Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	80C51
Core Size	8-Bit
Speed	40MHz
Connectivity	SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	48
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	1.25K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.23x24.23)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/t89c51rd2-smsim

- Instructions that use indirect addressing access the Upper 128 bytes of data RAM. **For example: MOV @R0, # data** where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).
- The XRAM bytes can be accessed by indirect addressing, with EXTRAM bit cleared and MOVX instructions. This part of memory which is physically located on-chip, logically occupies the first bytes of external data memory. The bits XRS0 and XRS1 are used to hide a part of the available XRAM as explained in Table 7. This can be useful if external peripherals are mapped at addresses already used by the internal XRAM.
- With EXTRAM = 0, the XRAM is indirectly addressed, using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. **An access to XRAM will not affect ports P0, P2, P3.6 (WR) and P3.7 (RD)**. **For example, with EXTRAM = 0, MOVX @R0, # data** where R0 contains 0A0H, accesses the XRAM at address 0A0H rather than external memory. An access to external data memory locations higher than the accessible size of the XRAM will be performed with the MOVX DPTR instructions in the same way as in the standard 80C51, so with P0 and P2 as data/address busses, and P3.6 and P3.7 as write and read timing signals. Accesses to XRAM above 0FFH can only be done thanks to the use of DPTR.
- With EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 80C51. MOVX @ Ri will provide an eight-bit address multiplexed with data on Port0 and any output port pins can be used to output higher order address bits. This is to provide the external paging capability. MOVX @DPTR will generate a sixteen-bit address. Port2 outputs the high-order eight address bits (the contents of DPH) while Port0 multiplexes the low-order eight address bits (DPL) with data. MOVX @ Ri and MOVX @DPTR will generate either read or write signals on P3.6 ($\overline{WR}$) and P3.7 ($\overline{RD}$).

The stack pointer (SP) may be located anywhere in the 256 bytes RAM (lower and upper RAM) internal data memory. The stack may not be located in the XRAM.

The M0 bit allows to stretch the XRAM timings; if M0 is set, the read and write pulses are extended from 6 to 30 clock periods. This is useful to access external slow peripherals.

Table 7. Auxiliary Register (08EH)

AUXR Address 08EH	-	-	M0	-	XRS1	XRS0	EXTRAM	AO
Reset value	X	X	0	X	1	0	0	0

Symbol	Function	
-	Not implemented, reserved for future use. (1)	
AO	Disable/Enable ALE	
	AO	Operating Mode
	0	ALE is emitted at a constant rate of 1/6 the oscillator frequency (or 1/3 if X2 mode is used)
	1	ALE is active only during a MOVX or MOVC instruction
EXTRAM	Internal/External RAM (00H-FFH) access using MOVX @ Ri/ @ DPTR	
	EXTRAM	Operating Mode
	0	Internal XRAM access using MOVX @ Ri/ @ DPTR
	1	External data memory access
XRS0 XRS1	XRAM size: Accessible size of the XRAM	
	XRS1:0	XRAM size
	0 0	256 bytes
	0 1	512 bytes
	1 0	768 bytes (default)
	1 1	1024 bytes
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

1. User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

Timer 2

The timer 2 in the T89C51RD2 is compatible with the timer 2 in the 80C52.
It is a 16-bit timer/counter: the count is maintained by two eight-bit timer registers, TH2 and TL2, connected in cascade. It is controlled by T2CON register (See Table 8) and T2MOD register (See Table 9). Timer 2 operation is similar to Timer 0 and Timer 1. C/$\overline{T2}$ selects $F_{OSC}/12$ (timer operation) or external pin T2 (counter operation) as the timer clock input. Setting TR2 allows TL2 to be incremented by the selected input.

Timer 2 has 3 operating modes: capture, autoreload and Baud Rate Generator. These modes are selected by the combination of RCLK, TCLK and CP/$\overline{RL2}$ (T2CON), as described in the ATMEL Wireless and Micrcontrollers 8-bit Microcontroller Hardware description.

Refer to the ATMEL Wireless and Micrcontrollers 8-bit Microcontroller Hardware description for the description of Capture and Baud Rate Generator Modes.

In T89C51RD2 Timer 2 includes the following enhancements:

