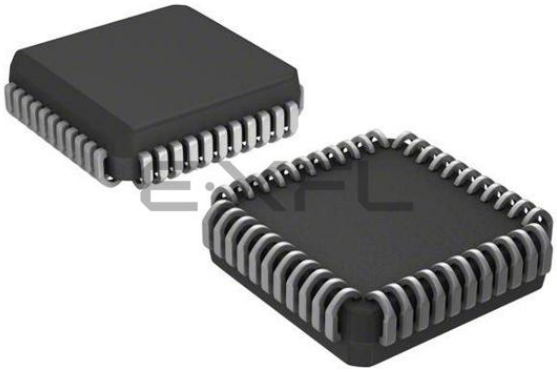


Welcome to E-XFL.COM

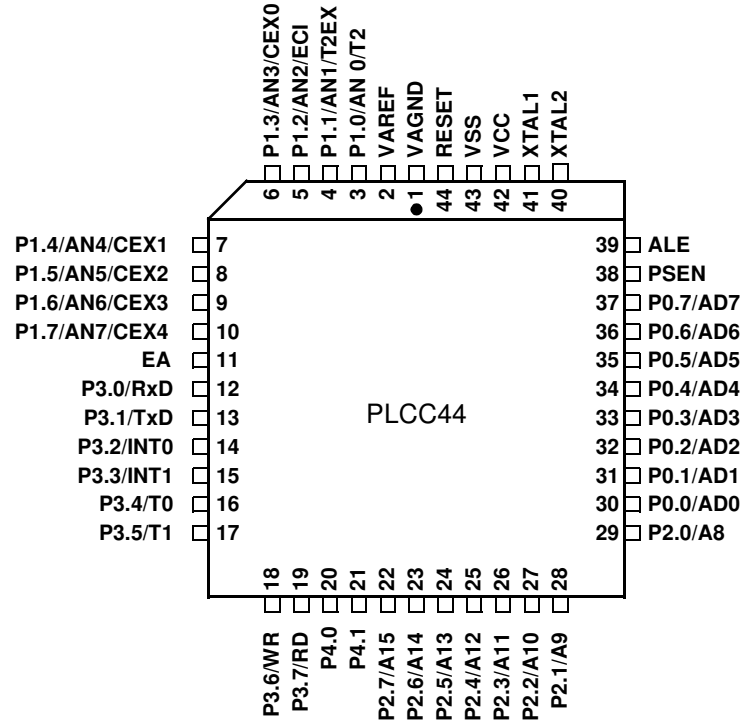
What is "Embedded - Microcontrollers"?

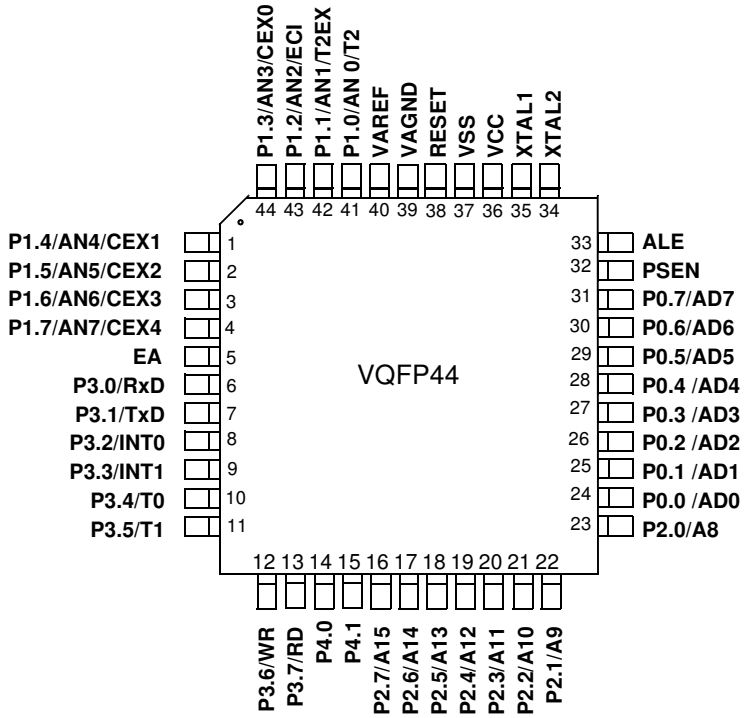
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	36
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
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/at89c51ac3-slsum

