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	8051
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
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-rdtum

Product Description

AT8xC5122/23 products are high-performance CMOS derivatives of the 80C51 8-bit microcontrollers designed for USB smart card reader applications.

The AT8xC5122 is proposed in four versions :

- ROM version referenced AT83R5122 is only factory programmable.

- CRAM version without internal data E2PROM. The CRAM device implements a volatile program memory which is programmed by means of an embedded ROMed bootloader which transfers the code from a remote software programming tool called FLIP through UART or USB interfaces.

- Flash version without internal data E2PROM. At power-up, the program located in the flash memory is transferred into the CRAM then executed.

The AT83C5123 is a low pin count of the AT8xC5122 and is proposed in ROM version with or without internal data E2PROM. The ROM device is only factory programmable.

The AT8xC5122DS is a secure version of the AT8xC5122 on which the external program memory access mode is disabled.

Table 1. Product versions

Features		**AT83C5122 AT83R5122**	**AT85C5122**	**AT89C5122**	**AT89C5122DS**	**AT83C5123**
Packages		VQFP64 QFN64 Die Form	PLCC68 VQFP64 Die Form	VQFP64 QFN64	VQFP64 QFN64	VQFP32 QFN32 Die Form
Program memory		32KB ROM	32KB CRAM	32KB E2PROM	32KB E2PROM	30KB ROM
Internal Data E2PROM		No	No	No	No	No
Embedded bootloader		No	Yes	Yes	Yes	No
Features	VQFP32, QFN32 packages					**Features not available :** - Keyboard Interface - Master/Slave SPI Interface - External Program Memory Access **Reduced features :** - Only 12 I/O with up to 4 LED Outputs with Programmable Current
	PLCC68, VQFP64,QFN64 packages	All features are available			All features are available except External Program Memory Access	

Table 6. Auxiliary Register 1 AUXR1- (0A2h) for AT8xC5122

7	6	5	4	3	2	1	0
-	-	ENBOOT	-	GF3	0	-	DPS

Bit Number	Bit Mnemonic	Description
7 - 6	-	**Reserved** The value read from this bit is indeterminate. Do not change these bits.
5	ENBOOT	**Enable Boot ROM (CRAM / E2PROM version only)** Set this bit to map the Boot ROM from 8000h to FFFFh. If the PC increments beyond 7FFFh address, the code is fetch from internal ROM Clear this bit to disable Boot ROM. If the PC increments beyond 7FFFh address, the code is fetch from external code memory (C51 standard roll over function) This bit is forced to 1 at reset
4	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
3	GF3	This bit is a general-purpose user flag.
2	0	Always cleared.
1	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
0	DPS	**Data Pointer Selection** Cleared to select DPTR0. Set to select DPTR1.

Reset Value = XX1X XX0X0b (Not bit addressable)

Table 7. Auxiliary Register 1 AUXR1- (0A2h) for AT83C5123

7	6	5	4	3	2	1	0
-	-	-	-	GF3	0	-	DPS

Bit Number	Bit Mnemonic	Description
7 - 6	-	**Reserved** The value read from this bit is indeterminate. Do not change these bits.
5		**Reserved** The value read from this bit is indeterminate. Do not change these bits.
4	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
3	GF3	This bit is a general-purpose user flag.
2	0	Always cleared.
1	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
0	DPS	**Data Pointer Selection** Cleared to select DPTR0. Set to select DPTR1.

Reset Value = XXXX XX0X0b (Not bit addressable)

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

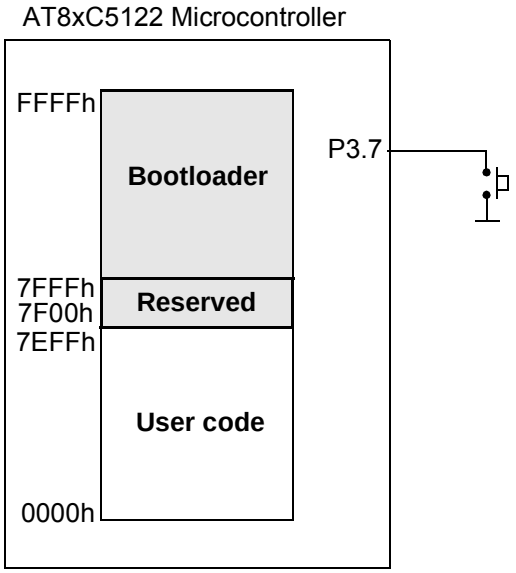
In E2PROM version :

- 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.



When pin EA =1 and after the reset, the MCU begins the execution of the embedded bootloader from location F800h of the ROM. The bootloader implements an In-System Programming (ISP) mode which manages the transfer of the code in the volatile Program Memory (CRAM).

For CRAM version, the code is supplied by the ATMEL's FLexible In-system Programming software (FLIP) through USB or UART interface

For E2PROM version, the code is supplied from the internal code E2PROM or by FLIP. The state of pin P3.7 at reset determines the code source. If P3.7=1 (reset condition) the source is the internal E2PROM and the transfer takes about 1.5 seconds. If P3.7=0 the source is FLIP and the transfer time depends mainly on external conditions not related to bootloader.

Once the code is running in CRAM, the roll-over condition (code fetched beyond address 7FFFh) depends on the state of ENBOOT bit of AUXR1 register (Table 6 on page 23).

If ENBOOT=1 (reset condition) the MCU fetches the code from bootloader ROM. If ENBOOT=0, the MCU fetches the code from the external Program Memory. In this last case, PSEN is activated and Ports P0 and P2 are used to emit data and address signals.

Warning : external Program Memory access is not allowed on Low Pin Count Packages.

Smart Card Interface Block (SCIB)

The SCIB provides all signals to interface directly with a smart card. The compliance with the ISO7816, EMV'2000, GSM and WHQL standards has been certified.

Both synchronous (e.g. memory card) and asynchronous smart cards (e.g. microprocessor card) are supported. The component supplies the different voltages requested by the smart card. The power off sequence is directly managed by the SCIB.

