DATA SHEET

P83C557E4/P80C557E4/P89C557E4
Single-chip 8-bit microcontroller

Product specification

1999 Mar 02

Supersedes data of 1999 Feb 15

Philips
Semiconductors

PHILIPS

1. FEATURES

- 80C51 central processing unit
- 32 K × 8 ROM respectively FEEPROM (Flash-EEPROM), expandable externally to 64 Kbytes
- ROM/FEEPROM Code protection
- 1024 × 8 RAM, expandable externally to 64 Kbytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- A 10-bit ADC with eight multiplexed analog inputs and programmable autoscan
- Two 8-bit resolution, pulse width modulation outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- I^2C-bus serial I/O port with byte oriented master and slave functions
- Full-duplex UART compatible with the standard 80C51
- On-chip watchdog timer
- 15 interrupt sources with 2 priority levels (2 to 6 external sources possible)
- Extended temperature range (–40 to +85°C)
- 4.5 to 5.5 V supply voltage range
- Frequency range for 80C51-family standard oscillator: 3.5 MHz to 16 MHz
- PLL oscillator with 32 kHz reference and software-selectable system clock frequency
- Seconds Timer
- Software enable/disable of ALE output pulse
- Electromagnetic compatibility improvements
- Wake-up from Power-down by external or seconds interrupt

2. GENERAL DESCRIPTION

The P80C557E4/P83C557E4/P89C557E4 (hereafter generically referred to as P8xC557E4) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family. The P8xC557E4 has the same instruction set as the 80C51. Three versions of the derivative exist:

- P83C557E4 — 32 Kbytes mask programmable ROM
- P80C557E4 — ROMless version of the P83C557E4
- P89C557E4 — 32 Kbytes FEEPROM (Flash-EEPROM)

The P8xC557E4 contains a non-volatile 32 Kbytes mask programmable ROM (P83C557E4) or electrically erasable FEEPROM respectively (P89C557E4), a volatile 1024 × 8 read/write data memory, five 8-bit I/O ports, one 8-bit input port, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a 15-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I^2C-bus), a “watchdog” timer, an on-chip oscillator and timing circuits. For systems that require extra capability the P8xC557E4 can be expanded using standard TTL compatible memories and logic.

In addition, the P8xC557E4 has two software selectable modes of power reduction — Idle Mode and power-down mode. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic as well as bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte, and 17 three- byte. With a 16 MHz system clock, 58% of the instructions are executed in 0.75 μs and 40% in 1.5 μs. Multiply and divide instructions require 3 μs.

3. ORDERING INFORMATION

EXTENDED TYPE NUMBER	PACKAGE			FREQUENCY RANGE (MHz)	TEMPERATURE RANGE (°C)
	NAME	DESCRIPTION	CODE		
ROMless					
P80C557E4EBB	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	0 to +70
P80C557E4EFB	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	–40 to +85
ROM coded					
P83C557E4EBB/YYY[1]	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	0 to +70
P83C557E4EFB/YYY[1]	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	–40 to +85
FEEPROM					
P89C557E4EBB	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	0 to +70
P89C557E4EFB	QFP80	Plastic Quad Flat Pack; 80 leads	SOT318-1	3.5 to 16	–40 to +85

NOTE:
1. YYY denotes the ROM code number

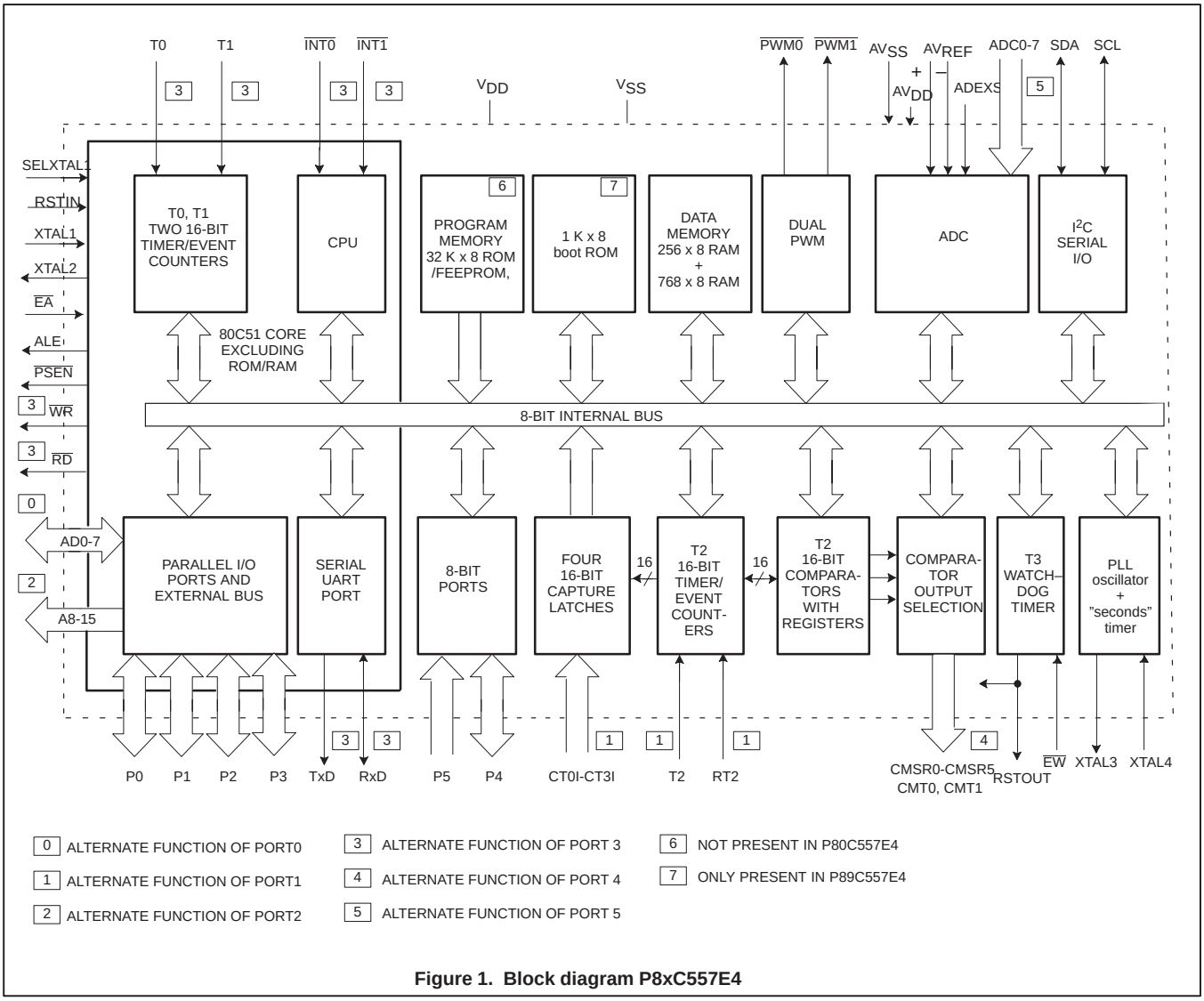
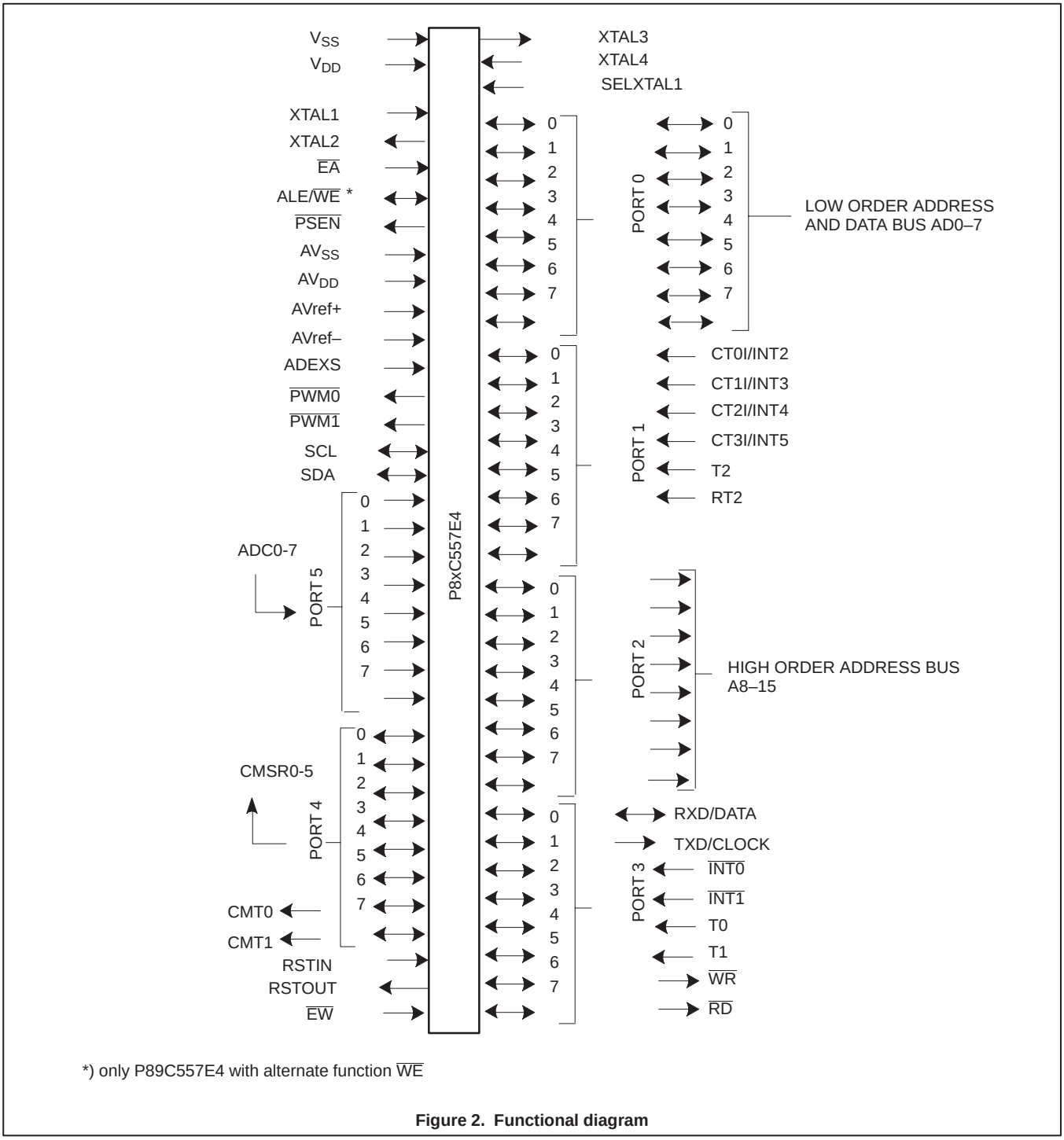


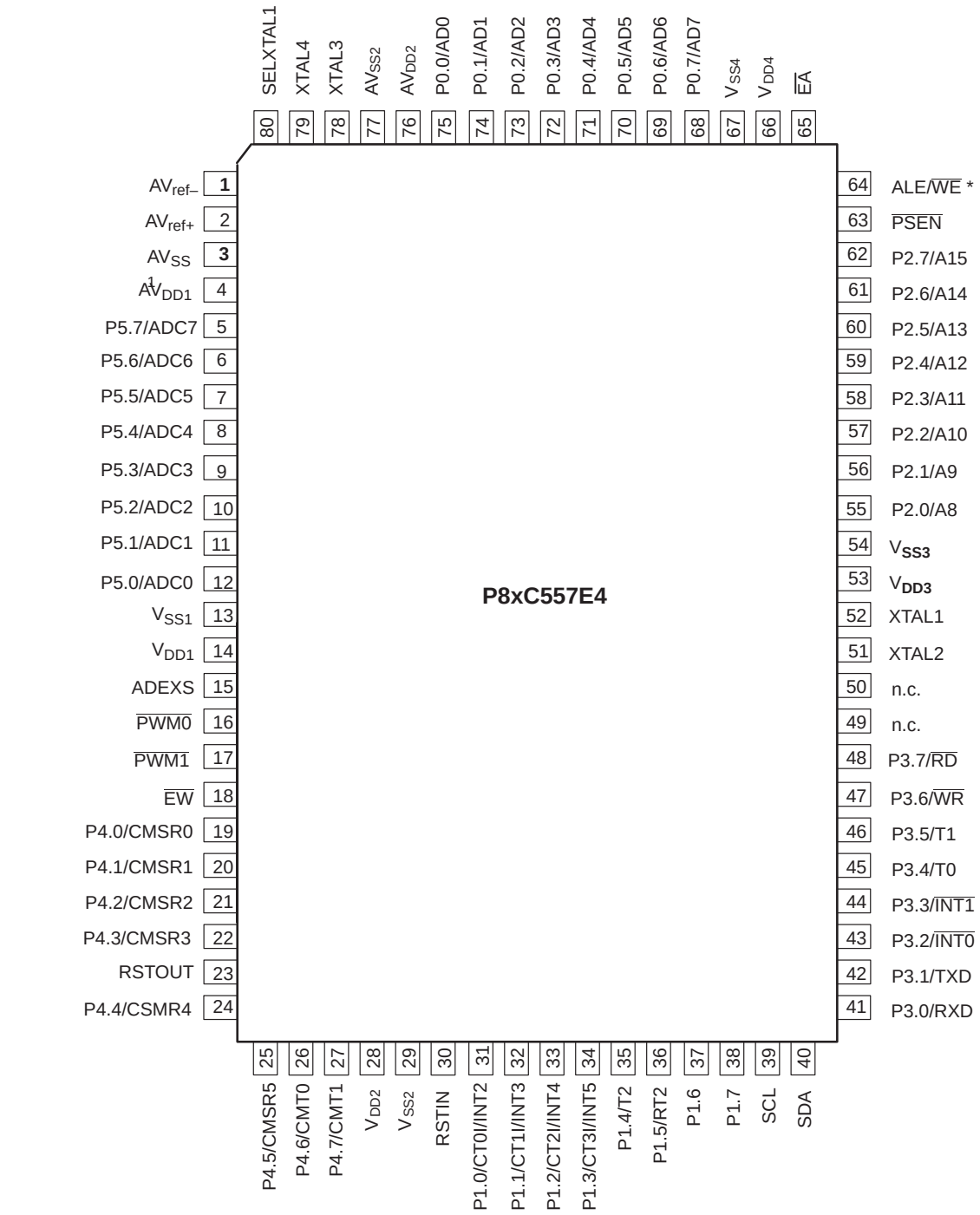
Figure 1. Block diagram P8xC557E4



*) only P89C557E4 with alternate function WE

Figure 2. Functional diagram

4. PINNING



n.c. = not connected

* = only P89C557E4 with alternate function $\overline{WE}$

Figure 3. Pinning diagram for QFP80 (SOT318)

4.1 PIN DESCRIPTION

SYMBOL	PIN	DESCRIPTION
AV_{ref-} AV_{ref+}	1 2	**Low end** of analog to digital conversion reference resistor **High end** of analog to digital conversion reference resistor.
AV_{SS1} AV_{DD1}	3 4	**Analog** ground for ADC **Analog** power supply (+5 V) for ADC
AV_{SS2} AV_{DD2}	77 76	**Analog** ground; for PLL oscillator **Analog** power supply; (+5 V) for PLL oscillator
P5.7 – P5.0	5 – 12	**Port 5** 8-bit input port Port pin — Alternative function P5.0–P5.7 — Eight input channels to ADC (ADC0–ADC7)
V_{DD1}, V_{DD2}, V_{DD3}, V_{DD4}	14, 28, 53, 66	**Digital power supply:** +5 V power supply pins during normal operation and power reduction modes. All pins must be connected.
V_{SS1}, V_{SS2} V_{SS3}, V_{SS4}	13, 29, 54, 67	**Digital ground:** circuit ground potential. All pins must be connected.
ADEXS	15	**Start ADC operation:** Input starting analog to digital conversion triggered by a programmable edge (ADC operation can also be started by software). This pin must not float
$\overline{PWM0}$	16	**Pulse width modulation** output 0
$\overline{PWM1}$	17	**Pulse width modulation** output 1
$\overline{EW}$	18	**Enable watchdog timer:** Enable for T3 watchdog timer and disable Power-down Mode.This pin must not float.
P4.0 – P4.7	19 – 22 24 – 27	**Port 4** 8-bit quasi-bidirectional I/O port Port pin — Alternative function P4.0 — CMSR0 } P4.1 — CMSR1 } P4.2 — CMSR2 } compare and set/reset P4.3 — CMSR3 } outputs on a match with timer T2 P4.4 — CMSR4 } P4.5 — CMSR5 } P4.6 — CMT0 } compare and toggle outputs P4.7 — CMT1 } on a match with timer T2
RSTIN	30	**Reset:** Input to reset the P8xC557E4.
RSTOUT	23	**Reset:** Output of the P8xC557E4 for resetting peripheral devices during initialization and Watchdog Timer overflow.
P1.0 – P1.7	31 – 38	**Port 1** 8-bit quasi-bidirectional I/O port Port pin — Alternative function P1.0 — CT0I/INT2} P1.1 — CT1I/INT3} : Capture timer inputs for P1.2 — CT2I/INT4} timer T2 or external interrupt inputs P1.3 — CT3I/INT5} P1.4 — T2 : T2 event input, rising edge triggered P1.5 — RT2 : T2 timer reset input, rising edge triggered P1.6 P1.7
SCL	39	**I^2C-bus serial clock** I/O port
SDA	40	**I^2C-bus serial data** I/O port If SCL and SDA are not used, they must be connected to V_{SS}.

PIN DESCRIPTION (Continued)

SYMBOL	PIN	DESCRIPTION
P3.0 – P3.7	41 – 48	8-bit quasi-bidirectional I/O port Port pin — Alternative function P3.0 RXD : Serial input port P3.1 TXD : Serial output port P3.2 INT0 : External interrupt P3.3 INT1 : External interrupt P3.4 T0 : Timer 0 external input P3.5 T1 : Timer 1 external input P3.6 WR : External data memory write strobe P3.7 RD : External data memory read strobe
N.C.	49 – 50	Not connected pins.
XTAL2	51	**Crystal pin 2**: output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used.
XTAL1	52	**Crystal pin 1**: input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used. Must be connected to logic HIGH if the PLL oscillator is selected (SELXTAL1 = LOW).
P2.0 – P2.7	55 – 62	**Port2:** 8-bit quasi-bidirectional I/O port with internal pull-ups.During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX@DPTR) Port 2 emits the high order address byte. The alternative function of P2.7 for the P89C557E4 is the output enable signal for verify/read modes (active low). Port 2 can sink/source one TTL (=4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
$\overline{PSEN}$	63	**Program Store Enable output:** read strobe to the external program memory via Port 0 and 2. Is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without external pull-ups.
ALE/$\overline{WE}$	64	**Address Latch Enable output**: latches the low byte of the address during access of external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE/$\overline{WE}$ can sink/-source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up. The alternative function for the P89C557E4 is the programming pulse input $\overline{WE}$. To prohibit the toggling of ALE pin (RFI noise reduction) the bit RFI in the PCON Register (PCON.5) must be set by software. This bit is cleared on RESET and can be set and cleared by software. When set, ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE if external memory is accessed. ALE will retain its normal high value during Idle Mode and a low value during Power-down Mode while in the "RFI" mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag "RFI" is set or not.
$\overline{EA}$	65	**External Access Input:** If, during RESET, $\overline{EA}$ is held at a TTL level HIGH the CPU executes out of the internal program memory, provided the program counter is less than 32768. If, during RESET, $\overline{EA}$ is held at a TTL level LOW the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. $\overline{EA}$ is latched during RESET and don't care after RESET.
P0.7–P0.0	68 –75	**Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during theses accesses internal pull-ups are activated). Port 0 can sink/source 8 LSTTL inputs.
XTAL3	78	**Crystal pin**, output of the inverting amplifier that forms the 32 kHz oscillator
XTAL4	79	**Crystal pin**, input to the inverting amplifier that forms the 32 kHz oscillator. XTAL3 and XTAL4 are pulled LOW if the PLL oscillator is not selected (SELXTAL1 = HIGH) or if Reset is active.
SELXTAL1	80	Must be connected to logic HIGH level to select the HF oscillator, using the XTAL1/XTAL2 crystal. If pulled low the PLL is selected for clocking of the controller, using the XTAL3/ XTAL4 crystal.

NOTE:
1. To avoid a 'latch-up' effect at Power-on, the voltage at any pin at any time must not be higher or lower than V_{DD}+ 0.5 V or V_{SS}– 0.5 V respectively.

5. ELECTROMAGNETIC COMPATIBILITY (EMC) IMPROVEMENTS

Primary attention was paid on the reduction of electromagnetic emission of the microcontroller P8xC557E4.

The following features effect in reducing the electromagnetic emission and additionally improve the electromagnetic susceptibility:

- Four supply voltage pins (V_{DD}) and four ground pins (V_{SS}) with pairs of V_{DD} and V_{SS} at two adjacent pins at each side of the package.
- Separated V_{DD} pins for the internal logic and the port buffers
- Internal decoupling capacitance improves the EMC radiation behavior and the EMC immunity
- External capacitors are to be located as close as possible between pins V_{DD1} and V_{SS1}, V_{DD2} and V_{SS2}, V_{DD3} and V_{SS3} as well as V_{DD4} and V_{SS4} ; ceramic chip capacitors are recommended (100nF).

Useful in applications that require no external memory or temporarily no external memory:

- The ALE output signal (pulses at a frequency of $f_{CLK}/6$) can be disabled under software control (bit 5 in the SFR PCON: “RFI”); if disabled, no ALE pulse will occur. ALE pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE (external data memory is accessed). ALE will retain its normal HIGH value during Idle Mode and a LOW value during Power-down mode while in the “RFI” reduction mode. Additionally during internal access ($\overline{EA}$ = 1) ALE will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE will always toggle normally, whether the flag “RFI” is set or not.

6. FUNCTIONAL DESCRIPTION

6.1 General

The P8xC557E4 is a stand-alone high-performance microcontroller designed for use in real time applications such as instrumentation, industrial control, medium to high-end consumer applications and specific automotive control applications.

In addition to the 80C51 standard functions, the device provides a number of dedicated hardware functions for these applications.

The P8xC557E4 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 Kbytes. It can also access up to 64 Kbytes of external data memory. For systems requiring extra capability, the P8xC557E4 can be expanded using standard memories and peripherals.

The P8xC557E4 has two software selectable modes of reduced activity for further power reduction – Idle and Power-down. The Idle Mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down Mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative. The Power-down Mode can be terminated by an external Reset, by the seconds interrupt and by any one of the two external interrupts. (See description Wake-up from Power-down Mode.)

6.2 Memory organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 Kbytes external data memory, 1024 bytes internal data memory (consisting of 256 bytes standard RAM and 768 bytes AUX-RAM) and the 32 Kbytes internal and/or 64 Kbytes external program memory (see Figure 4).

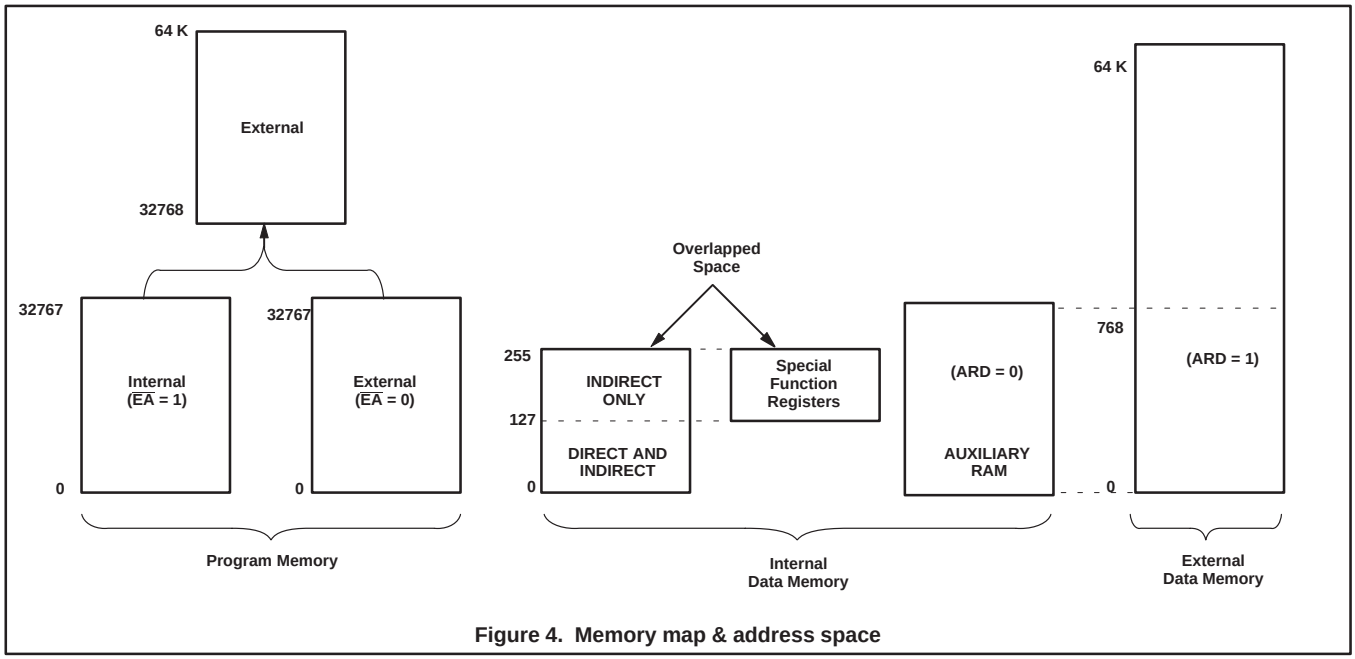


Figure 4. Memory map & address space

6.2.1 Program Memory

The program memory of the P8xC557E4 consists of 32 Kbytes ROM respectively FEEPROM ("Flash Memory") on-chip, externally expandable up to 64 Kbytes. If, during RESET, the $\overline{EA}$ pin was held HIGH, the P8xC557E4 executes out of the internal program memory unless the address exceeds 7FFFH. Locations 8000H through 0FFFFH are then fetched from the external program memory. If the $\overline{EA}$ pin was held LOW during RESET the P8xC557E4 fetches all instructions from the external program memory. The $\overline{EA}$ input is latched during RESET and is don't care after RESET.

The internal program memory content is protected, by setting a mask programmable security bit (ROM) or by the software programmable security byte (FEEPROM) respectively, i.e. it cannot be read out at any time by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The $\overline{EA}$ input is latched during RESET and is 'don't care' after RESET. This implementation prevents from reading internal program code by switching from external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 1 lists the access to the internal and external program memory with MOVC instructions when the security feature has been activated.

6.2.2 Internal Data Memory

The internal data memory is divided into three physically separated parts:

256 bytes of RAM, 768 bytes of AUX-RAM, and a 128 bytes special function area. These can be addressed each in a different way (see also Table 2).

– RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected registerbank.
– RAM 128 to 255 can only be addressed indirectly. Address pointers are R0 and R1 of the selected registerbank.
– AUX-RAM 0 to 767 is also indirectly addressable as external DATA MEMORY locations 0 to 767 via MOVX-Datapointer instruction, unless it is disabled by setting ARD = 1.
AUX-RAM 0 to 767 is indirectly addressable via pageregister (XRAMP) and MOVX-Ri instructions, unless it is disabled by setting ARD = 1 (see Figure 5).
When executing from internal program memory, an access to AUX-RAM 0 to 767 will not affect the ports P0, P2, P3.6 and P3.7.

  An access to external DATA MEMORY locations higher than 767 will be performed with the MOVX @ DPTR instructions in the same way as in the 80C51 structure, so with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read timing signals. Note that the external DATA MEMORY cannot be accessed with R0 and R1 as address pointer if the AUX-RAM is enabled (ARD = 0, default).
– The Special Function Registers (SFR) can only be addressed directly in the address range from 128 to 255 (see Table 5).
– Four register banks, each 8 registers wide, occupy locations 0 through 31 in the lower RAM area. Only one of these banks may be enabled at a time. The next 16 bytes, locations 32 through 47, contain 128 directly addressable bit locations.The stack can be located anywhere in the internal 256 bytes RAM.The stack depth is only limited by the available internal RAM space of 256 bytes (see Figure 7).

