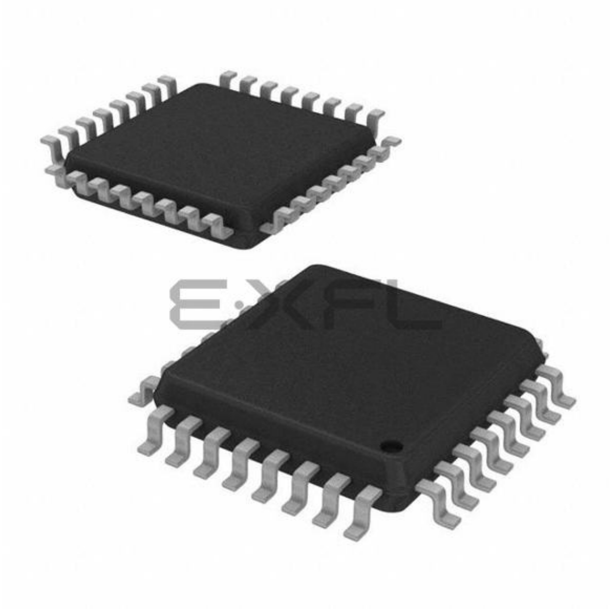
Microchip Technology - T89C5115-RATIM Datasheet



Welcome to E-XFL.COM

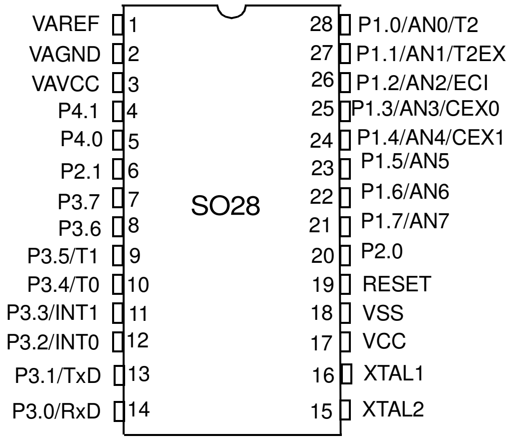
What is "Embedded - Microcontrollers"?

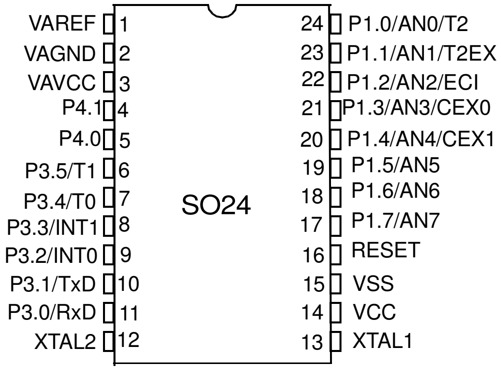
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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.

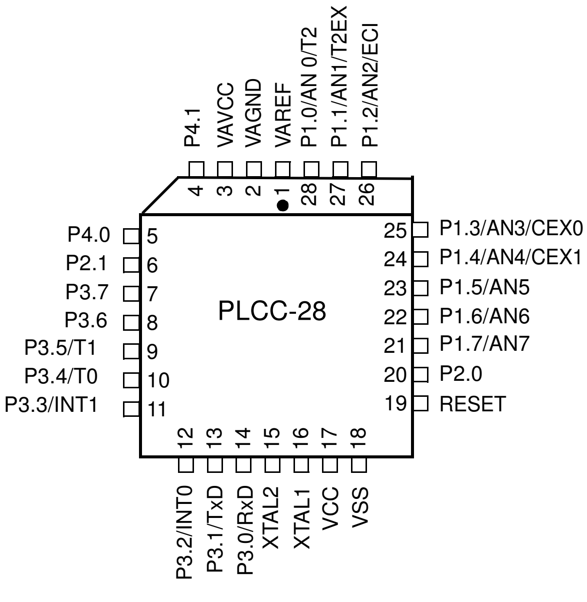
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	80C51
Core Size	8-Bit
Speed	40MHz
Connectivity	UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	20
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-VQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/t89c5115-ratim

Pin Configurations







Clock

The T89C5115 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

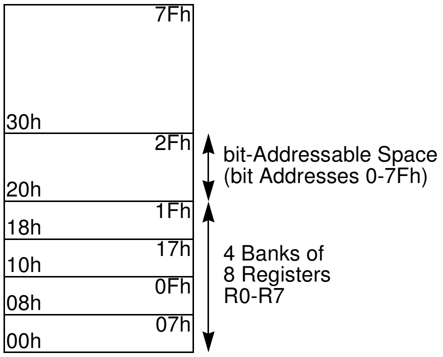
The X2 bit in the CKCON register (See Table 11) allows switching from 12 clock cycles per instruction to 6 clock cycles and vice versa. At reset, the standard speed is activated (STD mode).

Setting this bit activates the X2 feature (X2 Mode) for the CPU Clock only (See Figure 3).

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 3. 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 4 shows the mode switching waveforms.

Figure 10. Lower 128 Bytes Internal RAM Organization



Upper 128 Bytes RAM

The upper 128 Bytes of RAM are accessible from address 80h to FFh using only indirect addressing mode.

Expanded RAM

The on-chip 256 Bytes of expanded RAM (XRAM) are accessible from address 0000h to 00FFh using indirect addressing mode through MOVX instructions. In this address range.

Note: Lower 128 Bytes RAM, Upper 128 Bytes RAM, and expanded RAM are made of volatile memory cells. This means that the RAM content is indeterminate after power-up and must then be initialized properly.