Pin Configuration





Pin Name	Type	Description
P3.0:7	I/O	**Port 3:** Is an 8-bit bi-directional I/O port with internal pull-ups. Port 3 pins that have 1's written to them are pulled high by the internal pull-up transistors and can be used as inputs in this state. As inputs, Port 3 pins that are being pulled low externally will be a source of current (I_{IL}, see section "Electrical Characteristic") because of the internal pull-ups. The output latch corresponding to a secondary function must be programmed to one for that function to operate (except for TxD and $\overline{WR}$). The secondary functions are assigned to the pins of port 3 as follows: P3.0/RxD: Receiver data input (asynchronous) or data input/output (synchronous) of the serial interface P3.1/TxD: Transmitter data output (asynchronous) or clock output (synchronous) of the serial interface P3.2/$\overline{INT0}$: External interrupt 0 input/timer 0 gate control input P3.3/$\overline{INT1}$: External interrupt 1 input/timer 1 gate control input P3.4/T0: Timer 0 counter input P3.5/T1/$\overline{SS}$: Timer 1 counter input SPI Slave Select P3.6/$\overline{WR}$: External Data Memory write strobe; latches the data byte from port 0 into the external data memory P3.7/$\overline{RD}$: External Data Memory read strobe; Enables the external data memory. It can drive CMOS inputs without external pull-ups.
P4.0:4	I/O	**Port 4:** Is an 2-bit bi-directional I/O port with internal pull-ups. Port 4 pins that have 1's written to them are pulled high by the internal pull-ups and can be used as inputs in this state. As inputs, Port 4 pins that are being pulled low externally will be a source of current (IIL, on the datasheet) because of the internal pull-up transistor. The secondary functions are assigned to the 5 pins of port 4 as follows: P4.0: Regular Port I/O P4.1: Regular Port I/O P4.2/MISO: Master Input Slave Output of SPI controller P4.3/SCK: Serial Clock of SPI controller P4.4/MOSI: Master Ouput Slave Input of SPI controller It can drive CMOS inputs without external pull-ups.

write the new byte back to the latch. These Read-Modify-Write instructions are directed to the latch rather than the pin in order to avoid possible misinterpretation of voltage (and therefore, logic) levels at the pin. For example, a Port bit used to drive the base of an external bipolar transistor can not rise above the transistor's base-emitter junction voltage (a value lower than VIL). With a logic one written to the bit, attempts by the CPU to read the Port at the pin are misinterpreted as logic zero. A read of the latch rather than the pins returns the correct logic-one value.

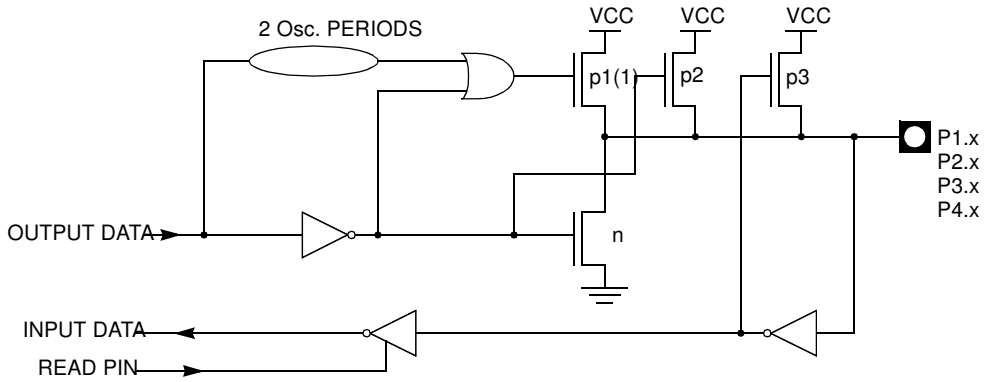
Quasi-Bidirectional Port Operation

Port 1, Port 2, Port 3 and Port 4 have fixed internal pull-ups and are referred to as "quasi-bidirectional" Ports. When configured as an input, the pin impedance appears as logic one and sources current in response to an external logic zero condition. Port 0 is a "true bidirectional" pin. The pins float when configured as input. Resets write logic one to all Port latches. If logical zero is subsequently written to a Port latch, it can be returned to input conditions by a logical one written to the latch.

Note: Port latch values change near the end of Read-Modify-Write instruction cycles. Output buffers (and therefore the pin state) update early in the instruction after Read-Modify-Write instruction cycle.

Logical zero-to-one transitions in Port 1, Port 2, Port 3 and Port 4 use an additional pull-up (p1) to aid this logic transition (see Figure 4.). This increases switch speed. This extra pull-up sources 100 times normal internal circuit current during 2 oscillator clock periods. The internal pull-ups are field-effect transistors rather than linear resistors. Pull-ups consist of three p-channel FET (pFET) devices. A pFET is on when the gate senses logical zero and off when the gate senses logical one. pFET #1 is turned on for two oscillator periods immediately after a zero-to-one transition in the Port latch. A logical one at the Port pin turns on pFET #3 (a weak pull-up) through the inverter. This inverter and pFET pair form a latch to drive logical one. pFET #2 is a very weak pull-up switched on whenever the associated nFET is switched off. This is traditional CMOS switch convention. Current strengths are 1/10 that of pFET #3.

Figure 4. Internal Pull-Up Configurations



Note: Port 2 p1 assists the logic-one output for memory bus cycles.

Clock

The AT89C51AC3 core needs only 6 clock periods per machine cycle. This feature, called"X2", provides the following advantages:

- Divides frequency crystals by 2 (cheaper crystals) while keeping the same CPU power.
- Saves power consumption while keeping the same CPU power (oscillator power saving).
- Saves power consumption by dividing dynamic operating frequency by 2 in operating and idle modes.
- Increases CPU power by 2 while keeping the same crystal frequency.

In order to keep the original C51 compatibility, a divider-by-2 is inserted between the XTAL1 signal and the main clock input of the core (phase generator). This divider may be disabled by the software.

An extra feature is available to start after Reset in the X2 mode. This feature can be enabled by a bit X2B in the Hardware Security Byte. This bit is described in the section "In-System Programming".