  All registers except the program counter and the four register banks reside in the Special Function Register address space.

Table 1. Memory Access by the MOVC Instruction for Protected ROMs

MOVC LOCATION	ACCESS TO INTERNAL PROGRAM MEMORY	ACCESS TO EXTERNAL PROGRAM MEMORY
MOVC in internal program memory	YES	YES
MOVC in external program memory	NO	YES

NOTE:
1. If the security feature has not been activated, there are no restrictions for MOVC instructions**.**

Table 2. Internal Data Memory Map

LOCATION		ADDRESSED
RAM	0 to 127	Direct and indirect
AUX-RAM	0 to 767	Indirect only with MOVX
RAM	128 to 255	Indirect only
SFR	128 to 255	Direct only

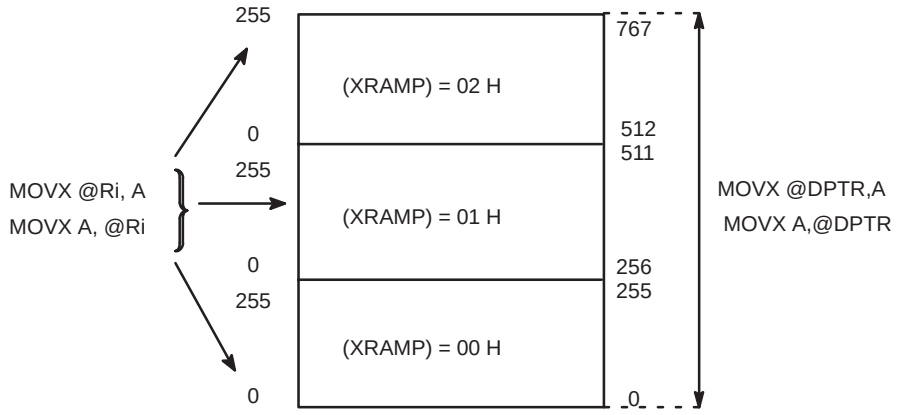


Figure 5. Indirect addressing of AUX-RAM (768 Bytes), ARD bit in PCON = 0

6.2.2.1 AUX-RAM Page Register XRAMP

The AUX-RAM Page Register is used to select one of three 256 bytes pages of the internal 768 bytes AUX-RAM for MOVX-accesses via R0 or R1. Its reset value is (XXXXXX00).

	7	6	5	4	3	2	1	0
XRAMP (FAH)	x	x	x	x	x	x	XRAMP1	XRAMP0

x: undefined during read, a write operation must write “0” to these locations

Figure 6. AUX-RAM page register.

Table 3. Description of XRAMP Bits

BIT	SYMBOL	FUNCTION
XRAMP.2–7	XRAMPx	reserved for future use
XRAMP.1	XRAMP1	AUX-RAM page select bit 1
XRAMP.0	XRAMP0	AUX-RAM page select bit 0

Table 4. Memory Locations for All Possible MOVX Accesses

ARD[1]	XRAMP1	XRAMP0	MOVX @Ri,A and MOVX A,@Ri instructions access:
0	0	0	AUX-RAM locations 0 .. 255 (reset condition)
0	0	1	AUX-RAM locations 256 .. 511
0	1	0	AUX-RAM locations 512 .. 767
0	1	1	no valid memory access; reserved for future use
1	X	X	External RAM locations 0 .. 255
			MOVX @DPTR,A and MOVX A,@DPTR instructions access:
0	X	X	AUX-RAM locations 0 .. 767 (reset condition) External RAM locations 768 .. 65535
1	X	X	External RAM locations 0 .. 65535

NOTE:
1. ARD (AUX-RAM Disable) is a bit in the Special Function Register PCON

Table 5. Special Function Register Memory Map and Reset Values

HIGH NIBBLE OF SFR ADDRESS								
LOW	8	9	A	B	C	D	E	F
0	P0 % 11111111	P1 % 11111111	P2 % 11111111	P3 % 11111111	P4 % 11111111	PSW % 00000000	ACC % 00000000	B % 00000000
1	SP 00000111							
2	DPL 00000000							
3	DPH 00000000							
4								
5								
6	ADRSL0 # XXXXXXXX	ADRSL1 # XXXXXXXX	ADRSL2 # XXXXXXXX	ADRSL3 # XXXXXXXX	ADRSL4 # XXXXXXXX	ADRSL5 # XXXXXXXX	ADRSL6 # XXXXXXXX	ADRSL7 # XXXXXXXX
7	PCON 00000000				P5 # XXXXXXXX	ADCON 00000000	ADPSS 00000000	ADRSH # 000000XX
8	TCON % 00000000	S0CON % 00000000	IEN0 % 00000000	IP0 % X0000000	TM2IR % 00000000	S1CON % 00000000	IEN1 % 00000000	IP1 % 00000000
9	TMOD 00000000	S0BUF XXXXXXXX	CML0 00000000		CMH0 00000000	S1STA # 11111000		PLLCON 00001101
A	TL0 00000000		CML1 00000000		CMH1 00000000	S1DAT 00000000	TM2CON 00000000	XRAMP XXXXXX00
B	TL1 00000000		CML2 00000000		CMH2 00000000	S1ADR 00000000	CTCON 00000000	FMCON * 000X0000
C	TH0 00000000		CTL0 # XXXXXXXX		CTH0 # XXXXXXXX		TML2 # 00000000	PWM0 00000000
D	TH1 00000000		CTL1 # XXXXXXXX		CTH1 # XXXXXXXX		TMH2 # 00000000	PWM1 00000000
E			CTL2 # XXXXXXXX		CTH2 # XXXXXXXX		STE 11000000	PWMP 00000000
F			CTL3 # XXXXXXXX		CTH3 # XXXXXXXX		RTE 00000000	T3 00000000

NOTES:
% = Bit addressable register
# = Read only register
X = Undefined
* = FMCON only in P89C557E4

6.3 Addressing

The P8xC557E4 has five methods for addressing:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a “destination/source” field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addresses is as follows:

- Register in one of the four register banks through Register, Direct or Register-Indirect addressing
- 1024 bytes of internal RAM through Direct or Register-Indirect addressing.
  - Bytes 0–127 of internal RAM may be addressed directly/indirectly. Bytes 128–255 of internal RAM share their address location with the SFRs and so may only be addressed indirectly as data RAM.
  - Bytes 0–767 of AUX-RAM can only be addressed indirectly via MOVX.
- Special Function Register through direct addressing at address locations 128–255 (see Figure 8).
- External data memory through Register-Indirect addressing
- Program memory look-up tables through Base-Register plus Index-Register-Indirect addressing

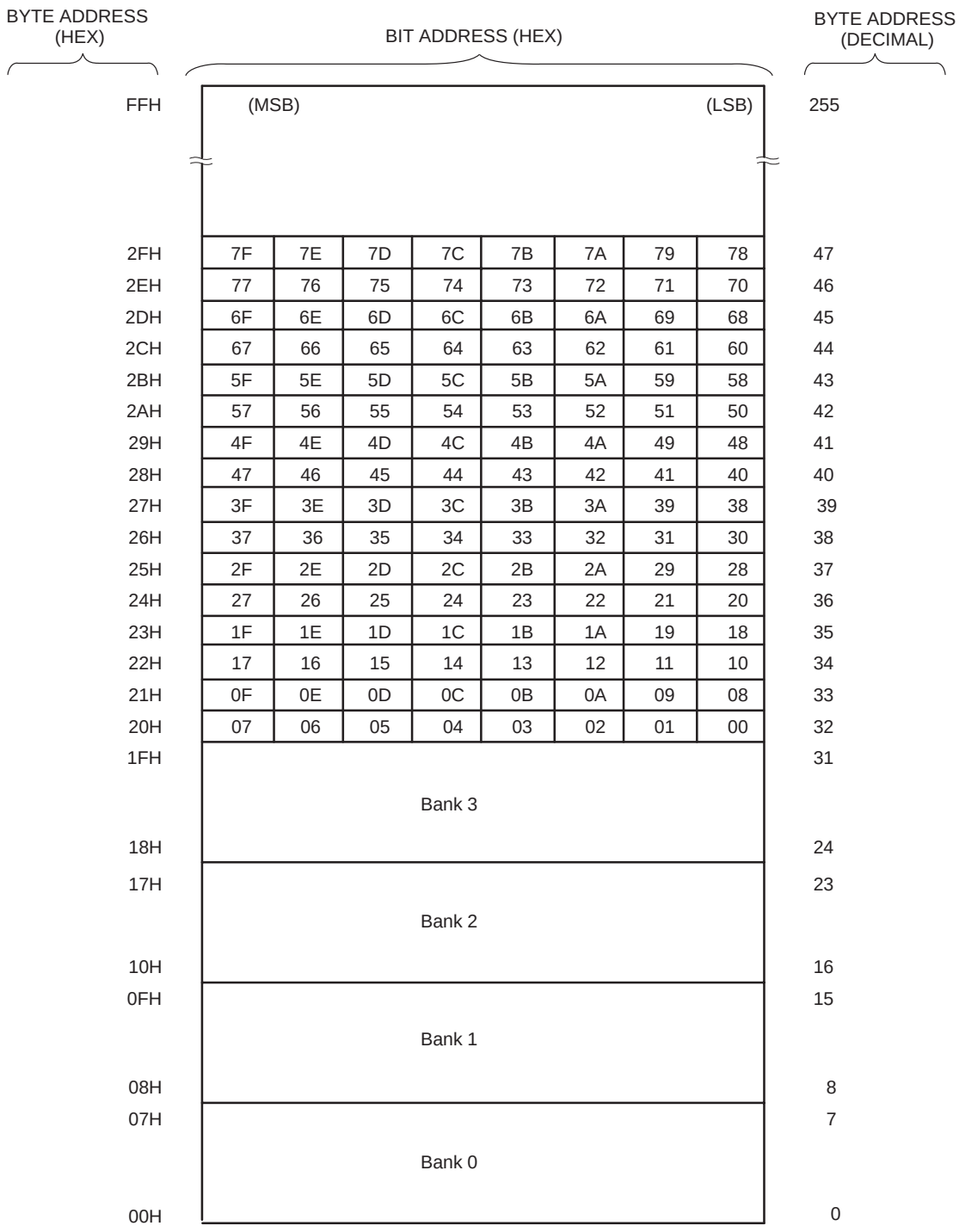


Figure 7. RAM bit addresses

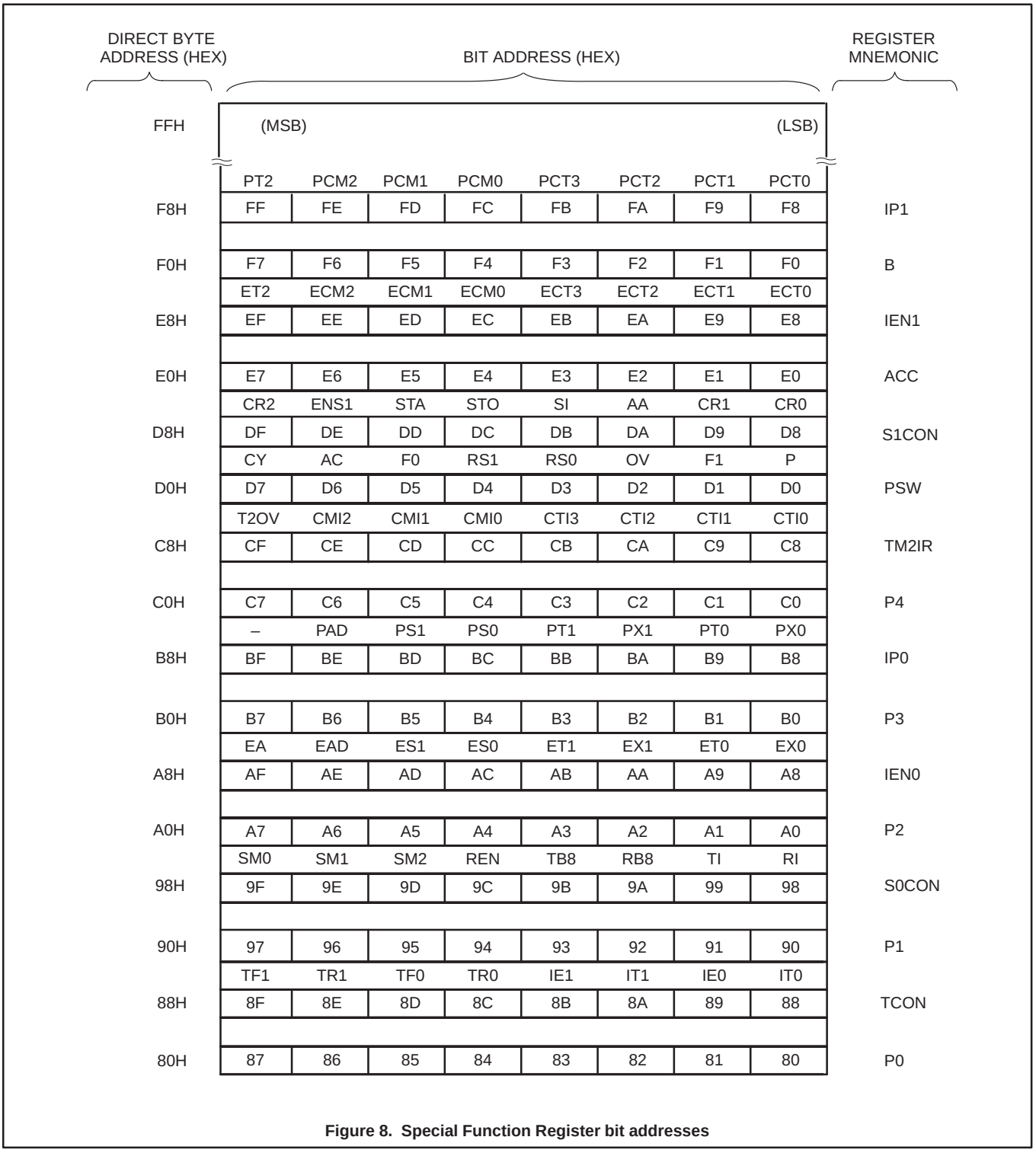


Figure 8. Special Function Register bit addresses

6.4 I/O Facilities

The P8xC557E4 has six 8-bit ports. Ports 0 to 3 are the same as in the 80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port.

The SDA and SCL lines serve the serial port SIO1 (I^2C). Because the I^2C-bus may be active while the device is disconnected from $V_{DD,}$ these pins, are provided with open drain drivers.

Ports 0, 1, 2, 3, 4 and 5 perform the following alternative functions:

Port 0 : provides the multiplexed low-order address and data bus used for expanding the P8xC557E4 with standard memories and peripherals.

Port 1 : Port 1 is used for a number of special functions:

4 capture inputs (or external interrupt request inputs if capture information is not utilized)
- external counter input
- external counter reset input

Port 2 : provides the high-order address bus when the P8xC557E4 is expanded with external Program Memory and/or external Data Memory.

Port 3 : pins can be configured individually to provide:
- external interrupt request inputs
- counter inputs
- receiver input and transmitter output of seri port SIO 0 (UART)
- control signals to read and write external Data Memory

Port 4 : can be configured to provide signals indicating a match between timer counter T2 and its compare registers.

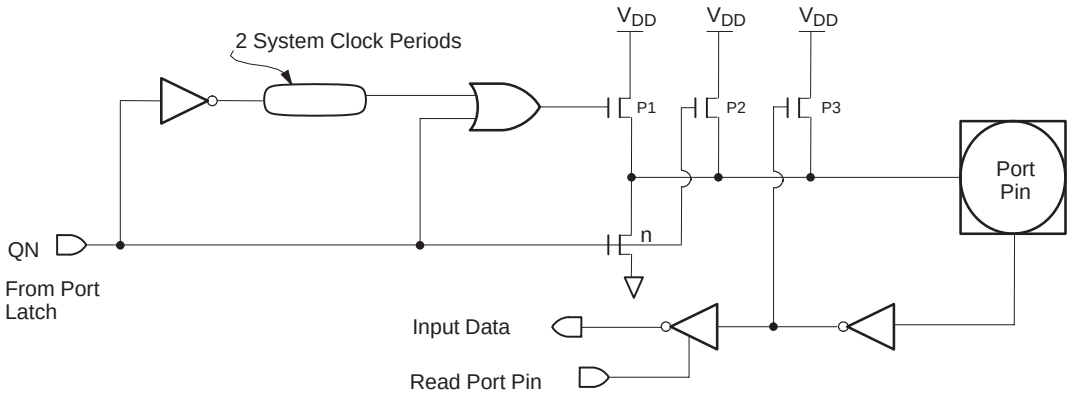
Port 5 : may be used in conjunction with the ADC interface. Unused analog inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analog signals. Channel to channel crosstalk should be taken into consideration when both digital and analog signals are simultaneously input to Port 5 (see DC characteristics).

All ports are bidirectional with the exception of Port 5 which is an input port.

Pins of which the alternative function is not used may be used as normal bidirectional I/Os.

The generation or use of a Port 1, Port 3 or Port 4 pin as an alternative function is carried out automatically by the P8xC557E4 provided the associated Special Function Register bit is set HIGH.

The pull-up arrangements of Ports 1 – 4 are shown in Figure 9.



P1 is turned on for 2 system clock periods after QN makes a 1-to-0 transition.
During this time, P1 also turns on P3 through the inverter to form an additional pull up.

Figure 9. I/O buffers in the P8xC557E4 (Ports 1, 2, 3 and 4)

6.5 Pulse Width Modulated Outputs

The P8xC557E4 contains two pulse width modulated output channels (see Figure 13). These channels generate pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP, which supplies the clock for the counter. The prescaler and counter are common to both PWM channels. The 8-bit counter counts module 255, i.e., from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the corresponding $\overline{PWM0}$ or $\overline{PWM1}$ output is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0/255 to 255/255 and may be programmed in increments of 1/255.

Buffered PWM outputs may be used to drive DC motors. The rotation speed of the motor would be proportional to the contents of PWMn. The PWM outputs may also be configured as a dual DAC. In this application, the PWM outputs must be integrated using conventional operational amplifier circuitry. If the resulting output voltages have to be accurate, external buffers with their own analog supply should be used to buffer the PWM outputs before they are integrated. The repetition frequency fpwm, at the PWMn outputs is give by:

$$fpwm = \frac{f_{CLK}}{2 \times (1 + PWMP) \times 255}$$

This gives a repetition frequency range of 123 Hz to 31.4 kHz (f_{CLK} = 16 MHz). By loading the PWM registers with either 00H or FFH, the PWM channels will output a constant HIGH or LOW level, respectively. Since the 8-bit counter counts modulo 255, it can never actually reach the value of the PWM registers when they are loaded with FFH.

When a compare register (PWM0 or PWM1) is loaded with a new value, the associated output is updated immediately. It does not have to wait until the end of the current counter period. Both $\overline{PWMn}$ output pins are driven by push-pull drivers. These pins are not used for any other purpose.

	7	6	5	4	3	2	1	0
PWMP (FEH)	PWMP.7	PWMP.6	PWMP.5	PWMP.4	PWMP.3	PWMP.2	PWMP.1	PWMP.0

Figure 10. Prescaler frequency control register PWMP.

Table 6. Description of PWMP Bits

BIT	FUNCTION
PWMP.0 to 7	Prescaler division factor = (PWMP) + 1

NOTE:
1. Reading PWMP gives the current reload value. The actual count of the prescaler cannot be read.

	7	6	5	4	3	2	1	0
PWM0 (FCH)	PWM0.7	PWM0.6	PWM0.5	PWM0.4	PWM0.3	PWM0.2	PWM0.1	PWM0.0

Figure 11. Pulse width register PWM0.

Table 7. Description of PWM0 bits

BIT	FUNCTION
PWM0.0 to 7	LOW/HIGH ration of $\overline{PWM0}$ signal = $\frac{(PWM0)}{255 - (PWM0)}$

	7	6	5	4	3	2	1	0
PWM1 (FDH)	PWM1.7	PWM1.6	PWM1.5	PWM1.4	PWM1.3	PWM1.2	PWM1.1	PWM1.0

Figure 12. Pulse width register PWM1.

Table 8. Description of PWM1 bits

BIT	FUNCTION
PWM1.0 to 7	LOW/HIGH ration of $\overline{PWM1}$ signal = $\frac{(PWM1)}{255 - (PWM1)}$

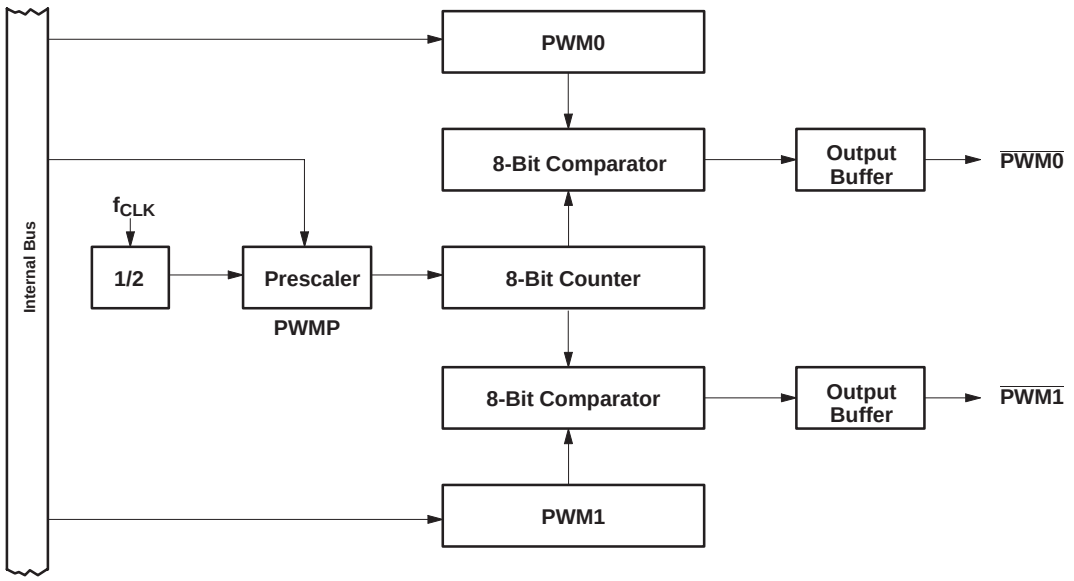


Figure 13. Functional Diagram of Pulse Width Modulated Outputs.

6.6 Analog/Digital Converter (ADC)

The P8xC557E4 A/D Converter is a 10-bit, successive approximation ADC with 8 multiplexed analog input channels. It additionally contains a high input impedance comparator, a DAC built with 1024 series resistors and analog switches, registers and control logic.

Input voltage range is from AV_{ref-} (typical 0V) to AV_{ref+} (typical +5V). A set of 8 buffer registers (10-bit) store the conversion results of the proper analog input channel each.

11 Special Function Registers (SFR) perform the user software interface to the ADC: a control SFR (ADCON), an analog port scan-select SFR (ADPSS), 8 input channel related conversion result SFR with the 8 lower result bits (ADRSL0...ADRSL7), one common result SFR for the upper 2 result bits (ADRSH). An extra SFR (P5) allows for reading digital input port data as an alternative function of the 8 analog input pins.

In order to have a minimum of ADC service overhead in the microcontroller program, the ADC is able to operate autonomously within its user configurable autoscan function.

The functional diagram of the ADC is shown in Figure 15.

Feature Overview:

- 10-bit resolution.
- 8 multiplexed analog inputs.
- Programmable autoscan of the analog inputs.
- Bit oriented 8-bit scan-select register to select analog inputs.
- Continuous scan or one time scan configurable from 1 to 8 analog inputs.
- Start of a conversion by software or with an external signal.
- Eight 10-bit buffer registers, one register for each analog input channel.
- Interrupt request after one channel scan loop.
- Programmable prescaler (dividing by 2, 4, 6, 8) to adapt to different system clock frequencies.
- Conversion time for one A/D conversion: 15 µs ... 50 µs
- Differential non-linearity : DLe ±1 LSB.
- Integral non-linearity : ILe ±2 LSB.
- Offset error : OSe ±2LSB.
- Gain error : Ge ±0.4 %.
- Absolute voltage error : Ae ±3 LSB.
- Channel to channel matching : Mctc ±1LSB.
- Crosstalk between analog inputs : Ct < –60dB. @100 kHz.
- Monotonic and no missing codes.
- Separated analog (AV_{DD}, AV_{SS}) and digital (V_{DD}, V_{SS}) supply voltages.
- Reference voltage at two special pins : AV_{REF-} and AV_{REF+}.

For further information on the ADC characteristics, refer to the “DC CHARACTERISTICS” section.

6.1.1 Functional description:

Table 9. A/D Special Function Registers

SYMBOL	NAME	ACCESS
ADCON	A/D control register	read/write
ADPSS	Analog port scan-select register	read/write
ADRSLn	8 A/D result registers, the 8 lower bits (n: 0...7)	read only
ADRSH	A/D result register, the 2 higher bits	read only
P5	Digital input port (shared with analog inputs)	read only

A/D Control Register ADCON

The Special Function Register ADCON contains control and status bits for the A/D Converter peripheral block. The reset value of ADCON is (00000000). Its hardware address is D7H. ADCON is not bit addressable.

	7	6	5	4	3	2	1	0
ADCON (D7H)	ADPR1	ADPR0	ADPOS	ADINT	ADSST	ADCSA	ADSRE	ADSFE

Figure 14. ADC control register.

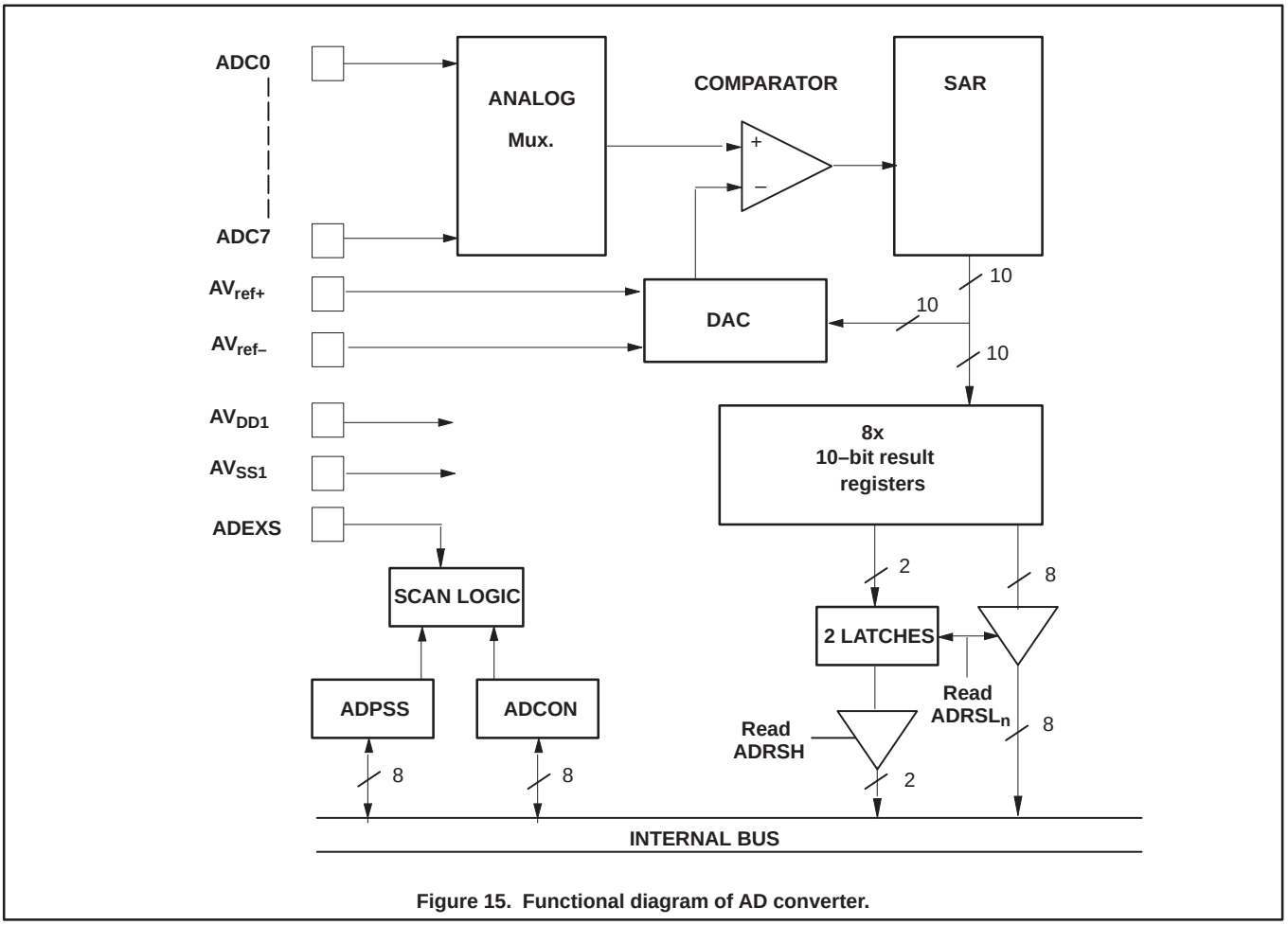


Figure 15. Functional diagram of AD converter.