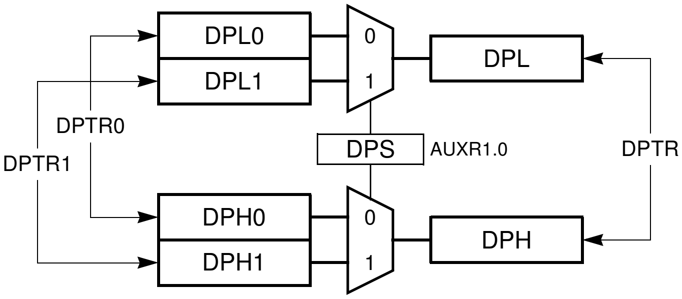
Dual Data Pointer

Description

The T89C5115 implements a second data pointer for speeding up code execution and reducing code size in case of intensive usage of external memory accesses.

DPTR0 and DPTR1 are Seen by the CPU as DPTR and are accessed using the SFR addresses 83h and 84h that are the DPH and DPL addresses. The DPS bit in AUXR1 register (See Figure 18) is used to select whether DPTR is the data pointer 0 or the data pointer 1 (See Figure 11).

Figure 11. Dual Data Pointer Implementation



Application

Software can take advantage of the additional data pointers to both increase speed and reduce code size, for example, block operations (copy, compare…) are well served by using one data pointer as a “source” pointer and the other one as a “destination” pointer. Hereafter is an example of block move implementation using the two pointers and coded in assembler. The latest C compiler takes also advantage of this feature by providing enhanced algorithm libraries.

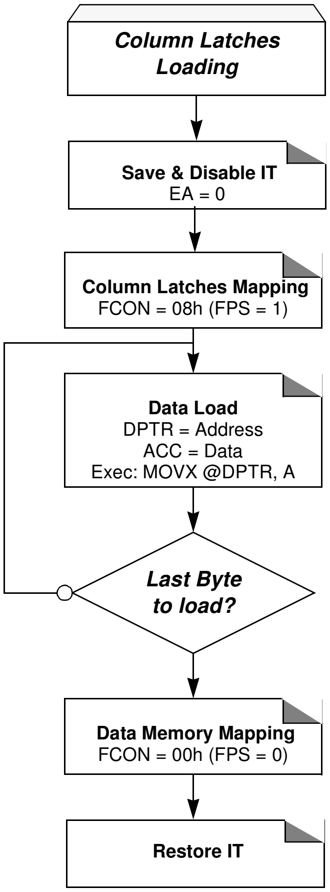
The INC instruction is a short (2 Bytes) and fast (6 machine cycle) way to manipulate the DPS bit in the AUXR1 register. However, note that the INC instruction does not directly force the DPS bit to a particular state, but simply toggles it. In simple routines, such as the block move example, only the fact that DPS is toggled in the proper sequence matters, not its actual value. In other words, the block move routine works the same whether DPS is 0 or 1 on entry.

```
; ASCII block move using dual data pointers
; Modifies DPTR0, DPTR1, A and PSW
; Ends when encountering NULL character
; Note: DPS exits opposite to the entry state unless an extra INC AUXR1 is
added

AUXR1EQU0A2h

move:movDPTR,#SOURCE ; address of SOURCE
  incAUXR1 ; switch data pointers
  movDPTR,#DEST ; address of DEST
mv_loop:incAUXR1; switch data pointers
  movxA,@DPTR; get a byte from SOURCE
  incDPTR; increment SOURCE address
  incAUXR1; switch data pointers
  movx@DPTR,A; write the byte to DEST
  incDPTR; increment DEST address
  jnzmv_loop; check for NULL terminator
end_move:
```

Figure 14. Column Latches Loading Procedure[1]



Note: 1. 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 15:

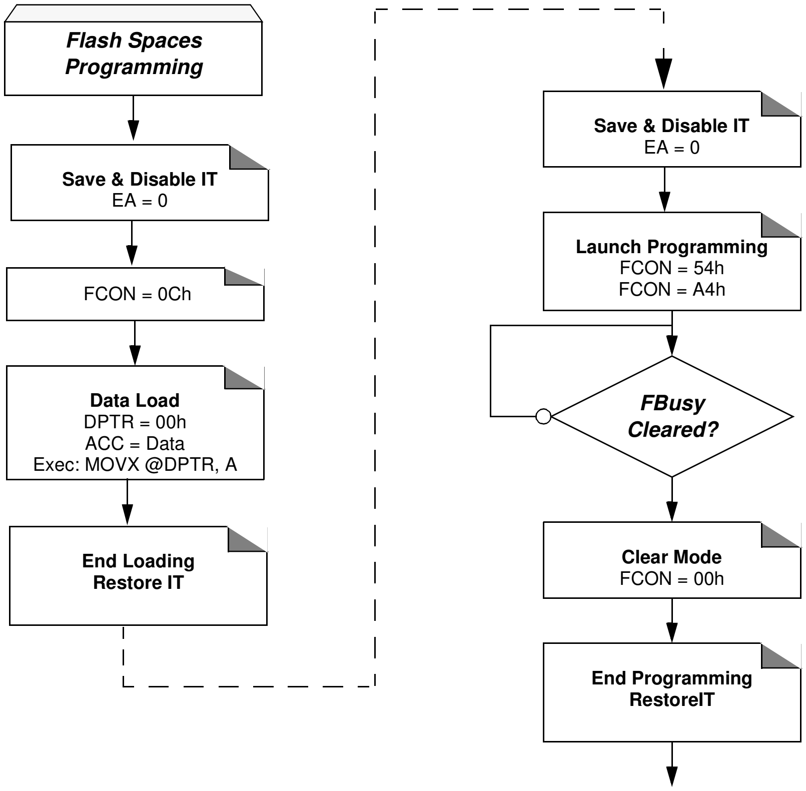
- Load up to one page of data in the column latches from address 0000h to 3FFFh.
- Save then disable the interrupts.
- Launch the programming by writing the data sequence 50h followed by A0h in FCON register.This step must be executed from FM1.
  The end of the programming indicated by the FBUSY flag cleared.
- Restore the interrupts.