The card presence switch of the smart card connector is used to detect card insertion or card removal. In case of card removal, the SCIB de-activates the smart card using the de-activation sequence. An interrupt can be generated when a card is inserted or removed.

Any malfunction is reported to the microcontroller (interrupt + control register).

The different operating modes are configured by internal registers.

- Support of ISO/IEC 7816
- character mode
- one transmit/receive buffer
- 11 bits ETU counter
- 9 bits guard time counter
- 32 bits waiting time counter
- Auto character repetition on error signal detection in transmit mode
- Auto error signal generation on parity error detection in receive mode
- Power on and power off sequence generation
- Manual mode to drive directly the card I/O

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

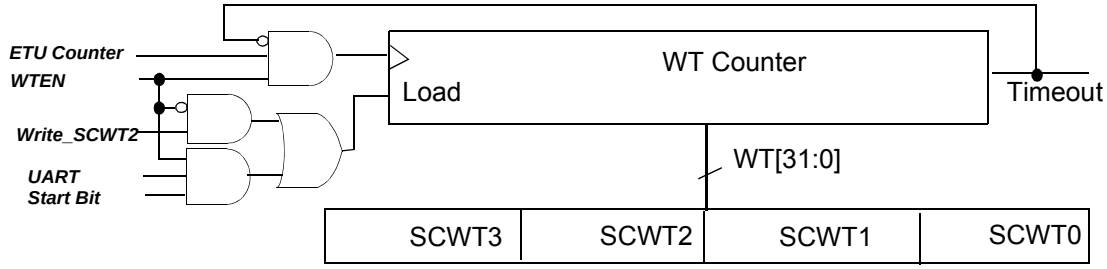
Guard Time Counter

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.

When the WT counter times out, an interrupt is generated and the SCIB function is locked: reception and emission are disabled. It can be enabled by resetting the macro or reloading the counter.

The Waiting Time Counter can be used in T=0 protocol for the Work Waiting Time. It can be used in T=1 protocol for the Character Waiting Time and for the Block Waiting Time. See the detailed explanation below.

Figure 33. Waiting Time Counter



In the so called manuel mode, the counter is loaded, if WTEN = 0, during the write of SCWT2 register. The counter is loaded with a 32 bit word built with SCWT3 SCWT2 SCWT1 SCWT0 registers (SCWT0 contain WT[7-0] byte. WTEN is located in the SCICR register.

When WTEN=1 and in UART mode, the counter is re-loaded at the occurence of a start bit. This mode will be detailed below in T=0 protocol and T=1 protocol.

In manual mode, the WTEN signal controls the start of the counter (rising edge) and the stop of the counter (falling edge). After a timeout of the counter, a falling edge on WTEN, a reload of SCWT2 and a rising edge of WTEN are necessary to start again the counter and to release the SCIB macro. The reload of SCWT2 transfers all SCWT0, SCWT1, SCWT2 and SCWT3 registers to the WT counter.

In UART mode there is an automatic load on the start bit detection. This automatic load is very useful for changing on-the-fly the timeout value since there is a register to hold the load value. This is the case for T=1 protocol.

In T=0 protocol the maximun interval between the start leading edge of any character sent by the ICC and the start of the previous character sent by either the ICC or the Terminal is the maximum Work Waiting Time. The Work Waiting Time shall not exceed 960*D*WI ETUs with D and WI parameters are returned by the field TA1 and TC2 respectively in the Answer To Reset (ATR). This is the value the user shall write in the SCWT0,1,2,3 register. This value will be reloaded in the Waiting Time counter every start bit.

Table 48. Smart Card UART Interrupt Enabling Register - SCIER (S:AEh, SCRS=1)

7	6	5	4	3	2	1	0
ESCTBI	-	ICARDER	EVCARDER	ESCWTI	ESCTI	ESCRI	ESCPI

Bit Number	Bit Mnemonic	Description
7	ESCTBI	**UART Transmit Buffer Empty Interrupt Enabled** Clear this bit to disable the Smart Card UART Transmit Buffer Empty interrupt. Set this bit to enable the Smart Card UART Transmit Buffer Empty interrupt.
6	-	**Reserved** The value read from this bit is indeterminate. Do not change this bit.
5	ICARDER	**Card Current Overflow Interrupt Enabled** Clear this bit to disable the Card Current Overflow interrupt. Set this bit to enable the Card Current Overflow interrupt.
4	EVCARDER	**Card Voltage Error Interrupt Enabled** Clear this bit to disable the Card Voltage Error interrupt. Set this bit to enable the Card Voltage Error interrupt.
3	ESCWTI	**WaitingTime Counter Timeout Interrupt Enabled** Clear this bit to disable the Waiting Time Counter timeout interrupt. Set this bit to enable the Waiting Time Counter timeout interrupt.
2	ESCTI	**UART Transmitted Character Interrupt Enabled** Clear this bit to disable the Smart Card UART Transmitted Character interrupt. Set this bit to enable the Smart Card UART Transmitted Character interrupt.
1	ESCRI	**UART Received Character Interrupt Enabled** Clear this bit to disable the Smart Card UART Received Character interrupt. Set this bit to enable the Smart Card UART Received Character interrupt.
0	ESCPI	**Character Reception Parity Error Interrupt Enabled** Clear this bit to disable the Smart Card Character Reception Parity Error interrupt. Set this bit to enable the Smart Card Character Reception Parity Error interrupt.