- Auto-reload mode with up or down counter
- Programmable clock-output

Auto-Reload Mode

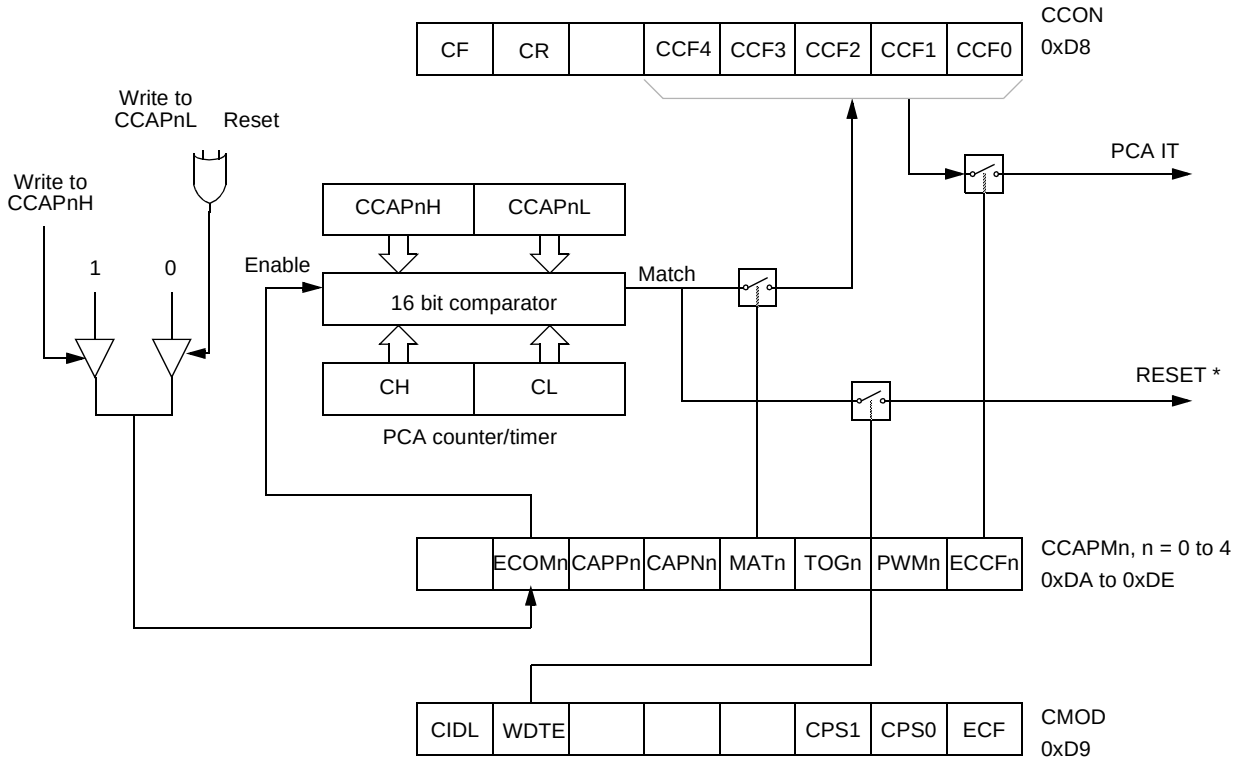
The auto-reload mode configures timer 2 as a 16-bit timer or event counter with automatic reload. If DCEN bit in T2MOD is cleared, timer 2 behaves as in 80C52 (refer to the ATMEL Wireless and Micrcontrollers 8-bit Microcontroller Hardware description). If DCEN bit is set, timer 2 acts as an Up/down timer/counter as shown in Figure 5. In this mode the T2EX pin controls the direction of count.

When T2EX is high, timer 2 counts up. Timer overflow occurs at FFFFh which sets the TF2 flag and generates an interrupt request. The overflow also causes the 16-bit value in RCAP2H and RCAP2L registers to be loaded into the timer registers TH2 and TL2.

When T2EX is low, timer 2 counts down. Timer underflow occurs when the count in the timer registers TH2 and TL2 equals the value stored in RCAP2H and RCAP2L registers. The underflow sets TF2 flag and reloads FFFFh into the timer registers.

The EXF2 bit toggles when timer 2 overflows or underflows according to the the direction of the count. EXF2 does not generate any interrupt. This bit can be used to provide 17-bit resolution.

Figure 10. PCA Compare Mode and PCA Watchdog Timer



* Only for Module 4

Before enabling ECOM bit, CCAPnL and CCAPnH should be set with a non zero value, otherwise an unwanted match could happen. Writing to CCAPnH will set the ECOM bit.

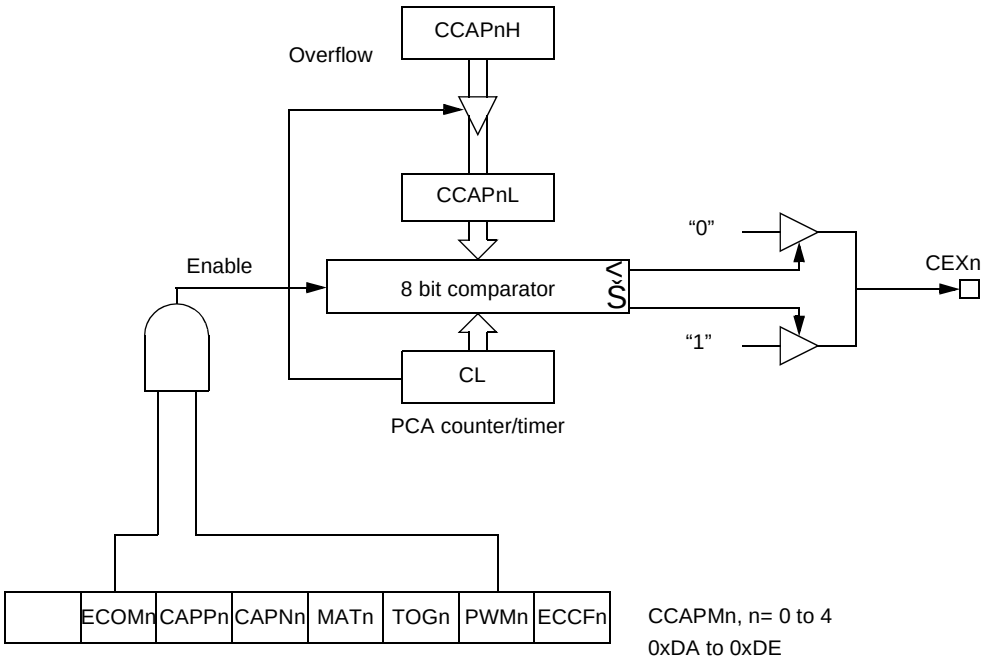
Once ECOM set, writing CCAPnL will clear ECOM so that an unwanted match doesn't occur while modifying the compare value. Writing to CCAPnH will set ECOM. For this reason, user software should write CCAPnL first, and then CCAPnH. Of course, the ECOM bit can still be controlled by accessing to CCAPMn register.