Table 10. Description of ADCON bits

SYMBOL	BIT	FUNCTION
ADCON.7	ADPR1	Control bit for the prescaler.
ADCON.6	ADPR0	Control bit for the prescaler. ADPR1=0 ADPR0=0 Prescaler divides by 2 (default by reset) ADPR1=0 ADPR0=1 Prescaler divides by 4 ADPR1=1 ADPR0=0 Prescaler divides by 6 ADPR1=1 ADPR0=1 Prescaler divides by 8
ADCON.5	ADPOS	ADPOS is reserved for future use. Must be '0' if ADCON is written.
ADCON.4	ADINT	ADC interrupt flag. This flag is set when all selected analog inputs are converted, as well in continuous scan as in one-time scan mode. An interrupt is invoked if this interrupt is enabled. ADINT must be cleared by software. It cannot be set by software.
ADCON.3	ADSST	ADC start and status. Setting this bit by software or by hardware (via ADEXS input) starts the A/D conversion of the selected analog inputs. ADSST stays a 'one' in continuous scan mode. In one-time scan mode, ADSST is cleared by hardware when the last selected analog input channel has been converted. As long as ADSST is '1', new start commands to the ADC-block are ignored. An A/D conversion in progress is aborted if ADSST is cleared by software.
ADCON.2	ADCSA	1 = Continuous scan of the selected analog inputs after a start of an A/D conversion. 0 = One-time scan of the selected analog inputs after a start of an A/D conversion.
ADCON.1	ADSRE	1 = A rising edge at input ADEXS will start the A/D conversion and generate a capture signal. 0 = A rising edge at input ADEXS has no effect.
ADCON.0	ADSFE	1 = A falling edge at input ADEXS will start the A/D conversion and generate a capture signal. 0 = A falling edge at input ADEXS has no effect.

A/D Input Port Scan-Select Register ADPSS

The Special Function Register ADPSS contains control bits to select the analog input channel(s) to be scanned for A/D conversion. The reset value of ADPSS is (00000000). Its hardware address is E7H. ADPSS is not bit addressable.

If all bits are '0' then no A/D conversion can be started. If ADPSS is written while an A/D conversion is in progress (ADSST in the ADCON register is '1') then the autoscan loop with the previous selected analog inputs is completed first. The next autoscan loop is performed with the new selected analog inputs.

	7	6	5	4	3	2	1	0
ADPSS (E7H)	ADPSS7	ADPSS6	ADPSS5	ADPSS4	ADPSS3	ADPSS2	ADPSS1	ADPSS0

ADPSS7–0 For each individual bit position: 0 = The corresponding analog input is skipped in the auto-scan loop.
1 = The corresponding analog input is included in the auto-scan loop.

Figure 16. A/D input port scan-select register.

A/D Result Registers ADRSLn and ADRSH:

The binary result code of A/D conversions is accessed by these Special Function Registers. The result SFR are read only registers. The read value after reset is indeterminate. Their data are not affected by chip reset. They are not bit addressable.

There are 8 Special Function Registers ADRSLn (ADRSL0...ADRSL7) – A/D Result Low byte – and one general SFR ADRSH – A/D Result High byte – . Each of ADRSLn is associated with the coincidently indexed analog input channel ADCn (ADC0/P5.0...ADC7/P5.7). Reading an ADRSLn register by software copies at the same time the two highest bits of the 10-bit conversion result into two latches, thus preserving them until the next read of any ADRSLn register. These two latches form bit positions 0 and 1 of SFR ADRSH, the upper 6 bits of ADRSH are always read as '0'.

Thus it is ensured to get the 10-bit result of the same single A/D conversion by reading any register ADRSLn first and after it the register ADRSH.

	7	6	5	4	3	2	1	0
ADRSLn	ADRSn.7	ADRSn.6	ADRSn.5	ADRSn.4	ADRSn.3	ADRSn.2	ADRSn.1	ADRSn.0

(n: 0...7)

	7	6	5	4	3	2	1	0
ADRSH	0	0	0	0	0	0	ADRSn.9	ADRSn.8

Figure 17. A/D Result Registers.

Digital Input Port Register P5

Port 5 Special Function Register P5 always represents the binary value of the logic level at input pins P5.0/ADC0...P5.7/ADC7. P5 is not affected by chip reset. P5 is a read only register. Its hardware address is C7H. P5 is not bit addressable.

Reading Special Function Register P5 does not affect A/D conversions. But it is recommended to use the digital input port function of the hardware Port 5 only as an alternative to analog input voltage conversions. Simultaneous mixed operation is discouraged for the sake of A/D conversion result reliability and accuracy.

For further information on Port 5, refer to the “I/O facilities” section.

For further information on A/D Special Function Registers, refer to the “Internal Data Memory” section.

	7	6	5	4	3	2	1	0
P5 (C7H)	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0

Figure 18. Digital input port register P5.

Reset

After a RESET of the microcontroller the ADCON and ADPSS register bits are initialized to zero. Registers ADRSLn and ADRSH are not initialized by a RESET.

Idle and Power-down Mode

The A/D Converter is active only when the microcontroller is in normal operating mode. If the Idle or Power-down Mode is activated, then the ADC is switched off and put into a power saving idle state – a conversion in progress is aborted, a previously set ADSST flag is cleared and the internal clock is halted. The conversion result registers are not affected.

The interrupt flag ADINT will not be set by activation of Idle or Power-down Mode. A previously set flag ADINT will not be cleared by the hardware. (Note: ADINT cannot be cleared by hardware at all, except for a RESET – it must be cleared by the user software.)

After a wakeup from Idle or Power-down Mode a set flag ADINT indicates that at least one autoscan loop was finished completely before the microcontroller was put into the respective power reduction mode and it indicates that the stored result data may be fetched now – if desired.

For further information on Idle and Power-down Mode, refer to the “Power reduction modes” section.

Timing

A programmable prescaler is controlled by the bits ADPR1 and ADPR0 in register ADCON to adapt the conversion time for different microcontroller clock frequencies.

Table 11 shows conversion times (tconv) for one A/D conversion at some convenient system clock frequencies (fclk) and ADC prescaler divisors (m), which are user selectable by the bits ADCON.7/ADPR1 and ADCON.6/ADPR0.

For conversion times outside the limits for tconv the specified ADC characteristics are not guaranteed; (prohibited conversion times are put in brackets):

Table 11. Conversion time configuration examples (tconv/μs)

m	f_{CLK}			
	6 MHz	8 MHz	12 MHz	16 MHz
2	26	19.5	[13]	[9.75]
4	50	37.5	25	18.75
6	[74]	[55.5]	37	27.75
8	[98]	[73.5]	49	36.75

Conversion time tconv = (6 m + 1) machine cycles

A conversion time tconv consists of one sample time period (which equals two bit conversion times), 10 bit conversion time periods and one machine cycle to store the result.

After result storage an extra initializing time period follows to select the next analog input channel (according to the contents of SFR ADPSS), before the input signal is sampled.

Thus the time period between two adjacent conversions within an autoscan loop is larger than the pure time tconv. This autoscan cycle time is (7 m) machine cycles.

At the start of an autoscan conversion the time between writing to SFR ADCON and the first analog input signal sampling depends on the current prescaler value (m) and the relative time offset between this write operation and the internal (divided) ADC clock. This gives a variation range for the A/D conversion start time of (m / 2) machine cycles.

6.6.2 Configuration and Operation

Every A/D conversion is an autoscan conversion. The two user selectable general operation modes are continuous scan and one-time scan mode.

The desired analog input port channel/s for conversion is/are selected by programming A/D input port scan-select bits in SFR ADPSS. An analog input channel is included in the autoscan loop if the corresponding bit in ADPSS is 1, a channel is skipped if the corresponding bit in ADPSS is 0.

An autoscan is always started according to the lowest bit position of ADPSS that contains a 1.

An autoscan conversion is started by setting the flag ADSST in register ADCON either by software or by an external start signal at input pin ADEXS, if enabled. Either no edge (external start totally disabled), a rising edge or/and a falling edge of ADEXS is selectable for external conversion start by the bits ADSRE and ADSFE in register ADCON.

After completion of an A/D conversion the 10-bit result is stored in the corresponding 10-bit buffer register. Then the next analog input is selected according to the next higher set bit position in ADPSS, converted and stored, and so on. When the result of the last conversion of this autoscan loop is stored, flag ADCON.4/ADINT, the ADC interrupt flag, is set. It is not cleared by interrupt hardware – it must be cleared by software.

In continuous scan mode (ADCON.2/ADCSA=1) the ADC start and status flag ADCON.3/ADSST retains the set state and the autoscan loop restarts from the beginning. In one-time scan mode (ADCSA=0) conversions stop after the last selected analog input was converted, ADINT is set and ADSST is cleared automatically.

ADSST cannot be set (neither externally nor by software) as long as ADINT=1, i.e. as long as ADINT is set, a new conversion start – by setting flag ADSST – is inhibited; actually it is only delayed until ADINT is cleared.

(If a '1' is written to ADSST while ADINT=1, this new value is internally latched and preserved, not setting ADSST until ADCON.4/ADINT=0. In this state, a read of SFR ADCON will display ADCON.3/ADSST=0, because always the effective ADC status is read.)

Note that under software control the analog inputs can also be converted in arbitrary order, when one-time scan mode is selected and in SFR ADPSS only one bit is set at a time. In this case ADINT is set and ADSST is cleared after every conversion.

6.6.3 Resolution and Characteristics

The ADC system has its own analog supply pins AV_{DD} and AV_{SS}. It is referenced by two special reference voltage input pins sourcing the resistance ladder of the DAC: AV_{ref+} and AV_{ref-}. The voltage between AV_{REF+} and AV_{REF-} defines the full-scale range. Due to the 10-bit resolution the full scale range is divided into 1024 unit steps. The unit step voltage is 1 LSB, which is typically 5 mV (AV_{ref+} = 5.12 V, AV_{ref-} = 0 V = AV_{SS}).

The DAC's resistance ladder has 1023 equally spaced taps, separated by a unit resistance 'R'. The first tap is located 0.5 x R above AV_{ref-}, the last tap is located 1.5 x R below AV_{ref+}. This results in a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error. For input voltages between AV_{ref-} and (AV_{ref-} + 1/2 LSB) the 10-bit conversion result code will be 00 0000 0000 B = 000H = 0D. For input voltages between (AV_{ref+} – 3/2 LSB) and AV_{ref+} the 10-bit conversion result code will be 11 1111 1111 B = 3FFH = 1023D.

The result code corresponding to an analog input voltage (AV_{in}) can be calculated from the formula:

$$\text{ResultCode} = 1024 \times \frac{AV_{IN} - AV_{ref-}}{AV_{ref+} - AV_{ref-}}$$

The analog input voltage should be stable when it is sampled for conversion. At any times the input voltage slew rate must be less than 10 V/ms (5 V conversion range) in order to prevent an undefined result.

This maximum input voltage slew rate can be ensured by an RC low pass filter with R = 2k2 and C = 100 nF. The capacitor between analog input pin and analog ground pin shall be placed close to the pins in order to have maximum effect in minimizing input noise coupling.

6.7 Timer/Counters

The P8xC557E4 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

6.7.1 Timer 0 and Timer 1

Timers 0 and 1 each have a control bit in SFR TMOD that selects the timer or counter function of the corresponding timer.

In the timer function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the counter function, the register is incremented in response to a 1-to-0 transition at the corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a HIGH in one cycle and a LOW in the next cycle, the counter is incremented. Thus, it takes two machine cycles (24 oscillator periods) to recognize a 1-to-0 transition. There are no restrictions on the duty cycle of the external input signal, but to insure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

Timer 0 and Timer 1 can be programmed independently to operate in one of four modes:

- **Mode 0:**
  8-bit timer or 8-bit counter each with divide-by-32 prescaler
- **Mode 1:**
  16-bit time-interval or event counter
- **Mode 2:**
  8-bit time-interval or event counter with automatic reload upon overflow
- **Mode 3:**
  –Timer 0: one 8-bit time-interval or event counter and one 8-bit time-interval counter
  –Timer 1: stopped

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port baud-rate generator.

With a 16 MHz crystal, the counting frequency of these timer/counters is as follows:

- In the timer function, the timer is incremented at a frequency of 1.33 MHz – a division by 12 of the system clock frequency
- 0 Hz to an upper limit of 0.66 MHz (1/24 of the system clock frequency) when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

When configured as a counter, the register is incremented on every falling edge on the corresponding input pin, T0 or T1. The incremented register value can be read earliest during the second machine cycle after that one, during which the incrementing pulse occurred.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all HIGHs to all LOWs (or automatic reload value), with the exception of mode 3 as previously described.

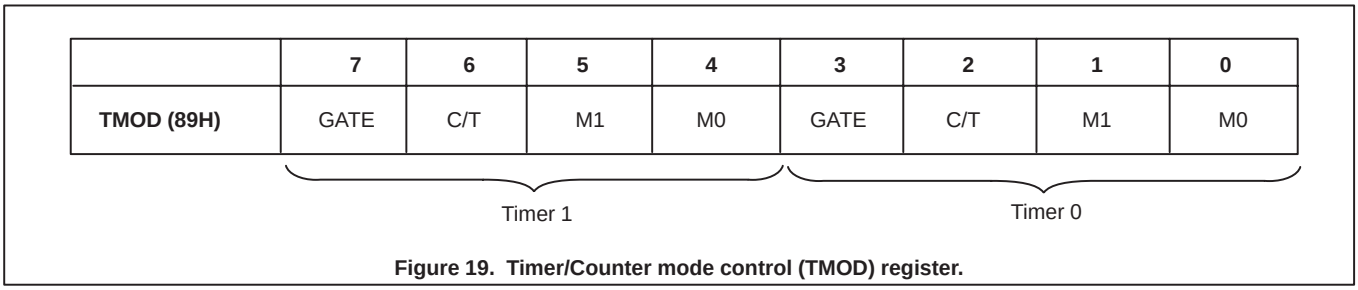


Figure 19. Timer/Counter mode control (TMOD) register.

Table 12. Description of TMOD bits

SYMBOL	BIT	FUNCTION
Gate	TMOD.7 TMOD.3	Gating control when set. Timer/Counter “x” is enabled only while “$\overline{\text{INTx}}$” pin is high and “TRx” control pin is set. When cleared Timer “x” is enabled whenever “TRx” control bit is set.
C/T	TMOD.6 TMOD.2	Timer or Counter Selector cleared for Timer operation (input from internal system clock). Set for Counter operation (input from “Tx” input pin).
M1 M0	TMOD.5 TMOD.1 TMOD.4 TMOD.0	Timer 0, Timer 1 mode select see Table 13.

Table 13. Timer 0 / Timer 1 operation select

M1	M0	OPERATING
0	0	8048 Timer “TLx” serves as 5-bit prescaler.
0	1	16-bit Timer/Counter “THx” and “TLx” are cascaded; there is no prescaler.
1	0	8-bit auto-reload Timer/Counter “THx” holds a value which is to be reloaded into “TLx” each time it overflows.
1	1	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits.
1	1	(Timer 1) Timer/Counter 1 stopped.

	7	6	5	4	3	2	1	0
TCON (88H)	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Figure 20. Timer/Counter mode control (TCON) register.

Table 14. Description of TCON bits

SYMBOL	BIT	FUNCTION
TF1	TCON.7	Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR1	TCON.6	Timer 1 run control bit. Set/cleared by software to turn Timer/Counter on/off.
TF0	TCON.5	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR0	TCON.4	Timer 0 run control bit. Set/cleared by software to turn Timer/Counter on/off.
IE1	TCON.3	Interrupt 1 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT1	TCON.2	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
IE0	TCON.1	Interrupt 0 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT0	TCON.0	Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

6.7.2 Timer T2

Timer T2 is a 16 bit timer/counter which has capture and compare facilities. The operational diagram is shown in Figure 21.

The 16 bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with 1/12 of the clock frequency, or by an external source connected to the T2 input, or it is switched off. The maximum repetition rate of the external clock source is $f_{CLK}/12$, twice that of Timer 0 and Timer 1. The prescaler is incremented on a rising edge. It is cleared if its division factor or its input source is changed, or if the timer/counter is reset (see also Figure 22: TM2CON). T2 is readable 'on the fly', without any extra read latches; this means that software precautions have to be taken against misinterpretation at overflow from least to most significant byte while T2 is being read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle or Power-down Mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit Capture Registers: CT0, CT1, CT2 and CT3. A rising or falling edge on the inputs CT0I, CT1I, CT2I or CT3I (alternative function of Port 1) results in loading the contents of T2 into the respective Capture Registers and an interrupt request.

Using the Capture Register CTCON (see Figure 23), these inputs may invoke capture and interrupt request on a positive, a negative edge or on both edges. If neither a positive nor a negative edge is selected for capture input, no capture or interrupt request can be generated by this input.

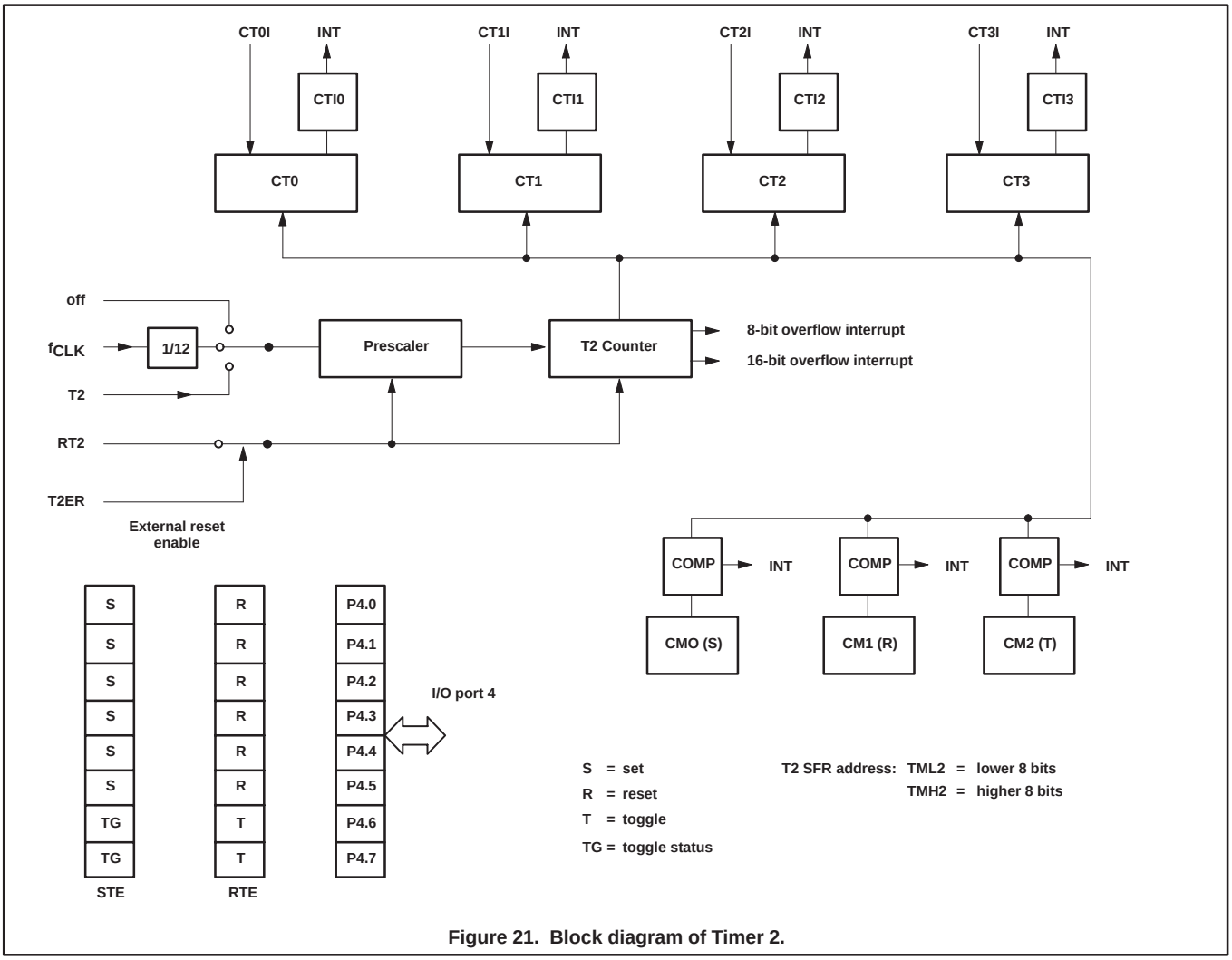


Figure 21. Block diagram of Timer 2.

	7	6	5	4	3	2	1	0
TM2CON (EAH)	T2IS1	T2IS0	T2ER	T2BO	T2P1	T2P0	T2MS1	T2MS0

Figure 22. T2 control register (TM2CON).

Table 15. Description of TM2CON bits

SYMBOL	BIT	FUNCTION
T2IS1	TM2CON.7	Timer T2 16-bit overflow interrupt select
T2IS0	TM2CON.6	Timer T2 byte overflow interrupt select
T2ER	TM2CON.5	Timer T2 external reset enable. When this bit is set, Timer T2 may be reset by a rising edge on RT2 (P1.5).
T2BO	TM2CON.4	Timer T2 byte overflow interrupt flag
T2P1	TM2CON.3	Timer T2 prescaler select
T2P0	TM2CON.2	
T2MS1	TM2CON.1	Timer T2 mode select
T2MS0	TM2CON.0	

Table 16. Timer 2 prescaler select

T2P1	T2P0	TIMER T2 CLOCK
0	0	Clock source
0	1	Clock source/2
1	0	Clock source/4
1	1	Clock source/8

Table 17. Timer 2 mode select

T2MS1	T2MS0	MODE SELECTED
0	0	Timer T2 halted (off)
0	1	T2 clock source = $f_{CLK}/12$
1	0	Test mode; do not use
1	1	T2 clock source = pin T2

	7	6	5	4	3	2	1	0
CTCON (EBH)	CTN3	CTP3	CTN2	CTP2	CTN1	CTP1	CTN0	CTP0

Figure 23. Capture control register (CTCON).

Table 18. Description of CTCON bits

SYMBOL	BIT	FUNCTION
CTN3	CTCON.7	Capture Register 3 triggered by a falling edge on CT3I
CTP3	CTCON.6	Capture Register 3 triggered by a rising edge on CT3I
CTN2	CTCON.5	Capture Register 2 triggered by a falling edge on CT2I
CTP2	CTCON.4	Capture Register 2 triggered by a rising edge on CT2I
CTN1	CTCON.3	Capture Register 1 triggered by a falling edge on CT1I
CTP1	CTCON.2	Capture Register 1 triggered by a rising edge on CT1I
CTN0	CTCON.1	Capture Register 0 triggered by a falling edge on CT0I
CTP0	CTCON.0	Capture Register 0 triggered by a rising edge on CT0I

The contents of the Compare Registers CM0, CM1 and CM2 are continuously compared with the counter value of Timer T2. When a match occurs, an interrupt may be invoked. A match of CM0 sets the bits 0–5 of Port 4, a CM1 match resets these bits and a CM2 match toggles bits 6 and 7 of Port 4, provided these functions are enabled by the STE respectively RTE registers. A match of CM0 and CM1 at the same time results in resetting bits 0–5 of Port 4. CM0, CM1 and CM2 are reset by the RSTIN signal.

	7	6	5	4	3	2	1	0
TM2IR (C8H)	T2OV	CMI2	CMI1	CMI0	CTI3	CTI2	CTI1	CTI0

Figure 24. Interrupt flag register (TM2IR).

Table 19. Description of TM2IR bits

SYMBOL	BIT	FUNCTION
T2OV	TM2IR.7	Timer T2 16-bit overflow interrupt flag
CMI2	TM2IR.6	CM2 interrupt flag
CMI1	TM2IR.5	CM1 interrupt flag
CMI0	TM2IR.4	CM0 interrupt flag
CTI3	TM2IR.3	CT3 interrupt flag
CTI2	TM2IR.2	CT2 interrupt flag
CTI1	TM2IR.1	CT1 interrupt flag
CTI0	TM2IR.0	CT0 interrupt flag

	7	6	5	4	3	2	1	0
STE (EEH)	TG47	TG46	SP45	SP44	SP43	SP42	SP41	SP40

Figure 25. Set enable register (STE).

Table 20. Description of STE bits

SYMBOL	BIT	FUNCTION
TG47	STE.7	If “1” then P4.7 is reset on the next toggle, if LOW P4.7 is set on the next toggle
TG46	STE.6	If “1” then P4.6 is reset on the next toggle, if LOW P4.6 is set on the next toggle
SP45	STE.5	If “1” then P4.5 is set on a match between CM0 and Timer T2
SP44	STE.4	If “1” then P4.4 is set on a match between CM0 and Timer T2
SP43	STE.3	If “1” then P4.3 is set on a match between CM0 and Timer T2
SP42	STE.2	If “1” then P4.2 is set on a match between CM0 and Timer T2
SP41	STE.1	If “1” then P4.1 is set on a match between CM0 and Timer T2
SP40	STE.0	If “1” then P4.0 is set on a match between CM0 and Timer T2

	7	6	5	4	3	2	1	0
RTE (EFH)	TP47	TP46	RP45	RP44	RP43	RP42	RP41	RP40

Figure 26. Reset/Toggle enable register (RTE).

Table 21. Description of RTE bits

SYMBOL	BIT	FUNCTION
TP47	RTE.7	If “1” then P4.7 toggles on a match between CM2 and Timer T2
TP46	RTE.6	If “1” then P4.6 toggles on a match between CM2 and Timer T2
RP45	RTE.5	If “1” then P4.5 toggles on a match between CM1 and Timer T2
RP44	RTE.4	If “1” then P4.4 toggles on a match between CM1 and Timer T2
RP43	RTE.3	If “1” then P4.3 toggles on a match between CM1 and Timer T2
RP42	RTE.2	If “1” then P4.2 toggles on a match between CM1 and Timer T2
RP41	RTE.1	If “1” then P4.1 toggles on a match between CM1 and Timer T2
RP40	RTE.0	If “1” then P4.0 toggles on a match between CM1 and Timer T2

For more information concerning the TM2CON, CTCON, TM2IR and the STE/RTE registers see IC20 handbook, chapter “80C51 family hardware description”.

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At a byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either one or both of these overflows can be programmed to request an interrupt.

All interrupt flags must be reset by software.

6.8 Watchdog Timer T3

In addition to Timer T2 and the standard timers, a watchdog timer (T3) consisting of an 11-bit prescaler and an 8-bit timer is also incorporated (see Figure 27).

The timer is incremented every 1.5 ms, derived from the system clock frequency of 16 MHz by the following:

$$f_{timer} = \frac{f_{CLK}}{12 \times 2048}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at pin RSTOUT. Also the PLL control register is reset.

To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The watchdog timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software.

At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer's reloading and the occurrence of a reset depends on the reloaded value. For example, this may range from 1.5 ms to 0.375 s when using an oscillator frequency of 16 MHz.

In the Idle state the watchdog timer and reset circuitry remain active.

The watchdog timer is controlled by the watchdog enable pin ($\overline{EW}$). A LOW level enables the watchdog timer and disables the Power-down Mode. A HIGH level disables the watchdog timer and enables the Power-down Mode.