Reset Value = 0X00 0000b

DC/DC Converter register

Table 61. DC/DC Converter Control Register - DCCKPS (S:BFh)

7	6	5	4	3	2	1	0
MODE	OVFADJ	BOOST1	BOOST0	DCCKPS3	DCCKPS2	DCCKPS1	DCCKPS0

Bit Number	Bit Mnemonic	Description	
7	MODE	Regulation mode 0 : Pump mode (External Inductance required) 1 : Regulator mode (No External inductance required if VCC > CVCC+0.3V)	
6	OVFADJ	Current Overflow Adjustment on Smart Card terminal 0 : normal: 100 mA average 1 : normal + 20%	
5 - 4	BOOST[1:0]	**VCARDOK=0** **Maximum Startup Current drawn from power supply** 00 : Normal: 30 mA average 01 : Normal + 30% 10 : Normal + 50% 11 : Normal + 80%	**VCARDOK=1** **Current Overflow Level on Smart Card terminal** 00 : Normal = OVFADJ 01 : Normal + 30% 10 : Normal + 50% 11 : Normal + 80%
3 - 0	DCCKPS[3:0]	DC/DC Clock Prescaler Value 0000 : Division factor: 2 (reset value) 0001 : Division factor: 3 0010 : Division factor: 4 0011 : Division factor: 5 0100 : Division factor: 6 0101 : Division factor: 8 0110 : Division factor: 10 0111 : Division factor: 12 1000 : Division factor: 24 Other values are reserved	

Reset Value = 0000 0000b

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

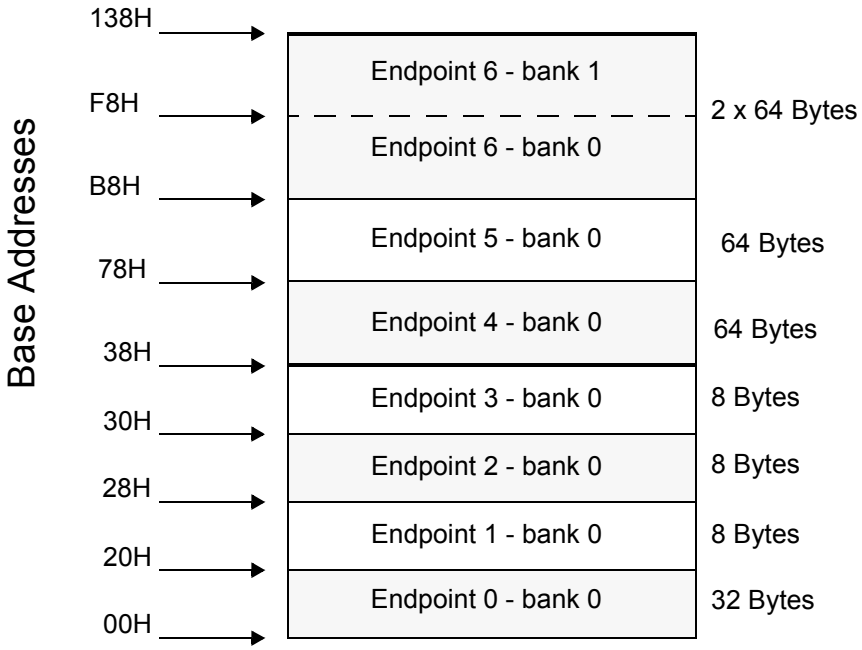
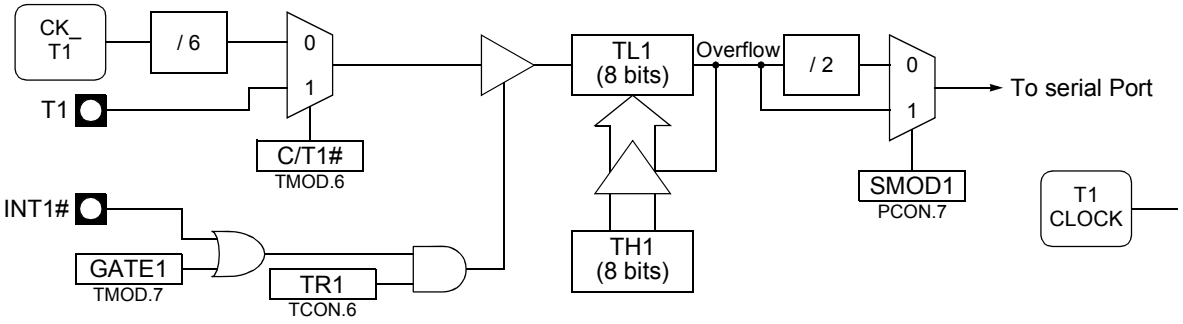


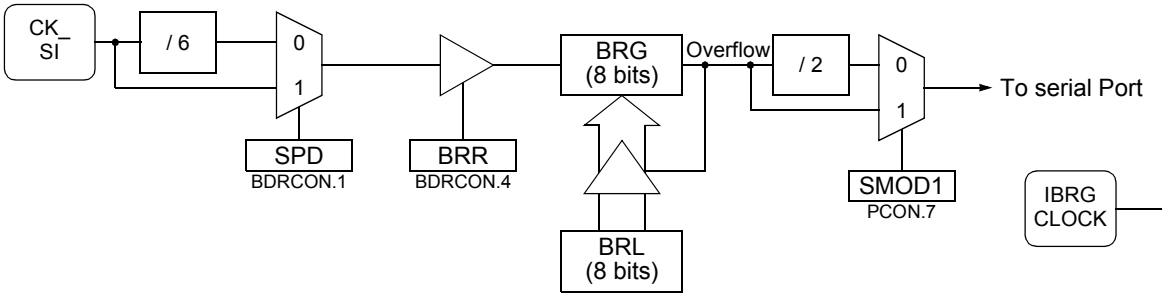
Figure 67. Timer 1 Baud Rate Generator Block Diagram



Internal Baud Rate Generator

When using the Internal Baud Rate Generator, the Baud Rate is derived from the overflow of the timer. As shown in Figure 68 the Internal Baud Rate Generator is an 8-bit auto-reload timer feed by the peripheral clock or by the peripheral clock divided by 6 depending on the SPD bit in BDRCON register (see Table 82 on page 134). The Internal Baud Rate Generator is enabled by setting BRR bit in BDRCON register. SMOD1 bit in PCON register allows doubling of the generated baud rate.