High Speed Output Mode

In this mode the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set (See Figure 11).

A prior write must be done to CCAPnL and CCAPnH before writing the ECOMn bit.

Figure 12. PCA PWM Mode



PCA Watchdog Timer

An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed. Figure 10 shows a diagram of how the watchdog works. The user pre-loads a 16-bit value in the compare registers. Just like the other compare modes, this 16-bit value is compared to the PCA timer value. If a match is allowed to occur, an internal reset will be generated. This will not cause the RST pin to be driven high.

In order to hold off the reset, the user has three options:

- 1. periodically change the compare value so it will never match the PCA timer,
- 2. periodically change the PCA timer value so it will never match the compare values, or
- 3. disable the watchdog by clearing the WDTE bit before a match occurs and then re-enable it.

The first two options are more reliable because the watchdog timer is never disabled as in option #3. If the program counter ever goes astray, a match will eventually occur and cause an internal reset. The second option is also not recommended if other PCA modules are being used. Remember, the PCA timer is the time base for all modules; changing the time base for other modules would not be a good idea. Thus, in most applications the first solution is the best option.

This watchdog timer won't generate a reset out on the reset pin.

The SADEN byte is selected so that each slave may be addressed separately.
For slave A, bit 0 (the LSB) is a don't-care bit; for slaves B and C, bit 0 is a 1. To communicate with slave A only, the master must send an address where bit 0 is clear (e.g. 1111 0000b).
For slave A, bit 1 is a 1; for slaves B and C, bit 1 is a don't care bit. To communicate with slaves B and C, but not slave A, the master must send an address with bits 0 and 1 both set (e.g. 1111 0011b).
To communicate with slaves A, B and C, the master must send an address with bit 0 set, bit 1 clear, and bit 2 clear (e.g. 1111 0001b).

Broadcast Address

A broadcast address is formed from the logical OR of the SADDR and SADEN registers with zeros defined as don't-care bits, e.g.:

```
SADDR 0101 0110b
SADEN 1111 1100b
Broadcast =SADDR OR SADEN1111 111Xb
```

The use of don't-care bits provides flexibility in defining the broadcast address, however in most applications, a broadcast address is FFh. The following is an example of using broadcast addresses:

```
Slave A:SADDR1111 0001b
    SADEN1111 1010b
    Broadcast1111 1X11b,

Slave B:SADDR1111 0011b
    SADEN1111 1001b
    Broadcast1111 1X11B,

Slave C:SADDR=1111 0010b
    SADEN1111 1101b
    Broadcast1111 1111b
```

For slaves A and B, bit 2 is a don't care bit; for slave C, bit 2 is set. To communicate with all of the slaves, the master must send an address FFh. To communicate with slaves A and B, but not slave C, the master can send and address FBh.

Reset Addresses

On reset, the SADDR and SADEN registers are initialized to 00h, i.e. the given and broadcast addresses are XXXX XXXXb (all don't-care bits). This ensures that the serial port will reply to any address, and so, that it is backwards compatible with the 80C51 microcontrollers that do not support automatic address recognition.

Table 18. SADEN - Slave Address Mask Register (B9h)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b
Not bit addressable

Table 33. EETIM Register

EETIM (S:*0D3*h)
EEPROM timing Control Register

7	6	5	4	3	2	1	0
EETIM							

Bit Number	Bit Mnemonic	Description
7-0	EETIM	Write Timer Register The write timer register value is required to adapt the write time to the oscillator frequency Value = 5 * Fxtal (MHz) in normal mode, 10 * Fxtal in X2 mode. Example : Fxtal = 33 MHZ, EETIM = 0A5h

Reset Value = 0000 0000b

these devices is similar to that used by EPROM 87C51 but it is not identical and the commercially available programmers need to have support for the T89C51RD2.

The bootloader and the In Application Programming (IAP) routines are located in the last kilobyte of the Flash, leaving 63k bytes available for the application with ISP.

Flash Registers and Memory Map

The T89C51RD2 Flash memory uses several registers for his management:

- Flash control register is used to select the Flash memory spaces and launch the Flash programming sequence.
- Hardware registers can only be accessed through the parallel programming modes which are handled by the parallel programmer.
- Software registers are in a special page of the Flash memory which can be accessed through the API or with the parallel programming modes. This page, called "Extra Flash Memory", is not in the internal Flash program memory addressing space.