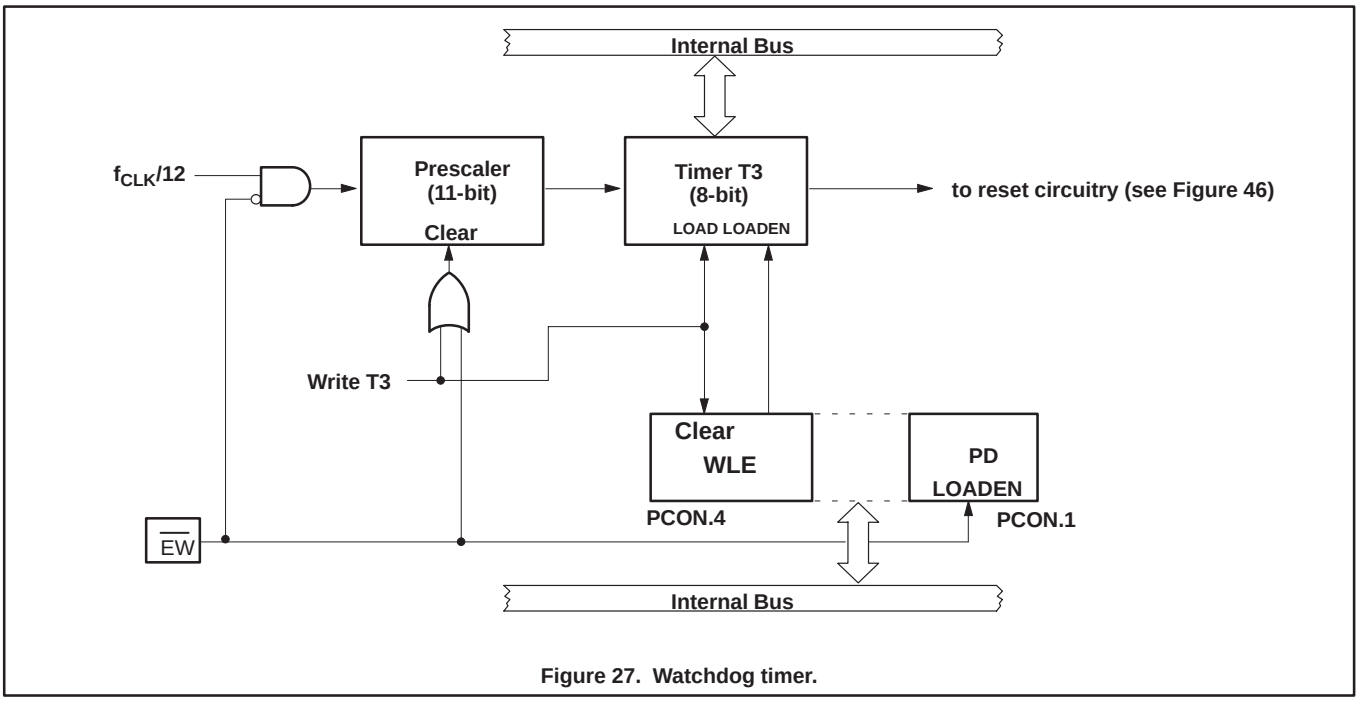


Figure 27. Watchdog timer.

6.9 Serial I/O

The P8xC557E4 is equipped with two independent serial ports: SIO0 and SI01. SIO0 is the full duplex UART port, identical to the PCB80C51 serial port. SIO1 is an I^2C-bus serial I/O interface with byte oriented master and slave functions.

6.9.1 SIO0 (UART)

SIO 0 is a full duplex serial I/O port – it can transmit and receive simultaneously. This serial port is also receive-buffered. It can commence reception of a second byte before the previously received byte has been read from the receive register. If, however, the first byte has still not been read by the time reception of the second byte is complete, one of the bytes will be lost. The SIO0 receive and transmit registers are both accessed via the S0BUF special function register. Writing to S0BUF loads the transmit register, and reading S0BUF accesses to a physically separate receive register. SIO0 can operate in 4 modes:

Mode 0: Serial data is transmitted and received through RXD. TXD outputs the shift clock. 8 data bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 of the oscillator frequency. A write into S0CON should be avoided during a transmission to avoid spikes on RXD/TXD.

Mode 1: 10 bits are transmitted via TXD or received through RXD: a start bit (0), 8 data bits (LSB first), and a stop bit(1). On receive, the stop bit is put into RB8 (S0CON special function register). The baud rate is variable.

Mode 2: 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. With nominal software, TB8 can be the parity bit (P in PSW). During a receive, the 9th data bit is stored in RB8 (S0CON), and the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 of the oscillator frequency.

Mode 3: 11 bits are transmitted through TXD or received through RXD: a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). Mode 3 is the same as Mode 2 except the baud rate which is variable in Mode 3.

In all four modes, transmission is initiated by any instruction that writes to the S0BUF function register. Reception is initiated in Mode 0 when RI = 0 and REN = 1. In the other three modes, reception is initiated by the incoming start bit provided that REN = 1.

Modes 2 and 3 are provided for multiprocessor communications. In these modes, 9 data bits are received with the 9th bit written to RB8. The 9th bit is followed by the stop bit. The port can be programmed so that with receiving the stop bit, the serial port interrupt will be activated if, and only if RB8 = 1.

This feature is enabled by setting bit SM2 in S0CON. This feature may be used in multiprocessor systems.

For more information about how to use the UART in combination with the registers S0CON, PCON, IEN0, S0BUF and Timer register refer to the 80C51 Data Handbook IC20.

	7	6	5	4	3	2	1	0
S0CON (98H)	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Figure 28. Serial port control (S0CON) register.

Table 22. Description of S0CON bits

SYMBOL	BIT	FUNCTION
SM0	S0CON.7	This bit is used to select the serial port mode. See Table 23.
SM1	S0CON.6	This bit is used to select the serial port mode. See Table 23.
SM2	S0CON.5	Enables the multiprocessor communication feature in modes 2 and 3. In mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1, then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0.
REN	S0CON.4	Enables serial reception. Set by software to enable reception. Clear by software to disable reception.
TB8	S0CON.3	The 9th data bit that will be transmitted in modes 2 and 3. Set or clear by software as desired.
RB8	S0CON.2	In modes 2 and 3, RB8 is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used.
TI	S0CON.1	The transmit interrupt flag. 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, in any serial transmission. Must be cleared by software.
RI	S0CON.0	The receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

Table 23. Description of S0CON bits

SM0	SM1	MODE	DESCRIPTION	BAUD RATE
0	0	0	Shift register	$f_{CLK}/12$
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	$f_{CLK}/64$ or $f_{CLK}/32$
1	1	3	9-bit UART	variable

6.9.2 SIO1 (I^2C-bus Interface)

The SIO1 of the P8xC557E4 provides the fast-mode, which allows a fourthfold increase of the bitrate up to 400 kHz. Nevertheless it is downward compatible, i.e. it can be used in a 0 to 100 Kbit/s I^2C bus system.

Except from the bit rate selection (see Table 25) and the timing of the SCL and SDA signals (see AC electrical characteristics in section 11) the SIO circuit is the same as described in detail in the 80C51 Data Handbook IC20 for the 8xC552 microcontroller.

The I^2C-bus is a simple bidirectional 2-wire bus for efficient inter-IC data exchange. Features of the I^2C-bus are:

- Only two bus lines are required: a serial clock line (SCL) and a serial data line (SDA)
- Each device connected to the bus is software addressable by a unique address
- Masters can operate as Master-transmitter or as Master-receiver
- It's a true multi-master bus including collision detection and arbitration to prevent data corruption if two or more masters simultaneously initiate data transfer
- Serial clock synchronization allows devices with different bit rates to communicate via the same serial bus
- ICs can be added to or removed from an I^2C-bus system without affecting any other circuit on the bus
- Fault diagnostics and debugging are simple; malfunctions can be immediately traced

For more information on the I^2C-bus specification (including fast-mode) please refer to the Philips publication number 9398 393 40011 and/or the 80C51 Data Handbook IC20.

The on-chip I^2C logic provides a serial interface that meets the I^2C-bus specification, supporting all I^2C-bus modes of operation, they are:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver

The SI01 logic performs a byte oriented data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware. Via two pins the external I^2C-bus is interfaced to the SIO1 logic:
SCL serial clock I/O and SDA serial data I/O, (see Special Function Register bit S1CON.6/ENS1 for enabling the SIO1 logic).

The SIO1 logic handles byte transfer autonomously. It keeps track of the serial transfers, and a status register (S1STA) reflects the status of SIO1 and the I^2C-bus.

Via the following four Special Function Registers the CPU interfaces to the I^2C logic.

S1CON	control register. Bit addressable by the CPU
S1STA	status register whose contents may be used as a vector to service routines.
S1DAT	data shift register. The data byte is stable as long as S1CON.3/SI=1.
S1ADR	slave address register. It's LSB enables/ disables general call address recognition.

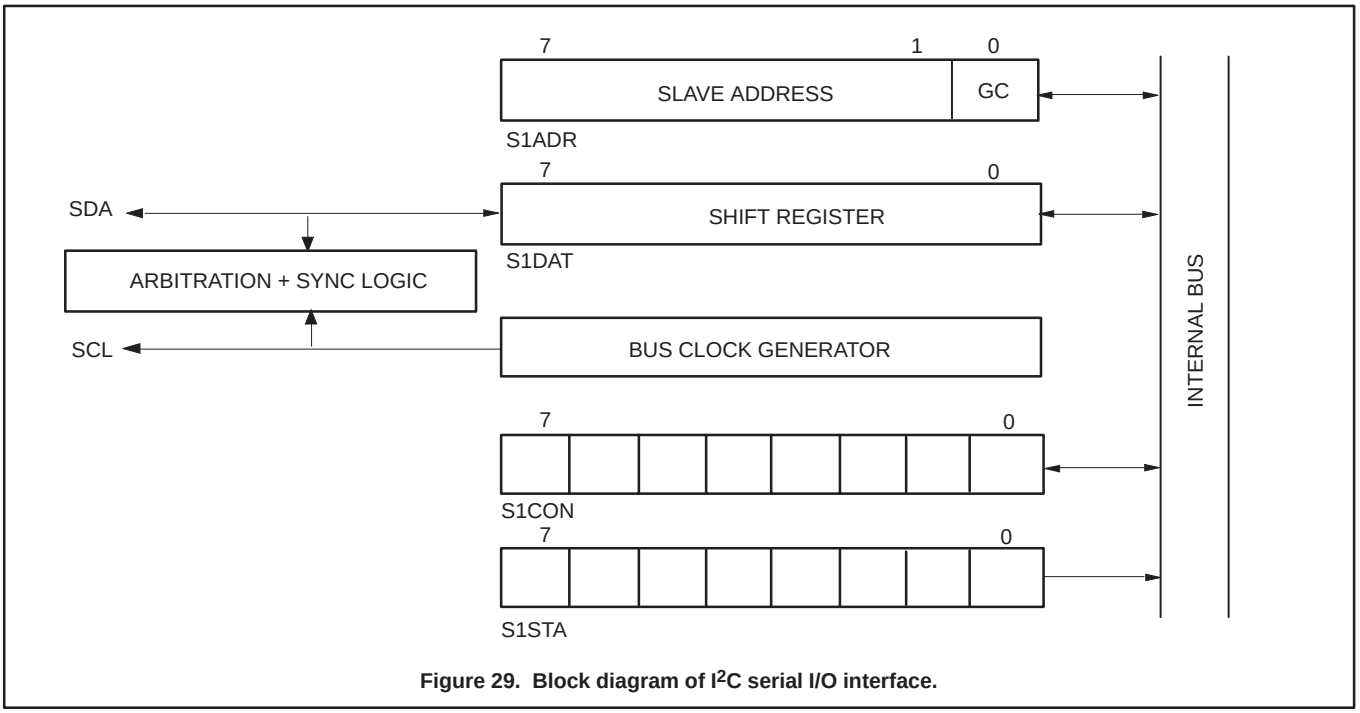


Figure 29. Block diagram of I^2C serial I/O interface.

The Control Register, S1CON:
The CPU can read from and write to this 8-bit, directly addressable SFR. Two bits are affected by the SIO1 hardware: the SI bit is set when a serial interrupt is requested, and the STO bit is cleared when a STOP condition is present on the I^2C bus. The STO bit is also cleared when ENS1 = 0.

	7	6	5	4	3	2	1	0
S1CON (D8H)	CR2	ENS1	STA	STO	SI	AA	CR1	CR0

Figure 30. Serial control (S1CON) register.

Table 24. Description of S1CON bits

SYMBOL	BIT	FUNCTION
CR2	S1CON.7	Clock rate bit 2, see Table 25.
ENS1	S1CON.6	ENS1 = 0: Serial I/O disabled and reset. SDA and SCL outputs are high-Z. ENS1 = 1: Serial I/O enabled.
STA	S1CON.5	START flag. When this bit is set in slave mode, the hardware checks the I^2C bus and generates a START condition if the bus is free or after the bus becomes free. If the device operates in master mode it will generate a repeated START condition.
STO	S1CON.4	STOP flag. If this bit is set in a master mode a STOP condition is generated. A STOP condition detected on the I^2C bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. In this case no STOP condition is generated to the I^2C bus, but the hardware releases the SDA and SCL lines and switches to the not selected receiver mode. The STOP flag is cleared by the hardware.
SI	S1CON.3	Serial Interrupt flag. This flag is set, and an interrupt request is generated, after any of the following events occur: – A START condition is generated in master mode. – The own slave address has been received during AA = 1. – The general call address has been received while S1ADR.0 and AA = 1. – A data byte has been received or transmitted in master mode (even if arbitration is lost). – A data byte has been received or transmitted as selected slave. – A STOP or START condition is received as selected slave receiver or transmitter. While the SI flag is set, SCL remains LOW and the serial transfer is suspended. SI must be reset by software.
AA	S1CON.2	Assert Acknowledge flag. When this bit is set, an acknowledge is returned after any one of the following conditions: – Own slave address is received. – General call address is received (S1ADR.0 = 1). – A data byte is received, while the device is programmed to be a master receiver. – A data byte is received. while the device is a selected slave receiver. When the bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received.
CR1 CR0	S1CON.1 S1CON.0	Clock rate bits 1 and 0, see Table 25.

When SIO1 is in a master mode serial clock frequency is determined by the clock rate bits CR2, CR1 and CR0. The various bit rates are shown in Table 25.

Table 25. Selection of I²C-bus bit rate

CR2	CR1	CR0	BIT RATE (kHz) at f_{CLK}	
			12MHz	16MHz
1	0	0	50	66.7
1	0	1	3.75	5
1	1	0	75	100
1	1	1	100	–
0	0	0	200 [1]	266.7 [1]
0	0	1	7.5	10
0	1	0	300 [1]	400 [1]
0	1	1	400 [1]	–

NOTE:
1. These bit rates are for “fast-mode” I²C bus applications and cannot be used for bit rates up to 100 kbit/sec.

The data shown in Table 25 do not apply to SIO1 in a slave mode. In the slave modes, SIO1 will automatically synchronize with any clock frequency up to 400kHz.

Serial status register S1STA

S1STA is a read only register.

The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I²C-bus.

	7	6	5	4	3	2	1	0
S1STA (D9H)	SC4	SC3	SC2	SC1	SC0	0	0	0

Figure 31. Serial status (S1STA) register.

Table 26. Description of S1STA bits

BIT	FUNCTION
S1STA.7 to 3	5-bit status code
S1STA.2 to 0	These bits are held LOW (for service routine vector increment 8)

The following is a list of the status codes:

Table 27. MST/TRX mode

S1STA VALUE	DESCRIPTION
08H	A START condition has been transmitted
10H	A repeated START condition has been transmitted
18H	SLA and W have been transmitted, ACK has been received
20H	SLA and W have been transmitted, $\overline{ACK}$ received
28H	DATA and S1DAT has been transmitted, ACK received
30H	DATA and S1DAT has been transmitted, $\overline{ACK}$ received
38H	Arbitration lost in SLA, R/W or DATA

Table 28. MST/REC mode

S1STA VALUE	DESCRIPTION
38H	Arbitration lost while returning $\overline{ACK}$
40H	SLA and R have been transmitted, ACK received
48H	SLA and R have been transmitted, $\overline{ACK}$ received
50H	DATA has been received, ACK returned
58H	DATA has been received, $\overline{ACK}$ returned

Table 29. SLV/REC mode

S1STA VALUE	DESCRIPTION
60H	Own SLA and W have been received, ACK returned
68H	Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, $\overline{ACK}$ returned
70H	General CALL has been received, ACK returned
78H	Arbitration lost in SLA, R/W as MST. General call has been received
80H	Previously addressed with own SLA. DATA byte received, ACK returned
88H	Previously addressed with own SLA. DATA byte received, $\overline{ACK}$ returned
90H	Previously addressed with general call. DATA byte has been received, ACK has been returned
98H	Previously addressed with general call. DATA byte has been received, $\overline{ACK}$ has been returned
A0H	A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX

Table 30. SLV/TRX mode

S1STA VALUE	DESCRIPTION
A8H	Own SLA and R have been received, ACK returned
B0H	Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
B8H	DATA byte has been transmitted, ACK returned
C0H	DATA byte has been transmitted, $\overline{ACK}$ returned
C8H	Last DATA byte has been transmitted (AA = logic 0), ACK received

Table 31. Miscellaneous

S1STA VALUE	DESCRIPTION
00H	Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition
F8H	No relevant information available, SI not set

Abbreviations used:

SLA : 7-bit slave address
R : Read bit
W : Write bit
ACK : Acknowledgement (acknowledge bit = 0)
$\overline{ACK}$: Not acknowledgement (acknowledge bit = 1)
DATA : 8-bit data byte to or from I^2C-bus
MST : Master
SLV : Slave
TRX : Transmitter
REC : Receiver

The data shift register S1DAT
This register contains the serial data to be transmitted or data which has been received. Bit 7 is transmitted or received first; i.e., data is shifted from right to left.

	7	6	5	4	3	2	1	0
S1DAT (DAH)	S1DAT.7	S1DAT.6	S1DAT.5	S1DAT.4	S1DAT.3	S1DAT.2	S1DAT.1	S1DAT.0

Figure 32. Data shift register.

The address register S1ADR
This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

	7	6	5	4	3	2	1	0
S1ADR (DBH)	SLA6	SLA5	SLA4	SLA3	SLA2	SLA1	SLA0	GC

Figure 33. Address register.

Table 32. Description of S1ADR bits

SYMBOL	BIT	FUNCTION
SLA6 to 0	S1ADR.7 to 1	Own slave address
GC	S1ADR.0	0 = general call address is not recognized 1 = general call address is recognized

6.10 Interrupt System

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response time in a single-interrupt system is in the range from 2.25µs to 6.75µs when using a 16MHz crystal. The latency time depends on the sequence of instructions executed directly after an interrupt request.

The P8xC557E4 acknowledges interrupt requests from 15 sources as follows (see Figure 34):

- $\overline{INT0}$ and $\overline{INT1}$ external interrupts
- Timer 0 and Timer 1 internal timer/counter interrupts
- Timer 2 internal timer/counter byte and/or 16-bit overflow, 3 compare and 4 capture interrupts (or 4 additional external interrupts) [1]
- UART serial I/O port receive/transmit interrupt
- I^2C-bus interface serial I/O interrupt
- ADC autoscan completion interrupt
- 'Seconds' timer interrupt SEC (ored with $\overline{INT1}$).

For details about seconds timer interrupts, please refer to chapter 6.13.4.

The External Interrupts $\overline{INT0}$ and $\overline{INT1}$ can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in register TCON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON. When an external interrupt is generated, the corresponding request flag is cleared by the hardware when the service routine is vectored to only if the interrupt was transition-activated. If the interrupt was level-activated then the interrupt request flag remains set until the external interrupt pin $\overline{INTx}$ goes high. Consequently the external source has to hold the request active until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated. As these external interrupts are active LOW a "wire-ORing" of several interrupt sources to one input pin allows expansion.

The Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1, which are set by a rollover in their respective timer/counter register (except for Timer 0 in Mode 3 of the serial interface). When a Timer interrupt is generated, the flag that generated it is cleared by the on-chip hardware when the service routine is vectored to.

The eight Timer/Counter T2 Interrupt sources are: 4 capture Interrupts (1), 3 compare interrupts and an overflow interrupt. The appropriate interrupt request flags must be cleared by software.

The UART Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware. The service routine will normally have to determine whether it was RI or TI that generated the interrupt, and the bit will have to be cleared by software.

The I^2C Interrupt is generated by bit SI in register S1CON. This flag has to be cleared by software.

The ADC Interrupt is generated by bit ADINT, which is set when of all selected analog inputs to be scanned, the conversion is finished. ADINT must be cleared by software. It cannot be set by software.

The 'Seconds' timer Interrupt is generated by bit SECINT in register PLLCON. This flag has to be cleared by software. Note that the 'Seconds' timer can only be used with the 32 kHz PLL oscillator.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it had been set or cleared by hardware (except the ADC interrupt request flag ADINT, which cannot be set by software). That is, interrupts can be generated or pending interrupts can be cancelled in software.

The Interrupts X0, T0, X1, T1, SEC, S0 and S1 are capable to terminate the Idle Mode.

Interrupt Enable Registers

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the interrupt enable special function registers IEN0 and IEN1. All interrupt sources can also be globally disabled by clearing bit EA in IEN0. The interrupt enable registers are described in Figures 34 and 36.

Interrupt Priority Structure

Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined by the interrupt priority special function registers IP0 and IP1. IP0 and IP1 are described in Figures 37 and 38.

Interrupt priority levels are as follows:
"0"—low priority
"1"—high priority

A low priority interrupt may be interrupted by a high priority interrupt. A high priority interrupt cannot be interrupted by any other interrupt source. If two requests of different priority occur simultaneously, the high priority level request is serviced. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced. Thus, within each priority level, there is a second priority structure determined by the polling sequence. This second priority structure is shown in Table 37.

Interrupt Handling

The interrupt sources are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the previous machine cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware- generated LCALL is not blocked by any of the following conditions:

1. An interrupt of higher or equal priority level is already in progress.
2. The current machine cycle is not the final cycle in the execution of the instruction in progress. (No interrupt request will be serviced until the instruction in progress is completed.)
3. The instruction in progress is RETI or any access to the interrupt priority or interrupt enable registers. (No interrupt will be serviced after RETI or after a read or write to IP0, IP1, IE0, or IE1 until at least one other instruction has been subsequently executed.)

NOTE:
1. If a capture register is unused and it's contents is of no interest, then the corresponding input pin CTnI/P1.n (n: 0...3) may be used as a (configurable) positive and/or negative edge triggered additional external interrupt input (INT2, INT3, INT4, INT5).

The polling cycle is repeated with every machine cycle, and the values polled are the values present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but is not being responded to because of one of the above conditions, and if the flag is inactive when the blocking condition is removed, then the blocked interrupt will not be serviced. Thus, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate service routine. In some cases it also clears the flag which generated the interrupt, and in others it does not. It clears the Timer 0, Timer 1, and external interrupt flags. An external interrupt flag (IE0 or IE1) is cleared only if it was transition-activated. All other interrupt flags are not cleared by hardware and must be cleared by the software. The LCALL pushes the contents of the program counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to as shown in Table 38.

Execution proceeds from the vector address until the RETI instruction is encountered. The RETI instruction clears the “priority level active” flip-flop that was set when this interrupt was acknowledged. It then pops the top two bytes from the stack and reloads the program counter. Execution of the interrupted program continues from where it was interrupted.

	7	6	5	4	3	2	1	0
IEN0 (A8H)	EA	EAD	ES1	ES0	ET1	EX1	ET0	EX0

Figure 34. Interrupt enable register (IEN0).

Table 33. Description of IEN0 bits

SYMBOL	BIT	FUNCTION
EA	IEN0.7	Global enable/disable control 0 = No interrupt is enabled 1 = Any individually enabled interrupt will be accepted
EAD	IEN0.6	Enable ADC interrupt
ES1	IEN0.5	Enable SIO1 (I^2C) interrupt
ES0	IEN0.4	Enable SIO0 (UART) interrupt
ET1	IEN0.3	Enable Timer 1 interrupt
EX1	IEN0.2	Enable External interrupt 1 / Seconds interrupt
ET0	IEN0.1	Enable Timer 0 interrupt
EX0	IEN0.0	Enable External interrupt 0

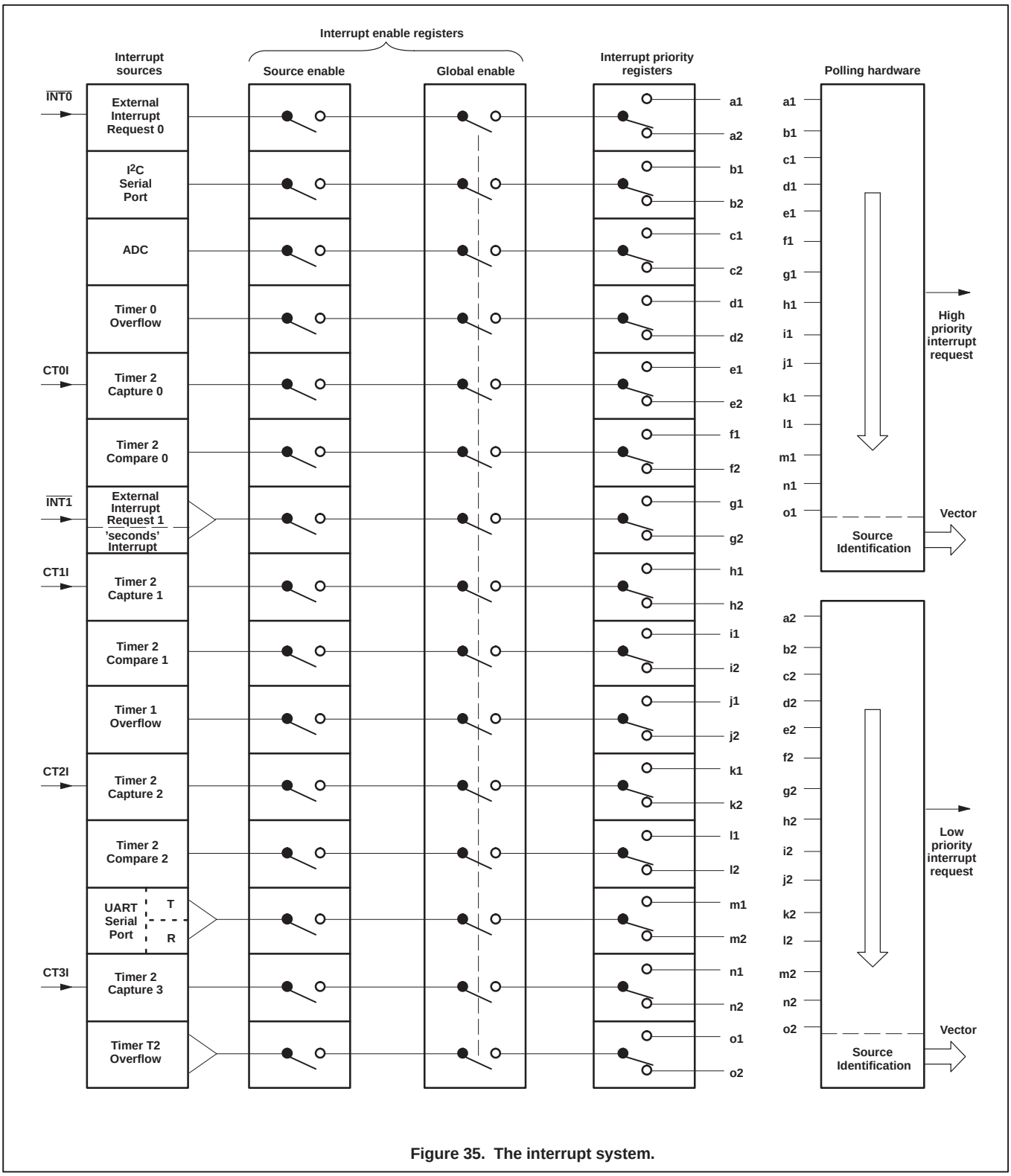


Figure 35. The interrupt system.

	7	6	5	4	3	2	1	0
IEN1 (E8H)	ET2	ECM2	ECM1	ECM0	ECT3	ECT2	ECT1	ECT0

Figure 36. Interrupt enable register (IEN1).