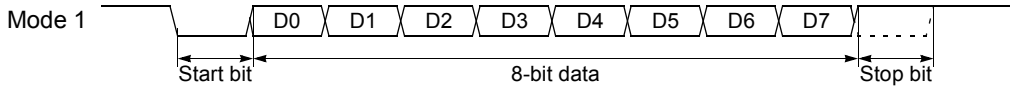
Figure 68. Internal Baud Rate Generator Block Diagram



Synchronous Mode (Mode 0)

Mode 0 is a half-duplex, synchronous mode, which is commonly used to expand the I/0 capabilities of a device with shift registers. The transmit data (TXD) pin outputs a set of eight clock pulses while the receive data (RXD) pin transmits or receives a byte of data. The 8-bit data are transmitted and received least-significant bit (LSB) first. Shifts occur at a fixed Baud Rate (see Section “Baud Rate Selection (Mode 0)”). Figure 69 shows the serial port block diagram in Mode 0.

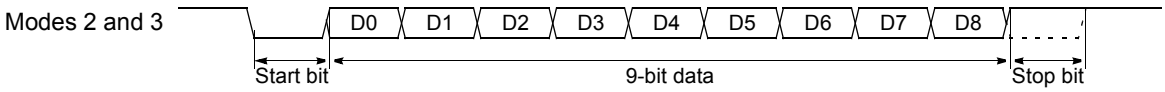
Figure 75. Data Frame Format (Mode 1)



Modes 2 and 3

Modes 2 and 3 are full-duplex, asynchronous modes. The data frame (see Figure 76) consists of 11 bits: one start bit, eight data bits (transmitted and received LSB first), one programmable ninth data bit and one stop bit. Serial data is transmitted on the TXD pin and received on the RXD pin. On receive, the ninth bit is read from RB8 bit in SCON register. On transmit, the ninth data bit is written to TB8 bit in SCON register. Alternatively, you can use the ninth bit as a command/data flag.

Figure 76. Data Frame Format (Modes 2 and 3)



Transmission (Modes 1, 2 and 3)

To initiate a transmission, write to SCON register, setting SM0 and SM1 bits according to Figure 69 on page 128, and setting the ninth bit by writing to TB8 bit. Then, writing the byte to be transmitted to SBUF register starts the transmission.

Reception (Modes 1, 2 and 3)

To prepare for a reception, write to SCON register, setting SM0 and SM1 bits according to Figure 69 on page 128, and setting REN bit. The actual reception is then initiated by a detected high-to-low transition on the RXD pin.

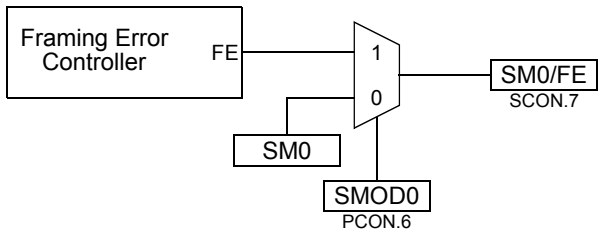
Framing Error Detection (Modes 1, 2 and 3)

Framing error detection is provided for the three asynchronous modes. To enable the framing bit error detection feature, set SMOD0 bit in PCON register as shown in Figure 77.

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 devices. If a valid stop bit is not found, the software sets FE bit in SCON register.

Software may examine FE bit after each reception to check for data errors. Once set, only software or a chip reset clear FE bit. Subsequently received frames with valid stop bits cannot clear FE bit. When the framing error detection feature is enabled, RI rises on stop bit instead of the last data bit as detailed in Figure 75 and Figure 76.

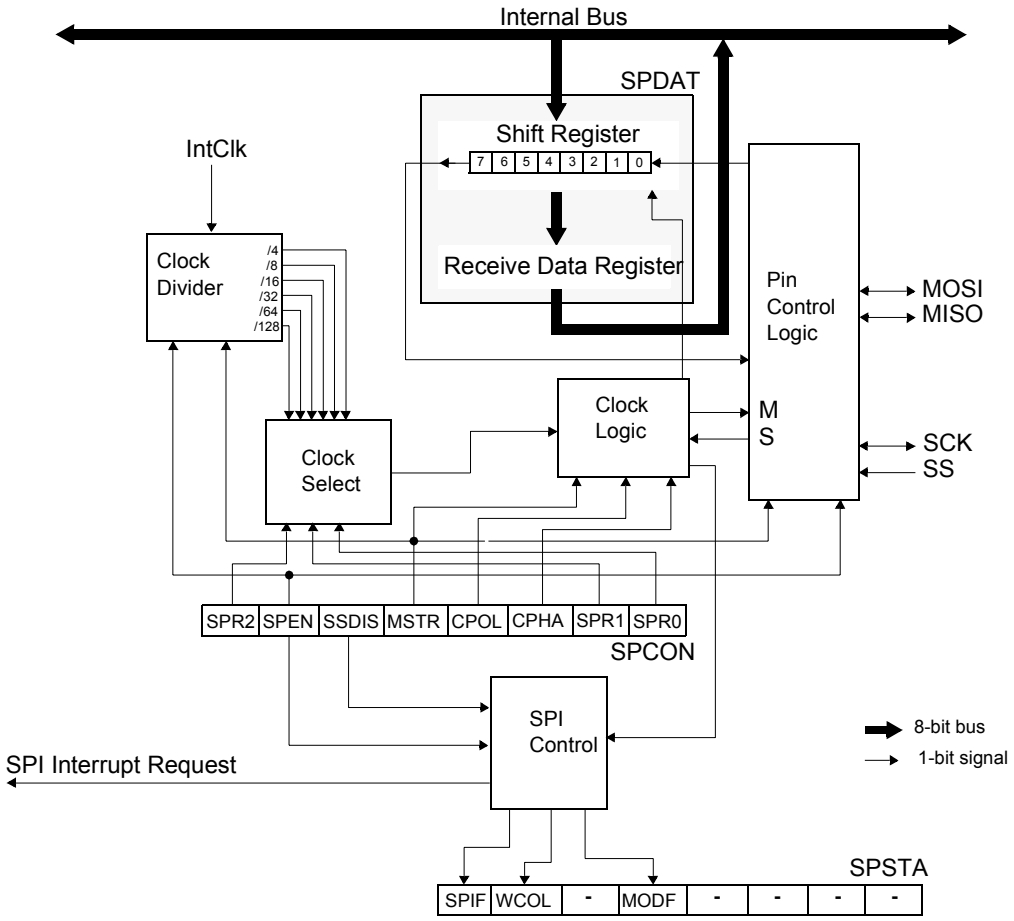
Figure 77. Framing Error Block Diagram



Functional Description

Figure 83 shows a detailed structure of the SPI module.