Extra Row

The following procedure is used to program the Extra Row space and is summarized in Figure 15:

- Load data in the column latches from address FF80h to FFFFh.
- Save then disable the interrupts.
- Launch the programming by writing the data sequence 52h followed by A2h in FCON register. This step of the procedure must be executed from FM1.
  The end of the programming indicated by the FBUSY flag cleared.
- Restore the interrupts.

Figure 16. Hardware Programming Procedure



Reading the Flash Spaces

User

The following procedure is used to read the User space:

- Read one byte in Accumulator by executing MOVC A,@A+DPTR with A+DPTR is the address of the code byte to read.

Note: FCON must be cleared (00h) when not used.

Extra Row

The following procedure is used to read the Extra Row space and is summarized in Figure 17:

- Map the Extra Row space by writing 02h in FCON register.
- Read one byte in Accumulator by executing MOVC A,@A+DPTR with A= 0 & DPTR= FF80h to FFFFh.
- Clear FCON to unmap the Extra Row.

Hardware Security Byte

The following procedure is used to read the Hardware Security Byte and is summarized in Figure 17:

- Map the Hardware Security space by writing 04h in FCON register.
- Read the byte in Accumulator by executing MOVC A,@A+DPTR with A= 0 & DPTR= 0000h.
- Clear FCON to unmap the Hardware Security Byte.

Sharing Instructions

Table 26. Instructions shared

Action	RAM	ERAM	EEPROM DATA	Boot FLASH	FM0	Hardware Byte	XROW
Read	MOV	MOVX	MOVX	MOVC	MOVC	MOVC	MOVC
Write	MOV	MOVX	MOVX	-	by cl	by cl	by cl

Note: by cl : using Column Latch

Table 27. Read MOVX A, @DPTR

EEE bit in EECON Register	FPS in FCON Register	ENBOOT	ERAM	EEPROM DATA	Flash Column Latch
0	0	X	OK		
0	1	X	OK		
1	0	X		OK	
1	1	X	OK		

Table 28. Write MOVX @DPTR,A

EEE bit in EECON Register	FPS bit in FCON Register	ENBOOT	ERAM	EEPROM Data	Flash Column Latch
0	0	X	OK		
0	1	X			OK
1	0	X		OK	
1	1	X			OK

Hardware Security Byte

Table 31. 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 Jump bit** - 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** (see Table 22)

After erasing the chip in parallel mode, the default value is : FFh

The erasing in ISP mode (from bootloader) does not modify this byte.

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

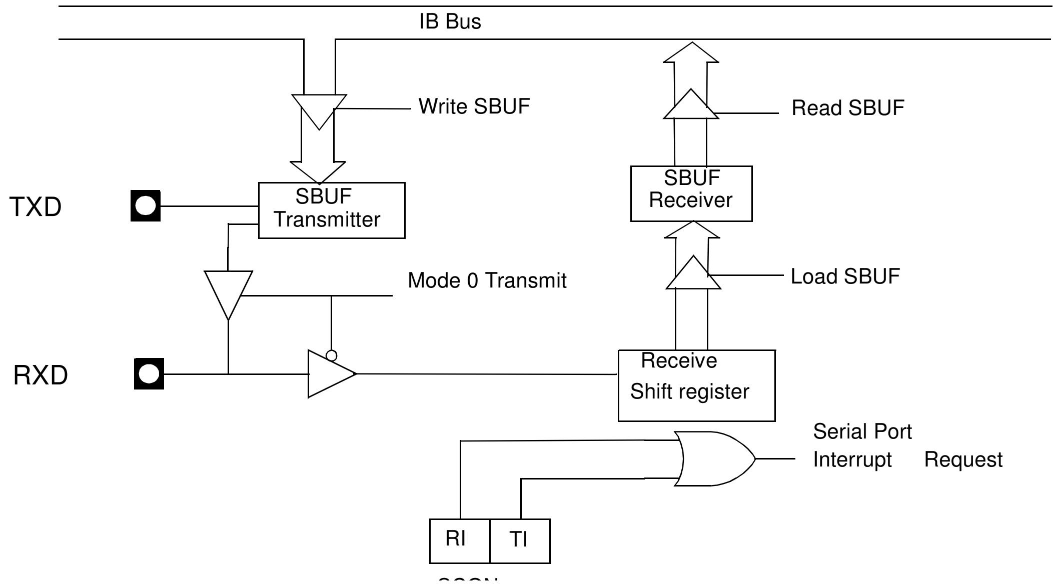
Serial I/O Port

The T89C5115 I/O serial port is compatible with the I/O serial port in the 80C52.
It provides both synchronous and asynchronous communication modes. It operates as a Universal Asynchronous Receiver and Transmitter (UART) in three full-duplex modes (Modes 1, 2 and 3). Asynchronous transmission and reception can occur simultaneously and at different baud rates

Serial I/O port includes the following enhancements:

