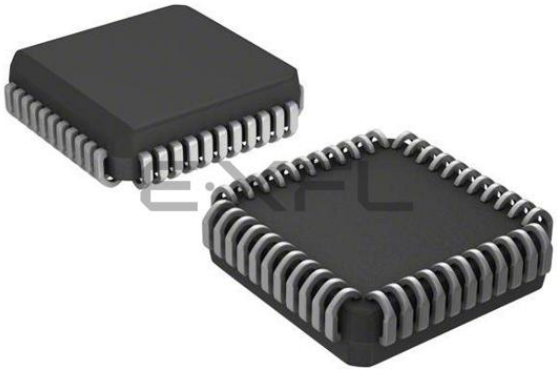


Welcome to E-XFL.COM

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	Active
Core Processor	80C51
Core Size	8-Bit
Speed	60MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	34
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.6x16.6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at89c51id2-slrum

The AT89C51ID2 retains all features of the Atmel 80C52 with 256 bytes of internal RAM, a 10-source 4-level interrupt controller and three timer/counters.

In addition, the AT89C51ID2 has a Programmable Counter Array, an XRAM of 1792 bytes, a Hardware Watchdog Timer, SPI and Keyboard, a more versatile serial channel that facilitates multiprocessor communication (EUART) and a speed improvement mechanism (X2 mode).

The fully static design of the AT89C51ID2 allows to reduce system power consumption by bringing the clock frequency down to any value, even DC, without loss of data.

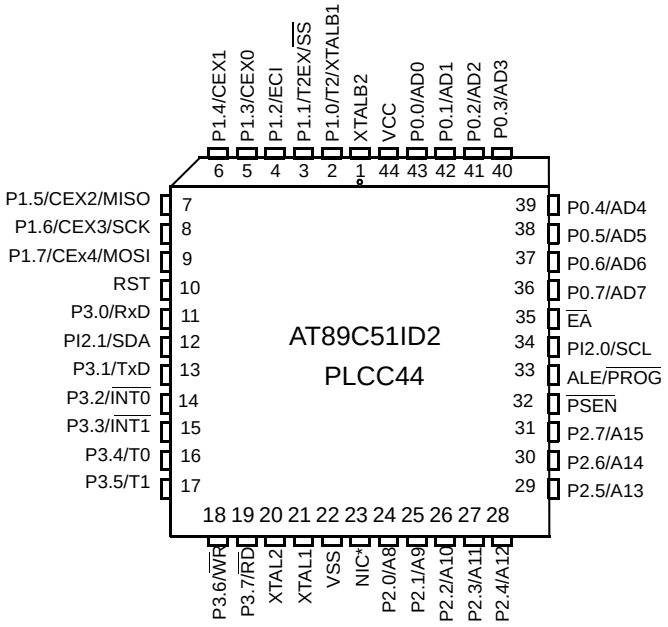
The AT89C51ID2 has 2 software-selectable modes of reduced activity and 8-bit clock prescaler for further reduction in power consumption. In the Idle mode the CPU is frozen while the peripherals and the interrupt system are still operating. In the power-down mode the RAM is saved and all other functions are inoperative.

The added features of the AT89C51ID2 make it more powerful for applications that need pulse width modulation, high speed I/O and counting capabilities such as alarms, motor control, corded phones, smart card readers.

Table 1. Memory Size and I/O pins

AT89C51ID2	Flash (bytes)	XRAM (bytes)	TOTAL RAM (bytes)	I/O
PLCC44/VQFP44	64K	1792	2048	34