Figure 83. SPI Module Block Diagram



Operating Modes

The Serial Peripheral Interface can be configured as one of the two modes: Master mode or Salve mode. The configuration and initialization of the SPI module is made through one register:

- The Serial Peripheral Control register (SPCON)

Once the SPI is configured, the data exchange is made using:

- SPCON
- The Serial Peripheral Status register (SPSTA)
- The Serial Peripheral Data register (SPDAT)

During an SPI transmission, data is simultaneously transmitted (shifted out serially) and received (shifted in serially). A serial clock line (SCK) synchronizes shifting and sampling on the two serial data lines (MOSI and MISO). A Slave Select line ($\overline{SS}$) allows individual selection of a Slave SPI device; Slave devices that are not selected do not interfere with SPI bus activities.

When the Master device transmits data to the Slave device via the MOSI line, the Slave device responds by sending data to the Master device via the MISO line. This implies full-duplex transmission with both data out and data in synchronized with the same clock (Figure 84).

Mode 3 (Two 8-bit Timers)

Mode 3 configures Timer 0 so that registers TL0 and TH0 operate as 8-bit Timers (see Figure 95). This mode is provided for applications requiring an additional 8-bit Timer or Counter. TL0 uses the Timer 0 control bits C/T0# and GATE0 in the TMOD register, and TR0 and TF0 in the TCON register in the normal manner. TH0 is locked into a Timer function (counting F_{UART}) and takes over use of the Timer 1 interrupt (TF1) and run control (TR1) bits. Thus, operation of Timer 1 is restricted when Timer 0 is in mode 3.

Figure 96 gives the autoreload period calculation formulas for both TF0 and TF1 flags.

Figure 95. Timer/Counter 0 in Mode 3: Two 8-bit Counters

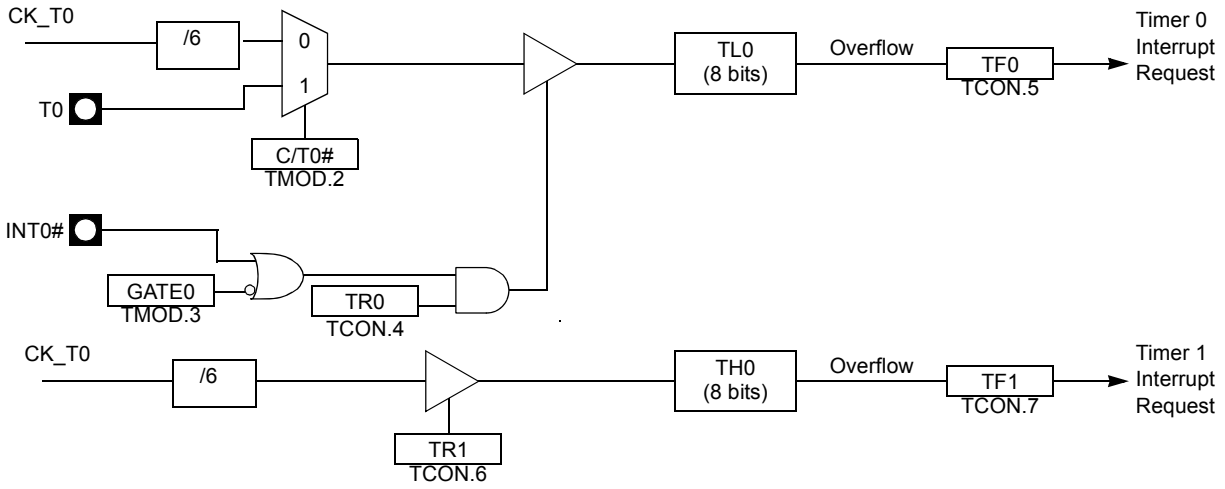


Figure 96. Mode 3 Overflow Period Formula

$$TF0_{PER} = \frac{6 \cdot (256 - TL0)}{F_{CK\_T0}} \qquad TF1_{PER} = \frac{6 \cdot (256 - TH0)}{F_{CK\_T0}}$$

Timer 1

Timer 1 is identical to Timer 0 except for Mode 3 which is a hold-count mode. The following comments help to understand the differences:

- Timer 1 functions as either a Timer or an event Counter in three operating modes. Figure 89 through Figure 93 show the logical configuration for modes 0, 1, and 2. Mode 3 of Timer 1 is a hold-count mode.
- Timer 1 is controlled by the four high-order bits of the TMOD register (see Table 89 on page 151) and bits 2, 3, 6 and 7 of the TCON register (see Table 88 on page 150). The TMOD register selects the method of Timer gating (GATE1), Timer or Counter operation (C/T1#) and the operating mode (M11 and M01). The TCON register provides Timer 1 control functions: overflow flag (TF1), run control bit (TR1), interrupt flag (IE1) and the interrupt type control bit (IT1).
- Timer 1 can serve as the Baud Rate Generator for the Serial Port. Mode 2 is best suited for this purpose.
- For normal Timer operation (GATE1 = 0), setting TR1 allows TL1 to be incremented by the selected input. Setting GATE1 and TR1 allows external pin INT1# to control Timer operation.
- Timer 1 overflow (count rolls over from all 1s to all 0s) sets the TF1 flag and generates an interrupt request.

Power Management

Before activating the Idle Mode or Power Down Mode, the CPU clock must be switched to on-chip oscillator source if the PLL is used to fed the CPU clock.