Flash Register

Figure 20. FCON register

FCON (S:D1h) Flash control register

7	6	5	4	3	2	1	0
FPL3	**FPL2**	**FPL1**	**FPL0**	**FPS**	**FMOD1**	**FMOD0**	**FBUSY**

Bit Number	Bit Mnemonic	Description
7-4	FPL3:0	**Programming Launch command bits** Write 5h followed by Ah to launch the programming.
3	FPS	**Flash Map Program Space** Clear to map the data space during MOVX Set to map the Flash space during MOVX (write) or MOVC (read) instructions (Write in the column latches)
2-1	FMOD1:0	**Flash Mode** **Select the addressed space** 00: User (0000h-FFFFh) 01: XAF 10: Hardware byte 11: reserved
0	FBUSY	**Flash Busy** Set by hardware when programming is in progress. Clear by hardware when programming is done. Can not be cleared by software

Reset Value = xxxx 0000b

The Flash programming application note and API source code are available on request.

Hardware register

The only hardware register of the T89C51RD2 is called Hardware Security Byte (HSB). After full Flash erasure, the content of this byte is FFh; each bit is active at low level.

Table 34. Program Lock bits

Program Lock Bits				Protection Description
Security level	LB0	LB1	LB2	
1	U	U	U	No program lock features enabled. MOVC instruction executed from external program memory returns non encrypted data.
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.
3	X	P	U	Same as 2, also verify through parallel programming interface is disabled.
4	X	X	P	Same as 3, also external execution is disabled.

Note: U: unprogrammed or "one" level.

Note: P: programmed or "zero" level.

Note: X:do not care

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

These security bits protect the code access through the parallel programming interface. They are set by default to level 4. The code access through the ISP is still possible and is controlled by the "software security bits" which are stored in the extra Flash memory accessed by the ISP firmware.

To load a new application with the parallel programmer, a chip erase must first be done. This will set the HSB in its inactive state and will erase the Flash memory, including the boot loader and the "Extra Flash Memory" (XAF). If needed, the 1K boot loader and the XAF content must be programmed in the Flash; the code is provided by ATMEL Wireless and Microcontrollers (see section 8.7.); the part reference can always be read using Flash parallel programming modes.

Default Values

The default value of the HSB provides parts ready to be programmed with ISP:

- SB: Cleared to secure the content of the HSB.
- BLJB: Cleared to force ISP operation.
- BLLB: Clear to protect the default boot loader.
- LB2-0: Security level four to protect the code from a parallel access with maximum security.

Software Registers

Several registers are used, in factory and by parallel programmers, to make copies of hardware registers contents. These values are used by ATMEL Wireless and Microcontrollers ISP (see section 8.7.).

These registers are in the "Extra Flash Memory" part of the Flash memory. This block is also called "XAF" or eXtra Array Flash. They are accessed in the following ways:

- Commands issued by the parallel memory programmer.
- Commands issued by the ISP software.

Calls of API issued by the application software.

They are several software registers described in Table 35

	Level 0	Level 1	Level 2
Manufacturer Info	Read only access allowed	Read only access allowed	Read only access allowed
Bootloader Info	Read only access allowed	Read only access allowed	Read only access allowed
Erase Block	Allowed	Not allowed	Not allowed
Full-chip Erase	Allowed	Allowed	Allowed
Blank Check	Allowed	Allowed	Allowed

Autobaud Performances

The ISP feature allows a wide range of baud rates in the user application. It is also adaptable to a wide range of oscillator frequencies. This is accomplished by measuring the bit-time of a single bit in a received character. This information is then used to program the baud rate in terms of timer counts based on the oscillator frequency. The ISP feature requires that an initial character (an uppercase U) be sent to the T89C51RD2 to establish the baud rate. Table 39 shows the autobaud capability.

Table 39. Autobaud Performances (Bootloader Revision 2.4)

Frequency (MHz) **Baudrate (bit/s)**	**1.8432**	**2**	**2.4576**	**3**	**3.6864**	**4**	**5**	**6**	**7.3728**	**8**
4800	OK	OK	OK	OK	OK	OK	OK	OK	OK	-
9600	OK	OK	OK	OK	OK	OK	-	OK	OK	OK
19200	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK
38400	-	OK	-	-	OK	OK	OK	OK	OK	OK
57600	-	-	-	-	OK	-	-	-	OK	-
115200	-	-	-	-	-	-	-	-	OK	-
Frequency (MHz) **Baudrate (bit/s)**	**10**	**11.0592**	**12**	**14.318**	**14.746**	**16**	**20**	**24**	**26.6**	**32**
4800	OK	OK	OK	OK	OK	-	-	-	-	-
9600	OK	OK	OK	OK	OK	OK	OK	OK	OK	-
19200	-	OK	OK	OK	OK	OK	OK	OK	OK	OK
38400	OK	OK	OK	OK	OK			OK	OK	OK
57600	-	OK	-		OK	OK	OK	OK	OK	OK
115200	-	OK	-		OK	-	-	-	-	OK

ISP Commands Summary

Table 40. ISP Commands Summary

Command	Command Name	Data[0]	Data[1]	Command Effect
00h	Program Data			Program Nb Data Byte. Bootloader will accept up to 16 (10h) data Bytes. The data Bytes should be 128 Byte page Flash boundary.
03h	Write Function	04h	00h	Erase SBV
		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
04h	Display Function	Data[0:1] = start address Data [2:3] = end address Data[4] = 00h -> Display data Data[4] = 01h -> Blank check		Display Data
				Blank Check
05h	Read Function	00h	00h	Manufacturer ID
			01h	Device ID #1
			02h	Device ID #2
			03h	Device ID #3
		07h	00h	Read SSB
			01h	Read BSB
			02h	Read SBV
			03h	Read Hardware Byte Copy
		08h	00h	Read Bootloader Version
		0Eh	00h	Read Device Boot ID1
			01h	Read Device Boot ID2

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 at Atmel's web site.