Table 34. Description of IEN1 bits

SYMBOL	BIT	FUNCTION
ET2	IEN1.7	Enable T2 overflow interrupt(s)
ECM2	IEN1.6	Enable T2 comparator 2 interrupt
ECM1	IEN1.5	Enable T2 comparator 1 interrupt
ECM0	IEN1.4	Enable T2 comparator 0 interrupt
ECT3	IEN1.3	Enable T2 capture register 3 interrupt
ECT2	IEN1.2	Enable T2 capture register 2 interrupt
ECT1	IEN1.1	Enable T2 capture register 1 interrupt
ECT0	IEN1.0	Enable T2 capture register 0 interrupt

If the enable bit is 0, then the interrupt is disabled, if the enable bit is 1, then the interrupt is enabled.

	7	6	5	4	3	2	1	0
IP0 (B8H)	–	PAD	PS1	PS0	PT1	PX1	PT0	PX0

Figure 37. Interrupt priority register (IP0).

Table 35. Description of IP0 bits

SYMBOL	BIT	FUNCTION
–	IP0.7	Reserved for future use
PAD	IP0.6	ADC interrupt priority level
PS1	IP0.5	SIO1 (I^2C) interrupt priority level
PS0	IP0.4	SIO0 (UART) interrupt priority level
PT1	IP0.3	Timer 1 interrupt priority level
PX1	IP0.2	External interrupt 1/Seconds interrupt priority level
PT0	IP0.1	Timer 0 interrupt priority level
PX0	IP0.0	External interrupt 0 priority level

	7	6	5	4	3	2	1	0
IP1 (F8H)	PT2	PCM2	PCM1	PCM0	PCT3	PCT2	PCT1	PCT0

Figure 38. Interrupt priority register (IP1).

Table 36. Description of IP1 bits

SYMBOL	BIT	FUNCTION
PT2	IP1.7	T2 overflow interrupt(s) priority level
PCM2	IP1.6	T2 comparator 2 interrupt priority level
PCM1	IP1.5	T2 comparator 1 interrupt priority level
PCM0	IP1.4	T2 comparator 0 interrupt priority level
PCT3	IP1.3	T2 capture register 3 interrupt priority level
PCT2	IP1.2	T2 capture register 2 interrupt priority level
PCT1	IP1.1	T2 capture register 1 interrupt priority level
PCT0	IP1.0	T2 capture register 0 interrupt priority level

Table 37. Interrupt Priority Structure

SOURCE	NAME	PRIORITY WITHIN LEVEL
		(highest) ↑
External interrupt 0	X0	
SIO1 (I^2C)	S1	
ADC completion	ADC	
Timer 0 overflow	T0	
Timer 2 capture 0	CT0	
Timer 2 compare 0	CM0	
External interrupt 1/Seconds interrupt	X1/SEC	
Timer 2 capture 1	CT1	
Timer 2 compare 1	CM1	
Timer 1 overflow	T1	
Timer 2 capture 2	CT2	
Timer 2 compare 2	CM2	
SIO0 (UART)	S0	
Timer 2 capture 3	CT3	
Timer 2 overflow	T2	↓ (lowest)

Table 38. Interrupt Vector Addresses

SOURCE	NAME	VECTOR ADDRESS
External interrupt 0	X0	0003H
Timer 0 overflow	T0	000BH
External interrupt 1/Seconds interrupt	X1/SEC	0013H
Timer 1 overflow	T1	001BH
SIO0 (UART)	S0	0023H
SIO1 (I^2C)	S1	002BH
Timer 2 capture 0	CT0	0033H
Timer 2 capture 1	CT1	003BH
Timer 2 capture 2	CT2	0043H
Timer 2 capture 3	CT3	004BH
ADC completion	ADC	0053H
Timer 2 compare 0	CM0	005BH
Timer 2 compare 1	CM1	0063H
Timer 2 compare 2	CM2	006BH
Timer 2 overflow	T2	0073H

	7	6	5	4	3	2	1	0
PCON (87H)	SMOD	ARD	RFI	WLE	GF1	GF0	PD	IDL

Figure 39. Power control register (PCON).

Table 39. Description of PCON bits

SYMBOL	BIT	FUNCTION
SMOD	PCON.7	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2, or 3.
ARD	PCON.6	AUX-RAM disable bit. When set to a 1 the internal 768 bytes AUX-RAM is disabled, so that all MOVX-Instructions access the external data memory – as it is with the standard PCB80C51.
RFI	PCON.5	Reduced radio frequency interference bit. When set to a 1 the toggling of ALE pin is prohibited. This bit is cleared on RESET (see also sections Features (EMC) and Pinning).
WLE	PCON.4	Watchdog load enable. This flag must be set by software prior to loading timer T3 (watchdog timer). It is cleared when timer T3 is loaded.
GF1	PCON.3	General-purpose flag bit
GF0	PCON.2	General-purpose flag bit
PD	PCON.1	Power-down bit. Setting this bit activates the power-down mode. It can only be set if input $\overline{EW}$ is high.
IDL	PCON.0	Idle Mode bit. Setting this bit activates the Idle Mode.

6.11 Power Reduction Modes

Two software-selectable modes of reduced power consumption are implemented. These are the Idle Mode and the Power-down Mode.

Idle Mode operation permits the interrupt, serial ports and timer blocks T0, T1 and T3 to function while the CPU is halted. The following functions are switched off when the microcontroller enters the Idle Mode:

- CPU (halted)
- Timer 2 (stopped and reset)
- PWM0, PWM1 (reset, output = HIGH)
- ADC (aborted if conversion in progress)

The following functions remain active during Idle Mode. These functions may generate an interrupt or reset and thus terminate the Idle Mode:

- Timer 0, Timer 1, Timer 3 (Watchdog timer)
- UART
- I^2C
- External interrupt
- Seconds Timer

In Power-down Mode the system clock is halted. If the PLL oscillator is selected (SELXTAL1 = 0) and the RUN32 bit is set, the 32 kHz oscillator keeps running, otherwise it is stopped. If the HF-oscillator (SELXTAL1 = 1) is selected, it is stopped after setting the bit PD in the PCON register.

Table 40. External Pin Status During Idle and Power-Down Modes

MODE	MEMORY	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3	PORT 4	SCL/SDA	$\overline{PWM0}$/$\overline{PWM1}$
Idle	Internal	1	1	data	data	data	data	data	operative (1)	HIGH
Idle	External	1	1	high-Z	data	address	data	data	operative (1)	HIGH
Power-down	Internal	0	0	data	data	data	data	data	high-Z	HIGH
Power-down	External	0	0	high-Z	data	data	data	data	high-Z	HIGH

NOTE:
1. In Idle Mode SCL and SDA can be active as outputs only if SIO1 is enabled; if SIO1 is disabled (S1CON.6/ENS1 = 0) these pins are in a high-impedance state.

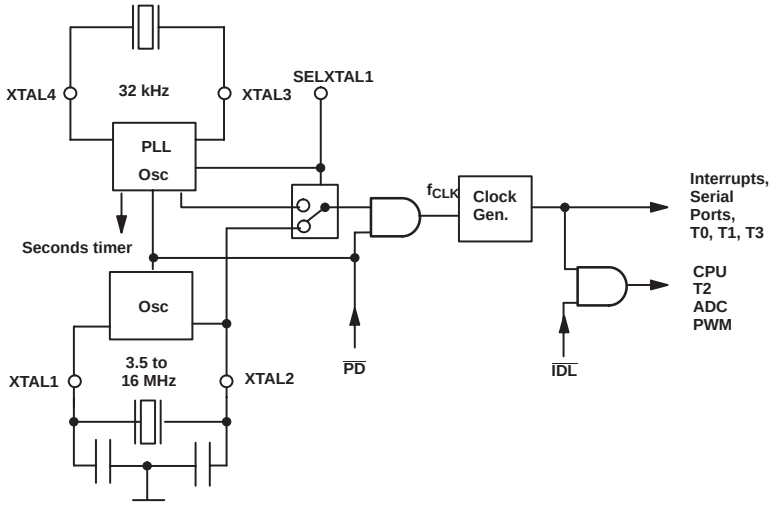


Figure 40. Idle and Power Down Hardware for Clock Generation

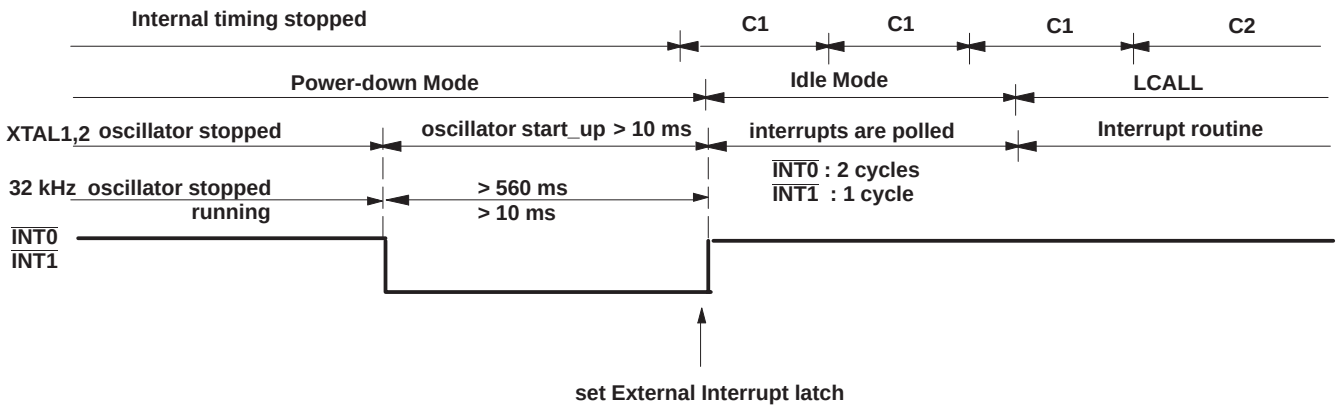


Figure 41. Wake-up by interrupt

6.11.1 Power Control Register

The modes Idle and Power-down are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable. The reset value of PCON is (00000000).

6.11.2 Idle Mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle Mode is activated. Once in the Idle Mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle Mode. The status of external pins during Idle Mode is shown in Table 40.

There are three ways to terminate the Idle Mode:

Activation of any enabled interrupt X0, T0, X1, SEC, T1, S0 or S1 will cause PCON.0 to be cleared by hardware terminating Idle Mode but only, if there is no interrupt in service with the same or higher priority. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during Idle Mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle Mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle Mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 HF oscillator periods) to complete the reset operation if the HF oscillator is selected.

When the PLL oscillator is selected a hardware reset of $\geq$ 1 µsec (but no longer than 10 ms) is required and the microcontroller will typically restart within 63 msec after the reset has finished.

The third way of terminating the Idle Mode is by internal watchdog reset. The microcontroller restarts after 3 machine cycles in all cases.

6.11.3 Power-down Mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down Mode. Once in Power-down Mode, the HF oscillator is stopped. The 32 kHz oscillator may stay running. The content of the on-chip RAM and the Special Function Registers are preserved. Note that the Power-down Mode can not be entered when the watchdog has been enabled.

The Power-down Mode can be terminated by an external RESET in the same way as in the 80C51 (RAM is saved, but SFRs are cleared due to RESET) or in addition by any one of the external interrupts ($\overline{INT0}$, $\overline{INT1}$) or Seconds interrupt.

The status of the external pins during Power-down Mode is shown in Table 40. If the Power-down Mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2.

If the data is a logic1, the port pin is held HIGH during the Power-down Mode by the strong pull-up transistor P1 (see Figure 9).

The Power-down Mode should not be entered within an interrupt routine because Wake-up with an external or 'Seconds' interrupt is not possible in that case.

6.11.4 Wake-up from Power-down Mode

The Power-down Mode of the P8xC557E4 can also be terminated by any one of the three enabled interrupts, $\overline{INT0}$, $\overline{INT1}$ or Seconds interrupt.

If there is an interrupt already in service, which has same or higher priority as the Wake-up interrupt, Power-down Mode will switch over to Idle Mode and stay there until an interrupt of higher priority terminates Idle Mode.

A termination with these interrupts does not affect the internal data memory and does not affect the Special Function Registers. This gives the possibility to exit Power-down without changing the port output levels. To terminate the Power-down Mode with an external interrupt, $\overline{INT0}$ or $\overline{INT1}$ must be switched to be level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ or $\overline{INT1}$ must be kept LOW till the oscillator has restarted and stabilized (see Figure 41). A Seconds interrupt will terminate the Power-down Mode if it is enabled and $\overline{INT1}$ is level sensitive. In order to prevent any interrupt priority problems during Wake-up, the priority of the desired Wake-up interrupt should be higher than the priorities of all other enabled interrupt sources.

The instruction following the one that put the device into the Power-down Mode will be the first one which will be executed after the interrupt routine has been serviced.

6.12 Oscillator Circuits

The input signal SELXTAL1 connected to logic "1" selects the XTAL1, 2 oscillator (standard 80C51) instead of the XTAL3, 4 oscillator, which is halted and XTAL3, 4 must not be connected.

6.12.1 XTAL1, 2 Oscillator circuit (standard 80C51)

The oscillator circuit of the P8xC557E4 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between the XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 is the high gain amplifier input, and XTAL2 is the output (see Figure 42). To drive the P8xC557E4 externally, XTAL1 is driven from an external source and XTAL2 left open-circuit (see Figure 43).

6.12.2 XTAL3, 4 Circuitry

Please refer to chapter 6.13.1

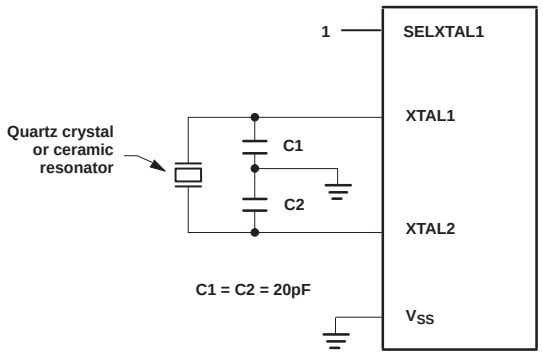


Figure 42. Using the On-Chip Oscillator.

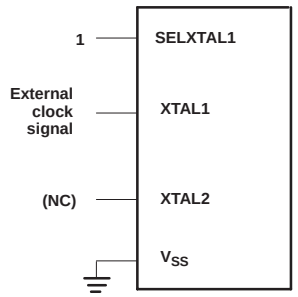


Figure 43. Using an external clock.

6.13 32kHz PLL Oscillator with Seconds Timer

6.13.1 XTAL3,4 Oscillator Circuitry

The input signal SELXTAL1 connected to logic “0” selects the 32kHz oscillator together with the PLL instead of the XTAL1,2 oscillator, which is halted. XTAL2 is floating in that case.

The 32kHz oscillator consists of an inverter, which forms a Pierce oscillator with the on-chip components C1,C2,Rf and an external crystal of 32768 Hz.

During the following situations, the inverter is switched to tristate and XTAL3 is pulled to Vss :

- during Power-down Mode, when the PLL control register bit RUN32 (PLLCON.7) was set to ’0’;
- during Reset (RSTIN = HIGH) ;
- when the XTAL1,2 oscillator is selected (SELXTAL1 = HIGH).

6.13.2 PLL CCO

A current controlled oscillator (CCO) generates a clock frequency f_{CCO} of approx. 32 , 38 , 44 or 50 MHz , controlled by the PLL, with the 32kHz oscillator as the reference clock. The system clock frequency f_{CLK} can be varied under software control by changing the contents of the PLL control register (PLLCON):

f_{CCO} can be changed via the PLLCON bits FSEL(1:0) (see Table 41). The maximum locking time is 10 ms[1].

During the stabilization phase, no time critical routines should be executed.

The system clock frequency f_{CLK} is derived from f_{CCO} under control of the PLLCON bits FSEL(4:0) (see Table 41).

If only FSEL(4:2) is changed but not FSEL(1:0), then it takes about 1us until the new frequency is available.

Changing the system clock frequency has to be done in two steps.

From HIGH to LOW frequencies:
First change (FSEL(4:2), then FSEL (1:0).

From LOW to HIGH frequencies:
First change only FSEL (1:0) and after a stabilization phase of 10 ms change FSEL (4:2).

6.13.3 PLL Control Register – PLLCON

PLLCON is a special function register, which can be read and written by software. It contains the control bits:

- to select one of several system clock frequencies (see Table 41);
- the seconds interrupt flag: SECINT
- to enable the seconds interrupt flag: ENSECI
- the RUN32 bit, which defines if during Power-down Mode the 32kHz oscillator is halted or stays running.

PLLCON is initialized to 0DH upon Reset (RSTIN = ‘1’) or Watchdog Timer Overflow. PLLCON = 0DH corresponds to a system clock frequency of 11.01 MHz.

	7	6	5	4	3	2	1	0
PLLCON (F9H)	RUN32	ENSECI	SECINT	FSEL.4	FSEL.3	FSEL.2	FSEL.1	FSEL.0

Figure 44. PLL control register (PLLCON).

Table 41. PLLCON

SYMBOL	BIT	FUNCTION
RUN32	PLLCON.7	RUN32 = 0: The 32 kHz oscillator halts during Power-down. RUN32 = 1: The 32 kHz oscillator stays running during Power-down.
ENSECI	PLLCON.6	Enable the seconds interrupt. (enabling INT1 is also required)
SECINT	PLLCON.5	Seconds interrupt requested by an overflow of the seconds timer (which occurs every second) or via writing a ‘1’ to this bit. SECINT can only be cleared by writing a ’0’ to this bit .
FSEL.4 to FSEL.0	PLLCON.4 to PLLCON.0	System clock frequency in MHz

		FSEL[4:2] = 100	011	010
FSEL[1:0]	11	3.93	7.86	15.73
	10	4.72	9.44	
	01	5.51	11.01	
	00	6.29	12.58	

Other combinations, than mentioned above, are reserved and may not be selected. This allows to generate the standard baudrates 19200, 9600, 4800, 2400 and 1200 Baud, when using the UART and Timer1.

NOTE:
1. This parameter is characterized.

6.13.4 Seconds Timer

This counter provides an overflow signal every second, when the 32kHz oscillator is running.

The overflow output sets the interrupt flag SECINT. This interrupt can be disabled/enabled by ENSECI. If SECINT is enabled, it is logically ORed with INT1 (external interrupt 1).

Seconds interrupt and INT1 therefor share the same priority and vector. The software has to check both flags SECINT (PLLCON.5) and IE1 (TCON.3), to distinguish between the two interrupt sources. SECINT can only be cleared via writing a '0' to this bit .

The external interrupts INT0 , INT1 or the seconds interrupt can Wake-up the PLL oscillator and the microcontroller as described in chapter "Wake-up from Power-down Mode".

For a Wake-up via INT1 or seconds interrupt, IE1 must be enabled and level-sensitive.

A further function of the seconds timer is to control the start-up timing of the microcontroller after Reset or after Wake-up from Power-down. It controls the stretching of the reset pulse to the microcontroller and controls releasing the system clock to the microcontroller.

A RSTIN signal of 1us at minimum will reset the microcontroller.

In case of Reset or Wake-up with halted 32kHz oscillator: From RSTIN falling edge or Wake-up interrupt it takes 560ms at maximum for the start-up of the 32kHz oscillator itself and the stabilization of the PLL's.

In case of Wake-up with running 32kHz oscillator: From Wake-up interrupt it takes about 1ms for the stabilization of the PLL's.

After this start-up time, the microcontroller is supplied with the system clock and – in case of a reset – the internally stretched reset signal overlaps about 45us, to guarantee a proper initialization of the microcontroller.

For further information refer to section ***6.11 Power reduction modes***.

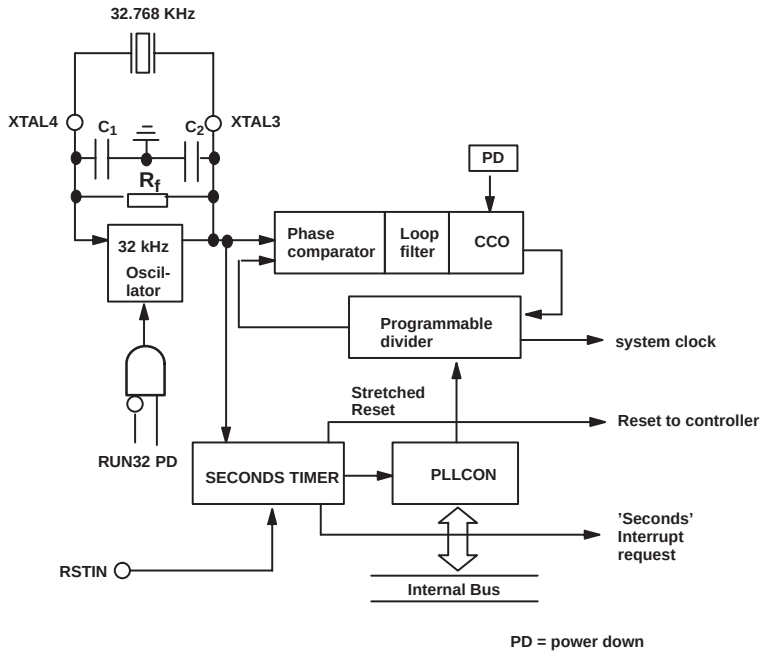


Figure 45. Block diagram PLL

6.14 Reset Circuitry

The reset input pin RSTIN is connected to a Schmitt trigger for noise reduction (see Figure 46). Is the HF-oscillator selected a Reset is accomplished by holding the RSTIN pin HIGH for at least 2 machine cycles (24 system clock periods). Is the PLL-oscillator selected the RSTIN-pulse must have a width of 1 µs at least, independent of the 32 kHz-oscillator is running or not (see PLL description). The CPU responds by executing an internal reset. The RSTOUT pin represents the signal resetting the CPU and can be used to reset peripheral devices.

The RSTOUT level also could be high due to a Watchdog timer overflow.

The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

During Reset, ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

A Reset leaves the internal registers as shown in Table 5.

The internal RAM is not affected by Reset. At power-on, the RAM content is indeterminate.

6.15 Power-on Reset

An automatic Reset can be obtained by switching on V_{DD}, if the RSTIN pin is connected to V_{DD} via a capacitor, as shown in Figure 47.

Is the HF oscillator selected the V_{DD} rise time must not exceed 10 ms and the capacitor should be at least 2.2 µF. The decrease of the RSTIN pin voltage depends on the capacitor and the internal resistor R_{RST}. That voltage must remain above the lower threshold for at minimum the HF-oscillator start-up time plus 2 machine cycles. Is the PLL-oscillator selected a 0.1 µF capacitor is sufficient to obtain an automatic reset.

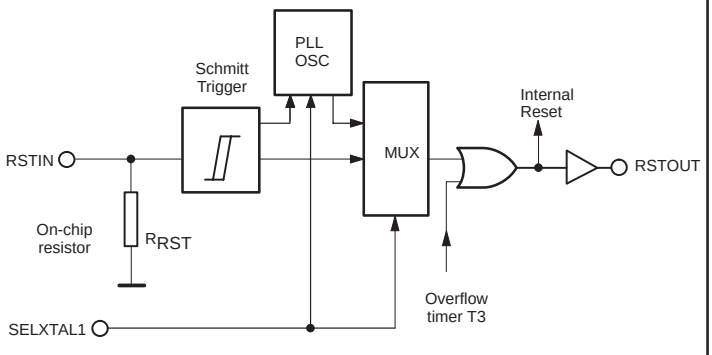


Figure 46. On-chip Reset Configuration

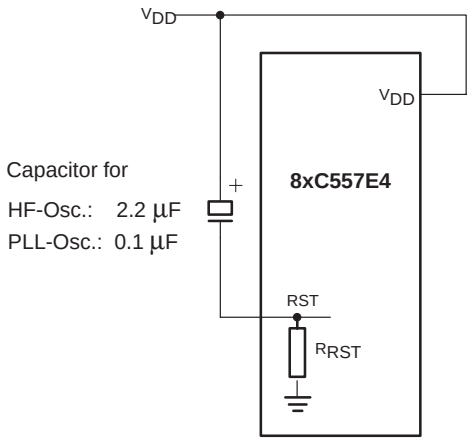


Figure 47. Power-on Reset

7. INSTRUCTION SET

The P8xC557E4 uses the powerful instruction set of the PCB80C51. It consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. Using a 16 MHz quartz, 64 of the instructions are executed in 0.75 μs, 45 in 1,5 μs and the multiply, divide instructions in 3 μs.

A summary of the instruction set is given in Table 43.

The P8xC557E4 has additional Special Function Registers to control the on-chip peripherals.

7.1 Addressing Modes

Most instructions have a “destination, source” field that specifies the data type, addressing modes and operands involved. For all these instructions, except for MOVs, the destination operand is also the source operand (e.g., ADD A,R7).

There are five kinds of addressing modes:

- Register Addressing
  - R0 – R7 (4 banks)
  - A,B,C (bit), AB (2 bytes), DPTR (double byte)
- Direct Addressing
  - lower 128 bytes of internal Main RAM (including the 4 R0–R7 register banks)
  - Special Function Registers
  - 128 bits in a subset of the internal Main RAM
  - 128 bits in a subset of the Special Function Registers
- Register-Indirect Addressing
  - internal Main RAM (@R0, @R1, @SP [PUSH/POP])
  - internal Auxiliary RAM (@R0, @R1, @DPTR)
  - external Data Memory (@R0, @R1, @DPTR)
- Immediate Addressing
  - Program Memory (in-code 8 bit or 16 bit constant)
- Base-Register-plus Index-Register-Indirect Addressing
  - Program Memory look-up table (@DPTR+A, @PC+A)

The first three addressing modes are usable for destination operands.

7.1.1 80C51 Family Instruction Set

Table 42. Instruction that affect Flag settings[1]

INSTRUCTION	FLAG		
	C	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	0	X	
DIV	0	X	
DA	X	X	
RRC	X		
RLC	X		
SETB C	1		
CLR C	0		
CPL C	X		
ANL C, bit	X		
ANL C,/bit	X		
ANL C, bit	X		
ORL C, bit	X		
MOV C, bit	X		
CJNE	X		

NOTES:
1. Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

Notes on instruction set and addressing modes:

Rn	Register R7-R0 of the currently selected Register Bank.
direct	8-bit internal data location’s address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
@Ri	8-bit RAM location addressed indirectly through register R1 or R0 of the actual register bank.
#data	8-bit constant included in the instruction.
#data 16	16-bit constant included in the instruction
addr 16	16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64 Kbytes Program Memory address space.
addr 11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 Kbytes page of program memory as the first byte of the following instruction.
rel	Signed (two’s complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction.
bit	Direct Addressed bit in Internal Data RAM or Special Function Register.

Hexadecimal opcode cross-reference to Table 43:

*	:	8, 9, A, B, C, D, E. F.
**	:	11, 31, 51, 71, 91, B1, D1, F1.
***	:	01, 21, 41, 61, 81, A1, C1, E1.

