LED1.0	LED0.1	LED0.0

Bit Number	Bit Mnemonic	Description
7 - 6	LED3	**Port LED3 Configuration bits** **00** LED control disabled **01** 2 mA current source when P3.7 is configured as Quasi-bi-directional mode **10** 4 mA current source when P3.7 is configured as Quasi-bi-directional mode **11** 10 mA current source when P3.7 is configured as Quasi-bidirect. mode
5 - 4	LED2	**Port LED2 Configuration bits** **00** LED control disabled **01** 2 mA current source when P3.6 is configured as Quasi-bi-directional mode **10** 4 mA current source when P3.6 is configured as Quasi-bi-directional mode **11** 10 mA current source when P3.6 is configured as Quasi-bidirect. mode
3 - 2	LED1	**Port LED1 Configuration bits** **00** LED control disabled **01** 2 mA current source when P3.4 is configured as Quasi-bi-directional mode **10** 4 mA current source when P3.4 is configured as Quasi-bi-directional mode **11** 10 mA current source when P3.4 is configured as Quasi-bidirect. mode
1 - 0	LED0	**Port LED0 Configuration bits** **00** LED control disabled **01** 2 mA current source when P3.2 is configured as Quasi-bi-directional mode **10** 4 mA current source when P3.2 is configured as Quasi-bi-directional mode **11** 10 mA current source when P3.2 is configured as Quasi-bidirect. mode

Reset Value = 0000 0000b

Table 41. LED Port Control Register 1- LEDCON1 (F1h) only for AT8xC5122

7	6	5	4	3	2	1	0
-	-	LED6.1	LED6.0	LED5.1	LED5.0	LED4.1	LED4.0

Bit Number	Bit Mnemonic	Description
7 - 6		**Reserved** The value read from this bit is indeterminate. Do not set this bit.
5 - 4	LED6	**Port LED6 Configuration bits** **00** LED control disabled **01** 2 mA current source when P4.5 is configured as Quasi-bi-directional mode **10** 4 mA current source when P4.5 is configured as Quasi-bi-directional mode **11** 10 mA current source when P4.5 is configured as Quasi-bidirect. mode
3 - 2	LED5	**Port LED5 Configuration bits** **00** LED control disabled **01** 2 mA current source when P4.4 is configured as Quasi-bi-directional mode **10** 4 mA current source when P4.4 is configured as Quasi-bi-directional mode **11** 10 mA current source when P4.4 is configured as Quasi-bidirect. mode
1 - 0	LED4	**Port LED0 Configuration bits** **00** LED control disabled **01** 2 mA current source when P4.3 is configured as Quasi-bi-directional mode **10** 4 mA current source when P4.3 is configured as Quasi-bi-directional mode **11** 10 mA current source when P4.3 is configured as Quasi-bidirect. mode

Reset Value = 0000 0000b

IEC7816-3 says this procedure is mandatory in ATR for card supporting T=0 while EMV says this procedure is mandatory for T=0 but does not apply for ATR.

Functional Description

The architecture of the Smart Card Interface Block can be detailed as follows:

Barrel Shifter

The Barrel Shifter performs the translation between 1 bit serial data and 8 bits parallel data

The barrel function is useful for character repetition since the character is still present in the shifter at the end of the character transmission.

This shifter is able to shift the data in both directions and to invert the input or output value in order to manage both direct and inverse ISO7816-3 convention.

Coupled with the barrel shifter is a parity checker and generator.

There are 2 registers connected to this barrel shifter, one for the transmission and one for the reception. They act as buffers to relieve the CPU of timing constraints.

SCART FSM

(Smart Card Asynchronous Receiver Transmitter Finite State Machine)

This is the core of the block. Its purpose is to control the barrel shifter. To sequence correctly the barrel shifter for a reception or a transmission, it uses the signals issued by the different counters. One of the most important counters is the guard time counter that gives time slots corresponding to the character frame.

The SCART FSM is enabled only in UART mode.

The transition from the receipt mode to the transmit mode is done automatically. Priority is given to the transmission. Transmission refers to Terminal transmission to the ICC. Reception refers to reception by the Terminal from the ICC.