Set-up modes configuration

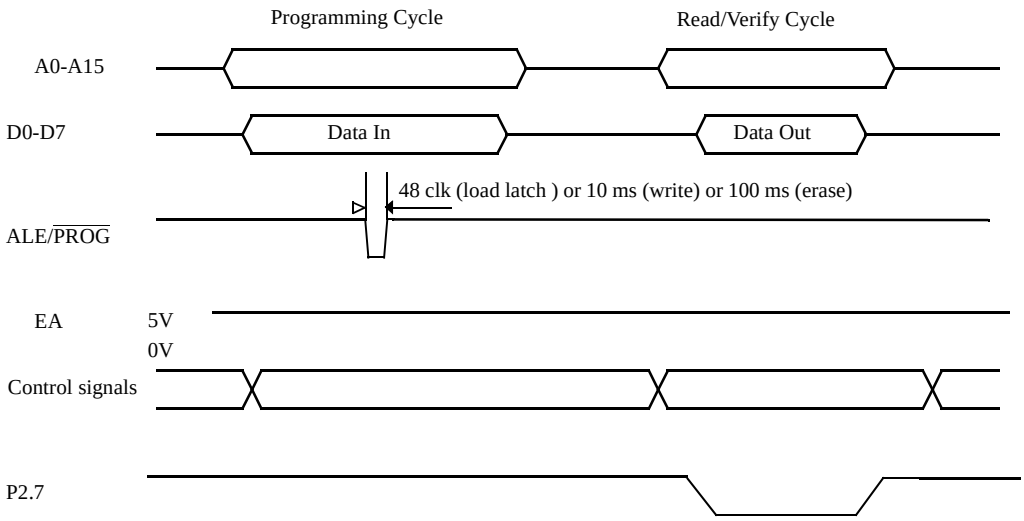
Control and program signals must be held at the levels indicated in the two following tables.

Mode Name	Mode	Rst	Psen	Ale _ _ |_|	EA	P2.6	P2.7	P3.6	P3.7	P0[7..0]
PELCK	Program or Erase Lock. Disable the Erasure or Programming access	1	0	_ _ |_|	1	1	0	1	0	xx
PEULCK	Program or Erase UnLock. Enable the Erasure or Programming access	1	0	Note 2	1	1	0	1	0	55-AA
PGMC	Write Code Data (byte) or write Page Always precedeed by PGML	1	0	_ _ |_| Internally timed	1	0	1	1	1	xx
PGML	Memory Page Load (up to 128 bytes)	1	0	Note 1	1	0	1	0	1	Din
PGMV	Read Code Data (byte)	1	0	1	1	0	_ _ |_|	1	1	Dout
VSB	Read Security Byte (=HSB)	1	0	1	1	0	_ _ |_|	0	1	Dout
PGMS	Write Security Byte (Note 3) (security byte = HSB)	1	0	10 ms	1	1	1	0	0	Din
CERR	Chip Erase User + XAF	1	0	100 ms	1	1	0	0	0	xx
PGXC	Write Byte or Page in Extra Memory (XAF) Always precedeed by PGXL	1	0	_ _ |_| Internally timed	1	1	1	0	1	xx
PGXL	Memory Page Load XAF (up to 128 bytes)	1	0	Note 1	1	1	1	0	1	Din
TMS	Read Signature bytes 30h (Manufacturer code) 31h (Device ID #1) 60h (Device ID #2) 61h (Device ID #3)	1	0	1	1	0	_ _ |_|	0	0	Dout = 58h D7h FCh FFh
RXAF	Read Extra Memory (XAF)	1	0	1	1	0	_ _ |_|	0	0	Dout

Mode Name	Mode	P1[7..0]	P2[5..0]	P3.0	P3.1	P3.2	P3.3	P3.4	P3.5
PELCK	Program or Erase Lock. Disable the Erasure or Programming access	xx	xx	x	x	x	1	x	x
PEULCK	Program or Erase UnLock. Enable the Erasure or Programming access	xx	xx	x	x	x	0	x	x
PGMC	Write Code Data (byte) or write Page Always precedeed by PGML	A7-A0	A13-A8	1	x	x	0	A14	A15
PGML	Memory Page Load (up to 128 bytes)	A7-A0	A13-A8	1	x	x	0	A14	A15
PGMV	Read Code Data (byte)	A7-A0	A13-A8	1	x	x	1	A14	A15
VSB	Read Security Byte (=HSB)	xx	xx	1	x	x	1	x	x
PGMS	Write lock Byte (Note 4) (security byte = HSB)	xx	xx	1	x	x	0	x	x
CERR	Chip Erase User + XAF	xx	xx	1	x	x	0	x	x
PGXC	Write Byte or Page Extra Memory (XAF) Always precedeed by PGXL	A7-A0 (0-7F)	xx	1	x	x	0	x	x
PGXL	Memory Page Load XAF (up to 128 bytes)	A7-A0 (0-7F)	xx	1	x	x	1	x	x
TMS	Read Signature bytes 30h (Manufacturer code) 31h (Device ID #1) 60h (Device ID #2) 61h (Device ID #3)	30h 31h 60h 61h	x	x	x	x	1	x	x
RXAF	Read Extra Memory (XAF)	Addr (0-7F)	00	1	x	x	0	x	x