Pin Configurations



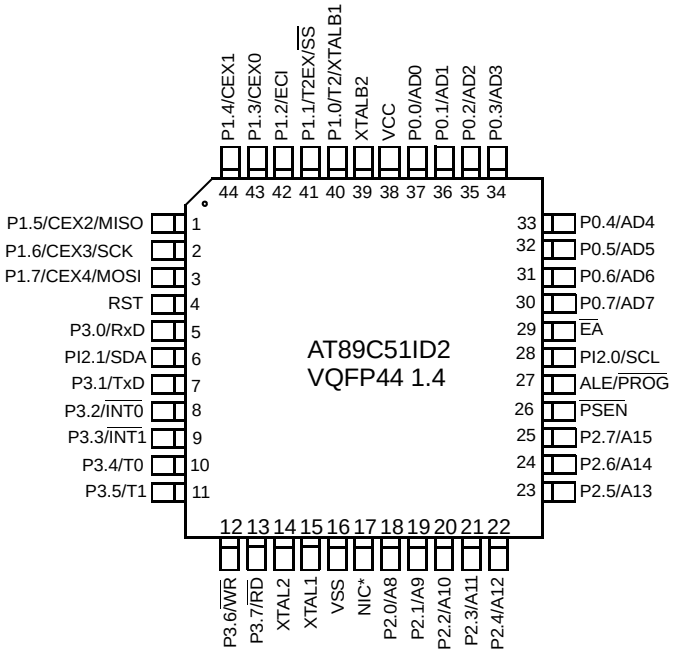


Table 15. Pin Description

Mnemonic	Pin Number PLCC44	Pin Number VQFP44	Type	Name and Function
V_{SS}	22	16	I	**Ground:** 0V reference
V_{CC}	44	38	I	**Power Supply:** This is the power supply voltage for normal, idle and power-down operation
P0.0 - P0.7	43 - 36	37 - 30	I/O	**Port 0**: Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high impedance inputs. Port 0 must be polarized to V_{CC} or V_{SS} in order to prevent any parasitic current consumption. Port 0 is also the multiplexed low-order address and data bus during access to external program and data memory. In this application, it uses strong internal pull-up when emitting 1s. Port 0 also inputs the code bytes during EPROM programming. External pull-ups are required during program verification during which P0 outputs the code bytes.
P1.0 - P1.7	2 - 9	40 - 44 1 - 3	I/O	**Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally pulled low will source current because of the internal pull-ups. Port 1 also receives the low-order address byte during memory programming and verification. Alternate functions for AT89C51ID2 Port 1 include:
	2	40	I/O	**P1.0**: Input/Output
			I/O	**T2 (P1.0):** Timer/Counter 2 external count input/Clockout
			I	**XTALB1 (P1.0):** Sub Clock input to the inverting oscillator amplifier
	3	41	I/O	**P1.1:** Input/Output
			I	**T2EX:** Timer/Counter 2 Reload/Capture/Direction Control
			I	$\overline{\textbf{SS}}$**:** SPI Slave Select
	4	42	I/O	**P1.2:** Input/Output
			I	**ECI:** External Clock for the PCA
	5	43	I/O	**P1.3:** Input/Output
			I/O	**CEX0:** Capture/Compare External I/O for PCA module 0
	6	44	I/O	**P1.4:** Input/Output
			I/O	**CEX1:** Capture/Compare External I/O for PCA module 1
	7	1	I/O	**P1.5:** Input/Output
			I/O	**CEX2:** Capture/Compare External I/O for PCA module 2
			I/O	**MISO:** SPI Master Input Slave Output line When SPI is in master mode, MISO receives data from the slave peripheral. When SPI is in slave mode, MISO outputs data to the master controller.
	8	2	I/O	**P1.6:** Input/Output
			I/O	**CEX3:** Capture/Compare External I/O for PCA module 3
			I/O	**SCK:** SPI Serial Clock
	9	3	I/O	**P1.7:** Input/Output:
			I/O	**CEX4:** Capture/Compare External I/O for PCA module 4

Table 20. Overview (Continued)

PCON.1	PCON.0	OscBEn	OscAEn	CKS	Selected Mode	Comment
0	1	1	X	0	IDLE MODE B	The CPU is off, OscB supplies the peripherals, OscA can be disabled (OscAEn = 0)
1	X	X	1	X	POWER DOWN MODE	The CPU and peripherals are off, OscA and OscB are stopped

Design Considerations

Oscillators Control

- PwdOscA and PwdOscB signals are generated in the Clock generator and used to control the hard blocks of oscillators A and B.
- PwdOscA ='1' stops OscA
- PwdOscB ='1' stops OscB
- The following tables summarize the Operating modes:

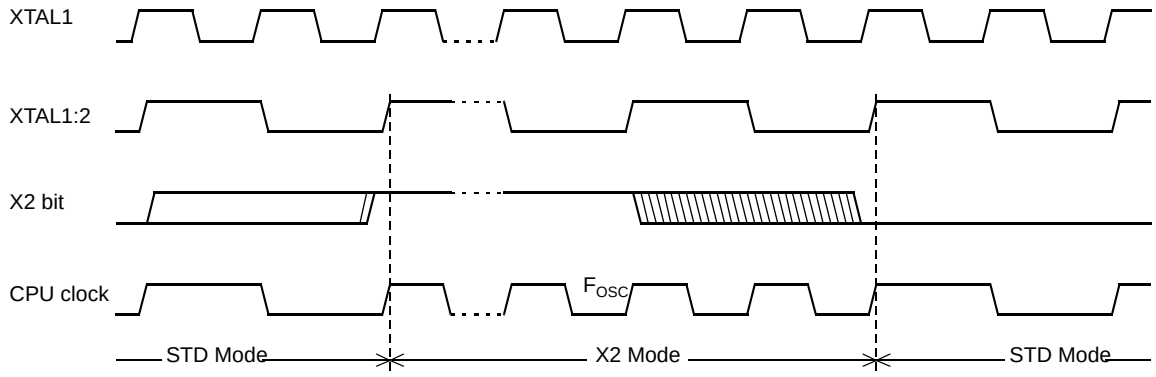
PCON.1	OscAEn	PwdOscA	Comments
0	1	0	OscA running
1	X	1	OscA stopped by Power-down mode
0	0	1	OscA stopped by clearing OscAEn