ETU Counter

The ETU (Elementary Timing Unit) counter controls the working frequency of the barrel shifter, in fact it generates the enable signal of the barrel shifter. It receives the Card Clock, and generates the ETU clock. The Card Clock frequency is called “f” below. The ETU counter is 11 bit wide.

A special compensation mode can be activated. It accomodates situations where the ETU is not an integer number of Card Clock (CK_ISO). The compensation mode is controlled by the COMP bit in SCETU1 register bit position 7. With COMP=1 the ETU of every character even bits is reduced by 1 Card Clock period. As a result, the average ETU is : ETU_average = (ETU - 0.5). One should bear in mind that the ETU counter should be programmed to deliver a faster ETU which will be reduced by the COMP mechanism, not the other way around. This allows to reach the required precision of the character duration specified by the ISO7816-3 standard.

Example1 : F=372, D=32 => ETU= F/D = 11.625 clock cycles.

We select ETU[10-0] = 12 , COMP=1. ETUaverage= 12 - (0.5*COMP) = 11.5

The result will be a full character duration (10 bit) = (10 - 0.107)*ETU. The EMV specification is (10 +/- 0.2)*ETU

Guard Time Counter

The minimum time between the leading edge of the start bit of 2 consecutive characters transmitted by the Terminal is controlled by the Guard Time counter, as described in Figure 32.

Additional Features

Clock

The CK_ISO input must be in the range 1 - 5 MHz according to ISO 7816.

The CK_ISO can be programmed up to 12 MHz. In this case, the timing specification of the output buffer will not comply to ISO 7816.

Figure 40. Clock Diagram of the SCIB Block

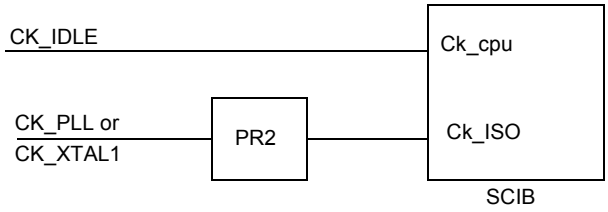
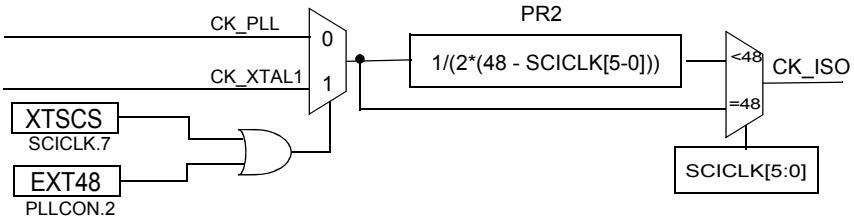


Figure 41. Prescaler 2 Description



The division factor SCICLK must be smaller than 49. If it is greater or equal to 49, the PR2 prescaler is locked.

See Figure 17 clock tree diagram in the clock controller chapter.