1. In Page Load Mode the current byte is loaded on ALE rising edge.
2. After a power up all external test mode to program or to erase the Flash are locked to avoid any untimely programming or erasure.
   After each programming or erasure test mode, it's advised to lock this feature (test mode PELCK).
   To validate the test mode mode PEULCK the following sequence has to be applied:Test Mode PEULCK with ALE = 1.
   Pulse on ALE (min width=25clk) with P0=55 (P0 latched on ALE rising edge)
   Pulse on ALE (min width=25clk) with P0=AA (P0 latched on ALE rising edge)
3. The highest security bit (bit 7) is used to secure the 7 lowest bit erasure. The only way to erase this bit is to erase the whole Flash memory.
   Procedure to program security bits (After array programming):
   - program bit7 to 0, program all other bits (1 = erased, 0 = programmed).
   - test mode PGMS (din = HSB).
   Procedure to erase security byte:
   - test mode CERR: erase all array included HSB.
   - program hardware security byte to FF: test mode PGMS (din = FF).

Figure 32. Programming and Verification Signal's Waveform



Extra Memory Mapping

The memory mapping the T89C51RD2 software registers in the Extra Flash memory is described in the table below.

Table 42. Extra Row Memory Mapping (XAF)

	Address	Default content
Copy of device ID #3	0061h	FFh
Copy of device ID #2	0060h	FCh
Copy of device ID #1	0031h	D7h
Copy of Manufacturer Code: ATMEL	0030h	58h
Software Security Byte (level 1 by default)	0005h	FFh
Copy of HSB (level 4 by default and BLJB = 0)	0004h	18h or 1Bh
Software Boot Vector	0001h	FCh
Boot Status Byte	0000h	FFh

All other addresses are reserved

DC Parameters for Low Voltage

T_A = 0°C to +70°C; V_{SS} = 0 V; V_{CC} = 2.7 V to 3.6 V; F = 0 to 25 MHz
T_A = -40°C to +85°C; V_{SS} = 0 V; V_{CC} = 2.7 V to 3.6 V; F = 0 to 25 MHz

Table 45. DC Parameters for Low Voltage

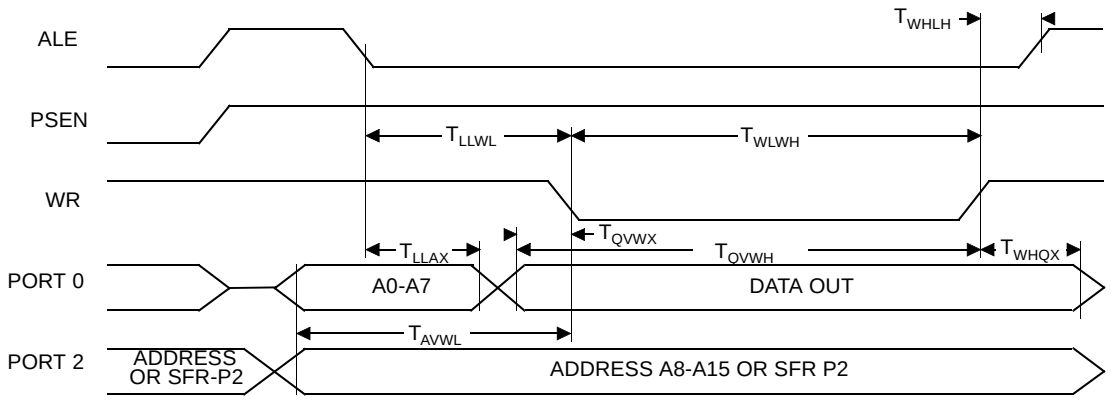
Symbol	Parameter	Min	Typ(5)	Max	Unit	Test Conditions
V_{IL}	Input Low Voltage	-0.5		0.2 V_{CC} - 0.1	V	
V_{IH}	Input High Voltage except XTAL1, RST	0.2 V_{CC} + 0.9		V_{CC} + 0.5	V	
V_{IH1}	Input High Voltage, XTAL1, RST	0.7 V_{CC}		V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage, ports 1, 2, 3, 4 and 5 (6)			0.45	V	I_{OL} = 0.8 mA(4)
V_{OL1}	Output Low Voltage, port 0, ALE, $\overline{PSEN}$ (6)			0.45	V	I_{OL} = 1.6 mA(4)
V_{OH}	Output High Voltage, ports 1, 2, 3, 4 and 5	0.9 V_{CC}			V	I_{OH} = -10 µA
V_{OH1}	Output High Voltage, port 0, ALE, $\overline{PSEN}$	0.9 V_{CC}			V	I_{OH} = -40 µA
I_{IL}	Logical 0 Input Current ports 1, 2, 3, 4 and 5			-50	µA	Vin = 0.45 V
I_{LI}	Input Leakage Current			±10	µA	0.45 V < Vin < V_{CC}
I_{TL}	Logical 1 to 0 Transition Current, ports 1, 2, 3, 4 and 5			-650	µA	Vin = 2.0 V
R_{RST}	RST Pulldown Resistor	50	90	200	kΩ	
CIO	Capacitance of I/O Buffer			10	pF	Fc = 1 MHz T_A = 25°C
I_{PD}	Power Down Current		1	50	µA	V_{CC} = 2.7 V to 3.6 V(3)
I_{CCOP}	Power Supply Current on normal mode			0.6 Freq (MHz) + 3	mA	V_{CC} = 3.6 V(1)
$I_{CCProgFlash}$	Power Supply Current during Flash Write / Erase		0.3 Freq (MHz) + 10	0.4 Freq (MHz) + 12	mA	V_{CC} = 3.6 V(1)
$I_{CCProgEE}$	Power Supply Current during EEprom data Write / Erase		0.7 Freq (MHz) + 3	0.7 Freq (MHz) + 18	mA	V_{CC} = 3.6 V(1)
I_{CCIDLE}	Power Supply Current on idle mode			0.3 Freq (MHz) + 2	mA	V_{CC} = 3.6 V(2)