Table 43. 80C51 Instruction Set Summary

MNEMONIC		DESCRIPTION	BYTE	CYCLES	OPCODE (HEX.)
ARITHMETIC OPERATIONS					
ADD	A,Rn	Add register to Accumulator	1	1	2*
ADD	A,direct	Add direct byte to Accumulator	2	1	25
ADD	A,@Ri	Add indirect RAM to Accumulator	1	1	26, 27
ADD	A,#data	Add immediate data to Accumulator	2	1	24
ADDC	A,Rn	Add register to Accumulator with carry	1	1	3*
ADDC	A,direct	Add direct byte to Accumulator with carry	2	1	35
ADDC	A,@Ri	Add indirect RAM to Accumulator with carry	1	1	36, 37
ADDC	A,#data	Add immediate data to ACC with carry	2	1	34
SUBB	A,Rn	Subtract Register from ACC with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from ACC with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from ACC with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from ACC with borrow	2	1	94
INC	A	Increment Accumulator	1	1	04
INC	Rn	Increment register	1	1	0*
INC	direct	Increment direct byte	2	1	05
INC	@Ri	Increment indirect RAM	1	1	06, 07
DEC	A	Decrement Accumulator	1	1	14
DEC	Rn	Decrement Register	1	1	1*
DEC	direct	Decrement direct byte	2	1	15
DEC	@Ri	Decrement indirect RAM	1	1	16, 17
INC	DPTR	Increment Data Pointer	1	2	A3
MUL	AB	Multiply A and B	1	4	A4
DIV	AB	Divide A by B	1	4	84
DA	A	Decimal Adjust Accumulator	1	1	D4
LOGICAL OPERATIONS					
ANL	A,Rn	AND Register to Accumulator	1	1	5*
ANL	A,direct	AND direct byte to Accumulator	2	1	55
ANL	A,@Ri	AND indirect RAM to Accumulator	1	1	56, 57
ANL	A,#data	AND immediate data to Accumulator	2	1	54
ANL	direct,A	AND Accumulator to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rn	OR register to Accumulator	1	1	4*
ORL	A,direct	OR direct byte to Accumulator	2	1	45
ORL	A,@Ri	OR indirect RAM to Accumulator	1	1	46, 47
ORL	A,#data	OR immediate data to Accumulator	2	1	44
ORL	direct,A	OR Accumulator to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rn	Exclusive-OR register to Accumulator	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to Accumulator	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to Accumulator	1	1	66, 67

Table 43. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	CYCLES	OPCODE (HEX.)
LOGICAL OPERATIONS (Continued)					
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	1	64
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear Accumulator	1	1	E4
CPL	A	Complement Accumulator	1	1	F4
RL	A	Rotate Accumulator left	1	1	23
RLC	A	Rotate Accumulator left through the carry	1	1	33
RR	A	Rotate Accumulator right	1	1	03
RRC	A	Rotate Accumulator right through the carry	1	1	13
SWAP	A	Swap nibbles within the Accumulator	1	1	C4
DATA TRANSFER					
MOV	A,Rn	Move register to Accumulator	1	1	E*
MOV	A,direct	Move direct byte to Accumulator	2	1	E5
MOV	A,@Ri	Move indirect RAM to Accumulator	1	1	E6, E7
MOV	A,#data	Move immediate data to Accumulator	2	1	74
MOV	Rn,A	Move Accumulator to register	1	1	F*
MOV	Rn,direct	Move direct byte to register	2	2	A*
MOV	RN,#data	Move immediate data to register	2	1	7*
MOV	direct,A	Move Accumulator to direct byte	2	1	F5
MOV	direct,Rn	Move register to direct byte	2	2	8*
MOV	direct,direct	Move direct byte to direct	3	2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3	2	75
MOV	@Ri,A	Move Accumulator to indirect RAM	1	1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1	76, 77
MOV	DPTR,#data16	Load Data Pointer with a 16-bit constant	3	2	90
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to ACC	1	2	93
MOVC	A,@A+PC	Move Code byte relative to PC to ACC	1	2	83
MOVX	A,@Ri	Move AUX-RAM (8-bit addr) to ACC	1	2	E2, E3
MOVX	A,@DPTR	Move AUX-RAM (16-bit addr) to A_{CC}	1	2	E0
MOVX	@Ri,A	Move ACC to AUX-RAM (8-bit addr)	1	2	F2, F3
MOVX	@DPTR,A	Move ACC to AUX-RAM (16-bit addr)	1	2	F0
PUSH	direct	Push direct byte onto stack	2	2	C0
POP	direct	Pop direct byte from stack	2	2	D0
XCH	A,Rn	Exchange register with Accumulator	1	1	C*
XCH	A,direct	Exchange direct byte with Accumulator	2	1	C5
XCH	A,@Ri	Exchange indirect RAM with Accumulator	1	1	C6, C7
XCHD	A,@Ri	Exchange low-order digit indirect RAM with ACC	1	1	D6, D7

Table 43. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	CYCLES	OPCODE (HEX.)
BOOLEAN VARIABLE MANIPULATION					
CLR	C	Clear carry	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry	2	2	B2
ANL	C,/bit	AND complement of direct bit to carry	2	2	B0
ORL	C,bit	OR direct bit to carry	2	2	72
ORL	C,/bit	OR complement of direct bit to carry	2	2	A0
MOV	C,bit	Move direct bit to carry	2	1	A2
MOV	bit,C	Move carry to direct bit	2	2	92
JC	rel	Jump if carry is set	2	2	40
JNC	rel	Jump if carry not set	2	2	50
JB	rel	Jump if direct bit is set	2	2	20
JNB	rel	Jump if direct bit is not set	2	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
PROGRAM BRANCHING					
ACALL	addr11	Absolute subroutine call	2	2	**1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	***1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative addr)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if Accumulator is zero	2	2	60
JNZ	rel	Jump if Accumulator is not zero	2	2	70
CJNE	A,direct,rel	Compare direct byte to ACC and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to ACC and jump if not equal	3	2	B4
CJNE	RN,#data,rel	Compare immediate to register and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immediate to indirect and jump if not equal	3	2	B6, B7
DJNZ	Rn,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct byte and jump if not zero	3	2	D5
NOP		No operation	1	1	00

NOTE:

1. All mnemonics copyrighted © Intel Corporation 1980

Table 44. Instruction map P8xC557E4

second hexadecimal character of opcode

first hexadecimal character of opcode	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @ Ri		INC Rr							
							0	1	0	1	2	3	4	5	6	7
1	JBC bit, rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @ Ri		DEC Rr							
							0	1	0	1	2	3	4	5	6	7
2	JB bit, rel	AJMP addr11	RET	RL A	ADD A, #data	ADD A, dir	ADD A, @ Ri		ADD A, Rr							
							0	1	0	1	2	3	4	5	6	7
3	JNB bit, rel	ACALL addr11	RETI	RLC A	ADDC A, #data	ADDC A, dir	ADDC A, @ Ri		ADDC A, Rr							
							0	1	0	1	2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir, A	ORL dir, #data	ORL A, #data	ORL A, dir	ORL A, @ Ri		ORL A, Rr							
							0	1	0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir, A	ANL dir, #data	ANL A, #data	ANL A, dir	ANL A, @ Ri		ANL A, Rr							
							0	1	0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir, A	XRL dir, #data	XRL A, #data	XRL A, dir	XRL A, @ Ri		XRL A, Rr							
							0	1	0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C, bit	JMP @A+DPTR	MOV A, #data	MOV dir,#data	MOV @ Ri, #data		MOV Rr, #data							
							0	1	0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C, bit	MOVC A, @A+PC	DIV AB	MOV dir, dir	MOV dir, @ Ri		MOV dir, Rr							
							0	1	0	1	2	3	4	5	6	7
9	MOV DPTR,#data16	ACALL addr11	MOV bit, C	MOVC A,@A+DPTR	SUBB A, #data	SUBB A, dir	SUBB A, @ Ri		SUBB A, Rr							
							0	1	0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C, bit	INC DPTR	MUL AB		MOV @ Ri, dir		MOV Rr, dir							
							0	1	0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A,#data,rel	CJNE A,dir, rel	CJNE @Ri,#data,rel		CJNE Rr, #data, rel							
							0	1	0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A, dir	XCH A, @ Ri		XCH A, Rr							
							0	1	0	1	2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DNJZ dir, rel	XCHD A, @ Ri		DJNZ Rr, rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A, @DPTR	AJMP addr11	MOVX A, @Ri		CLR A	MOV A, dir *)	MOV A, @ Ri		MOV A, Rr							
			0	1			0	1	0	1	2	3	4	5	6	7
F	MOVX @DPTR, A	ACALL addr11	MOVX A, @Ri, A		CPL A	MOV dir, A	MOV @ Ri, A		MOV Rr, A							
			0	1			0	1	0	1	2	3	4	5	6	7

***) MOV A, ACC is not a valid instruction**

8. FLASH EEPROM

8.1 General

- 32 Kbytes electrically erasable internal program memory with Block-and Page-Erase option (”Flash Memory”).
- Internal fixed boot ROM.
- Up to 32 Kbytes external program memory in combination with the internal FEEPROM ($\overline{EA}$=1).
- Up to 64 Kbytes external program memory if the internal program memory is switched off ($\overline{EA}$=0).

The FEEPROM can be read and written byte-wise. Full Erase, Block Erase, and Page erase will erase 32 Kbytes, 256 bytes and 32 bytes respectively. In-circuit programming and out-of-circuit programming is possible. On-chip erase and write timing generation and on chip high voltage generation contribute to a user friendly interface.

8.2 Features

- Read:
  byte-wise
- Write:
  byte-wise within 2.5 ms.
  (previously erased by a page, block or full erase).
- Erase:
  Page Erase (32 bytes) within 5 ms.
  Block Erase (256 bytes) within 5 ms.
  Full Erase (32 Kbytes) within 5 ms.
  Erased bytes contain FFH.
- Endurance:
  100 erase and write cycles each byte at T_{amb} = 22°C
- Retention:
  10 years
- Out-of-circuit programming:
  Parallel programming with 87C51 compatible hardware Interface to programmer.
- In-circuit programming:
  Serial programming via RS232 interface under boot ROM program control. Auto baud rate selection.
  Intel Hex Object file Format.
  The user program can call routines in the boot ROM for erase, write and verify of the FEEPROM.
- High programming voltage generation: on chip
- Zero point on-chip oscillator and timer to generate the write and erase time durations.
- Programmable security for the code in the FEEPROM to prevent software piracy. The Security Byte is located in the highest address (7FFFH) of the FEEPROM.
- Supply voltage monitoring circuit on-chip to prevent loss of information in the FEEPROM during power-on and power-off.

8.3 Memory Map

Figure 48 shows the memory map of the user program memory and the boot ROM. They are located in the same program address space. Two bits UBS1 and UBS0 of the FEEPROM control special function register FMCON select between the two memory blocks.

User program memory selection
If UBS1 and UBS0 are both 0, then the user program memory is mapped into the 64 K program memory space and the boot ROM cannot be selected. This is the situation after a reset when $\overline{PSEN}$ and ALE have not been pulled down during reset. Program execution starts at 0000H in the internal FEEPROM or in the external program memory dependent on the level of $\overline{EA}$ during reset.

Boot ROM selection
After a reset program execution starts in the boot ROM when during reset $\overline{PSEN}$ and $\overline{EA}$ are pulled down while ALE stay high. The boot ROM size is 1 Kbyte. Besides the serial in-circuit programming routine the boot ROM contains the routines for erase, write and verify of the FEEPROM, which can be called by the user program (LCALL to the address space between 63 K and 64 K).

Switching between user program memory and boot ROM
Switching between user program memory (internal or external) and boot ROM is possible if UBS1 and UBS0 are 0,1. Then in the program memory address space between 0 and 63k the user program memory is selected and in the memory space between 63 K and 64 K the boot ROM is selected.

To switch from user program memory to boot ROM first UBS0 must be set (UBS1 stay 0) and a jump or call instruction to a location >63 K must be executed.

At the moment of crossing the 63 K address border by a return instruction the switching from boot ROM to user memory (internal or external) is performed. After crossing the 63 K address border UBS1 and UBS0 are cleared and the total 64 K memory space is mapped as user program memory. By clearing UBS1 and UBS0, no special requirements to the user program are necessary to do that after a read or erase or write routine.

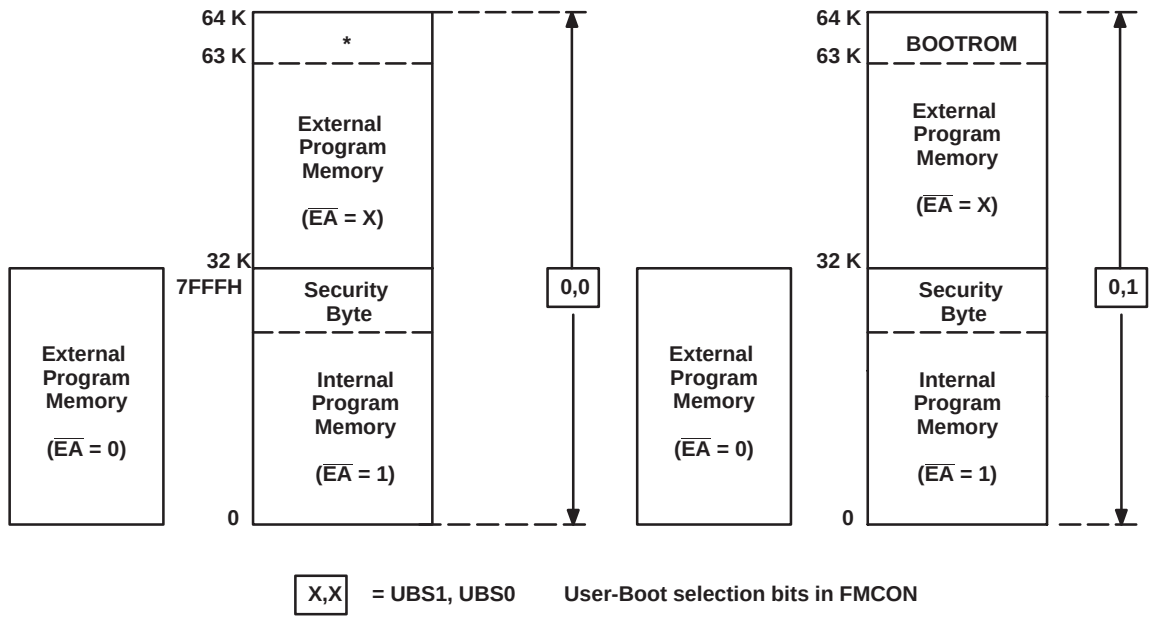
A small restriction for memory switching is that no memory switching is allowed from or to the address space between 63 K and 64 K of the user program memory because the UBS bits must stay 0 in this range. This restriction can be avoided if the memory switching is always done by a subroutine in the address range between 0 and 63 K.

Description

The user program code in the FEEPROM is executed as in the standard 80C51 microcontroller. Erase and write cycles in the FEEPROM are always performed under control of the boot program in the boot ROM in the address space between 63 K and 64 K. Address and data parameters are passed via DPTR and accumulator A respectively. During an erase or write cycle in the FEEPROM no other access or program execution in the FEEPROM is possible. All interrupts must be disabled when the user program calls a user routine in the boot ROM.

The boot routine for serial programming takes care of addressing, data transfer, verify, high voltage control, error message and return to the user program memory. It also contains the serial communication routine.

The FEEPROM control register FMCON is a special function register. It contains the control bits for verify, write, erase and boot ROM switching.



* In the program execution between 63k and 64k setting of UBS bits is not allowed.

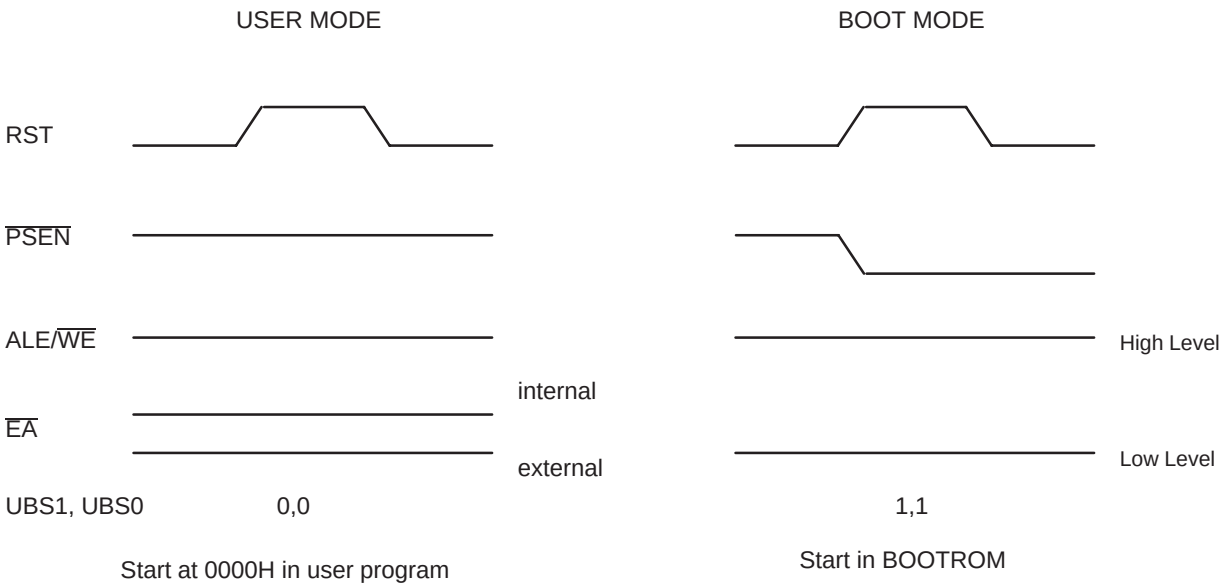


Figure 48. Program memory map and operation modes

	7	6	5	4	3	2	1	0
FMCON (FB)	UBS1	UBS0	HV	– [1)]	FCB3	FCB2	FCB1	FCB0

Figure 49. FEEPROM control register.

NOTE:
1. Reserved for future use; a write operation must write “0” to the location.

Table 45. Description of FMCON bits

UBS1	**UBS0**	**User - Boot selection bits**
0	0	User memory mapped from 0 to 64 K.
0	1	User memory mapped from 0 to 63 K. Boot ROM mapped from 63 K to 64 K.
1	0	User memory mapped from 0 to 63 K, but UBS1 bit cleared by hardware in this user address range. Boot ROM mapped from 63 K to 64 K. User software should not write “1” UBS1.
1	1	Boot ROM mapped from 0 to 64 K. User software should not write “1” UBS1.

HV	**High voltage indication bit.** Read only. Is “1” as long as the high voltage for an erase or write operation is present.

FCB3	**FCB2**	**FCB1**	**FCB0**	**Function Code Bits**
0	0	0	0	Value after Reset.
0	1	0	1	Byte Write or byte read (verify)
1	1	0	0	Page Erase (32 bytes boundaries).
0	0	1	1	Block Erase (256 bytes boundaries).
1	0	1	0	Full Erase (32 Kbytes).

The four FCB bits are write protected if the security feature is activated. Then only instructions in the internal program memory (FEEPROM) are able to write FCB (3–0), boot ROM and external program memory instructions cannot change FCB (3–0) except the full erase code can be loaded.

The duration of a write or erase operation is determined by the FEEPROM timer. This timer includes a zero point RC oscillator and cannot be controlled by software.

For calling a user routine in the boot ROM first all interrupts must be disabled and the DPTR and A have to be loaded with the desired values. After setting UBS0 = 1 and UBS1 = 0 and selecting the function via FCB-bits the respective user routine has to be called.

The table below lists the boot ROM user routines, which can be called by the user program. The content of FMCON, A and DPTR before the call is described by “(IN)” and the contents after the return is described by “(OUT)”. The boot ROM user routines do not change other registers or Data memory.

BOOT-ROM ROUTINE	**CALL ADDRESS**	**FMCON (IN)**	**FMCON (OUT)**	**ACC (IN)**	**ACC (OUT)**	**DPTR (IN)**	**DPTR (OUT)**
BYTE_READ	FFBAH	45H	15H	XXH	BYTE	BYTE ADDRESS	BYTE ADDRESS
BYTE_WRITE	FFADH	45H	15H	BYTE	BYTE [(V)]	BYTE ADDRESS	BYTE ADDRESS
PAGE_ERASE	FFAAH	4CH	1CH	XXH	08H	PAGE ADDRESS [1)]	PAGE ADDRESS [2)]
BLOCK_ERASE	FFA5H	43H	13H	XXH	02H	BLOCK ADDRESS [3)]	BLOCK ADDRESS [4)]
FULL_ERASE	FFA0H	4AH	1AH	XXH	0AH	XXXXH	0018H

X = don’t care or not defined
V = verified byte (read back)
1) = 5 LSB’s of DPTR are don’t care
2) = 5 LSB’s of DPTR are “0”
3) = 8 LSB’s of DPTR are don’t care
4) = 8 LSB’s of DPTR contain 08H.

Example of user software (internal or external) that calls the **Page Erase** routine in the boot ROM to erase a page in the FEEPROM (32 bytes) starting at address location 1260H.

```
CLR EA                  ; Disable all interrupts
MOV DPTR, # 1260H       ; Load page-address
MOV FMCON, # 4CH        ; Load Page-Erase code
LCALL 0FFAAH            ; Call Page-Erase routine
                        ; in boot ROM (inherent delay
                          5 ms)
MOV FMCON, #00H         ; Clear FMCON for security
SETB EA                 ; Enable interrupts again
```

Example of user software (internal or external) that calls the **Byte-Write** routine in the boot ROM to write the content of R5 into the FEEPROM address location 1263H.

```
CLR EA                  ; Disable all interrupts
MOV DPTR, # 1263H       ; Load byte address
MOV A, R5               ; Load byte to be written
MOV FMCON, # 45H        ; Load byte-write code
LCALL 0FFADH            ; Call byte-write routine
                        ; in boot ROM (inherent
                          delay 2.5 ms)
MOV FMCON, #00H         ; Clear FMCON for security
SETB EA                 ; Enable interrupts again
XRL A, R5               ; Compare the “read-back” byte
JNZ ERROR               ; Jump if verify error
```

8.4 Security

The security feature protects against software piracy and prevents that the content of the FEEPROM can be read undesirable. The Security Byte is located in the highest address location 7FFFH of the FEEPROM.

The Security Byte should be 50H to activate and 00H or FFH to deactivate the security feature. This security code is chosen in such a way that single bit failures will not deactivate the security feature.

If the security feature is deactivated, then there are no access restrictions to the FEEPROM.

If the security feature is activated, then the external program memory has no access to the FEEPROM with the MOVC instructions. Also bits FCB (3–0) of FMCON cannot be written by external program code or boot ROM code. This prevents in-circuit programming and verification. Only the Full Erase code can be written to FCB (0–3) of FMCON. Note that for the internal program code no restrictions exist if the security feature is activated. At the end of a full erase operation the security feature is deactivated. Also parallel programming and verify is inhibited if the security feature is activated, only a full erase is possible. Note that the security mode does not change immediately when the security code is written into the security byte 7FFFH, but after a reset or power-on. This allows the verification of the loaded code in the FEEPROM, including the Security Byte.

8.5 Parallel Programming

Unlike standard EPROM programming, no high programming supply voltage must be applied to the $\overline{EA}$ pin and only one programming pulse must be applied to the ALE/$\overline{WE}$ pin. The parallel programming mode is entered with the steady signals RST=1, PSEN=0, $\overline{EA}$=1 and SELXTAL1 = 1. The XTAL1,2 clock must have a frequency between 4 and 6MHz. The following table shows the logic levels for programming, erasing, verifying and read signature.

MODE	ALE/$\overline{WE}$	P2.7	P2.6	P3.7	P3.6
Full erase	⊓	1	1	0	1
Program FEEPROM	⊓	1	0	1	1
Verify FEEPROM	1	0	0	1	1
Read signature	1	0	0	0	0

ALE/$\overline{WE}$ Write Enable signal (program/erase), active low
P2.6, P2.7, P3.6, P3.7 control signals

Data and address bits:

P0.0 – P0.7	:	D0 – D7	Program data input / verify or read data output
P1.0 – P1.7	:	A0 – A7	Input low order address bits.
P2.0 – P2.5, P3.4	:	A8 – A14	Input high order address bits.

The P89C557E4 contains two signature bytes that can be read and used by an EPROM programming system to identify the device. These bytes are read by the same procedure as for a normal verification of locations 30H and 31H, except that P3.6 and P3.7 need to be pulled to LOW.

ADDRESS	CONTENT	MEANING
30H	15H	Philips
31H	B5H	P89C557E4

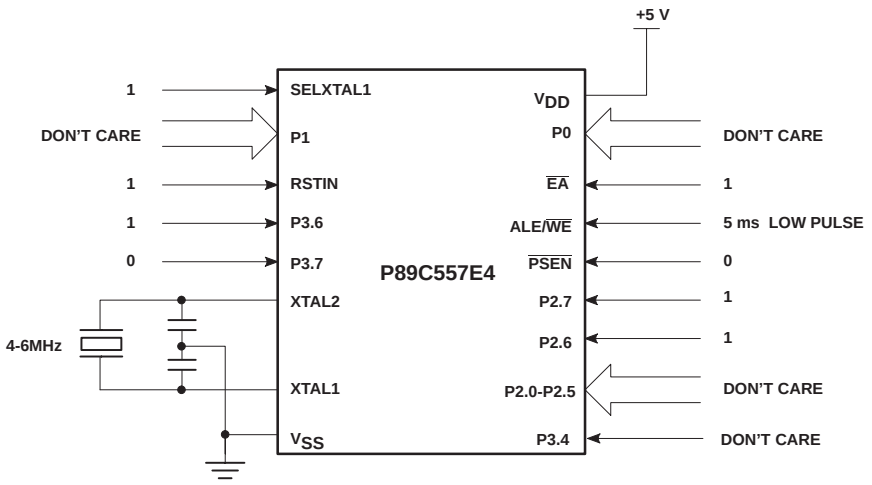


Figure 50. Erase Configuration

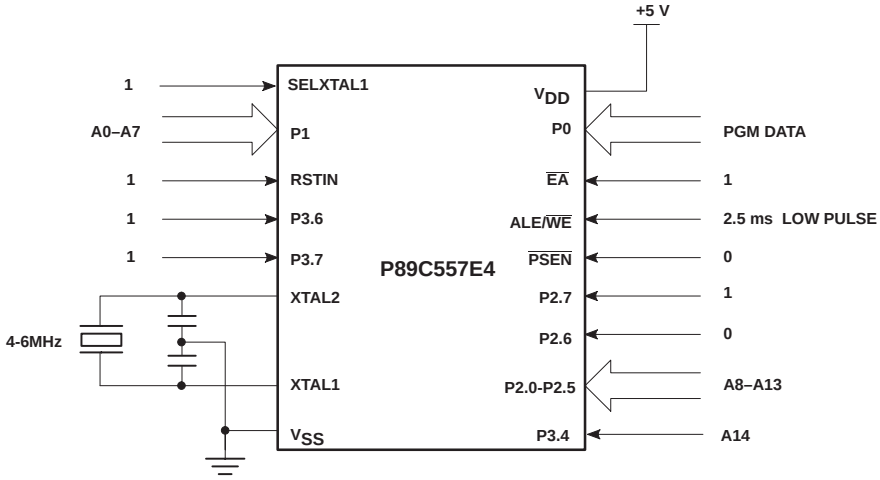


Figure 51. Programming Configuration

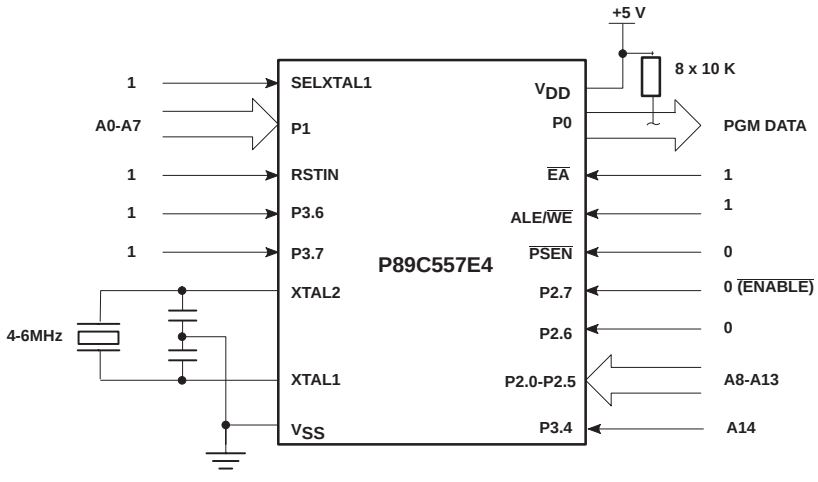
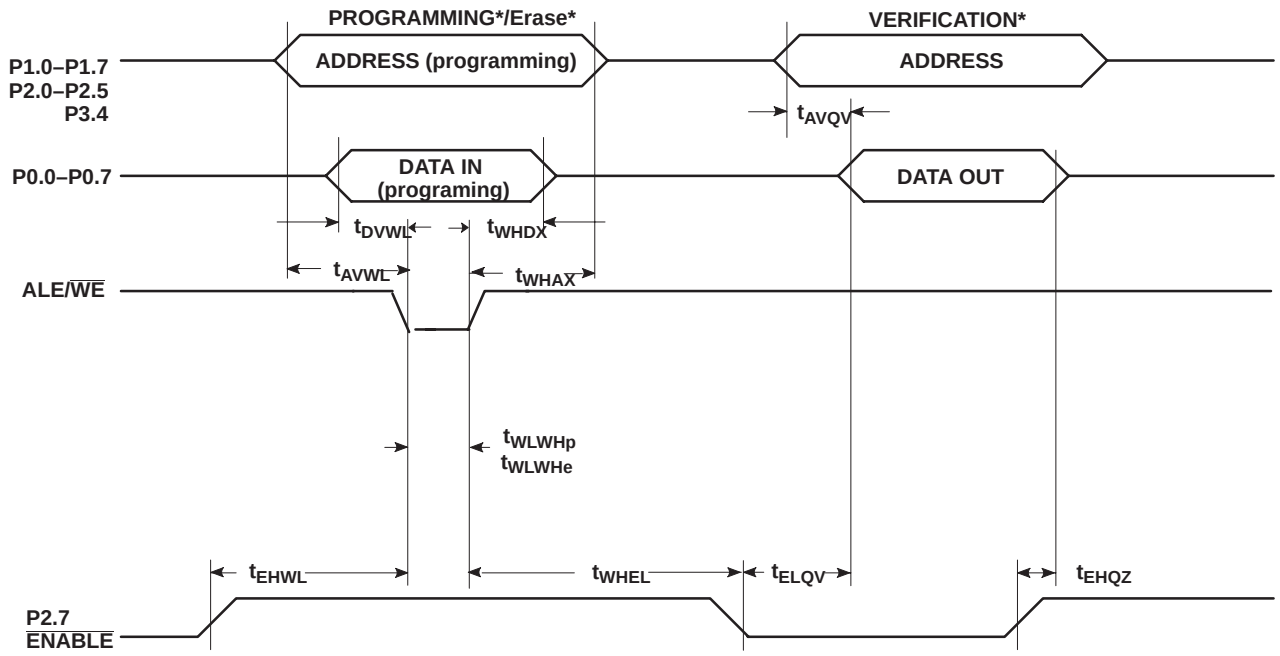