Description

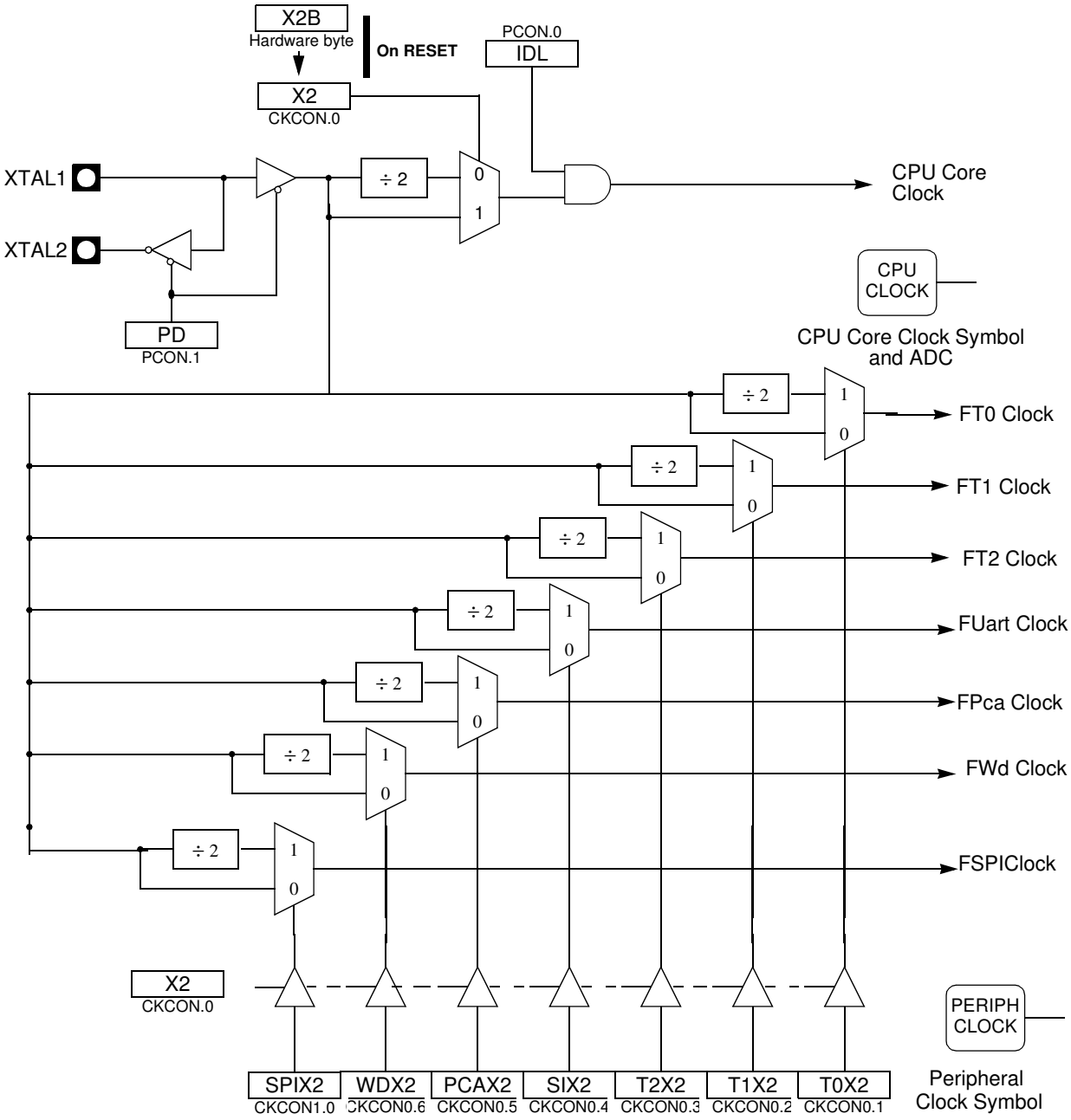
The X2 bit in the CKCON register (see Table 2) allows switching from 12 clock cycles per instruction to 6 clock cycles and vice versa. At reset, the standard speed is activated (STD mode).

Setting this bit activates the X2 feature (X2 mode) for the CPU Clock only (see Figure 5.).

The Timers 0, 1 and 2, Uart, PCA or WatchDog switch in X2 mode only if the corresponding bit is cleared in the CKCON register.

The clock for the whole circuit and peripheral is first divided by two before being used by the CPU core and peripherals. This allows any cyclic ratio to be accepted on the XTAL1 input. In X2 mode, as this divider is bypassed, the signals on XTAL1 must have a cyclic ratio between 40 to 60%. Figure 5. shows the clock generation block diagram. The X2 bit is validated on the XTAL1÷2 rising edge to avoid glitches when switching from the X2 to the STD mode. Figure 6 shows the mode switching waveforms.

Figure 5. Clock CPU Generation Diagram



Registers

Table 6. PSW Register

PSW (S:8Eh)
Program Status Word Register

7	6	5	4	3	2	1	0
CY	**AC**	**F0**	**RS1**	**RS0**	**OV**	**F1**	**P**

Bit Number	Bit Mnemonic	Description
7	CY	**Carry Flag** Carry out from bit 1 of ALU operands.
6	AC	**Auxiliary Carry Flag** Carry out from bit 1 of addition operands.
5	F0	**User Definable Flag 0.**
4-3	RS1:0	**Register Bank Select Bits** Refer to Table 4 for bits description.
2	OV	**Overflow Flag** Overflow set by arithmetic operations.
1	F1	**User Definable Flag 1**
0	P	**Parity Bit** Set when ACC contains an odd number of 1's. Cleared when ACC contains an even number of 1's.