- Framing error detection
- Automatic address recognition

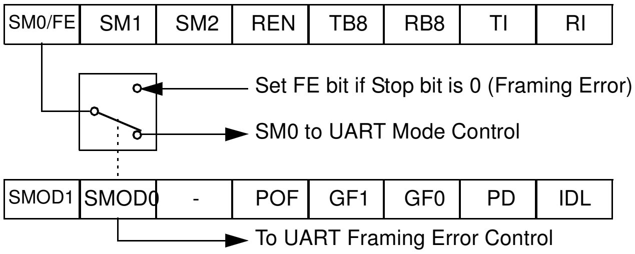
Figure 20. Serial I/O Port Block Diagram



Framing Error Detection

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

Figure 21. Framing Error Block Diagram



When this feature is enabled, the receiver checks each incoming data frame for a valid stop bit. An invalid stop bit may result from noise on the serial lines or from simultaneous transmission by two CPUs. If a valid stop bit is not found, the Framing Error bit (FE) in SCON register bit is set.

The software may examine the FE bit after each reception to check for data errors. Once set, only software or a reset clears the FE bit. Subsequently received frames with valid stop bits cannot clear the FE bit. When the FE feature is enabled, RI rises on the stop bit instead of the last data bit (See Figure 22 and Figure 23).

Registers

Table 32. SCON Register

SCON (S:98h)
Serial Control Register

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

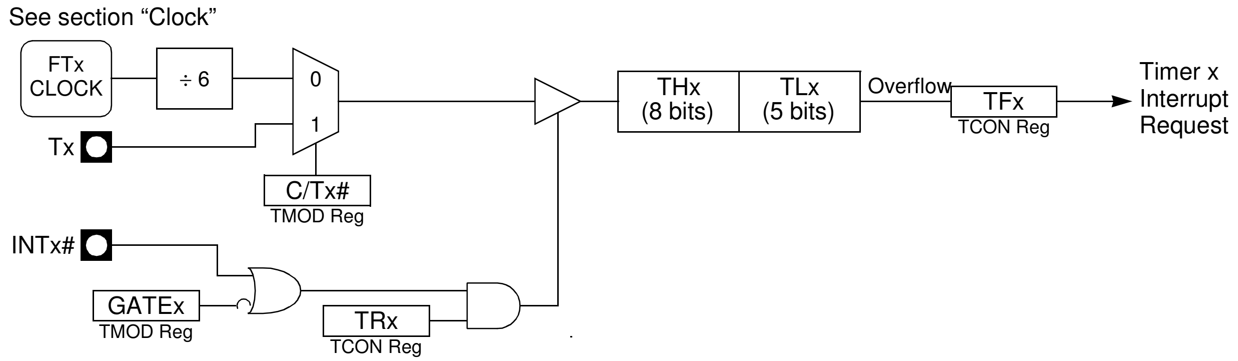
Bit Number	Bit Mnemonic	Description
7	FE	**Framing Error bit (SMOD0 = 1**) Clear to reset the error state, not cleared by a valid stop bit. Set by hardware when an invalid stop bit is detected.
6	SM0	**Serial port Mode bit 0 (SMOD0 = 0)** Refer to SM1 for serial port mode selection.
	SM1	**Serial port Mode bit 1** SM0 SM1 Mode Baud Rate 0 0 Shift Register F_{XTAL}/12 (or F_{XTAL}/6 in mode X2) 0 1 8-bit UART Variable 1 0 9bit UART F_{XTAL}/64 or F_{XTAL}/32 1 1 9bit UART Variable
5	SM2	**Serial port Mode 2 bit/Multiprocessor Communication Enable bit** Clear to disable multiprocessor communication feature. Set to enable multiprocessor communication feature in mode 2 and 3.
4	REN	**Reception Enable bit** Clear to disable serial reception. Set to enable serial reception.
3	TB8	**Transmitter bit 8/Ninth bit to Transmit in Modes 2 and 3** Clear to transmit a logic 0 in the 9th bit. Set to transmit a logic 1 in the 9th bit.
2	RB8	**Receiver bit 8/Ninth bit Received in Modes 2 and 3** Cleared by hardware if 9th bit received is a logic 0. Set by hardware if 9th bit received is a logic 1.
1	TI	**Transmit Interrupt Flag** Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0 or at the beginning of the stop bit in the other modes.
0	RI	**Receive Interrupt Flag** Clear to acknowledge interrupt. Set by hardware at the end of the 8th bit time in mode 0, See Figure 22. and Figure 23. in the other modes.

Reset Value = 0000 0000b
bit addressable

Mode 0 (13-bit Timer)

Mode 0 configures Timer 0 as an 13-bit Timer which is set up as an 8-bit Timer (TH0 register) with a modulo 32 prescaler implemented with the lower five bits of TL0 register (See Figure 24). The upper three bits of TL0 register are indeterminate and should be ignored. Prescaler overflow increments TH0 register.