Table 42. Examples of Clock settings

XTAL1 (MHz)	EXT48	SCICLK	CK_ ISO
8	0	36	4
8	0	44	12
8	0	42	8
8	0	40	6
8	0	24	2
8	0	0	1

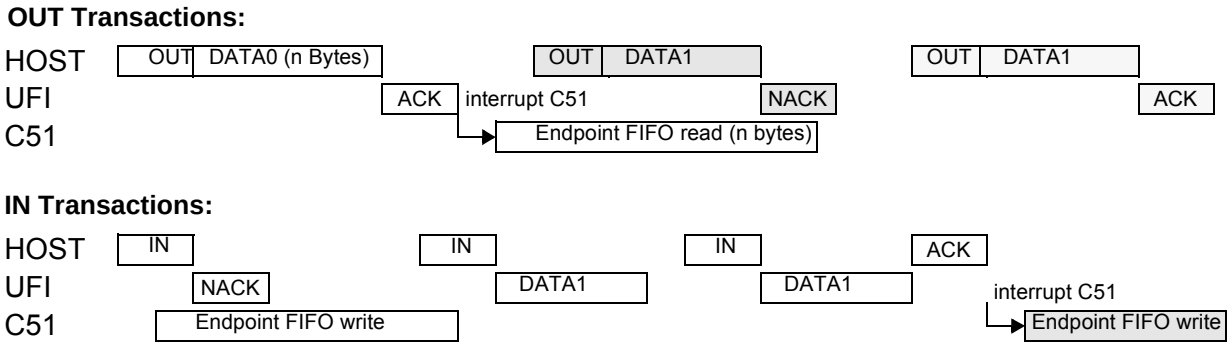
Card Presence Input

The internal pull-up (weak pull-up) on Card Presence input can be disconnected in order to reduce the consumption (CPRESRES, bit 3 in PMOD0).

In this case, an external resistor (typically 1 MΩ) must be externally tied to Vcc.

CPRES input can generate an interrupt (see Interrupt system section).
The detection level can be selected.

Figure 51. Minimum Intervention from the USB Device Firmware



Read/Write Data FIFO

Read Data FIFO

The read access for each OUT endpoint is performed using the UEPDATX register.

After a new valid packet has been received on an Endpoint, the data are stored into the FIFO and the byte counter of the endpoint is updated (UBYCTX register). The firmware has to store the endpoint byte counter before any access to the endpoint FIFO. The byte counter is not updated when reading the FIFO.

To read data from an endpoint, select the correct endpoint number in UEPNUM and read the UEPDATX register. This action automatically decreases the corresponding address vector, and the next data is then available in the UEPDATX register.

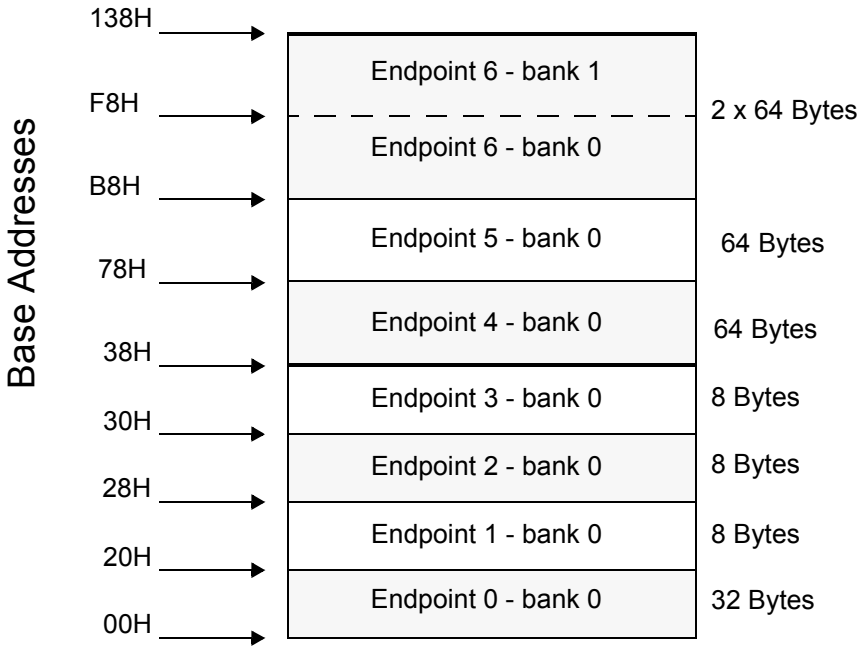
Write Data FIFO

The write access for each IN endpoint is performed using the UEPDATX register.

To write a byte into an IN endpoint FIFO, select the correct endpoint number in UEPNUM and write into the UEPDATX register. The corresponding address vector is automatically increased, and another write can be carried out.

Warning 1: The byte counter is not updated.
Warning 2: Do not write more bytes than supported by the corresponding endpoint.

Figure 53. Endpoint FIFO Configuration



Miscellaneous

USB Reset

The EORINT bit in the USBINT register is set by hardware when a End of Reset has been detected on the USB bus. This triggers a USB interrupt if enabled. The USB controller is still enabled, but all the USB registers are reset by hardware. The firmware should clear the EORINT bit to allow the next USB reset detection.