Reset Value = 0000 0000b

Table 7. AUXR Register

AUXR (S:8Eh)
Auxiliary Register

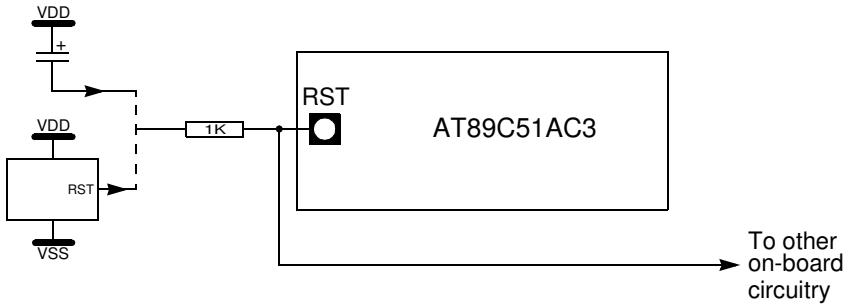
7	6	5	4	3	2	1	0
-	**-**	**M0**	**XRS2**	**XRS1**	**XRS0**	**EXTRAM**	**A0**

Bit Number	Bit Mnemonic	Description
7-6	-	**Reserved** The value read from these bits are indeterminate. Do not set this bit.
5	M0	**Stretch MOVX control:** the RD/ and the WR/ pulse length is increased according to the value of M0. **M0** **Pulse length in clock period** 0 6 1 30

Reset Output

As detailed in Section “Watchdog Timer”, page 79, the WDT generates a 96-clock period pulse on the RST pin. In order to properly propagate this pulse to the rest of the application in case of external capacitor or power-supply supervisor circuit, a 1 kΩ resistor must be added as shown Figure 18.

Figure 18. Recommended Reset Output Schematic



Power Management

Introduction

Two power reduction modes are implemented in the AT89C51AC3. The Idle mode and the Power-Down mode. These modes are detailed in the following sections. In addition to these power reduction modes, the clocks of the core and peripherals can be dynamically divided by 2 using the X2 mode detailed in Section “Clock”, page 15.

Idle Mode

Idle mode is a power reduction mode that reduces the power consumption. In this mode, program execution halts. Idle mode freezes the clock to the CPU at known states while the peripherals continue to be clocked. The CPU status before entering Idle mode is preserved, i.e., the program counter and program status word register retain their data for the duration of Idle mode. The contents of the SFRs and RAM are also retained. The status of the Port pins during Idle mode is detailed in Table 9.

Entering Idle Mode

To enter Idle mode, set the IDL bit in PCON register (see Table 10). The AT89C51AC3 enters Idle mode upon execution of the instruction that sets IDL bit. The instruction that sets IDL bit is the last instruction executed.

Note: If IDL bit and PD bit are set simultaneously, the AT89C51AC3 enters Power-Down mode. Then it does not go in Idle mode when exiting Power-Down mode.

Exiting Idle Mode

There are two ways to exit Idle mode:

1. Generate an enabled interrupt.
   - Hardware clears IDL bit in PCON register which restores the clock to the CPU. Execution resumes with the interrupt service routine. Upon completion of the interrupt service routine, program execution resumes with the instruction immediately following the instruction that activated Idle mode. The general purpose flags (GF1 and GF0 in PCON register) may be used to indicate whether an interrupt occurred during normal operation or during Idle mode. When Idle mode is exited by an interrupt, the interrupt service routine may examine GF1 and GF0.
2. Generate a reset.
   - A logic high on the RST pin clears IDL bit in PCON register directly and asynchronously. This restores the clock to the CPU. Program execution momentarily resumes with the instruction immediately following the instruction that activated the Idle mode and may continue for a number of clock cycles before the internal reset algorithm takes control. Reset initializes the AT89C51AC3 and vectors the CPU to address C:0000h.

Note: During the time that execution resumes, the internal RAM cannot be accessed; however, it is possible for the Port pins to be accessed. To avoid unexpected outputs at the Port pins, the instruction immediately following the instruction that activated Idle mode should not write to a Port pin or to the external RAM.

Power-Down Mode

The Power-Down mode places the AT89C51AC3 in a very low power state. Power-Down mode stops the oscillator, freezes all clock at known states. The CPU status prior to entering Power-Down mode is preserved, i.e., the program counter, program status word register retain their data for the duration of Power-Down mode. In addition, the SFR and RAM contents are preserved. The status of the Port pins during Power-Down mode is detailed in Table 9.

Note: VCC may be reduced to as low as V_{RET} during Power-Down mode to further reduce power dissipation. Take care, however, that VDD is not reduced until Power-Down mode is invoked.