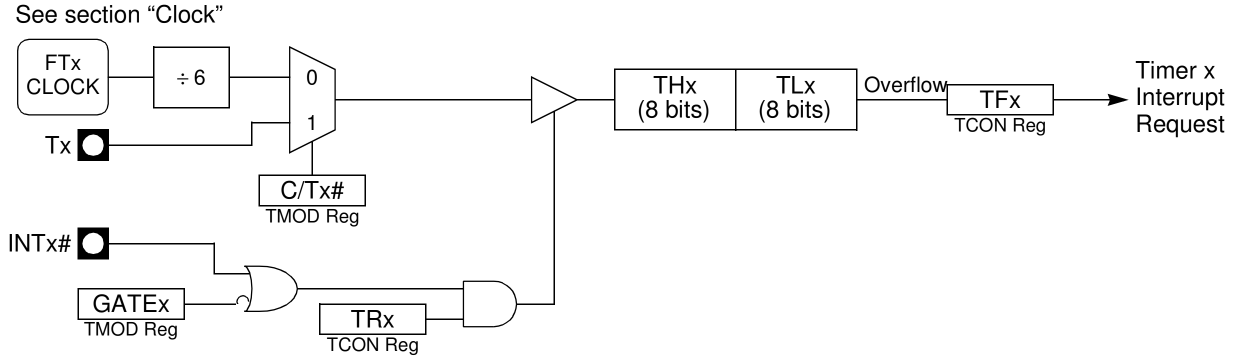
Figure 24. Timer/Counter x (x= 0 or 1) in Mode 0



Mode 1 (16-bit Timer)

Mode 1 configures Timer 0 as a 16-bit Timer with TH0 and TL0 registers connected in cascade (See Figure 25). The selected input increments TL0 register.

Figure 25. Timer/Counter x (x= 0 or 1) in Mode 1



Mode 2 (8-bit Timer with Auto-Reload)

Mode 2 configures Timer 0 as an 8-bit Timer (TL0 register) that automatically reloads from TH0 register (See Figure 26). TL0 overflow sets TF0 flag in TCON register and reloads TL0 with the contents of TH0, which is preset by software. When the interrupt request is serviced, hardware clears TF0. The reload leaves TH0 unchanged. The next reload value may be changed at any time by writing it to TH0 register.

- 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 generating an interrupt request.
- When Timer 0 is in mode 3, it uses Timer 1's overflow flag (TF1) and run control bit (TR1). For this situation, use Timer 1 only for applications that do not require an interrupt (such as a Baud Rate Generator for the Serial Port) and switch Timer 1 in and out of mode 3 to turn it off and on.
- It is important to stop Timer/Counter before changing mode.

Mode 0 (13-bit Timer)

Mode 0 configures Timer 1 as a 13-bit Timer, which is set up as an 8-bit Timer (TH1 register) with a modulo-32 prescaler implemented with the lower 5 bits of the TL1 register (See Figure 24). The upper 3 bits of TL1 register are ignored. Prescaler overflow increments TH1 register.

Mode 1 (16-bit Timer)

Mode 1 configures Timer 1 as a 16-bit Timer with TH1 and TL1 registers connected in cascade (See Figure 25). The selected input increments TL1 register.

Mode 2 (8-bit Timer with Auto-Reload)

Mode 2 configures Timer 1 as an 8-bit Timer (TL1 register) with automatic reload from TH1 register on overflow (See Figure 26). TL1 overflow sets TF1 flag in TCON register and reloads TL1 with the contents of TH1, which is preset by software. The reload leaves TH1 unchanged.

Mode 3 (Halt)

Placing Timer 1 in mode 3 causes it to halt and hold its count. This can be used to halt Timer 1 when TR1 run control bit is not available i.e. when Timer 0 is in mode 3.

Interrupt

Each Timer handles one interrupt source that is the timer overflow flag TF0 or TF1. This flag is set every time an overflow occurs. Flags are cleared when vectoring to the Timer interrupt routine. Interrupts are enabled by setting ETx bit in IEN0 register. This assumes interrupts are globally enabled by setting EA bit in IEN0 register.

Figure 28. Timer Interrupt System

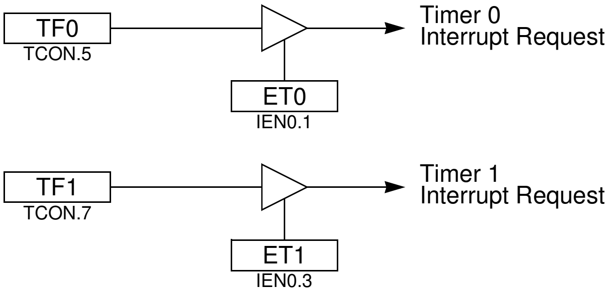


Table 38. TMOD Register
TMOD (S:89h)
Timer/Counter Mode Control Register

7	6	5	4	3	2	1	0
GATE1	C/T1#	M11	M01	GATE0	C/T0#	M10	M00

Bit Number	Bit Mnemonic	Description
7	GATE1	**Timer 1 Gating Control bit** Clear to enable Timer 1 whenever TR1 bit is set. Set to enable Timer 1 only while INT1# pin is high and TR1 bit is set.
6	C/T1#	**Timer 1 Counter/Timer Select bit** Clear for Timer operation: Timer 1 counts the divided-down system clock. Set for Counter operation: Timer 1 counts negative transitions on external pin T1.
5	M11	**Timer 1 Mode Select bits** M11 M01 Operating mode 0 0 Mode 0: 8-bit Timer/Counter (TH1) with 5bit prescaler (TL1). 0 1 Mode 1: 16-bit Timer/Counter. 1 1 Mode 3: Timer 1 halted. Retains count. 1 0 Mode 2: 8-bit auto-reload Timer/Counter (TL1).[1]
4	M01	
3	GATE0	**Timer 0 Gating Control bit** Clear to enable Timer 0 whenever TR0 bit is set. Set to enable Timer/Counter 0 only while INT0# pin is high and TR0 bit is set.
2	C/T0#	**Timer 0 Counter/Timer Select bit** Clear for Timer operation: Timer 0 counts the divided-down system clock. Set for Counter operation: Timer 0 counts negative transitions on external pin T0.
1	M10	**Timer 0 Mode Select bit** M10 M00 Operating mode 0 0 Mode 0: 8-bit Timer/Counter (TH0) with 5bit prescaler (TL0). 0 1 Mode 1: 16-bit Timer/Counter. 1 0 Mode 2: 8-bit auto-reload Timer/Counter (TL0).[2] 1 1 Mode 3: TL0 is an 8-bit Timer/Counter. TH0 is an 8-bit Timer using Timer 1's TR0 and TF0 bits.
0	M00	

Reset Value = 0000 0000b

Notes: 1. Reloaded from TH1 at overflow.
2. Reloaded from TH0 at overflow.