STALL Handshake

This function is only available for Control, Bulk, and Interrupt endpoints.

The firmware has to set the STALLRQ bit in the UEPSTAX register to send a STALL handshake at the next request of the Host on the endpoint selected with the UEPNUM register. The RXSETUP, TXRDY, TXCMPL, RXOUTB0 and RXOUTB1 bits must be first reset to 0. The bit STLCRC is set at 1 by the USB controller when a STALL has been sent. This triggers an interrupt if enabled.

The firmware should clear the STALLRQ and STLCRC bits after each STALL sent.
The STALLRQ bit is cleared automatically by hardware when a valid SETUP PID is received on a CONTROL type endpoint.

Start of Frame Detection

The SOFINT bit in the USBINT register is set when the USB controller detects a Start Of Frame PID. This triggers an interrupt if enabled. The firmware should clear the SOFINT bit to allow the next Start of Frame detection.

Frame Number

When receiving a Start of Frame, the frame number is automatically stored in the UFNUML and UFNUMH registers. The CRCOK and CRCERR bits indicate if the CRC of the last Start Of Frame is valid (CRCOK set at 1) or corrupt (CRCERR set at 1). The UFNUML and UFNUMH registers are automatically updated when receiving a new Start of Frame.

Data Toggle Bit

The Data Toggle bit is set by hardware when a DATA 0 packet is received and accepted by the USB controller and cleared by hardware when a DATA 1 packet is received and accepted by the USB controller. This bit is reset when the firmware resets the endpoint FIFO using the UEPRST register.

For Control endpoints, each SETUP transaction starts with a DATA 0 and data toggling is then used as for Bulk endpoints until the end of the Data stage (for a control write transfer). The Status stage completes the data transfer with a DATA 1 (for a control read transfer).

For Isochronous endpoints, the device firmware should ignore the data-toggle.

NAK Handshakes

When a NAK handshake is sent by the USB controller to a IN or OUT request from the Host, the NAKIN or NAKOUT bit is set by hardware. This information can be used to determine the direction of the communication during a Control transfer.
These bits are cleared by software.

Table 65. USB Global Interrupt Register - USBINT (S:BDh)

7	6	5	4	3	2	1	0
-	-	WUPCPU	EORINT	SOFINT	-	-	SPINT

Bit Number	Bit Mnemonic	Description
7 - 6	-	**Reserved** The value read from these bits is always 0. Do not change these bits.
5	WUPCPU	**Wake-up CPU Interrupt** This bit is set by hardware when the USB controller is in SUSPEND state and is re-activated by a non-idle signal FROM USB line (not by an upstream resume). This triggers a USB interrupt when EWUPCPU is set in the Table on page 117. When receiving this interrupt, user has to enable all USB clock inputs. This bit should be cleared by software (USB clocks must be enabled before).
4	EORINT	**End of Reset Interrupt** This bit is set by hardware when End of Reset has been detected by the USB controller. This triggers a USB interrupt when EEORINT is set in the Table on page 117. This bit should be cleared by software.
3	SOFINT	**Start Of Frame Interrupt** This bit is set by hardware when an USB Start Of Frame PID (SOF) has been detected. This triggers a USB interrupt when ESOFINT is set in the Table on page 117. This bit should be cleared by software.
2-1	-	**Reserved** The value read from these bits is always 0. Do not change these bits.
0	SPINT	**Suspend Interrupt** This bit is set by hardware when a USB Suspend (Idle bus for three frame periods: a J state for 3 ms) is detected. This triggers a USB interrupt when ESPINT is set in USBIEN register (Table 66 on page 117). This bit must be cleared by software before powering the microcontroller down as it disables the USB pads to reduce the power consumption.