Table 9. Pin Conditions in Special Operating Modes

Mode	Port 0	Port 1	Port 2	Port 3	Port 4	ALE	PSEN#
Reset	Floating	High	High	High	High	High	High
Idle (internal code)	Data	Data	Data	Data	Data	High	High
Idle (external code)	Floating	Data	Data	Data	Data	High	High
Power-Down(internal code)	Data	Data	Data	Data	Data	Low	Low
Power-Down (external code)	Floating	Data	Data	Data	Data	Low	Low

Cross Flash Memory Access

Code executing from		Action	FM0 (user Flash)	FM1 (boot Flash)
	FM0 (user Flash)	Read	ok	-
		Load column latch	ok	-
		Write	-	-
	FM1 (boot Flash)	Read	ok	ok
		Load column latch	ok	-
		Write	ok	-
	External memory EA = 0	Read	(a)	-
		Load column latch	-	-
		Write	-	-

(a) Depend upon general lock bit configuration.

FSTA Register

Table 14. FSTA Register

FSTA Register (S:D3h)
Flash Status Register

7	6	5	4	3	2	1	0
						SEQERR	FLOAD

Bit Number	Bit Mnemonic	Description
7-2		**unusesd**
1	SEQERR	**Flash activation sequence error** Set by hardware when the flash activation sequence(MOV FCON 5X and MOV FCON AX)is not correct (See Error Repport Section) Clear by software or clear by hardware if the last activation sequence was correct (previous error are canceled)
0	FLOAD	**Flash Colums latch loaded** Set by hardware when the first data is loaded in the column latches. Clear by hardware when the activation sequence suceed (flash write sucess, or reset column latch success)

Reset Value= 0000 0000b

Mapping of the Memory Space

By default, the user space is accessed by MOVC A, @DPTR instruction for read only. The column latches space is made accessible by setting the FPS bit in FCON register. Writing is possible from 0000h to FFFFh, address bits 6 to 0 are used to select an address within a page while bits 15 to 7 are used to select the programming address of the page.
Setting FPS bit takes precedence on the EXTRAM bit in AUXR register.

The other memory spaces (user, extra row, hardware security) are made accessible in the code segment by programming bits FMOD0 and FMOD1 in FCON register in accordance with Table 15. A MOVC instruction is then used for reading these spaces.

Table 15. FM0 Blocks Select Bits

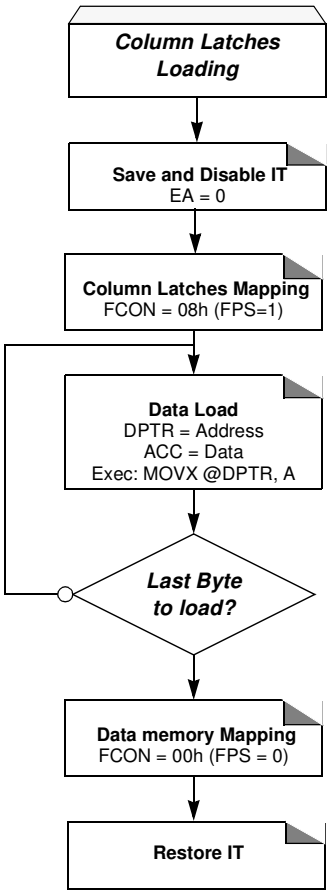
FMOD1	FMOD0	FM0 Adressable space
0	0	User (0000h-FFFFh)
0	1	Extra Row(FF80h-FFFFh)
1	0	Hardware Security Byte (0000h)
1	1	Column latches reset (note1)

Notes: 1. The column latches reset is a new option introduced in the AT89C51AC3, and is not available in T89C51CC01/2

Launching Programming

FPL3:0 bits in FCON register are used to secure the launch of programming. A specific sequence must be written in these bits to unlock the write protection and to launch the programming. This sequence is 5xh followed by Axh. Table 16 summarizes the memory spaces to program according to FMOD1:0 bits.

Figure 25. Column Latches Loading Procedure



Note: The last page address used when loading the column latch is the one used to select the page programming address.

Programming the Flash Spaces

User

The following procedure is used to program the User space and is summarized in Figure 26:

- Load up to one page of data in the column latches from address 0000h to FFFFh.
- Save and Disable the interrupts.
- Launch the programming by writing the data sequence 50h followed by A0h in FCON register (only from FM1).
  The end of the programming indicated by the FBUSY flag cleared.
- Restore the interrupts.

Extra Row

The following procedure is used to program the Extra Row space and is summarized in Figure 26:

- Load data in the column latches from address FF80h to FFFFh.
- Save and Disable the interrupts.
- Launch the programming by writing the data sequence 52h followed by A2h in FCON register (only from FM1).
  The end of the programming indicated by the FBUSY flag cleared.
- Restore the interrupts.

Table 17. Program Lock Bit

Program Lock Bits				
Security level	LB0	LB1	LB2	**Protection Description**
1	U	U	U	No program lock features enabled.
2	P	U	U	MOVC instruction executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further parallel programming of the Flash is disabled. ISP and software programming with API are still allowed. Writing EEprom Data from external parallel programmer is disabled but still allowed from internal code execution.
3	U	P	U	Same as 2, also verify through parallel programming interface is disabled. Writing And Reading EEPROM Data from external parallel programmer is disabled but still allowed from internal code execution..
4	U	U	P	Same as 3, also external execution is disabled

Program Lock bits

U: unprogrammed

P: programmed

WARNING: Security level 2 and 3 should only be programmed after Flash and Core verification.