4. Operating I_{CC} is measured with all output pins disconnected; XTAL1 driven with T_{CLCH}, T_{CHCL} = 5 ns (see Figure 36.), V_{IL} = V_{SS} + 0.5 V,
   V_{IH} = V_{CC} - 0.5V; XTAL2 N.C.; $\overline{EA}$ = RST = Port 0 = V_{CC}. I_{CC} would be slightly higher if a crystal oscillator used (see Figure 33.).
5. Idle I_{CC} is measured with all output pins disconnected; XTAL1 driven with T_{CLCH}, T_{CHCL} = 5 ns, V_{IL} = V_{SS} + 0.5 V, V_{IH} = V_{CC} - 0.5 V; XTAL2 N.C; Port 0 = V_{CC}; $\overline{EA}$ = RST = V_{SS} (see Figure 34.).
6. Power Down I_{CC} is measured with all output pins disconnected; $\overline{EA}$ = V_{SS}, PORT 0 = V_{CC}; XTAL2 NC.; RST = V_{SS} (see Figure 35.). In addition, the WDT must be inactive and the POF flag must be set.
7. Capacitance loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the V_{OL}s of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make 1 to 0

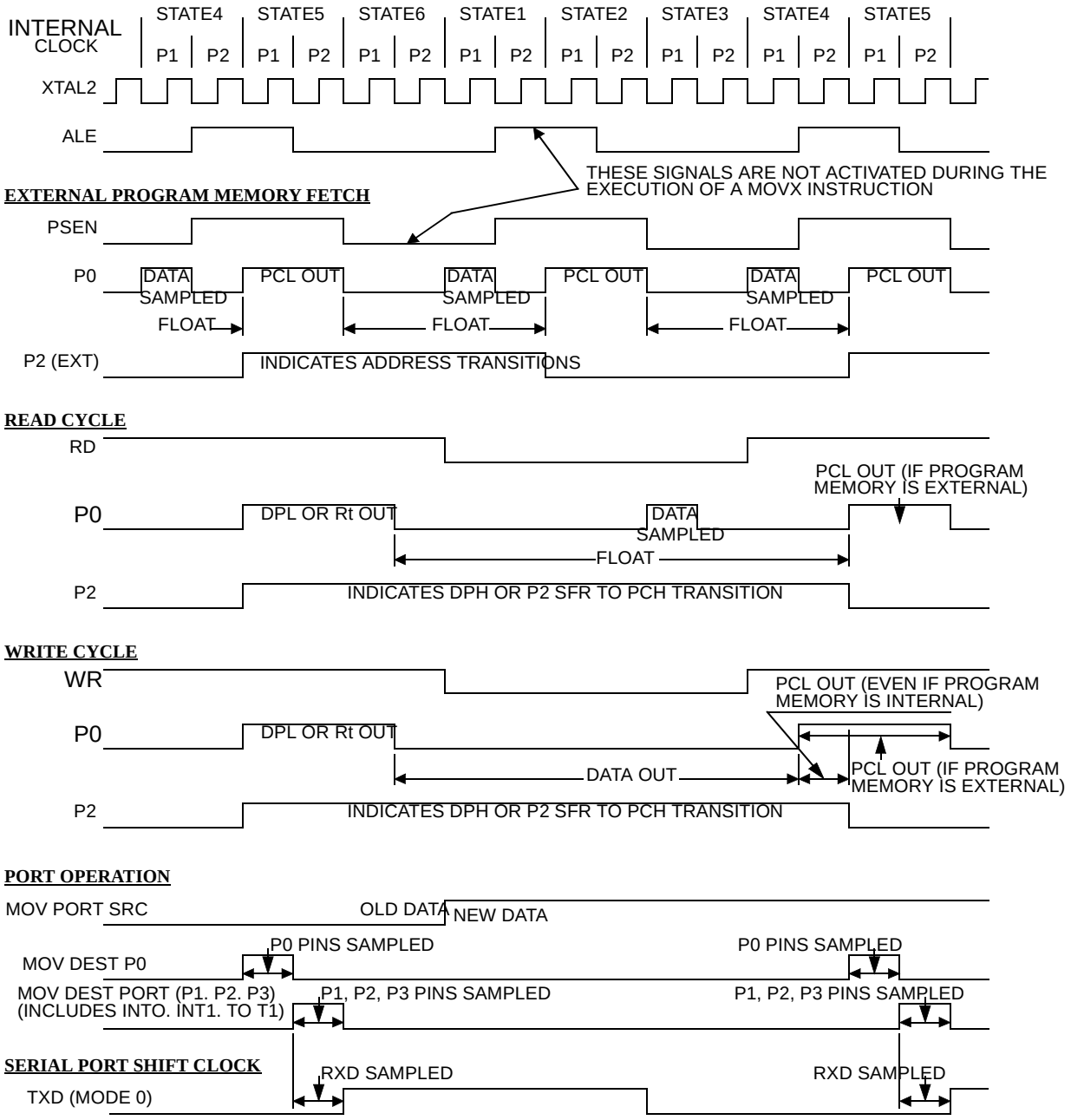
Table 50. AC Parameters for a Variable Clock