Table 39. TH0 Register
TH0 (S:8Ch)
Timer 0 High Byte Register

7	6	5	4	3	2	1	0

Bit Number	Bit Mnemonic	Description
7:0		**High Byte of Timer 0**

Reset Value = 0000 0000b

Timer 2

The T89C5115 Timer 2 is compatible with Timer 2 in the 80C52.

It is a 16-bit timer/counter: the count is maintained by two eightbit timer registers, TH2 and TL2 that are cascade-connected. It is controlled by T2CON register (See Table 44) and T2MOD register (See Table 45). Timer 2 operation is similar to Timer 0 and Timer 1. C/$\overline{T2}$ selects $F_{T2\ clock}$/6 (timer operation) or external pin T2 (counter operation) as timer clock. Setting TR2 allows TL2 to be incremented by the selected input.

Timer 2 includes the following enhancements:

- Auto-reload mode (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. This feature is controlled by the DCEN bit in T2MOD register (See Table 44). Setting the DCEN bit enables Timer 2 to count up or down as shown in Figure 29. In this mode the T2EX pin controls the counting direction.

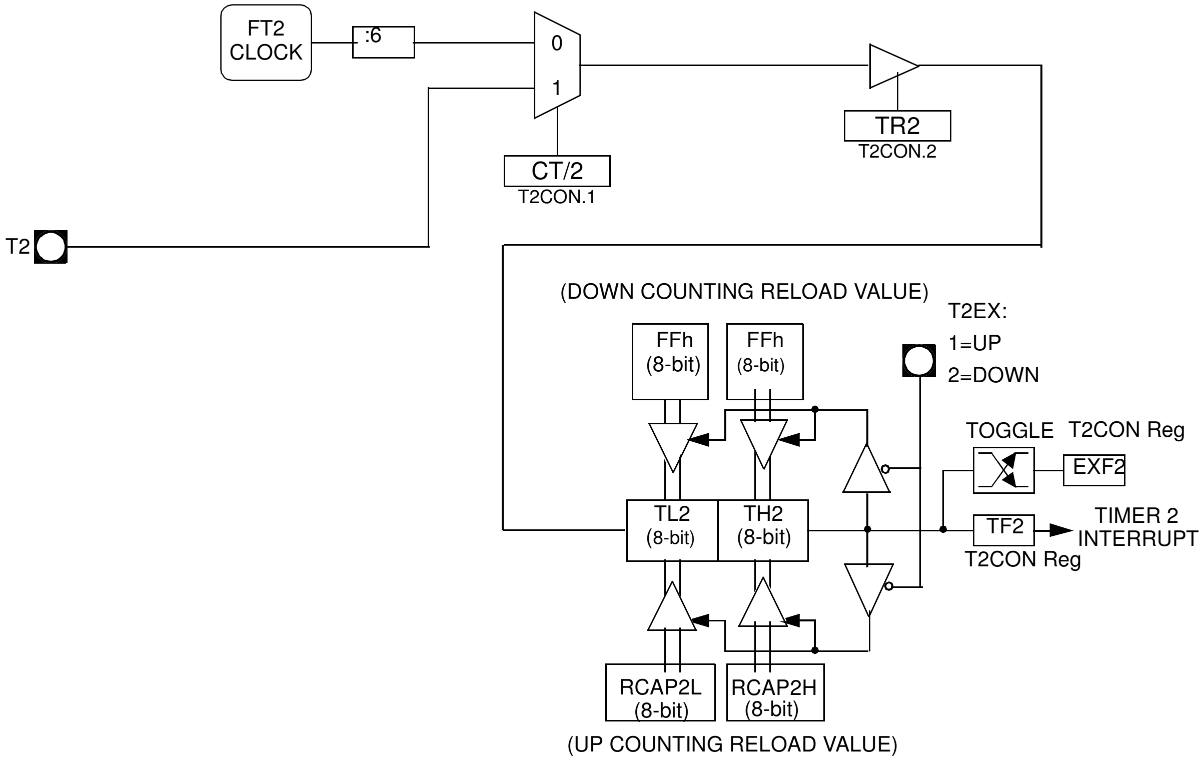
When T2EX is high, Timer 2 counts up. Timer overflow occurs at FFFFh which sets the TF2 flag and generates an interrupt request. The overflow also causes the 16-bit value in RCAP2H and RCAP2L registers to be loaded into the timer registers TH2 and TL2.

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 overflow or underflow, depending on the direction of the count. EXF2 does not generate an interrupt. This bit can be used to provide 17-bit resolution.