PCON.1	OscBEn	PwdOscB	Comments
0	1	0	OscB running
1	X	1	OscB stopped by Power-down mode
0	0	1	OscB stopped by clearing OscBEn

Prescaler Divider

- A hardware RESET puts the prescaler divider in the following state:
  - CKRL = FFh: $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSCA}/2$ (Standard C51 feature)
- CKS signal selects OSCA or OSCB: $F_{CLK\ OUT} = F_{OSCA}$ or F_{OSCB}
- Any value between FFh down to 00h can be written by software into CKRL register in order to divide frequency of the selected oscillator:
  - CKRL = 00h: minimum frequency
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSCA}/1020$ (Standard Mode)
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSCA}/510$ (X2 Mode)
  - CKRL = FFh: maximum frequency
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSCA}/2$ (Standard Mode)
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSCA}$ (X2 Mode)

Figure 4. Mode Switching Waveforms



The X2 bit in the CKCON0 register (see Table 21) allows a switch from 12 clock periods per instruction to 6 clock periods and vice versa. At reset, the speed is set according to X2 bit of Hardware Security Byte (HSB). By default, Standard mode is active. Setting the X2 bit activates the X2 feature (X2 mode).

The T0X2, T1X2, T2X2, UartX2, PcaX2, and WdX2 bits in the CKCON0 register (See Table 21.) and SPIX2 bit in the CKCON1 register (see Table 22) allows a switch from standard peripheral speed (12 clock periods per peripheral clock cycle) to fast peripheral speed (6 clock periods per peripheral clock cycle). These bits are active only in X2 mode.

Table 21. CKCON0 Register

CKCON0 - Clock Control Register (8Fh)

7	6	5	4	3	2	1	0
TWIX2	WDX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2

Bit Number	Bit Mnemonic	Description
7	TWIX2	**2-wire clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect) Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
6	WDX2	**Watchdog Clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
5	PCAX2	**Programmable Counter Array Clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
4	SIX2	**Enhanced UART Clock (Mode 0 and 2)** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
3	T2X2	**Timer2 Clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
2	T1X2	**Timer1 Clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
1	T0X2	**Timer0 Clock** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Cleared to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.
0	X2	**CPU Clock** Cleared to select 12 clock periods per machine cycle (STD mode) for CPU and all the peripherals. Set to select 6clock periods per machine cycle (X2 mode) and to enable the individual peripherals'X2' bits. Programmed by hardware after Power-up regarding Hardware Security Byte (HSB), Default setting, X2 is cleared.

Reset Value = 0000 000'HSB. X2'b (See "Hardware Security Byte")
Not bit addressable

Table 30. PCA Module Modes (CCAPMn Registers)

ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMm	ECCFn	Module Function
0	0	0	0	0	0	0	No Operation
X	1	0	0	0	0	X	16-bit capture by a positive-edge trigger on CEXn
X	0	1	0	0	0	X	16-bit capture by a negative trigger on CEXn
X	1	1	0	0	0	X	16-bit capture by a transition on CEXn
1	0	0	1	0	0	X	16-bit Software Timer / Compare mode.
1	0	0	1	1	0	X	16-bit High Speed Output
1	0	0	0	0	1	0	8-bit PWM
1	0	0	1	X	0	X	Watchdog Timer (module 4 only)

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode these registers are used to control the duty cycle of the output (See Table 31 & Table 32).

Table 31. CCAPnH Registers (n = 0-4)

CCAP0H - PCA Module 0 Compare/Capture Control Register High (0FAh)

CCAP1H - PCA Module 1 Compare/Capture Control Register High (0FBh)

CCAP2H - PCA Module 2 Compare/Capture Control Register High (0FCh)

CCAP3H - PCA Module 3 Compare/Capture Control Register High (0FDh)

CCAP4H - PCA Module 4 Compare/Capture Control Register High (0FEh)

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

Bit Number	Bit Mnemonic	Description
7-0	-	**PCA Module n Compare/Capture Control** CCAPnH Value

Reset Value = 0000 0000b
Not bit addressable

Table 32. CCAPnL Registers (n = 0-4)

CCAP0L - PCA Module 0 Compare/Capture Control Register Low (0EAh)

CCAP1L - PCA Module 1 Compare/Capture Control Register Low (0EBh)

CCAP2L - PCA Module 2 Compare/Capture Control Register Low (0ECh)