Hardware Security Byte

Table 24. Hardware Security Byte

7	6	5	4	3	2	1	0
X2B	**BLJB**	**-**	**-**	**-**	**LB2**	**LB1**	**LB0**

Bit Number	Bit Mnemonic	Description
7	X2B	**X2 Bit** Set this bit to start in standard mode Clear this bit to start in X2 mode.
6	BLJB	**Boot Loader JumpBit** - 1: To start the user's application on next RESET (@0000h) located in FM0, - 0: To start the boot loader(@F800h) located in FM1.
5-3	-	**Reserved** The value read from these bits are indeterminate.
2-0	LB2:0	**Lock Bits**

Default value after erasing chip: FFh

Notes: 1. Only the 4 MSB bits can be accessed by software.
2. The 4 LSB bits can only be accessed by parallel mode.

In a Master configuration, the $\overline{SS}$ line can be used in conjunction with the MODF flag in the SPI Status register (SPSCR) to prevent multiple masters from driving MOSI and SCK (see Error conditions).

A high level on the $\overline{SS}$ pin puts the MISO line of a Slave SPI in a high-impedance state.

The $\overline{SS}$ pin could be used as a general-purpose if the following conditions are met:

- The device is configured as a Master and the SSDIS control bit in SPCON is set. This kind of configuration can be found when only one Master is driving the network and there is no way that the $\overline{SS}$ pin could be pulled low. Therefore, the MODF flag in the SPSCR will never be set(1).
- The Device is configured as a Slave with CPHA and SSDIS control bits set(2). This kind of configuration can happen when the system includes one Master and one Slave only. Therefore, the device should always be selected and there is no reason that the Master uses the $\overline{SS}$ pin to select the communicating Slave device.

Note: 1. Clearing SSDIS control bit does not clear MODF.

2. Special care should be taken not to set SSDIS control bit when CPHA ='0' because in this mode, the $\overline{SS}$ is used to start the transmission.

Baud Rate

In Master mode, the baud rate can be selected from a baud rate generator which is controlled by three bits in the SPCON register: SPR2, SPR1 and SPR0.The Master clock is selected from one of seven clock rates resulting from the division of the internal clock by 4, 8, 16, 32, 64 or 128.

Table 46 gives the different clock rates selected by SPR2:SPR1:SPR0.

In Slave mode, the maximum baud rate allowed on the SCK input is limited to $F_{sys}/4$

Table 46. SPI Master Baud Rate Selection

SPR2	SPR1	SPR0	Clock Rate	Baud Rate Divisor (BD)
0	0	0	Don't Use	No BRG
0	0	1	$F_{CLK\ PERIPH}$ /4	4
0	1	0	$F_{CLK\ PERIPH}$/8	8
0	1	1	$F_{CLK\ PERIPH}$ /16	16
1	0	0	$F_{CLK\ PERIPH}$ /32	32
1	0	1	$F_{CLK\ PERIPH}$ /64	64
1	1	0	$F_{CLK\ PERIPH}$ /128	128
1	1	1	Don't Use	No BRG

Error Conditions

The following flags in the SPSCR register indicate the SPI error conditions:

Mode Fault Error (MODF)

Mode Fault error in Master mode SPI indicates that the level on the Slave Select ($\overline{SS}$) pin is inconsistent with the actual mode of the device.

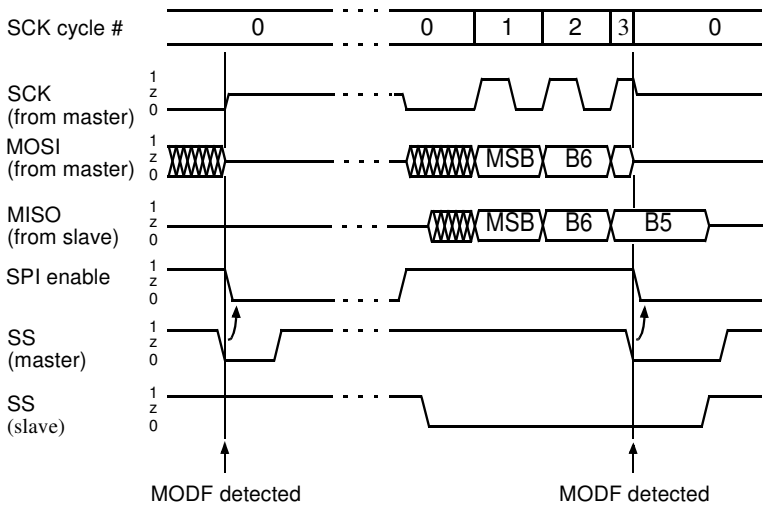
- Mode fault detection in Master mode:

MODF is set to warn that there may be 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 disables the SPI
  – The MSTR bit in SPCON is cleared

Clearing the MODF bit is accomplished by a read of SPSCR 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.