Figure 29. Auto-Reload Mode Up/Down Counter

See section "Clock"



Registers

Table 43. T2CON Register
T2CON (S:C8h)
Timer 2 Control Register

7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2#	CP/RL2#

Bit Number	Bit Mnemonic	Description
7	TF2	**Timer 2 Overflow Flag** TF2 is not set if RCLK=1 or TCLK = 1. Must be cleared by software. Set by hardware on Timer 2 overflow.
6	EXF2	**Timer 2 External Flag** Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. Set to cause the CPU to vector to Timer 2 interrupt routine when Timer 2 interrupt is enabled. Must be cleared by software.
5	RCLK	**Receive Clock bit** Clear to use timer 1 overflow as receive clock for serial port in mode 1 or 3. Set to use Timer 2 overflow as receive clock for serial port in mode 1 or 3.
4	TCLK	**Transmit Clock bit** Clear to use timer 1 overflow as transmit clock for serial port in mode 1 or 3. Set to use Timer 2 overflow as transmit clock for serial port in mode 1 or 3.
3	EXEN2	**Timer 2 External Enable bit** Clear to ignore events on T2EX pin for Timer 2 operation. Set to cause a capture or reload when a negative transition on T2EX pin is detected, if Timer 2 is not used to clock the serial port.
2	TR2	**Timer 2 Run Control bit** Clear to turn off Timer 2. Set to turn on Timer 2.
1	C/T2#	**Timer/Counter 2 Select bit** Clear for timer operation (input from internal clock system: f_{OSC}). Set for counter operation (input from T2 input pin).
0	CP/RL2#	**Timer 2 Capture/Reload bit** If RCLK=1 or TCLK=1, CP/RL2# is ignored and timer is forced to auto-reload on Timer 2 overflow. Clear to auto-reload on Timer 2 overflows or negative transitions on T2EX pin if EXEN2=1. Set to capture on negative transitions on T2EX pin if EXEN2=1.

Reset Value = 0000 0000b
bit addressable

Table 52. WDTRST Register
WDTRST (S:A6h Write Only) – Watchdog Timer Enable register

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

Bit Number	Bit Mnemonic	Description
7	-	Watchdog Control Value

Reset Value = 1111 1111b

Note: The WDRST register is used to reset/enable the WDT by writing 1EH then E1H in sequence without instruction between these two sequences.

Figure 38. ADC Description

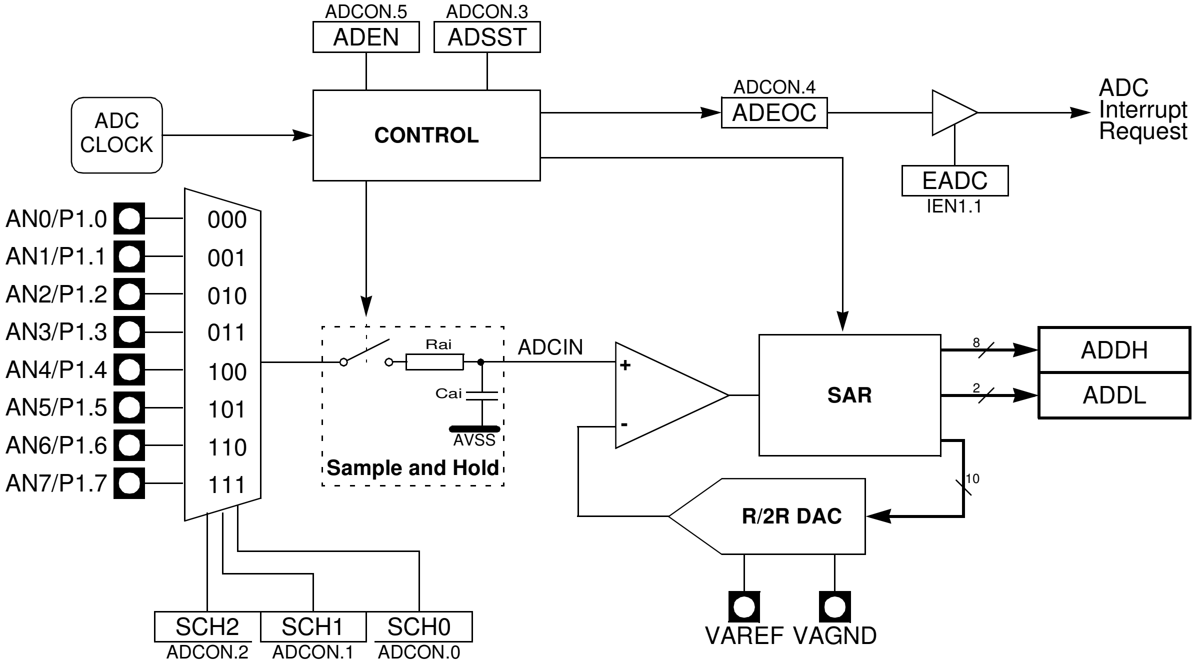
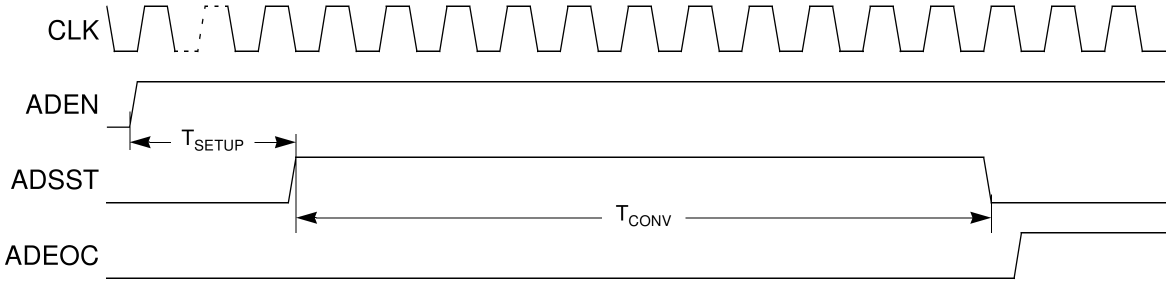


Figure 39 shows the timing diagram of a complete conversion. For simplicity, the figure depicts the waveforms in idealized form and do not provide precise timing information. For ADC characteristics and timing parameters refer to the section “AC Characteristics” of this datasheet.