CCAP3L - PCA Module 3 Compare/Capture Control Register Low (0EDh)

CCAP4L - PCA Module 4 Compare/Capture Control Register Low (0EEh)

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

Bit Number	Bit Mnemonic	Description
7-0	-	**PCA Module n Compare/Capture Control** CCAPnL Value

Reset Value = 0000 0000b
Not bit addressable

Table 33. CH Register

CH - PCA Counter Register High (0F9h)

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

Bit Number	Bit Mnemonic	Description
7-0	-	**PCA counter** CH Value

Reset Value = 0000 0000b
Not bit addressable

Table 34. CL Register

CL - PCA Counter Register Low (0E9h)

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

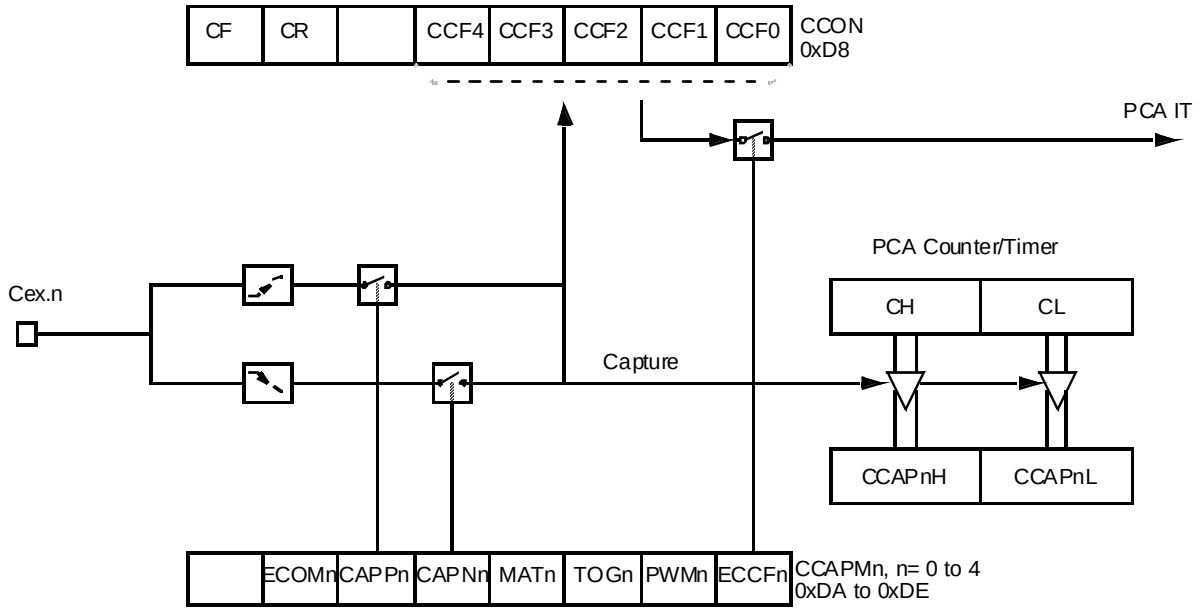
Bit Number	Bit Mnemonic	Description
7-0	-	**PCA Counter** CL Value

Reset Value = 0000 0000b
Not bit addressable

PCA Capture Mode

To use one of the PCA modules in the capture mode either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH). If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated (Refer to Figure 16).

Figure 16. PCA Capture Mode



16-bit Software Timer/ Compare Mode

The PCA modules can be used as software timers by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set (See Figure 17).

Serial I/O Port

The serial I/O port in the AT89C51ID2 is compatible with the serial I/O port in the 80C52. It provides both synchronous and asynchronous communication modes. It operates as a Universal Asynchronous Receiver and Transmitter (UART) in three full-duplex modes (Modes 1, 2 and 3). Asynchronous transmission and reception can occur simultaneously and at different baud rates

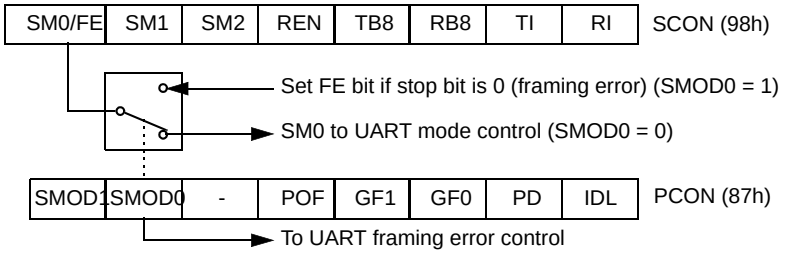
Serial I/O port includes the following enhancements:

- Framing error detection
- Automatic address recognition

Framing Error Detection

Framing bit error detection is provided for the three asynchronous modes (modes 1, 2 and 3). To enable the framing bit error detection feature, set SMOD0 bit in PCON register (See Figure 20).