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

External Data Memory Write Cycle

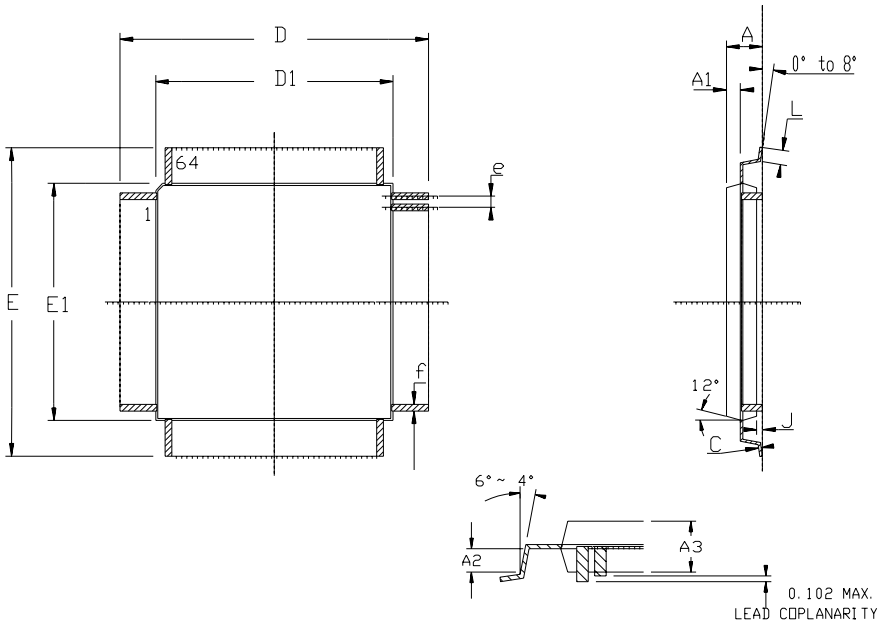


Clock Waveforms Valid in normal clock mode. In X2 mode XTAL2 must be changed to XTAL2/2.



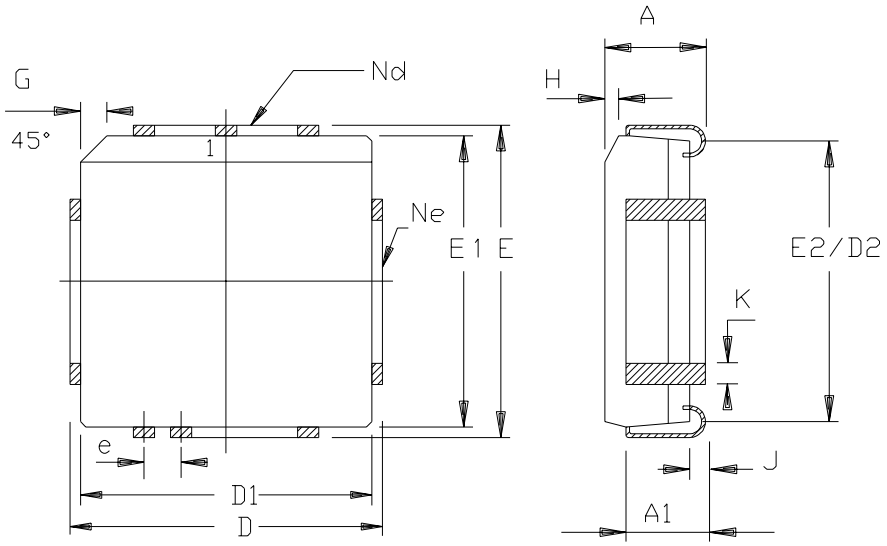
This diagram indicates when signals are clocked internally. The time it takes the signals to propagate to the pins, however, ranges from 25 to 125 ns. This propagation delay is dependent on variables such as temperature and pin loading. Propagation also varies from output to output and component. Typically though (T_A=25°C fully loaded) $\overline{RD}$ and $\overline{WR}$ propagation delays are approximately 50ns. The other signals are typically 85 ns. Propagation delays are incorporated in the AC specifications.

VQFP64



	MM		INCH	
	Min	Max	Min	Max
A	-	1.60	-	.063
A1	0.64 REF		.025 REF	
A2	0.64 REF		.025 REF	
A3	1.35	1.45	.053	.057
D	11.75	12.25	.463	.483
D1	9.90	10.10	.390	.398
E	11.75	12.25	.463	.483
E1	9.90	10.10	.390	.398
J	0.05	-	.002	-
L	0.45	0.75	.018	.030
e	0.50 BSC		.0197 BSC	
f	0.25 BSC		.010 BSC	

PLCC68



	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	
PKG STD	00			