Idle Mode

An instruction that sets PCON.0 indicates that it is the last instruction to be executed before going into the Idle mode. In the Idle mode, the internal clock signal is gated off to the CPU, but not to the interrupt, Timer, and Serial Port functions. The CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator and all other registers maintain their data during Idle. The port pins hold the logical states they had at the time Idle was activated. ALE and PSEN hold at logic high level.

There are two ways to terminate the Idle mode. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating the Idle mode. The interrupt will be serviced, and following RETI the next instruction to be executed will be the one following the instruction that put the device into idle.

The flag bits GF0 and GF1 can be used to give an indication if an interrupt occured during normal operation or during an Idle. For example, an instruction that activates Idle can also set one or both flag bits. When Idle is terminated by an interrupt, the interrupt service routine can examine the flag bits.

The other way of terminating the Idle mode is with a hardware reset. Since the clock oscillator is still running, the hardware reset needs to be held active for only two machine cycles (24 oscillator periods) to complete the reset.

Power Down Mode

To save maximum power, a power-down mode can be invoked by software (see Table 13, PCON register).

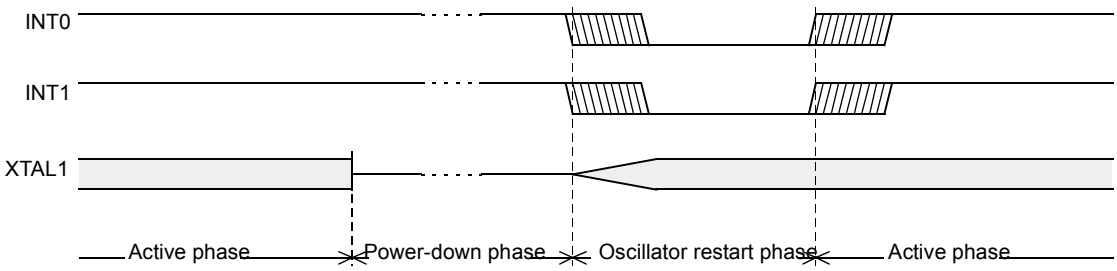
WARNING: To minimize power consumption, all peripherals and I/Os with static current consumption must be set in the proper state. I/Os programmed with low speed output configuration (KB_OUT) must be switch to push-pull or Standard C51 configuration before entering power-down. The CVCC generator must also be switch off.

In power-down mode, the oscillator is stopped and the instruction that invoked power-down mode is the last instruction executed. The internal RAM and SFRs retain their value until the power-down mode is terminated. V_{CC} can be lowered to save further power. Either a hardware reset or an external interrupt can cause an exit from power-down. To properly terminate power-down, the reset or external interrupt should not be executed before V_{CC} is restored to its normal operating level and must be held active long enough for the oscillator to restart and stabilize.

Only external interrupts $\overline{INT0}$, $\overline{INT1}$, Keyboard, Card insertion/removal and USB Interrupts are useful to exit from power-down. For that, interrupt must be enabled and configured as level or edge sensitive interrupt input. When Keyboard Interrupt occurs after a power-down mode, 1024 clocks are necessary to exit to power-down mode and enter in operating mode.

Holding the pin low restarts the oscillator but bringing the pin high completes the exit as detailed in Figure 108. When both interrupts are enabled, the oscillator restarts as soon as one of the two inputs is held low and power-down exit will be completed when the first input is released. In this case, the higher priority interrupt service routine is executed. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put AT83R5122, AT8xC5122/23 into power-down mode.

Figure 108. Power-down Exit Waveform



Exit from power-down by reset redefines all the SFRs, exit from power-down by external interrupt does no affect the SFRs.

Exit from power-down by either reset or external interrupt does not affect the internal RAM content.

Note: If idle mode is activated with power-down mode (IDL and PD bits set), the exit sequence is unchanged, when execution is vectored to interrupt, PD and IDL bits are cleared and idle mode is not entered.

Table shows the state of ports during idle and power-down modes.

Table State of Ports

Mode	Program Memory	ALE	PSEN	P0	P1	P2	P3	P4	P5
Idle	Internal	1	1	Port Data(1)	Port Data	Port Data	Port Data	Port Data	Port Data
Idle	External	1	1	Floating	Port Data	Address	Port Data	Port Data	Port Data
Power-down	Internal	0	0	Port Dat*	Port Data	Port Data	Port Data	Port Data	Port Data
Power-down	External	0	0	Floating	Port Data	Port Data	Port Data	Port Data	Port Data

Note: 1. Port 0 can force a 0 level. A "one" will leave port floating.

Reduced EMI Mode

The ALE signal is used to demultiplex address and data buses on port 0 when used with external program or data memory. Nevertheless, during internal code execution, ALE signal is still generated. In order to reduce EMI, ALE signal can be disabled by setting AO bit.

The AO bit is located in AUXR register at bit location 0. As soon as AO is set, ALE is no longer output but remains active during MOVX and MOVC instructions and external fetches. During ALE disabling, ALE pin is weakly pulled high.