Figure 20. Framing Error Block Diagram



When this feature is enabled, the receiver checks each incoming data frame for a valid stop bit. An invalid stop bit may result from noise on the serial lines or from simultaneous transmission by two CPUs. If a valid stop bit is not found, the Framing Error bit (FE) in SCON register (See Table 38.) bit is set.
Software may examine FE bit after each reception to check for data errors. Once set, only software or a reset can clear FE bit. Subsequently received frames with valid stop bits cannot clear FE bit. When FE feature is enabled, RI rises on stop bit instead of the last data bit (See Figure 21. and Figure 22.).

Figure 21. UART Timings in Mode 1

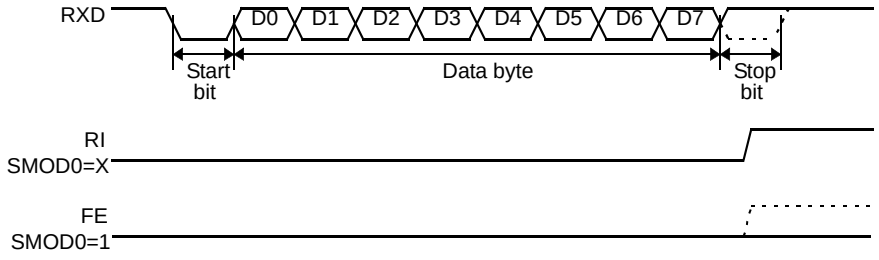


Table 38. SCON Register

SCON - Serial Control Register (98h)

7	6	5	4	3	2	1	0
FE/SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Bit Number	Bit Mnemonic	Description
7	FE	**Framing Error bit (SMOD0=1**) Clear to reset the error state, not cleared by a valid stop bit. Set by hardware when an invalid stop bit is detected. SMOD0 must be set to enable access to the FE bit.
	SM0	**Serial port Mode bit 0** Refer to SM1 for serial port mode selection. SMOD0 must be cleared to enable access to the SM0 bit.
6	SM1	**Serial port Mode bit 1** SM0 SM1 Mode Baud Rate 0 0 Shift Register $F_{XTAL}/12$ (or F_{XTAL} /6 in mode X2) 0 1 8-bit UART Variable 1 0 9-bit UART $F_{XTAL}/64$ or $F_{XTAL}/32$ 1 1 9-bit UART Variable
5	SM2	**Serial port Mode 2 bit / Multiprocessor Communication Enable bit** Clear to disable multiprocessor communication feature. Set to enable multiprocessor communication feature in mode 2 and 3, and eventually mode 1.This bit should be cleared in mode 0.
4	REN	**Reception Enable bit** Clear to disable serial reception. Set to enable serial reception.
3	TB8	**Transmitter Bit 8 / Ninth bit to transmit in modes 2 and 3** Clear to transmit a logic 0 in the 9th bit. Set to transmit a logic 1 in the 9th bit.
2	RB8	**Receiver Bit 8 / Ninth bit received in modes 2 and 3** Cleared by hardware if 9th bit received is a logic 0. Set by hardware if 9th bit received is a logic 1. In mode 1, if SM2 = 0, RB8 is the received stop bit. In mode 0 RB8 is not used.
1	TI	**Transmit Interrupt flag** Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0 or at the beginning of the stop bit in the other modes.
0	RI	**Receive Interrupt flag** Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0, see Figure 21. and Figure 22. in the other modes.

Reset Value = 0000 0000b
Bit addressable

Table 46. PCON Register

PCON - Power Control Register (87h)

7	6	5	4	3	2	1	0
SMOD1	**SMOD0**	**-**	**POF**	**GF1**	**GF0**	**PD**	**IDL**

Bit Number	Bit Mnemonic	Description
7	SMOD1	**Serial port Mode bit 1 for UART** Set to select double baud rate in mode 1, 2 or 3.
6	SMOD0	**Serial port Mode bit 0 for UART** Cleared to select SM0 bit in SCON register. Set to select FE bit in SCON register.
5	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
4	POF	**Power-Off Flag** Cleared to recognize next reset type. Set by hardware when VCC rises from 0 to its nominal voltage. Can also be set by software.
3	GF1	**General purpose Flag** Cleared by user for general purpose usage. Set by user for general purpose usage.
2	GF0	**General purpose Flag** Cleared by user for general purpose usage. Set by user for general purpose usage.
1	PD	**Power-Down mode bit** Cleared by hardware when reset occurs. Set to enter power-down mode.
0	IDL	**Idle mode bit** Cleared by hardware when interrupt or reset occurs. Set to enter idle mode.