Figure 52. Program Verification

FEEPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

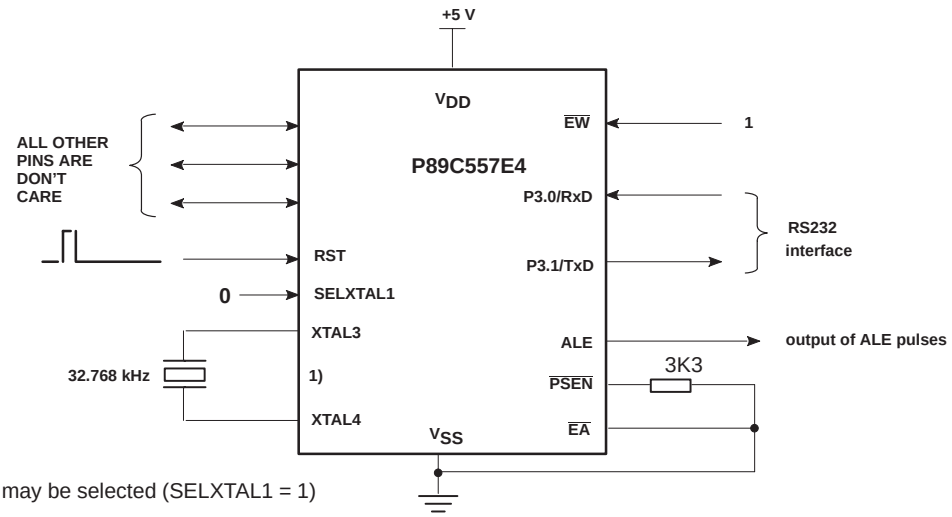
T_{amb} = –40 °C to +85 °C, V_{DD} = 5 V ± 10%, V_{SS} = 0 V (see Figure 53)

SYMBOL	PARAMETER	MIN	MAX	UNIT
$1/t_{CLK}$	System clock frequency (standard oscillator)	4	6	MHz
t_{AVWL}	Address setup to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHAX}	Address hold after $\overline{WE}$ HIGH	$48t_{CLK}$	–	
t_{DVWL}	Data setup to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHDX}	Data hold after $\overline{WE}$ HIGH	$48t_{CLK}$	–	
t_{EHWL}	P2.7 ($\overline{ENABLE}$) HIGH to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHEL}	$\overline{WE}$ HIGH to P2.7 ($\overline{ENABLE}$) LOW	$48t_{CLK}$	–	
t_{WLWHp}	$\overline{WE}$ width (programming)	2.25	2.75	ms
t_{WLWHe}	$\overline{WE}$ width (erase)	4.5	5.5	ms
t_{AVQV}	Address to data valid	–	$48t_{CLK}$	
t_{ELQV}	P2.7 ($\overline{ENABLE}$) Low to data valid	–	$48t_{CLK}$	
t_{EHQZ}	Data float after P2.7 ($\overline{ENABLE}$) HIGH	0	$48t_{CLK}$	



* For ERASE conditions see Figure 50.
For PROGRAM conditions see Figure 51.
For VERIFY conditions see Figure 52.

Figure 53. FEEPROM Programming/Erase and Verification Waveforms



1) Alternative XTAL1, 2 may be selected (SELXTAL1 = 1)

Figure 54. Serial programming (boot mode) Configuration

8.6 Serial Programming of FEEPROM

Serial in-circuit programming (boot-mode) is entered if during and after RESET $\overline{PSEN}$ and $\overline{EA}$ are pulled down, $\overline{PSEN}$ via a resistor of 3.3 k Ohm to VSS. The two UBS bits are set to 1 by hardware and program execution starts at 0000H of the boot ROM. P3.0 (RXD) and P3.1 (TXD) form the serial RS232 interface. A baud rate of 4800 or 9600 Baud is possible, if the PLL oscillator is selected. The receive and transmit channel have the same baudrate. The format is: Startbit, 8 data bits (last bit always 0), no parity bit and at least one stopbit. The boot routine inputs the Intel Hex Object Format. The baud rate will be selected automatically after reception of the first character (:) of the object file. No other characters are allowed to preceed the first (:) character. Programming is only started if the first received record has the right type indication (TT). If the security feature is activated (contents of the security byte = 50H) then the programming starts with a Full Erase, otherwise only the addressed page(s) will be erased and the not altered bytes are rewritten. During the erase or write operation the next string of bytes can be received. Xon and Xoff handshake codes are used to control the serial transfer. At the end of the programming a message that indicates a successful or not successful programming, will be returned over the RS232 interface channel. If the programming was successful then the user program can be started up at 0000H in FEEPROM by a reset for user mode ($\overline{EA}$ = high, $\overline{PSEN}$ not affected). If the programming was not successful the boot program halts and a retry can be started by a reset for the boot mode.

8.7 Boot Routine

The boot routine transmits the next “one ASCII character” messages via the RS232 interface:

“ . ”	After each record type TT = 00H indication in the HEX file.
“ X ”	Checksum error of a record in the HEX file detected.
“ Y ”	Wrong record type received
“ Z ”	Buffer overflow error (Check Xon/Xoff of terminal)
“ R ”	Verification error (of last written byte)
“ V ”	End record received and programming of FEEPROM was successful

No messages are transmitted if the baud rate of the first character (:) can not be detected.

The boot routine can also be started by the internal or external user program (LJMP FC07H). FMCON must be loaded previously with 40H. Interrupt registers, stack pointer, Timer 0, UART, P3.0 and P3.1 must be in the reset state. $\overline{EA}$ and $\overline{PSEN}$ must not be affected. A reset is needed to restart the user program after programming.

The following baudrates will be detected automatically within the specified µC clock range in MHz.

Baudrate	f_{CLK} (min)	f_{CLK} (max)
1200	1 [1)]	3.6
2400	2 [1)]	7.3
4800	4	14.7
9600	7.9	29.5 [1)]
19200	15.7	59 [1)]

NOTE:

1. Value outside the specified clock range

Note that the boot routines can (re) program any number of bytes from 1 byte to 32 Kbytes, independent in which order or at which location, but if the security feature is activated, a full erase is performed and all not programmed bytes become FFH.

Definitions:

: – Record start character

BC – Byte Count. The hexadecimal number of data bytes in the record. This may theoretically be any number from 0 to 255, although many assemblers prefer to deal with 16 data bytes per record (as shown in the example below).

AAAA – Load address in hexadecimal of first data byte in this record.

TT – Record type. The record type is 00 for data records and 01 for the end record.

HH – One hexadecimal data byte.

CC – Record checksum. This is the 2's complement of the summation of all of the bytes in the record from the byte count through the last data byte. While the summation is calculated, it is always truncated to a one byte result. Thus, if all of the bytes in the record are summed, including the checksum itself, the result will always be 00 if the record is valid.

Construction of data records (using the notation defined above, each letter corresponds to one hexadecimal digit in ASCII representation) is as follows:

: BCAAAATTHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHCC

The last record in a file is the end record and contains no data. Usually the end record will appear as shown in the first example below. However, in some cases a 16 bit checksum of all of the data bytes in the entire file may be inserted in the address field of the end record. This checksum would correspond to one generated by an EPROM programmer during file load, and its inclusion does not violate the rules for this format. This is shown in the second example.

:00000001FF

:00B12C0122

Successive hex records need not appear in sequential address order . For instance, a record for address 0000H might appear after a record for address 7FE0H. All of the bytes in a single record, however, must be in sequence. Any characters that appear outside of a record (i.e. after a checksum, but before the next ":") will be ignored, if present.

An example of a valid hex file follows:

:10010000C2F0E53030E704F404D2F08531F030F786

:100110000763F0FF05F0B2F0A430F00A63F0FFF4DB

:0C0120002401500205F085F032F5332276

:00000001FF

9. ABSOLUTE MAXIMUM RATINGS

ABSOLUTE MAXIMUM RATINGS [1, 2, 3]

PARAMETER	RATING	UNIT
Storage temperature range	–65 to +150	°C
Voltage on V_{DD} to V_{SS} and SCL, SDA to V_{SS}	–0.5 to +6.5	V
Input / output current on any I/O pin	10	mA
Power dissipation (based on package heat transfer limitations, not device power consumption)	1.0	W

NOTES:
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions are taken to avoid applying greater than the rated maxima.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to V_{SS} unless otherwise noted.

10. DC CHARACTERISTICS

DC ELECTRICAL CHARACTERISTICS

V_{DD} = 5V (± 10%), V_{SS} = 0V, T_{amb} = 0°C to +70°C (P8xC557E4EBx). All voltages with respect to V_{SS} unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			MIN	MAX	
V_{DD}	Supply voltage		4.5	5.5	V
I_{DD}	Supply current operating : P89C557E4 P83C557E4	See notes 1 and 2 f_{CLK} = 16MHz V_{DD} = 5.5 V		 40 40	 mA mA
I_{ID}	Supply current Idle Mode : P89C557E4 P83C557E4	See notes 1 and 3 f_{CLK} = 16MHz V_{DD} = 5.5 V		 15 12	 mA mA
I_{PD}	Supply current Power-down mode	See note 4 2 V < V_{PD} < V_{DDmax}		100	µA
	Supply current Power-down mode: 32 kHz / PLL operation	See note 17 V_{DD} = 5.5 V		100	µA
Inputs					
V_{IL}	Input LOW voltage, except $\overline{EA}$, SCL, SDA		–0.5	$0.2V_{DD}$–0.1	V
V_{IL1}	Input LOW voltage to $\overline{EA}$		–0.5	$0.2V_{DD}$–0.3	V
V_{IL2}	Input LOW voltage to SCL, SDA [5]		–0.5	$0.3V_{DD}$	V
V_{IH}	Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS		$0.2V_{DD}$+0.9	V_{DD}+0.5	V
V_{IH1}	Input HIGH voltage, XTAL1, RSTIN, ADEXS		$0.7V_{DD}$	V_{DD}+0.5	V
V_{IH2}	Input HIGH voltage, SCL, SDA [5]		$0.7V_{DD}$	6.0	V
I_{IL}	Input current LOW level, Ports 1, 2, 3, 4	V_{IN} = 0.45 V		–50	µA
I_{TL}	Transition current HIGH to LOW, Ports 1, 2, 3, 4	See note 6		–650	µA
$\pm I_{LI1}$	Input leakage current, Port 0, $\overline{EA}$, ADEXS, $\overline{EW}$, SELXTAL1	0.45 V < V_I < V_{DD}		10	µA
$\pm I_{LI2}$	Input leakage current, SCL, SDA	0 V < V_I < 6 V 0 V < V_{DD} < 5.5 V		10	µA
$\pm I_{LI3}$	Input leakage current, Port 5	0.45 V < V_I < V_{DD}		1	µA
Outputs					
V_{OL}	Output low voltage, Ports 1, 2, 3, 4	I_{OL} = 1.6mA[7]		0.45	V
V_{OL1}	Output low voltage, Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT	I_{OL} = 3.2mA[7]		0.45	V
V_{OL2}	Output low voltage, SCL, SDA	I_{OL} = 3.0mA[7, 19]		0.4	V
		I_{OL} = 6.0mA[7, 19]		0.6	
V_{OH}	Output high voltage, Ports 1, 2, 3, 4	V_{DD} = 5 V ± 10% $-I_{OH}$ = 60µA $-I_{OH}$ = 25µA $-I_{OH}$ = 10µA	 2.4 $0.75V_{DD}$ $0.9V_{DD}$		 V V V
V_{OH1}	Output high voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$, RSTOUT) [8]	V_{DD} = 5 V ± 10% $-I_{OH}$ = 800µA $-I_{OH}$ = 300µA $-I_{OH}$ = 80µA	 2.4 $0.75V_{DD}$ $0.9V_{DD}$		 V V V
V_{HYS}	Hysteresis of Schmitt Trigger inputs SCL, SDA (Fast-mode)		$0.05V_{DD}$ [20]		V

NOTES: See Page 62.

DC ELECTRICAL CHARACTERISTICS (Continued)

V_{DD} = 5 V (± 10%), V_{SS} = 0 V, T_{amb} = –40°C to +85°C (P8xC557E4EFx).
DC parameters not included here are the same as in the P8xC557E4EBx, DC electrical characteristics
All voltages with respect to V_{SS} unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS MIN	LIMITS MAX	UNIT
R_{RST}	Internal reset pull-down resistor		50	150	kΩ
C_{IO}	Pin capacitance	Test freq = 1MHz, T_{amb} = 25 °C		10	pF
Inputs					
V_{IL}	Input LOW voltage, except $\overline{EA}$, SCL, SDA		–0.5	$0.2V_{DD}$–0.15	V
V_{IL1}	Input LOW voltage to $\overline{EA}$		–0.5	$0.2V_{DD}$–0.35	V
V_{IH}	Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS		$0.2V_{DD}$+1.0	V_{DD}+0.5	V
V_{IH1}	Input HIGH voltage, XTAL1, RSTIN, ADEXS		$0.7V_{DD}$+0.1	V_{DD}+0.5	V
I_{IL}	Input current LOW level, Ports 1, 2, 3, 4	V_{IN} = 0.45 V		–75	µA
I_{TL}	Transition current HIGH to LOW, Ports 1, 2, 3, 4	See note 6		–750	µA

NOTES: See Page 62.

DC ELECTRICAL CHARACTERISTICS ANALOG

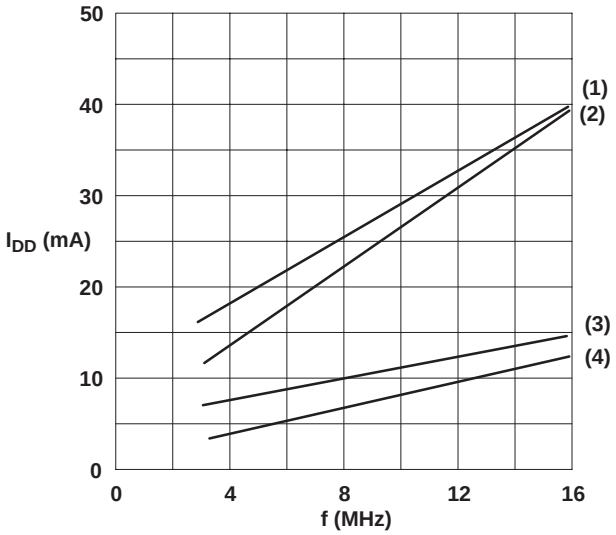
AV_{DD} = 5 V (± 10%), AV_{SS} = 0 V, Tamb = 0 °C to +70 °C (P8xC557E4EBx).
AV_{DD} = 5 V (± 10%), AV_{SS} = 0 V, Tamb = –40 °C to +85 °C (P8xC557E4EFx).
All voltages with respect to V_{SS} unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS MIN	LIMITS MAX	UNIT
AV_{DD}	Analog supply voltage	$AV_{DD} = V_{DD} \pm 0.2$ V	4.5	5.5	V
AI_{DD}	Analog supply current operating	Port 5 = 0 to AV_{DD} see notes 1 and 2		1.2	mA
	Analog supply current operating: 32 kHz/PLL operation	Port 5 = 0 to AV_{DD} see note 17, 18		7.2	mA
AI_{ID}	Analog supply current Idle Mode	see notes 1 and 3		70	µA
	Analog supply current Idle Mode: 32 kHz/PLL operation	see note 17		6.0	mA
AI_{PD}	Supply current Power-down mode	2 V < V_{PD} < V_{DDmax} see note 4		50	µA
	Supply current Power-down mode: 32 kHz / PLL operation	V_{DD} = 5.5V see note 17		200	µA
Analog Inputs					
AV_{IN}	Analog input voltage		AV_{SS}–0.2	AV_{DD}+0.2	V
AV_{REF}	Reference voltage: AV_{REF-}		AV_{SS}–0.2		V
	AV_{REF+}			AV_{DD}+0.2	V
R_{REF}	Resistance between AV_{REF+} and AV_{REF-}		10	50	kΩ
C_{IA}	Analog input capacitance			15	pF
DL_e	Differential non-linearity [9, 10, 11]			±1	LSB
IL_e	Integral non-linearity [9, 12]			±2	LSB
OS_e	Offset error [9, 13]			±2	LSB
G_e	Gain error [9, 14]			±0.4	%
A_e	Absolute voltage error [9, 15]			±3	LSB
M_{CTC}	Channel to channel matching			±1	LSB
C_t	Crosstalk between inputs of port 5 [16]	0–100kHz		–60	dB

NOTES: See Page 62.

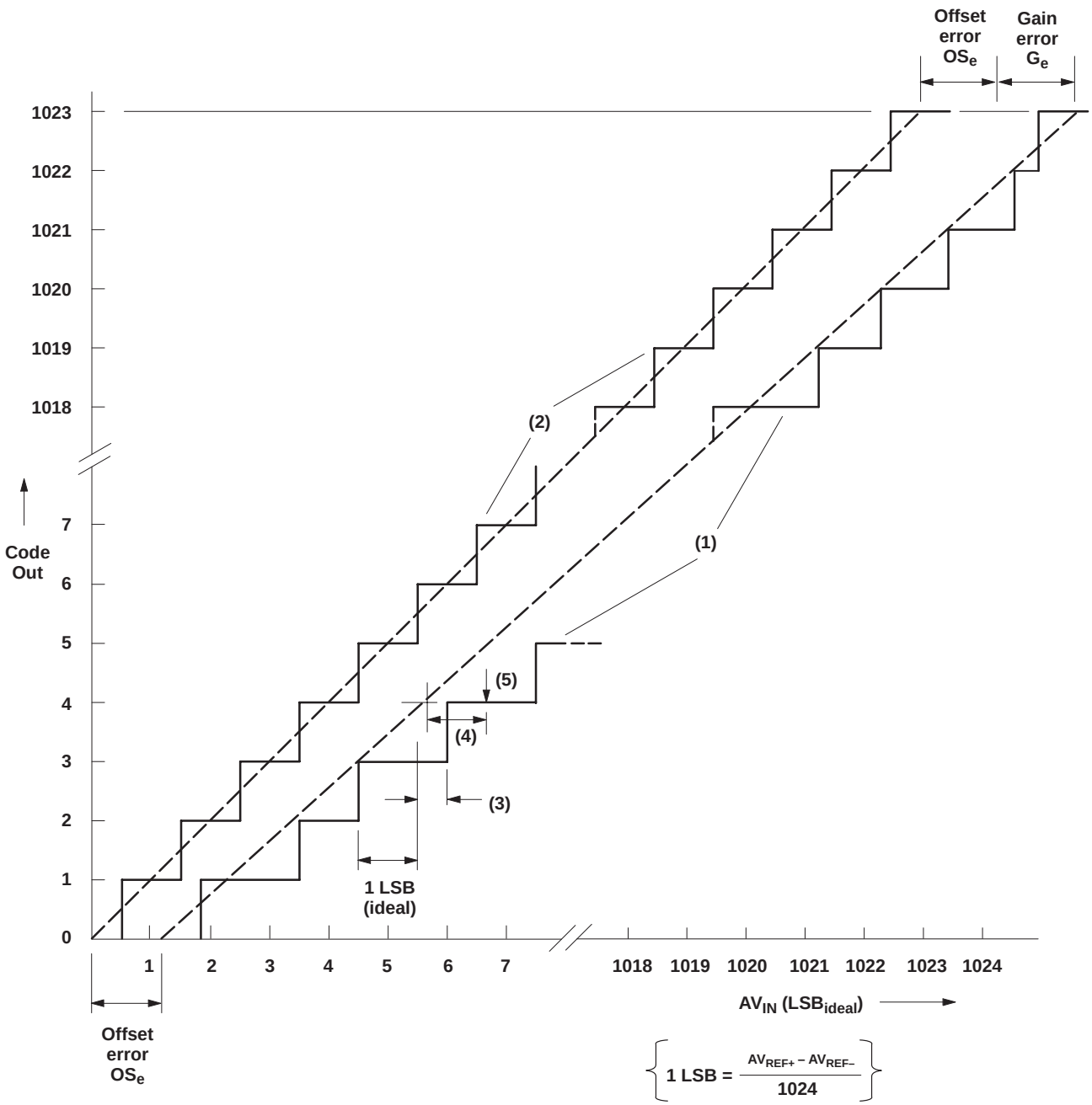
NOTES FOR DC ELECTRICAL CHARACTERISTICS:

1. See Figures 55 and 57 through 59 for I_{DD} test conditions.
2. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; $\overline{EA}$ = RSTIN = Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL1 = V_{DD}; ADEXS = XTAL4 = V_{SS}.
3. The Idle Mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2, XTAL3 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = V_{DD}; $\overline{EA}$ = RSTIN = ADEXS = XTAL4 = V_{SS}.
4. The Power-down current is measured with all output pins disconnected; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = SELXTAL 1 = V_{DD}; $\overline{EA}$ = RSTIN = ADEXS = XTAL1 = XTAL4 = V_{SS}.
5. The input threshold voltage of SCL and SDA (SIO1) meets the I^2C specification, so an input voltage below 0.3 V_{DD} will be recognized as a logic 0 while an input voltage above 0.7 V_{DD} will be recognized as a logic 1.
6. Pins of ports 1, 2, 3, and 4 source a transition current when they are being externally driven from HIGH to LOW. The transition current reaches its maximum value when V_{IN} is approximately 2 V.
7. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the V_{OL} of ALE and ports 1, 3 and 4. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE pin may exceed 0.8V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
8. Capacitive loading on ports 0 and 2 may cause the V_{OH} on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
9. Conditions: $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V, $AV_{REF+} = 5.12$ V. $V_{DD} = 5.0$ V, $V_{SS} = 0$ V, ADC is monotonic with no missing codes. Measurement by continuous conversion of $AV_{IN} = -20$mV to 5.12 V in steps of 0.5mV, derivating parameters from collected conversion results of ADC. ADC prescaler programmed according to the actual oscillator frequency, resulting in a conversion time within the specified range for t_{conv} (15µs ... 50µs).
10. The differential non-linearity (DL_e) is the difference between the actual step width and the ideal step width.
11. The ADC is monotonic; there are no missing codes.
12. The integral non-linearity (IL_e) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
13. The offset error (OS_e) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
14. The gain error (G_e) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve.
15. The absolute voltage error (A_e) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
16. This should be considered when both analog and digital signals are simultaneously input to port 5.
17. The supply current with 32 kHz oscillator running and PLL operation (SELXTAL1 = 0) is measured with all output pins disconnected; XTAL4 driven with $t_r = t_f = 5ns$; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2 not connected; Port 0 = $\overline{EW}$ = SCL = SDA = V_{DD}; $\overline{EA}$ = RSTIN = ADEXS = SELXTAL 1 = XTAL1 = V_{SS}.
18. Not 100% tested; sum of A_{IID} (PLL) and A_{IDD} (HF-Oscillator).
19. The parameter meets the I^2C bus specification for standard-mode and fast-mode devices.
20. Not 100% tested.



(1) Maximum operating mode P89C557E4 : V_{DD} = 5.5 V
(2) Maximum operating mode P83C557E4/P80C557E4 : V_{DD} = 5.5 V
(3) Maximum Idle Mode P89C557E4 : V_{DD} = 5.5 V
(4) Maximum Idle Mode P83C557E4/P80C557E4 : V_{DD} = 5.5 V

Figure 55. Supply Current (I_{DD}) as a Function of Frequency at XTAL1



(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity (DL_e).
(4) Integral non-linearity (IL_e).
(5) Center of a step of the actual transfer curve.

Figure 56. ADC Conversion Characteristic

11. AC CHARACTERISTICS

AC ELECTRICAL CHARACTERISTICS

V_{DD} = 5 V ± 10% (EBx), V_{SS} = 0 V, t_{CLK} min = 1/fmax (maximum operating frequency)
V_{DD} = 5 V ± 10% (EFx), V_{SS} = 0 V, t_{CLK} min = 1/fmax (maximum operating frequency)
T_{amb} = 0 °C to +70 °C, t_{CLK} min = 63 ns for P8xC557E4EBx
T_{amb} = –40 °C to +85 °C, t_{CLK} min = 63 ns for P8xC557E4EFx
C1 = 100 pF for Port 0, ALE and $\overline{PSEN}$; C1 = 80 pF for all other outputs unless otherwise specified.