Reset Value = 0000 0000b

Table 66. USB Global Interrupt Enable Register - USBIEN (S:BEh)

7	6	5	4	3	2	1	0
-	-	EWUPCPU	EEORINT	ESOFINT	-	-	ESPINT

Bit Number	Bit Mnemonic	Description
7 - 6	-	**Reserved** The value read from these bits is always 0. Do not change these bits.
5	EWUPCPU	**Enable Wake-up CPU Interrupt** Set this bit to enable Wake-up CPU Interrupt. Clear this bit to disable Wake-up CPU Interrupt.
4	EEORINT	**Enable End of Reset Interrupt** Set this bit to enable End of Reset Interrupt. This bit is set after reset. Clear this bit to disable End of Reset Interrupt.
3	ESOFINT	**Enable SOF Interrupt** Set this bit to enable SOF Interrupt. Clear this bit to disable SOF Interrupt.
2-1	-	**Reserved** The value read from these bits is always 0. Do not change these bits.
0	ESPINT	**Enable Suspend Interrupt** Set this bit to enable Suspend Interrupts (See Table 65 on page 116). Clear this bit to disable Suspend Interrupts.

Reset Value = 0001 0000b

Table 67. USB Address Register - USBADDR (S:C6h)

7	6	5	4	3	2	1	0
FEN	UADD6	UADD5	UADD4	UADD3	UADD2	UADD1	UADD0

Bit Number	Bit Mnemonic	Description
7	FEN	**Function Enable** Set this bit to enable the function. FADD is reset to 1. Cleared this bit to disable the function.
6-0	UADD[6:0]	**USB Address** This field contains the default address (0) after power-up or USB bus reset. It should be written with the value set by a SET_ADDRESS request received by the device firmware.

Reset Value = 1000 0000b

sions (Figure 87). This format may be preferable in systems having only one Master and only one Slave driving the MISO data line.

Error Conditions

The following flags in the SPSTA signal SPI error conditions.

Mode Fault (MODF)

MODF error bit in Master mode SPI indicates that the level on the Slave Select ($\overline{SS}$) pin is inconsistent with the actual mode of the device. MODF is set to warn that there may have a multi-master conflict for system control. In this case, the SPI system is affected in the following ways:

- An SPI receiver/error CPU interrupt request is generated.
- The SPEN bit in SPCON is cleared. This disable the SPI.
- The MSTR bit in SPCON is cleared.

When $\overline{SS}$ Disable (SSDIS) bit in the SPCON register is cleared, the MODF flag is set when the $\overline{SS}$ signal becomes ’0’.

However, as stated before, for a system with one Master, if the $\overline{SS}$ pin of the Master device is pulled low, there is no way that another Master is attempting to drive the network. In this case, to prevent the MODF flag from being set, software can set the SSDIS bit in the SPCON register and therefore making the $\overline{SS}$ pin as a general-purpose I/O pin.

Clearing the MODF bit is accomplished by a read of SPSTA register with MODF bit set, followed by a write to the SPCON register. SPEN Control bit may be restored to its original set state after the MODF bit has been cleared.

Write Collision (WCOL)

A Write Collision (WCOL) flag in the SPSTA is set when a write to the SPDAT register is done during a transmit sequence.

WCOL does not cause an interruption, and the transfer continues uninterrupted.

Clearing the WCOL bit is done through a software sequence of an access to SPSTA and an access to SPDAT.

Overrun Condition

An overrun condition occurs when the Master device tries to send several data bytes and the Slave device has not cleared the SPIF bit issuing from the previous data byte transmitted. In this case, the receiver buffer contains the byte sent after the SPIF bit was last cleared. A read of the SPDAT returns this byte. All others bytes are lost.

This condition is not detected by the SPI peripheral.

SS Error Flag (SSERR)

A Synchronous Serial Slave Error occurs when $\overline{SS}$ goes high before the end of a received data in slave mode. SSERR does not cause in interruption, this bit is cleared by writing 0 to SPEN bit (reset of the SPI state machine).

Interrupts

Two SPI status flags can generate a CPU interrupt requests:

Table 84. SPI Interrupts

Flag	Request
SPIF (SP data transfer)	SPI Transmitter Interrupt request
MODF (Mode Fault)	SPI Receiver/Error Interrupt Request (if SSDIS = ’0’)

Serial Peripheral data transfer flag, SPIF: This bit is set by hardware when a transfer has been completed. SPIF bit generates transmitter CPU interrupt requests.