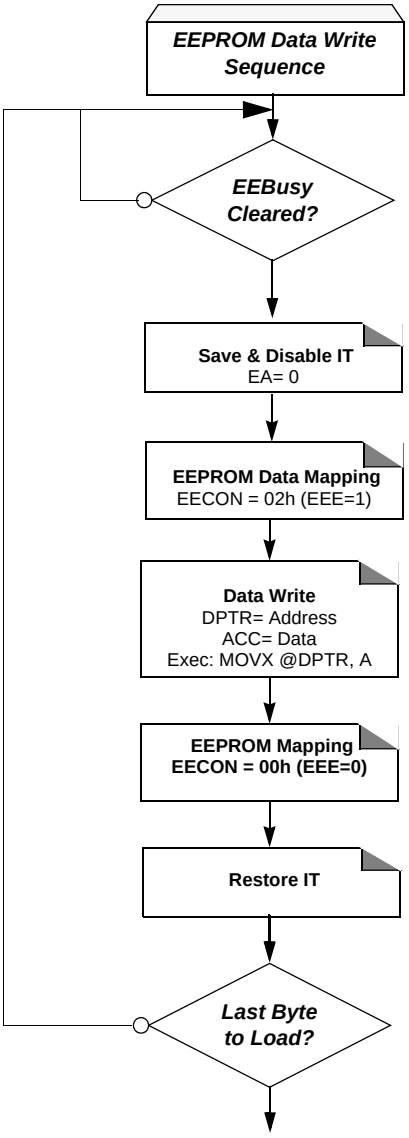
Reset Value = 00X1 0000b
Not bit addressable

Power-off flag reset value will be 1 only after a power on (cold reset). A warm reset doesn't affect the value of this bit.

Table 68. Status in Slave Receiver Mode

Status Code (SSCS)	Status of the 2-wire bus and 2-wire hardware	Application Software Response					Next Action Taken By 2-wire Software
		To/from SSDAT	To SSCON				
			STA	STO	SI	AA	
60h	Own SLA+W has been received; ACK has been returned	No SSDAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		No SSDAT action	X	0	0	1	Data byte will be received and ACK will be returned
68h	Arbitration lost in SLA+R/W as master; own SLA+W has been received; ACK has been returned	No SSDAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		No SSDAT action	X	0	0	1	Data byte will be received and ACK will be returned
70h	General call address has been received; ACK has been returned	No SSDAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		No SSDAT action	X	0	0	1	Data byte will be received and ACK will be returned
78h	Arbitration lost in SLA+R/W as master; general call address has been received; ACK has been returned	No SSDAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		No SSDAT action	X	0	0	1	Data byte will be received and ACK will be returned
80h	Previously addressed with own SLA+W; data has been received; ACK has been returned	No SSDAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		No SSDAT action	X	0	0	1	Data byte will be received and ACK will be returned
88h	Previously addressed with own SLA+W; data has been received; NOT ACK has been returned	Read data byte or	0	0	0	0	Switched to the not addressed slave mode; no recognition of own SLA or GCA
		Read data byte or	0	0	0	1	Switched to the not addressed slave mode; own SLA will be recognised; GCA will be recognised if GC=logic 1
		Read data byte or	1	0	0	0	Switched to the not addressed slave mode; no recognition of own SLA or GCA. A START condition will be transmitted when the bus becomes free
		Read data byte	1	0	0	1	Switched to the not addressed slave mode; own SLA will be recognised; GCA will be recognised if GC=logic 1. A START condition will be transmitted when the bus becomes free
90h	Previously addressed with general call; data has been received; ACK has been returned	Read data byte or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		Read data byte	X	0	0	1	Data byte will be received and ACK will be returned

Figure 43. Recommended EEPROM Data Write Sequence



Bootloader Functionality

Introduction

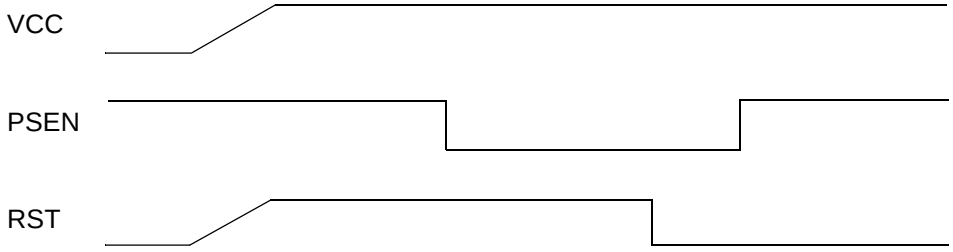
The bootloader can be activated by two means: Hardware conditions or regular boot process.