SYMBOL	FIGURE	PARAMETER	12MHz CLOCK		16MHz CLOCK		VARIABLE CLOCK		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
$1/t_{CLK}$	60	System clock frequency					3.5	16	MHz
t_{LHLL}	60	ALE pulse width	127		85		$2t_{CLK}$–40		ns
t_{AVLL}	60	Address valid to ALE LOW	43		23		t_{CLK}–40		ns
t_{LLAX}	60	Address hold after ALE LOW	53		33		t_{CLK}–30		ns
t_{LLIV}	60	ALE LOW to valid instruction in		234		150		$4t_{CLK}$–100	ns
t_{LLPL}	60	ALE LOW to $\overline{PSEN}$ LOW	53		33		t_{CLK}–30		ns
t_{PLPH}	60	$\overline{PSEN}$ pulse width	205		143		$3t_{CLK}$–45		ns
t_{PLIV}	60	$\overline{PSEN}$ LOW to valid instruction in		145		83		$3t_{CLK}$–105	ns
t_{PXIX}	60	Input instruction hold after $\overline{PSEN}$	0		0		0		ns
t_{PXIZ}	60	Input instruction float after $\overline{PSEN}$		59		38		t_{CLK}–25	ns
t_{AVIV}	60	Address to valid instruction in		312		208		$5t_{CLK}$–105	ns
t_{PLAZ}	60	$\overline{PSEN}$ LOW to address float		10		10		10	ns
Data Memory									
t_{AVLL}	61, 62	Address valid to ALE LOW	43		23		t_{CLK}–40		ns
t_{LLAX}	61, 62	Address hold after ALE LOW	48		28		t_{CLK}–35		ns
t_{RLRH}	61	$\overline{RD}$ pulse width	400		275		$6t_{CLK}$–100		ns
t_{WLWH}	62	$\overline{WR}$ pulse width	400		275		$6t_{CLK}$–100		ns
t_{RLDV}	61	$\overline{RD}$ LOW to valid data in		252		148		$5t_{CLK}$–165	ns
t_{RHDX}	61	Data hold after $\overline{RD}$	0		0		0		ns
t_{RHDZ}	61	Data float after $\overline{RD}$		97		55		$2t_{CLK}$–70	ns
t_{LLDV}	61	ALE LOW to valid data in		517		350		$8t_{CLK}$–150	ns
t_{AVDV}	61	Address to valid data in		585		398		$9t_{CLK}$–165	ns
t_{LLWL}	61, 62	ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW	200	300	138	238	$3t_{CLK}$–50	$3t_{CLK}$+50	ns
t_{AVWL}	61, 62	Address valid to $\overline{WR}$ LOW or $\overline{RD}$ LOW	203		120		$4t_{CLK}$–130		ns
t_{QVWX}	62	Data valid to $\overline{WR}$ transition	33		13		t_{CLK}–50		ns
t_{QVWH}	62	Data before $\overline{WR}$	433		288		$7t_{CLK}$–150		ns
t_{WHQX}	62	Data hold after $\overline{WR}$	33		13		t_{CLK}–50		ns
t_{RLAZ}	61	$\overline{RD}$ low to address float		0		0		0	ns
t_{WHLH}	61, 62	$\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	43	123	23	103	t_{CLK}–40	t_{CLK}+40	ns
UART Timing – Shift Register Mode (Test Conditions: T_{amb} = 0 °C to +70 °C; V_{SS} = 0 V; Load Capacitance = 80pF)									
t_{XLXL}	64	Serial port clock cycle time	1.0		0.75		$12t_{CLK}$		µs
t_{QVXH}	64	Output data setup to clock rising edge	700		492		$10t_{CLK}$–133		ns
t_{XHQX}	64	Output data hold after clock rising edge	50		8		$2t_{CLK}$–117		ns
t_{XHDX}	64	Input data hold after clock rising edge	0		0		0		ns
t_{XHDV}	64	Clock rising edge to input data valid		700		492		$10t_{CLK}$–133	ns

AC ELECTRICAL CHARACTERISTICS (Continued)

SYMBOL	PARAMETER	Standard-mode I²C-bus		Fast-mode I²C-bus		UNIT
		MIN	MAX	MIN	MAX	
I²C Interface timing (refer to Figure 63)						
f_{SCL}	SCL clock frequency	0	100	0	400	kHz
t_{BUF}	Bus free time between a STOP and START condition	4.7	–	1.3	–	µs
$t_{HD; STA}$	Hold time (repeated) START condition. After this period, the first clock pulse is generated	4.0	–	0.6	–	µs
t_{LOW}	LOW period of the SCL clock	4.7	–	1.3	–	µs
t_{HIGH}	High period of the SCL clock	4.0	–	0.6	–	µs
$t_{SU; STA}$	Set-up time for a repeated START condition	4.7	–	0.6	–	µs
$t_{HD; DAT}$	Data hold time: for CBUS competible masters (see Section 9, Notes 1, 3) for I²C-bus devices	 5.0 0 [1]		 – 0[1]	 – 0.9[2]	µs
$t_{SU; DAT}$	Data set-up time	250	–	100[3]	–	ns
t_{FD}, t_{FC}	Rise time of both SDA and SCL signals	–	1000	20 + 0.1C_b[4]	300	ns
t_{FD}, t_{FC}	Fall time of both SDA and SCL signals	–	300	20 + 0.1C_b[4]	300	ns
$t_{SU; STO}$	Set-up time for STOP condition	4.0	–	0.6	–	µs
C_b	Capacitive load for each bus line	–	400	–	400	pF
t_{SP}	Pulse width of spikes which must be suppressed by the input filter	–	–	0	50	ns

All values referred to V_{IH} and $V_{IL\ max}$ levels.

NOTES:

1. A device must internally provide a hold time of at least 300 ns from the SDA signal (referred to the $V_{IH\ min}$ of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
2. The maximum $t_{HD,DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
3. A fast-mode I²C-bus device can be used in a standard-mode I²C-bus system, but the requirement $t_{SU,DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{Rmax} + t_{SU,DAT} = 1000 + 250 = 1250$ ns (according to the standard-mode I²C-bus specification) before the SCL line is released.
4. C_b = total capacitance of one bus line in pF.

Table 46. External clock drive XTAL1 (refer to Figure 57)

SYMBOL	PARAMETER	VARIABLE CLOCK f_{CLK} = 3.5 to 16 MHz		UNIT
		MIN	MAX	
t_{CLK}	XTAL1 Period	63	286	ns
t_{CLKH}	XTAL1 HIGH time	20	–	ns
t_{CLKL}	XTAL1 LOW time	20	–	ns
t_{CLKR}	XTAL1 rise time	–	20	ns
t_{CLKF}	XTAL1 fall time	–	20	ns
t_{CYC} [1)]	Controller cycle time	0.75	3.4	µs

NOTE:

1. $t_{CYC} = 12\ f_{CLK}$

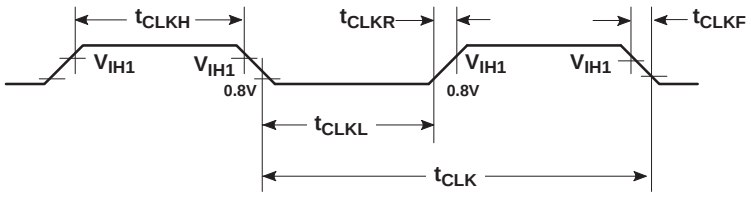
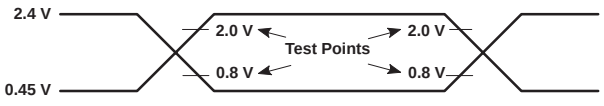
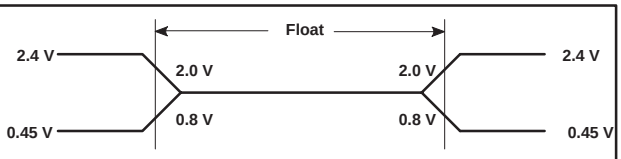


Figure 57. External Clock Drive waveform



NOTE:
AC inputs during testing are driven at 2.4V for a logic 'HIGH' and 0.45V for a logic 'LOW'. Timing measurements are made at 2.0 V for a logic 'HIGH' and 0.8 V for a logic 'LOW'.

Figure 58. AC Testing Input/Output



NOTE:
The float state is defined as the point at which a port 0 pins sinks 3.2 mA or sources 400μA at the voltage test levels.

Figure 59. AC Testing, Float Waveform

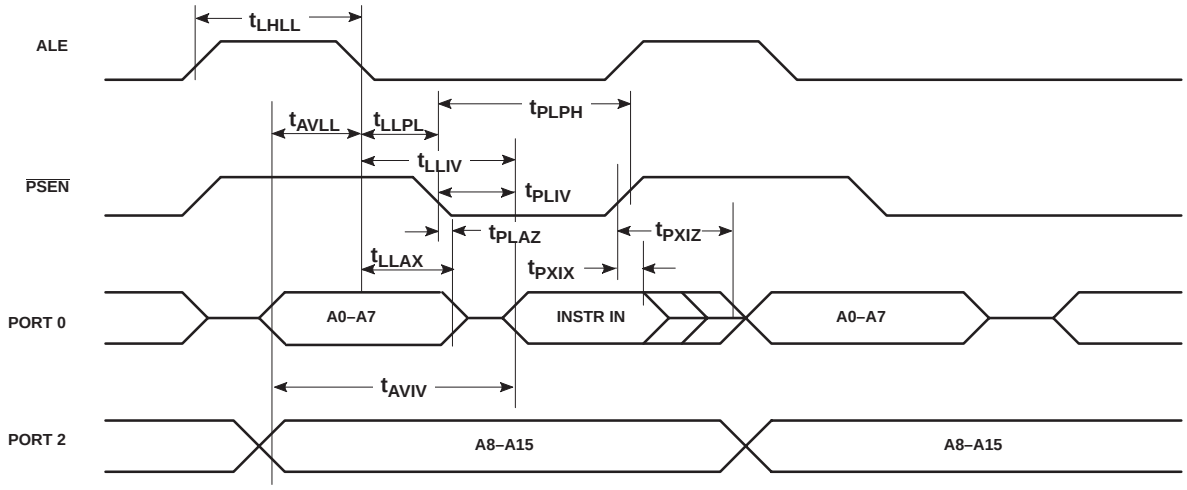


Figure 60. External Program Memory Read Cycle

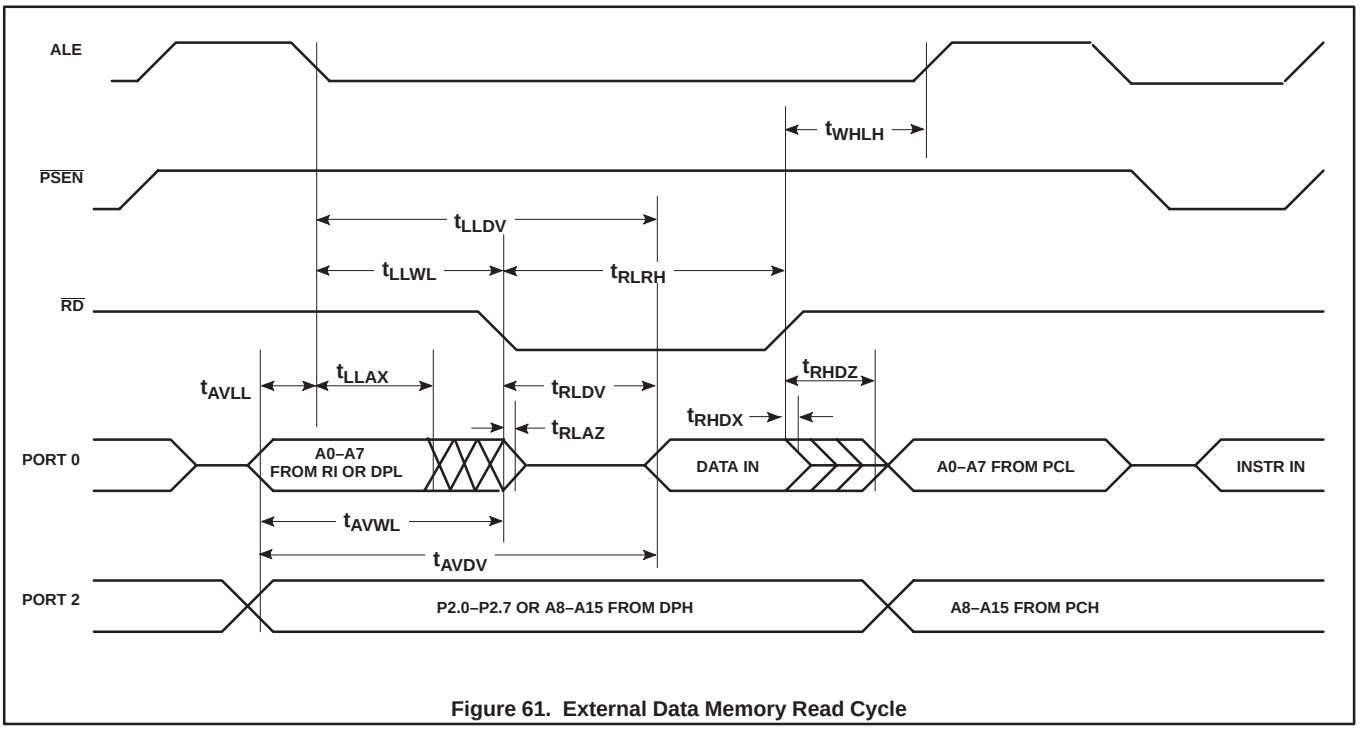


Figure 61. External Data Memory Read Cycle

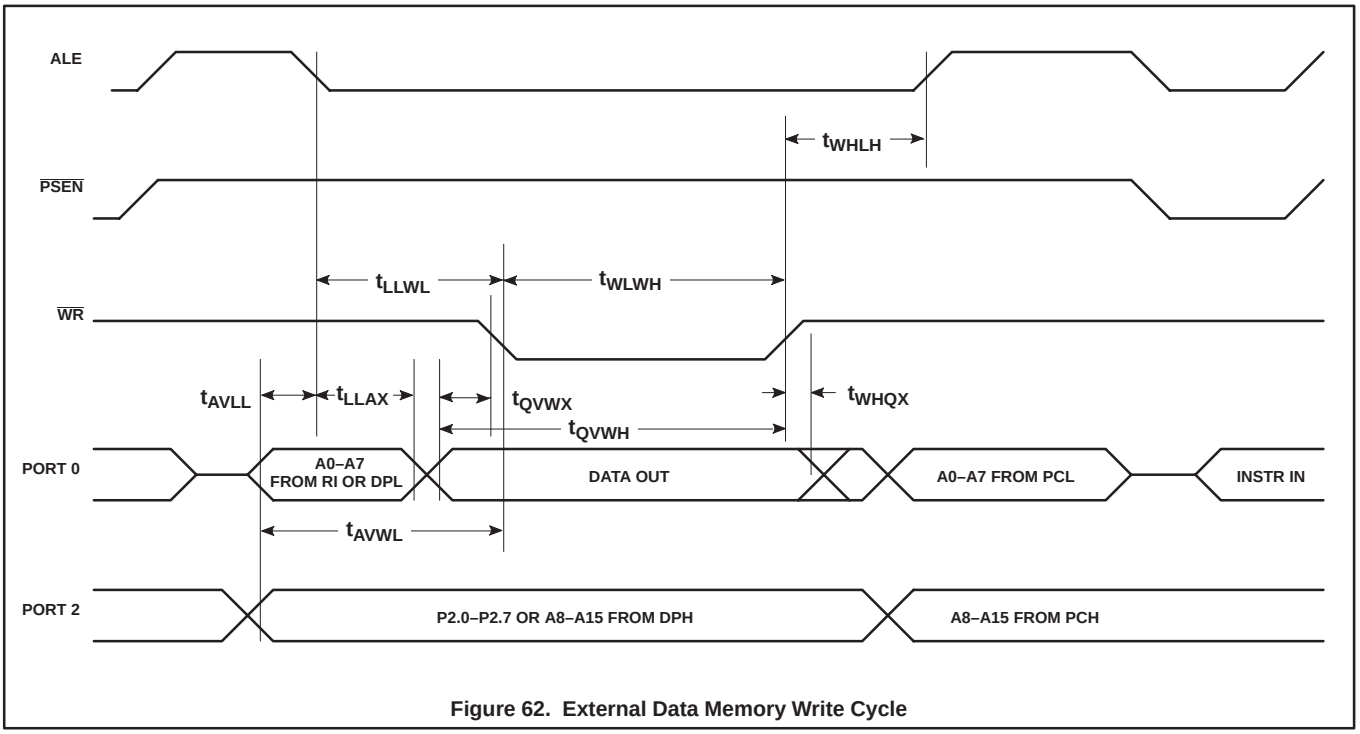


Figure 62. External Data Memory Write Cycle

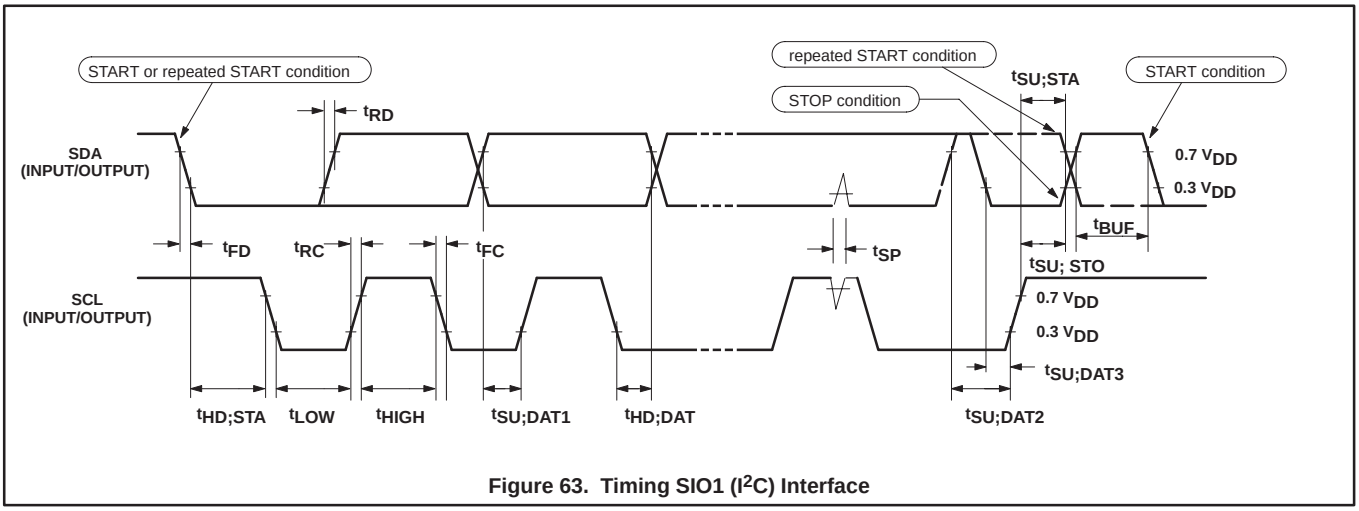


Figure 63. Timing SIO1 (I^2C) Interface

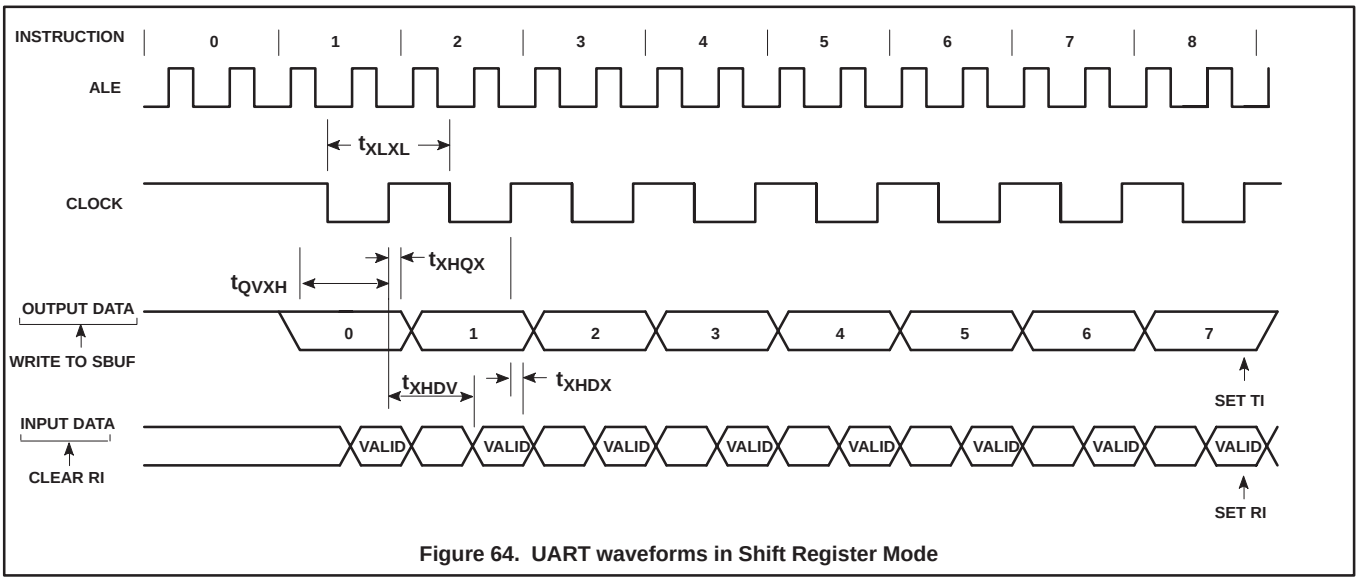


Figure 64. UART waveforms in Shift Register Mode

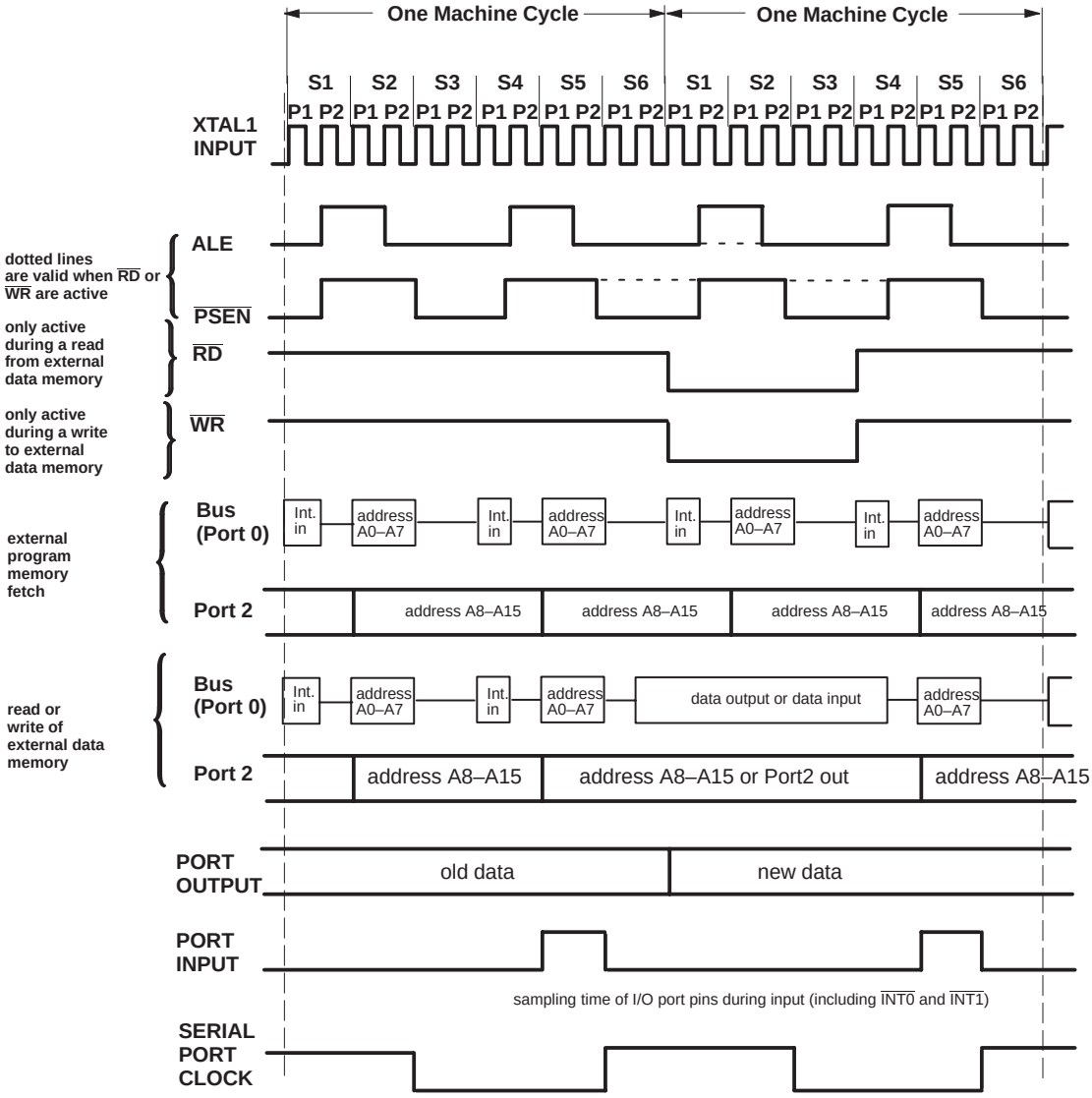
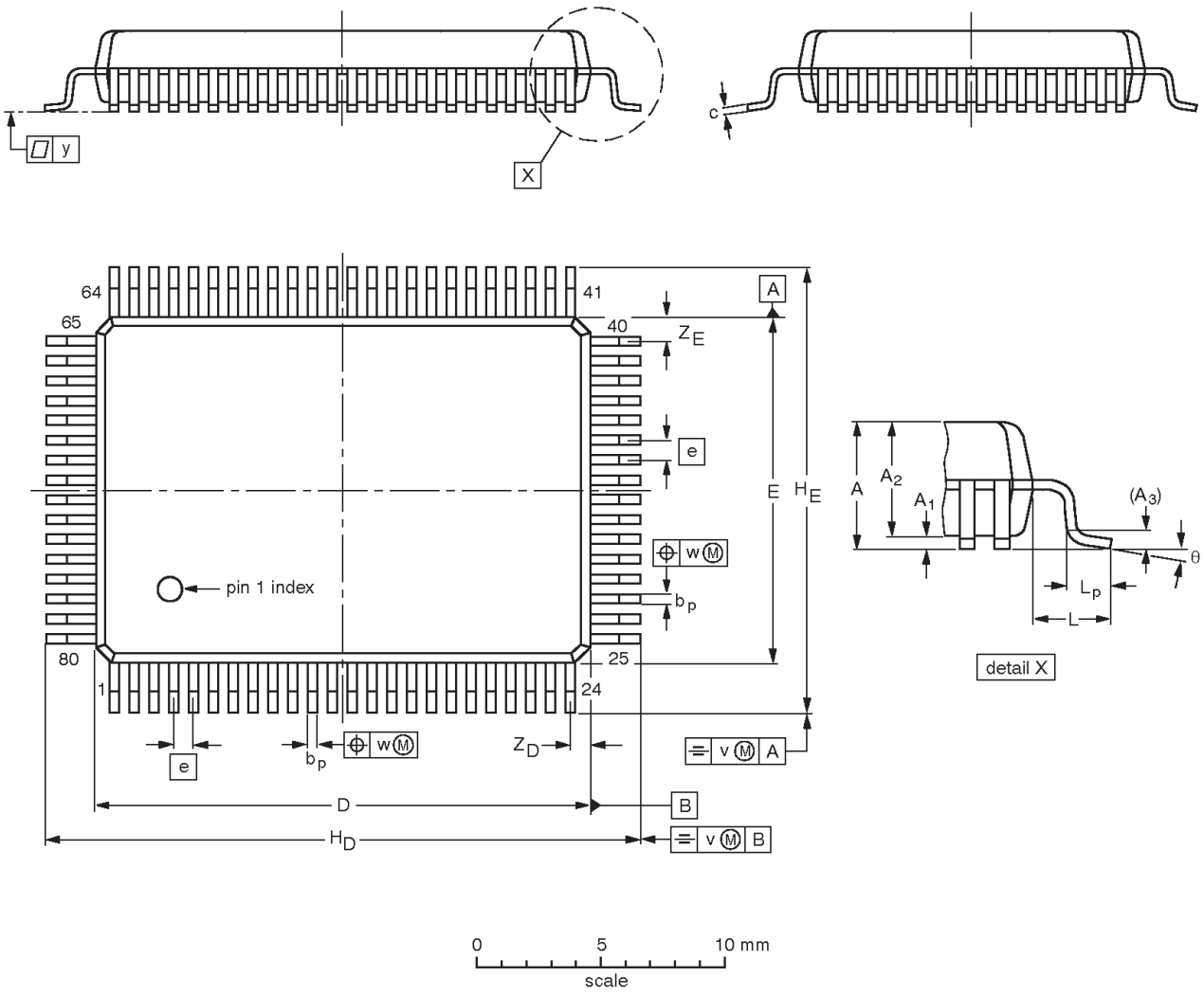


Figure 65. Instruction cycle timing

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specifications defined by Philips. This specification can be ordered using the code 9398 393 40011.

QFP80: plastic quad flat package;
80 leads (lead length 1.95 mm); body 14 x 20 x 2.7 mm; high stand-off height **SOT318-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	3.3	0.36 0.10	2.87 2.57	0.25	0.45 0.30	0.25 0.13	20.1 19.9	14.1 13.9	0.8	24.2 23.6	18.2 17.6	1.95	1.0 0.6	0.2	0.2	0.1	1.0 0.6	1.2 0.8	7^o 0^o

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT318-1						~~95-02-04~~ 97-08-01

Data sheet status

Data sheet status	Product status	Definition [1]
Objective specification	Development	This data sheet contains the design target or goal specifications for product development. Specification may change in any manner without notice.
Preliminary specification	Qualification	This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make chages at any time without notice in order to improve design and supply the best possible product.
Product specification	Production	This data sheet contains final specifications. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

[1] Please consult the most recently issued datasheet before initiating or completing a design.

Definitions

Short-form specification — The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

Limiting values definition — Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

Application information — Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Disclaimers

Life support — These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

Right to make changes — Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

Philips Semiconductors
811 East Arques Avenue
P.O. Box 3409
Sunnyvale, California 94088–3409
Telephone 800-234-7381

Date of release: 03-99

Document order number: 9397 750 05357

Let's make things better.

Philips Semiconductors

PHILIPS

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

NXP:

P80C557E4EFB/01,55 P80C557E4EFB/01,51