Table 105. Interrupt Priority High Register 1 - IPH1 (B3h) for AT8xC5122

7	6	5	4	3	2	1	0
-	PUSBH	-	-	PSCIH		-	

Bit Number	Bit Mnemonic	Description
7	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
6	PUSBH	**USB Interrupt Priotity High bit** PUSBH PUSBL Priority Level 0 0 Lowest 0 1 1 0 1 1 Highest
5-4	-	**Reserved** The value read from this bit is indeterminate. Do not change these bits.
3	PSCIH	**SCI Interrupt Priority High bit** PSCIH PSCIL Priority Level 0 0 Lowest 0 1 1 0 1 1 Highest
2	PSPIH	**SPI Interrupt Priority High bit** PSPIH PSPIL Priority Level 0 0 Lowest 0 1 1 0 1 1 Highest
1	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
0	PKBH	**Keyboard Interrupt Priority High bit** PKBDH PKBDL Priority Level 0 0 Lowest 0 1 1 0 1 1 Highest

Reset Value = XXXX X000b (Not bit addressable)

Microcontroller Reset

Introduction

The internal reset is used to start up (cold reset) or to re-start (warm reset) the microcontroller activity. When the reset is applied (active state), all internal registers are initialized so that the microcontroller starts from a known and clean state for the program always runs as expected.

The reset is released (inactive state) when the following conditions are internally met :

- The power supply has reatched a minimum level which garantees that the microcontroller works properly
- The on-chip oscillator has reached a minimum oscillation level which ensures a good noise to signal ratio and a correct internal duty cycle
- the active state duration is at least two machine cycles.

If one of the above conditions is not met the microcontroller is not correctly reset and might not work properly.

The internal reset comes from four different sources :

- Reset pin
- Power On Reset (POR)
- Power Fail Detector (PFD)
- Hardware Watch-Dog Timer (WDT)