The Hardware conditions (EA = 1, PSEN = 0) during the Reset# falling edge force the on-chip bootloader execution. This allows an application to be built that will normally execute the end user's code but can be manually forced into default ISP operation.

As PSEN is a an output port in normal operating mode after reset, user application should take care to release PSEN after falling edge of reset signal. The hardware conditions are sampled at reset signal falling edge, thus they can be released at any time when reset input is low.

To ensure correct microcontroller startup, the PSEN pin should not be tied to ground during power-on (See Figure 48).

Figure 48. Hardware conditions typical sequence during power-on.



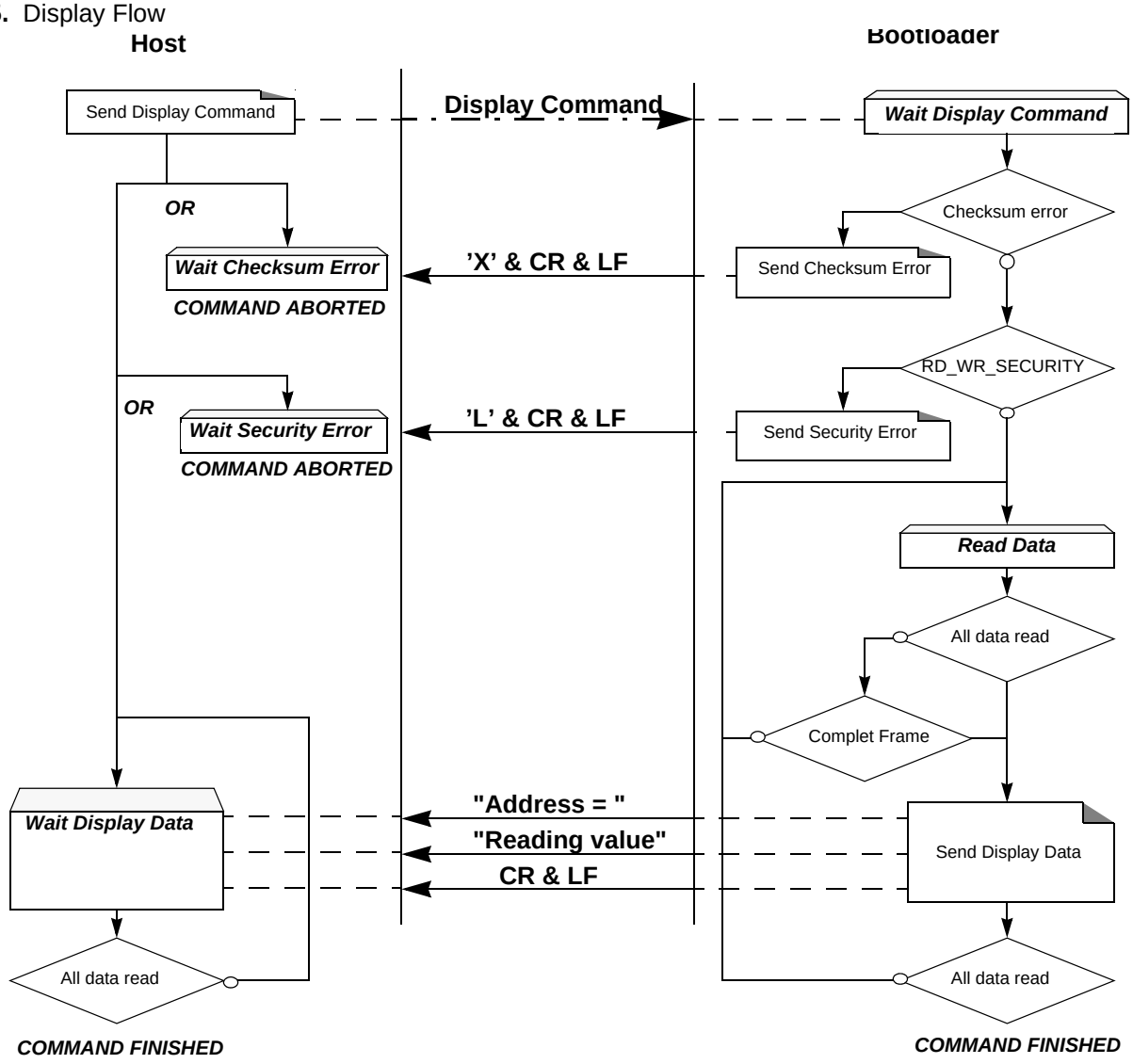
The on-chip bootloader boot process is shown Figure 49

	Purpose
Hardware Conditions	**The Hardware Conditions force the bootloader execution whatever BLJB, BSB and SBV values.**
BLJB	**The Boot Loader Jump Bit forces the application execution.** **BLJB = 0 => Boot loader execution.** **BLJB = 1 => Application execution** **The BLJB is a fuse bit in the Hardware Byte.** **That can be modified by hardware (programmer) or by software (API).** **Note:** **The BLJB test is perform by hardware to prevent any program execution..**

Display Data

Description

Figure 55. Display Flow



Note: The maximum size of block is 400h. To read more than 400h bytes, the Host must send a new command.