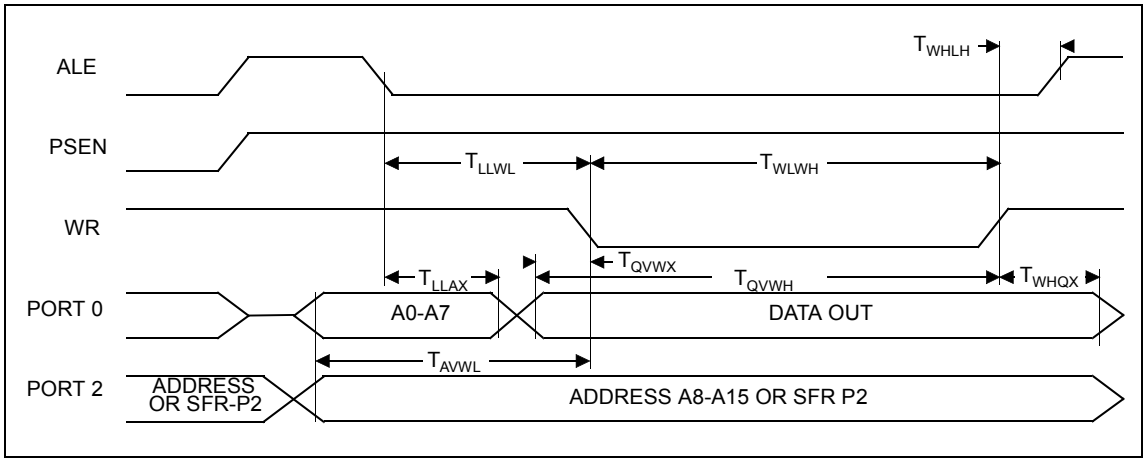
Symbol	Parameter	Min	Typ	Max	Unit	Test Conditions
R_{RST}	Internal reset pull-up resistor	5	10	30	kΩ	
I_{PD}	Power down consumption		60µA 40µA	200µA 200µA		Vcc = 5.5V Vcc = 3.6V
I_{CCIDLE}	Power Supply current in IDLE mode			0.4*F+2	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT89C5122) with DC/DC ON			1.6*F+3	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT85C5122) with DC/DC ON			1.6*F+3	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT83C5122) with DC/DC ON			1.6*F+2	mA	Vcc = 5.5V (F in MHz)
$I_{CCWRITE}$	Power Supply current in Active mode (AT89C5122) Flash or E2PROM write DC/DC ON			1.6*F+4	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT89C5122) with DC/DC OFF			0.8*F+3	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT85C5122) with DC/DC FF			0.8*F+3	mA	Vcc = 5.5V (F in MHz)
I_{CCOP}	Power Supply current in Active mode (AT83C5122) with DC/DC OFF			0.8*F+2	mA	Vcc = 5.5V (F in MHz)
$I_{CCWRITE}$	Power Supply current in Active mode (AT89C5122) Flash or E2PROM write DC/DC OFF			0.8*F+4	mA	Vcc = 5.5V (F in MHz)

Table 117. AC Parameters for a Variable Clock

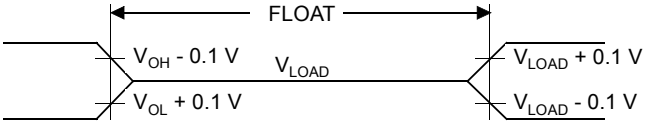
Symbol	Type	Standard Clock	X2 Clock	X parameter	Units
T_{RLRH}	Min	6T - x	3 T - x	20	ns
T_{WLWH}	Min	6T - x	3 T - x	20	ns
T_{RLDV}	Max	5T - x	2.5 T - x	25	ns
T_{RHDX}	Min	x	x	0	ns
T_{RHDZ}	Max	2T - x	T - x	20	ns
T_{LLDV}	Max	8T - x	4T - x	40	ns
T_{AVDV}	Max	9T - x	4.5 T - x	60	ns
T_{LLWL}	Min	3T - x	1.5 T - x	25	ns
T_{LLWL}	Max	3T + x	1.5 T + x	25	ns
T_{AVWL}	Min	4T - x	2 T - x	25	ns
T_{QVWX}	Min	T - x	0.5 T - x	15	ns
T_{QVWH}	Min	7 T - x	3.5 T - x	25	ns
T_{WHQX}	Min	T - x	0.5 T - x	10	ns
T_{RLAZ}	Max	x	x	0	ns
T_{WHLH}	Min	T - x	0.5 T - x	15	ns
T_{WHLH}	Max	T - x	0.5 T + x	15	ns

(warning x value differ from AT89C51RD2)