Figure 100. Reset bock diagram

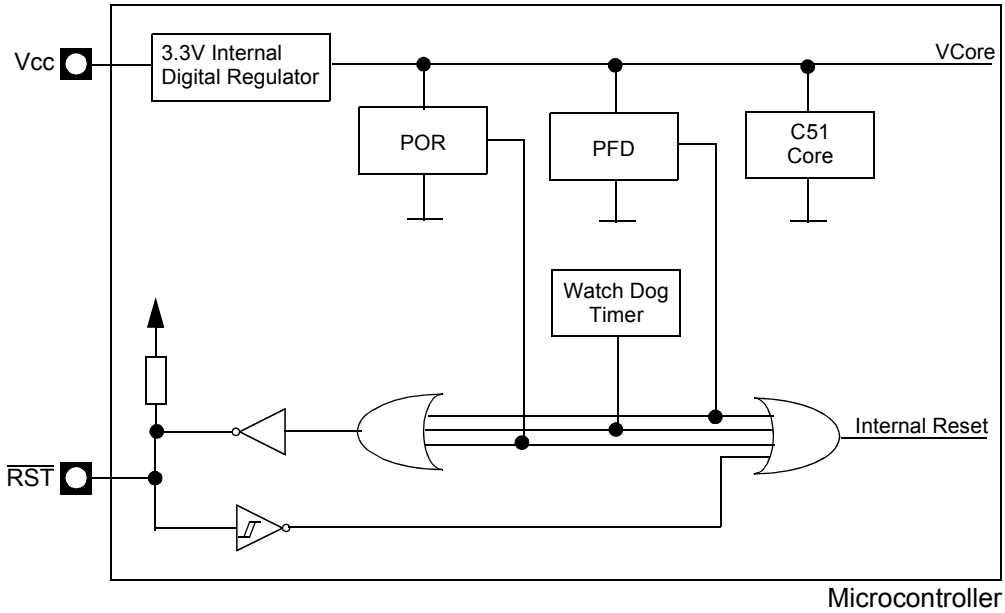


Figure 101. Static behaviour of POR and PFD

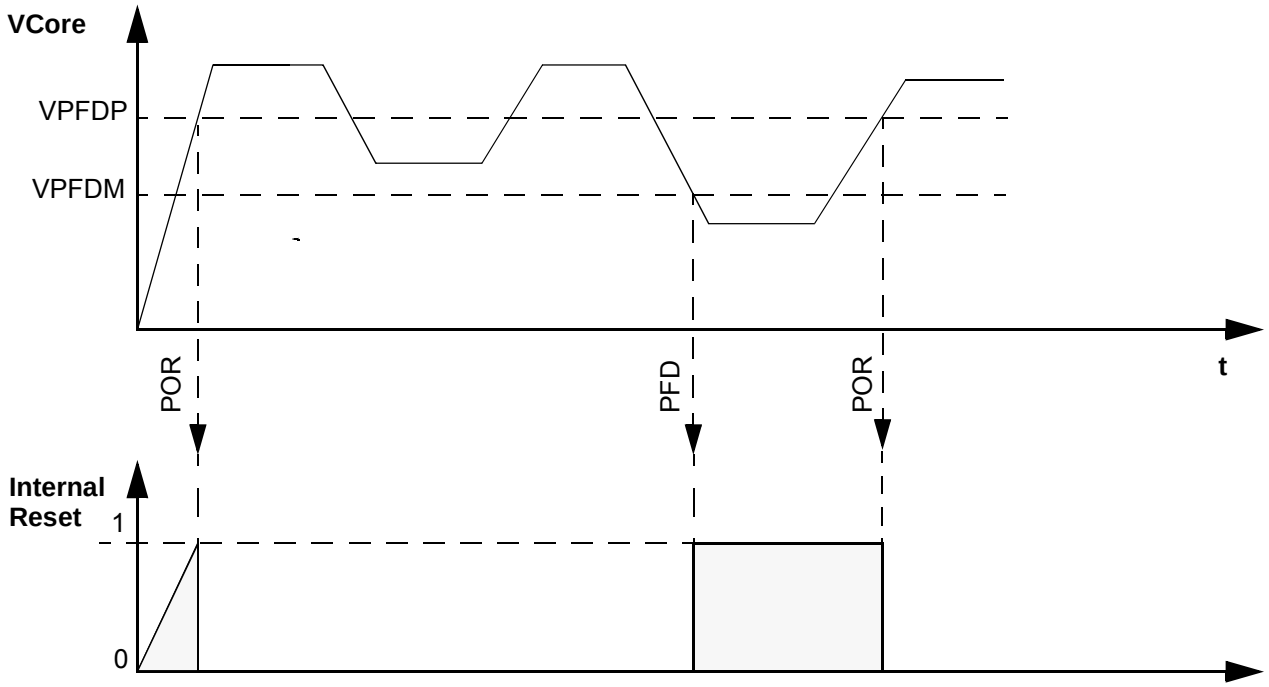


Figure 102. Dynamic behaviour of POR and PFD

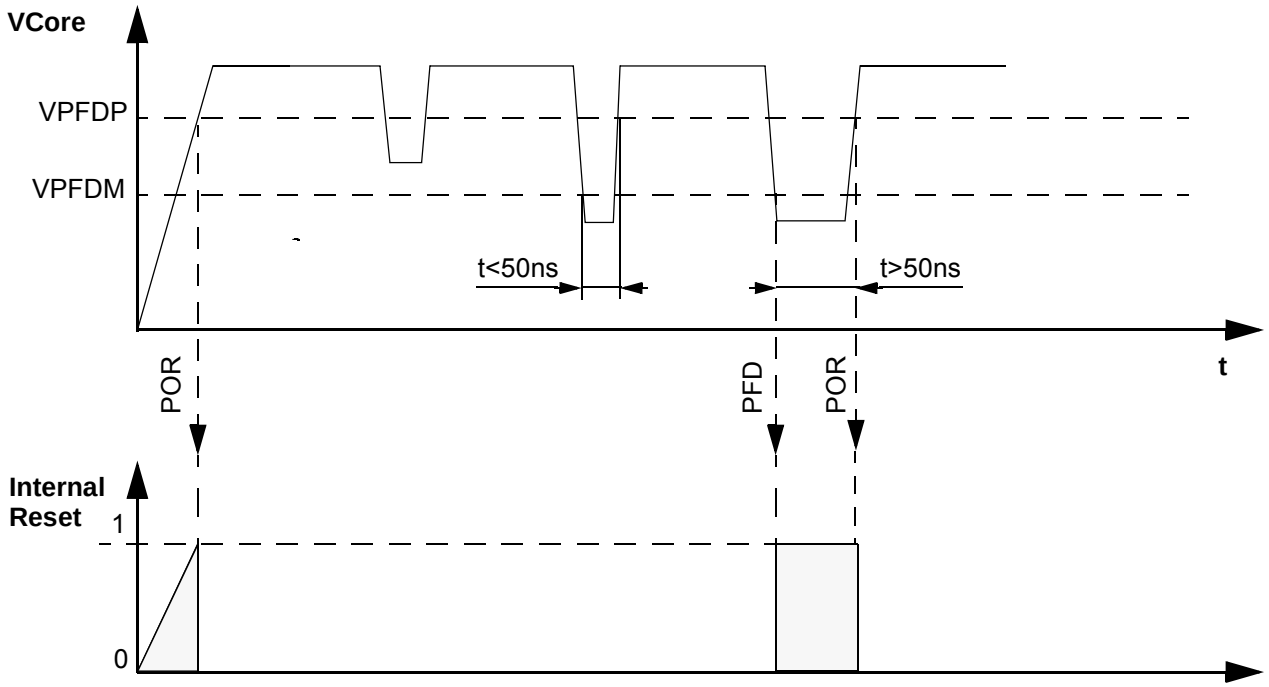


Table 111. Timeout value for F_{CK_WD} = 24 MHz / X2

S2	S1	S0	Timeout for F_{CK_WD}= 24 MHz / X2
0	0	0	4.10 ms
0	0	1	8.19 ms
0	1	0	16.38 ms
0	1	1	32.77 ms
1	0	0	65.54 ms
1	0	1	131.07 ms
1	1	0	262.14 ms
1	1	1	524.29 ms

Table 112. Watchdog Timer Enable register (Write Only) - WDTRST (A6h)

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

Reset Value = XXXX XXXXb

The WDTRST register is used to reset / enable the WDT by writing 1EH then E1H in sequence.

AC Parameters

Explanation of the AC Symbols

Each timing symbol has 5 characters. The first character is always a “T” (stands for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

Example:T_{AVLL} = Time for Address Valid to ALE Low.
T_{LLPL} = Time for ALE Low to $\overline{PSEN}$ Low.

T_A = -40°C to +85°C; V_{SS} = 0V; V_{CC} = 3.0V to 5.5V ; F_{CK_CPU} = 0 to 24 MHz.