ISP Commands Summary

Table 96. ISP Commands Summary

Command	Command Name	data[0]	data[1]	Command Effect
00h	Program Data			Program Nb Data Byte. Bootloader will accept up to 128 (80h) data bytes. The data bytes should be 128 byte page flash boundary.
03h	Write Function	01h	00h	Erase block0 (0000h-1FFFh)
			20h	Erase block1 (2000h-3FFFh)
			40h	Erase block2 (4000h-7FFFh)
			80h	Erase block3 (8000h- BFFFh)
			C0h	Erase block4 (C000h- FFFFh)
		03h	00h	Hardware Reset
		04h	00h	Erase SBV & BSB
		05h	00h	Program SSB level 1
			01h	Program SSB level 2
		06h	00h	Program BSB (value to write in data[2])
			01h	Program SBV (value to write in data[2])
		07h	-	Full Chip Erase (This command needs about 6 sec to be executed)
		0Ah	02h	Program Osc fuse (value to write in data[2])
			04h	Program BLJB fuse (value to write in data[2])
			08h	Program X2 fuse (value to write in data[2])
04h	Display Function	Data[0:1] = start address Data [2:3] = end address Data[4] = 00h -> Display data Data[4] = 01h -> Blank check Data[4] = 02h -> Display EEPROMk		Display Data
				Blank Check
				Display EEPROM data

API Call Description

Several Application Program Interface (API) calls are available for use by an application program to permit selective erasing and programming of Flash pages. All calls are made through a common interface, PGM_MTP. The programming functions are selected by setting up the microcontroller's registers before making a call to PGM_MTP at FFF0h. Results are returned in the registers.

When several bytes have to be programmed, it is highly recommended to use the Atmel API "PROGRAM DATA PAGE" call. Indeed, this API call writes up to 128 bytes in a single command.

All routines for software access are provided in the C Flash driver available on Atmel web site.

The API calls description and arguments are shown in Table

Table 97. API Call Summary

Command	R1	A	DPTR0	DPTR1	Returned Value	Command Effect
READ MANUF ID	00h	XXh	0000h	XXh	ACC = Manufacturer Id	Read Manufacturer identifier
READ DEVICE ID1	00h	XXh	0001h	XXh	ACC = Device Id 1	Read Device identifier 1
READ DEVICE ID2	00h	XXh	0002h	XXh	ACC = Device Id 2	Read Device identifier 2
READ DEVICE ID3	00h	XXh	0003h	XXh	ACC = Device Id 3	Read Device identifier 3
ERASE BLOCK	01h	XXh	DPH = 00h	00h	ACC = DPH	Erase block 0
			DPH = 20h			Erase block 1
			DPH = 40h			Erase block 2
			Address of byte to program			Program one Data Byte in user Flash
			XXh			Erase Software boot vector and boot status byte. (SBV = FCh and BSB = FFh)
PROGRAM SSB	05h	XXh	DPH = 00h DPL = 00h	00h	ACC = SSB value	Set SSB level 1
			DPH = 00h DPL = 01h			Set SSB level 2
			DPH = 00h DPL = 10h			Set SSB level 0
			DPH = 00h DPL = 11h			Set SSB level 1
PROGRAM BSB	06h	New BSB value	0000h	XXh	none	Program boot status byte
PROGRAM SBV	06h	New SBV value	0001h	XXh	none	Program software boot vector
READ SSB	07h	XXh	0000h	XXh	ACC = SSB	Read Software Security Byte
READ BSB	07h	XXh	0001h	XXh	ACC = BSB	Read Boot Status Byte
READ SBV	07h	XXh	0002h	XXh	ACC = SBV	Read Software Boot Vector

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.

(Load Capacitance for port 0, ALE and PSEN = 100 pF; Load Capacitance for all other outputs = 80 pF.)

Table 98 Table 101, and Table 104 give the description of each AC symbols.

Table 99, Table 100, Table 102 and Table 105 gives the range for each AC parameter.

Table 99, Table 100 and Table 106 give the frequency derating formula of the AC parameter for each speed range description. To calculate each AC symbols. take the x value in the correponding column (-M or -L) and use this value in the formula.

Example: T_{LLIU} for -M and 20 MHz, Standard clock.
x = 35 ns
T 50 ns
T_{CCIV} = 4T - x = 165 ns

External Program Memory Characteristics

Table 98. Symbol Description

Symbol	Parameter
T	Oscillator clock period
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