Figure 39. Timing Diagram



Note: Tsetup min, see the AC Parameter for A/D conversion.
Tconv = 11 clock ADC = 1sample and hold + 10-bit conversion
The user must ensure that Tsetup time between setting ADEN and the start of the first conversion.

ADC Converter Operation

A start of single A/D conversion is triggered by setting bit ADSST (ADCON.3).

After completion of the A/D conversion, the ADSST bit is cleared by hardware.

The end-of-conversion flag ADEOC (ADCON.4) is set when the value of conversion is available in ADDH and ADDL, it must be cleared by software. If the bit EADC (IEN1.1) is set, an interrupt occur when flag ADEOC is set (See Figure 41). Clear this flag for re-arming the interrupt.

Note: Always leave Tsetup time before starting a conversion unless ADEN is permanently high. In this case one should wait Tsetup only before the first conversion

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. $I_{OL}/I_{OH} \geq \pm$ 20mA.

Clock Waveforms

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

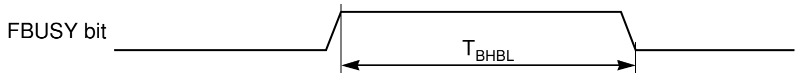
Flash/EEPROM Memory

Table 78. Memory AC Timing

V_{cc} = 3.0V to 5.5V, T_A = -40°C to +85°C

Symbol	Parameter	Min	Typ	Max	Unit
T_{BHBL}	Flash/EEPROM Internal Busy (Programming) Time		13	17	ms
N_{FCY}	Number of Flash/EEPROM Erase/Write Cycles	100 000			cycles
T_{FDR}	Flash/EEPROM Data Retention Time	10			years

Figure 49. Flash Memory - Internal Busy Waveforms

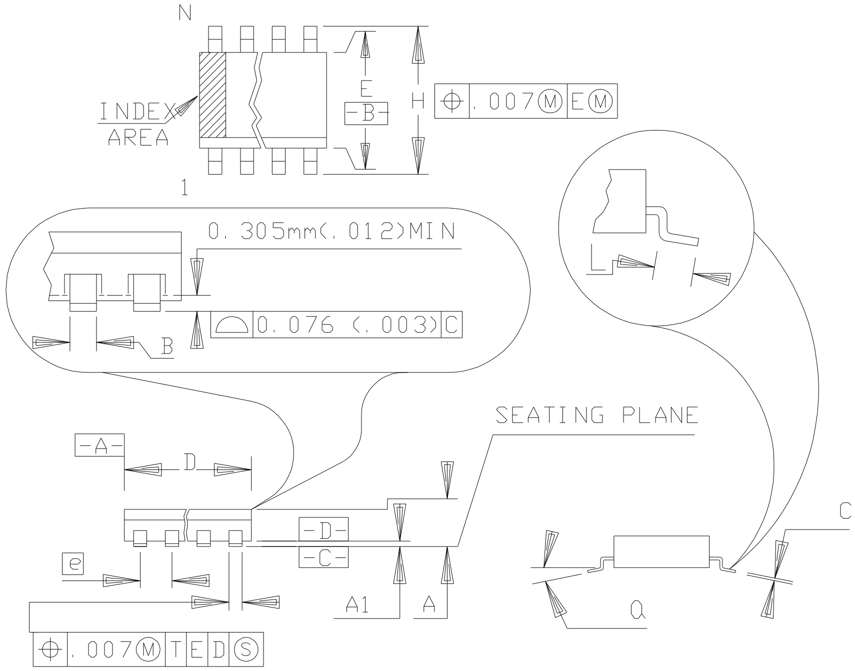


A/D Converter

Table 79. AC Parameters for A/D Conversion

Symbol	Parameter	Min	Typ	Max	Unit
T_{SETUP}		4			µs
ADC Clock Frequency			700		KHz

SOIC28



	MM		INCH	
A	2.29	2.54	.090	.100
A1	0.102	0.254	.004	.010
B	0.38	0.51	.015	.020
C	0.15	0.27	.006	.0105
D	20.83	21.08	.820	.830
E	10.03	10.29	.395	.405
e	1.27	BSC	.050	BSC
H	13.49	13.84	.531	.545
L	0.53	1.04	.021	.041
N	32		32	
α	0°	8°	0°	8°

Datasheet Revision History for T89C5115

Changes from 4128A-01/03 to 4128B-06/03

1. Changed the endurance of Flash to 100, 000 Write/Erase cycles.
2. Added note on Flash retention formula for V_{IH1}, in Section "Electrical Characteristics", page 97.

Changes from 4128B-06/03 to 4128C-12/03

1. Changed value of IPDMAX to 400, Section “Electrical Characteristics”, page 97.
2. PCA , CPS0, register correction, Section “PCA Registers”, page 78.
3. Cross Memory section added Section “Operation Cross Memory Access”, page 42.

Changes from 4128C - 12/03 to 4128D - 01/05

1. Various minor corrections throughout the document.

Changes from 4128D - 01/05 to 4128E - 10/05

1. Added green product ordering information.

Changes from 4128E - 10/05 to 4182F - 06/05

1. Minor corrections throughout the document.

Changes from 4182F - 06/05 to 4182G - 01/08

1. Removed non-green part numbers from ordering information.