(Load Capacitance for port 0, ALE and PSEN = 60 pF; Load Capacitance for all other outputs = 60 pF.)

Table and Table 118 give the description of each AC symbols.

Table 117 and Table 120 give for each range the AC parameter.

Table 115, Table 117 and Table 119 give the frequency derating formula of the AC parameter for each speed range description. To calculate each AC symbols. take the x value and use this value in the formula.

Example: T_{LLIV} and 20 MHz, Standard clock.
x = 30 ns
T = 50 ns
T_{CCIV} = 4T - x = 170 ns

External Program Memory Characteristics

Table 114. Symbol Description

Symbol	Parameter
T	CPU clock period (F_{CK_CPU})
T_{LHLL}	ALE pulse width
T_{AVLL}	Address Valid to ALE
T_{LLAX}	Address Hold After ALE
T_{LLIV}	ALE to Valid Instruction In
T_{LLPL}	ALE to $\overline{PSEN}$
T_{PLPH}	$\overline{PSEN}$ Pulse Width
T_{PLIV}	$\overline{PSEN}$ to Valid Instruction In
T_{PXIX}	Input Instruction Hold After $\overline{PSEN}$
T_{PXIZ}	Input Instruction Float After $\overline{PSEN}$
T_{AVIV}	Address to Valid Instruction In
T_{PLAZ}	$\overline{PSEN}$ Low to Address Float