External Data Memory Write Cycle



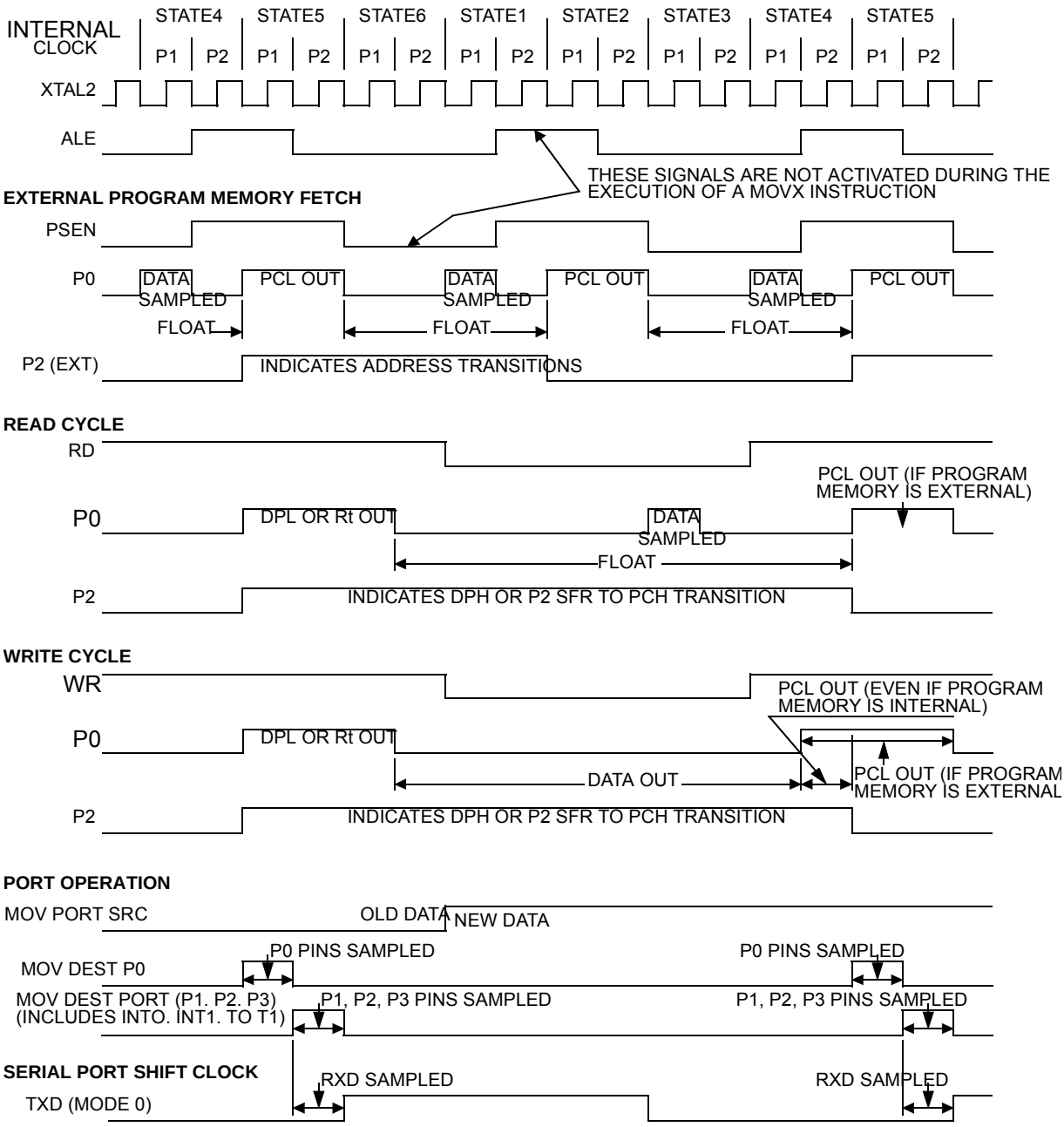
Float Waveforms



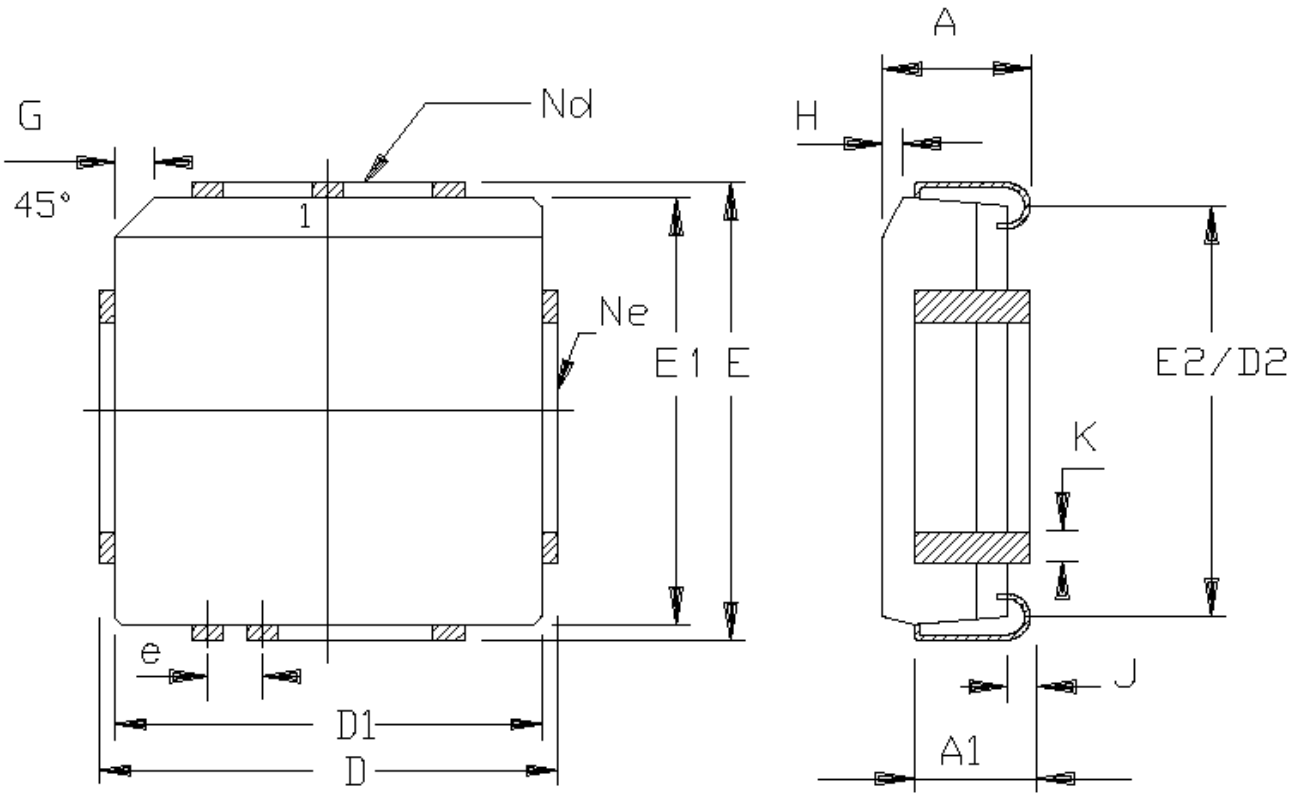
For timing purposes as port pin is no longer floating when a 100 mV change from load voltage occurs and begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs. $I_{OL}/I_{OH} \geq \pm 20$ mA.

Clock Waveforms

Valid in normal clock mode. In X2 mode XTAL2 must be changed to XTAL2/2.



PLCC68 Package



	MM		INCH	
A	4.20	5.08	.165	.200
A1	2.29	3.30	.090	.130
D	25.02	25.27	.985	.995
D1	24.13	24.33	.950	.958
D2	22.61	23.62	.890	.930
E	25.02	25.27	.985	.995
E1	24.13	24.33	.950	.958
E2	22.61	23.62	.890	.930
e	1.27	BSC	.050	BSC
G	1.07	1.22	.042	.048
H	1.07	1.42	.042	.056
J	0.51	-	.020	-
K	0.33	0.53	.013	.021
Nd	17		17	
Ne	17		17	