Figure 50. Mode Fault Conditions in Master Mode (Cpha ='1'/Cpol ='0')



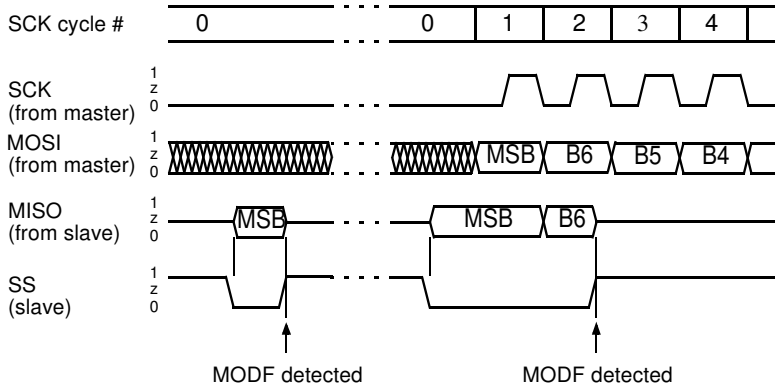
Note: When SS is discarded (SS disabled) it is not possible to detect a MODF error in master mode because the SPI is internally unselected and the SS pin is a general purpose I/O.

- Mode fault detection in Slave mode

In slave mode, the MODF error is detected when SS goes high during a transmission.
A transmission begins when SS goes low and ends once the incoming SCK goes back to its idle level following the shift of the eighteen data bit.
A MODF error occurs if a slave is selected (SS is low) and later unselected (SS is high) even if no SCK is sent to that slave.

At any time, a '1' on the SS pin of a slave SPI puts the MISO pin in a high impedance state and internal state counter is cleared. Also, the slave SPI ignores all incoming SCK clocks, even if it was already in the middle of a transmission. A new transmission will be performed as soon as SS pin returns low.

Figure 51. Mode Fault Conditions in Slave Mode



Note: when SS is discarded (SS disabled) it is not possible to detect a MODF error in slave mode because the SPI is internally selected. Also the SS pin becomes a general purpose I/O.

OverRun Condition

This error mean that the speed is not adapted for the running application:

An OverRun condition occurs when a byte has been received whereas the previous one has not been read by the application yet.
The last byte (which generate the overrun error) does not overwrite the unread data so that it can still be read. Therefore, an overrun error always indicates the loss of data.

Interrupts

Three SPI status flags can generate a CPU interrupt requests:

Table 47. SPI Interrupts

Flag	Request
SPIF (SPI data transfer)	SPI Transmitter Interrupt Request
MODF (Mode Fault)	SPI mode-fault Interrupt Request
SPTE (Transmit register empty)	SPI transmit register empty Interrupt Request

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

Mode Fault flag, MODF: This bit is set to indicate that the level on the $\overline{SS}$ is inconsistent with the mode of the SPI (in both master and slave modes).

Serial Peripheral Transmit Register empty flag, SPTE: This bit is set when the transmit buffer is empty (other data can be loaded is SPDAT). SPTE bit generates transmitter CPU interrupt request only when SPTEIE is enabled.

Note: While using SPTE interruption for “burst mode” transfers (SPTEIE=’1’), the user software application should take care to clear SPTEIE, during the last but one data reception (to be able to generate an interrupt on SPIF flag at the end of the last data reception).

Bit Number	Bit Mnemonic	Description
3	CPOL	**Clock Polarity** Cleared to have the SCK set to '0' in idle state. Set to have the SCK set to '1' in idle state.
2	CPHA	**Clock Phase** Cleared to have the data sampled when the SCK leaves the idle state (see CPOL). Set to have the data sampled when the SCK returns to idle state (see CPOL).
1	SPR1	**SPR2** **SPR1** **SPR0** **Serial Peripheral Rate** 0 0 0 Invalid 0 0 1 $F_{CLK\ PERIPH}/4$ 0 1 0 $F_{CLK\ PERIPH}/8$ 0 1 1 $F_{CLK\ PERIPH}/16$ 1 0 0 $F_{CLK\ PERIPH}/32$ 1 0 1 $F_{CLK\ PERIPH}/64$ 1 1 0 $F_{CLK\ PERIPH}/128$ 1 1 1 Invalid
0	SPR0	

Reset Value = 0001 0100b

Not bit addressable

Serial Peripheral Status Register and Control (SPSCR)

The Serial Peripheral Status Register contains flags to signal the following conditions:

- Data transfer complete
- Write collision
- Inconsistent logic level on $\overline{SS}$ pin (mode fault error)

Table 49. SPSCR Register

SPSCR - Serial Peripheral Status and Control register (0D5H)

7	6	5	4	3	2	1	0
SPIF	**-**	**OVR**	**MODF**	**SPTE**	**UARTM**	**SPTEIE**	**MODFIE**

Bit Number	Bit Mnemonic	Description
7	SPIF	**Serial Peripheral Data Transfer Flag** Cleared by hardware to indicate data transfer is in progress or has been approved by a clearing sequence. Set by hardware to indicate that the data transfer has been completed. This bit is cleared when reading or writing SPDATA after reading SPSCR.
6	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
5	OVR	**Overrun Error Flag** - Set by hardware when a byte is received whereas SPIF is set (the previous received data is not overwritten). - Cleared by hardware when reading SPSCR

Figure 59. ADC Description

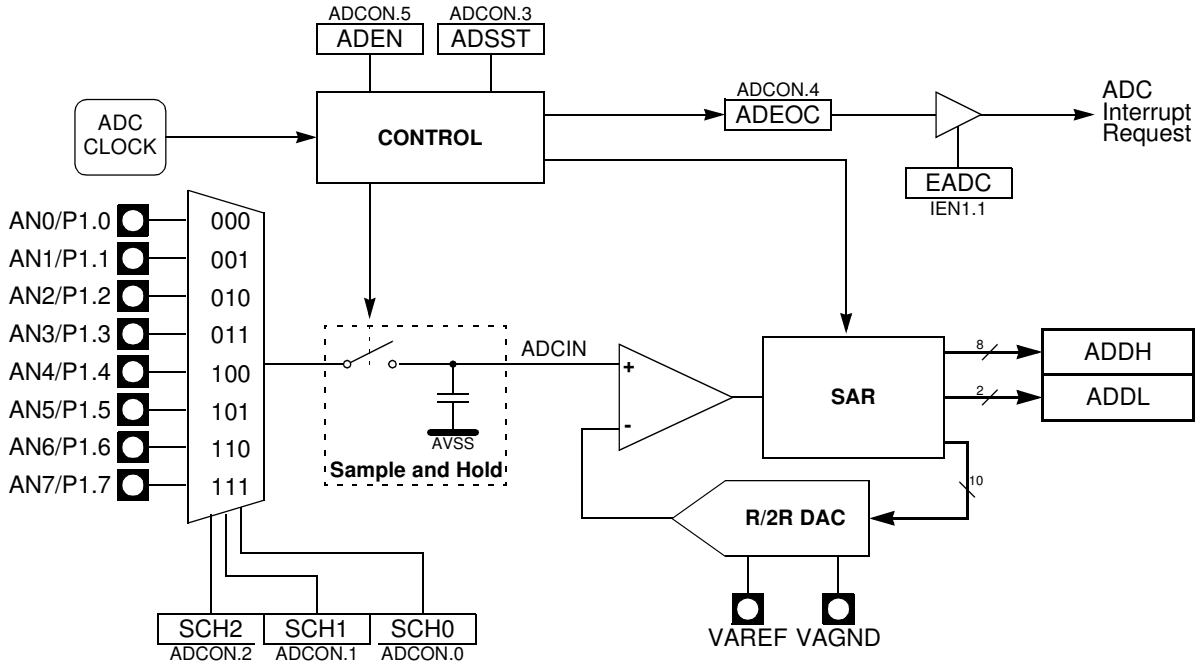
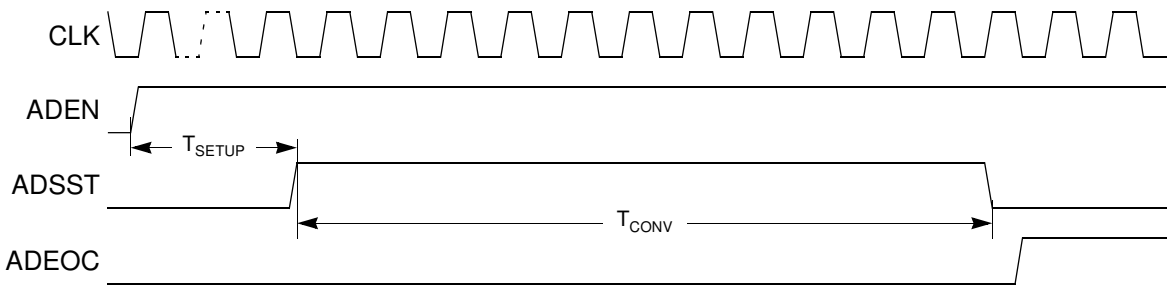


Figure 60 shows the timing diagram of a complete conversion. For simplicity, the figure depicts the waveforms in idealized form and do not provide precise timing information. For ADC characteristics and timing parameters refer to the Section “AC Characteristics” of the AT89C51AC3 datasheet.

Figure 60. Timing Diagram



Note: Tsetup min = 4 us
Tconv=11 clock ADC = 1sample and hold + 10 bit conversion
The user must ensure that 4 us minimum time between setting ADEN and the start of the first conversion.

Table 60. ADCLK Register

ADCLK (S:F2h)
ADC Clock Prescaler

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

Bit Number	Bit Mnemonic	Description
7-5	-	**Reserved** The value read from these bits are indeterminate. Do not set these bits.
4-0	PRS4:0	**Clock Prescaler** See Note (1)

Reset Value = XXX0 0000b

Note: 1. In X1 mode:

$$\text{For PRS} > 0\ F_{ADC} = \frac{FXTAL}{4 \times PRS}$$

$$\text{For PRS} = 0\ F_{ADC} = \frac{FXTAL}{128}$$

In X2 mode:

$$\text{For PRS} > 0\ F_{ADC} = \frac{FXTAL}{2 \times PRS}$$

$$\text{For PRS} = 0\ F_{ADC} = \frac{FXTAL}{64}$$

Table 61. ADDH Register

ADDH (S:F5h Read Only)
ADC Data High Byte Register

7	6	5	4	3	2	1	0
ADAT 9	ADAT 8	ADAT 7	ADAT 6	ADAT 5	ADAT 4	ADAT 3	ADAT 2

Bit Number	Bit Mnemonic	Description
7-0	ADAT9:2	**ADC result** bits 9-2

Reset Value = 00h

Table 62. ADDL Register

ADDL (S:F4h Read Only)
ADC Data Low Byte Register

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