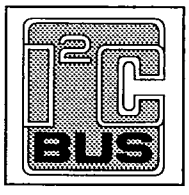
8-bit microcontroller with EMC and FEEPROM

P8xCE528

CONTENTS

1 FEATURES
2 GENERAL DESCRIPTION
3 ELECTROMAGNETIC COMPATIBILITY (EMC)
3.1 Recommendation on ALE
4 ORDERING INFORMATION
5 PINNING INFORMATION
5.1 Pinning
6 FUNCTIONAL DESCRIPTION
6.1 General
6.2 Memory Organization
6.2.1 Program Memory
6.2.2 Internal Data Memory
6.3 Addressing
6.4 I/O Facilities
6.5 Timers/Counters
6.5.1 Timer 0 and Timer 1
6.5.2 TMOD: Timer/counter Mode Control Register
6.5.3 TCON: Timer/counter Control Register
6.5.4 Timer 2
6.5.5 T2CON: Timer 2 Control Register
6.5.6 Capture Mode
6.5.7 Automatic Reload Mode
6.5.8 Baud Rate Generator Mode
6.6 Watchdog Timer T3
6.7 Full Duplex Serial Port (UART)
6.7.1 The Serial Port operating modes
6.7.2 SCON: Serial Port Control Register
6.7.3 SCON: SM0 and SM1 operating modes
6.8 Bit-Level I²C-bus Interface
6.8.1 S1INT: I²C-bus Interrupt Register
6.8.2 S1BIT: Single-bit Data Register with I²C-bus Auto-clock
6.8.3 Reading or Writing the S1BIT SFR
6.8.4 S1SCS: Control and Status Register for the I²C-bus
6.9 Idle and Power-down Operation
6.10 PCON: Power Control Register
6.11 Idle Mode
6.12 Power-down Mode
6.13 Wake-up from Power-down Mode
6.14 Interrupt System
6.14.1 Setting or clearing the Interrupt Enable Register
6.14.2 Interrupt priority structure
6.14.3 Interrupt Handling
6.14.4 IE: Interrupt Enable Register
6.14.5 IP: Interrupt Priority Register
6.14.6 Interrupt Vectors
6.15 Oscillator Circuit
6.16 Reset Circuit
6.17 Power-on Reset
7 INSTRUCTION SET
7.1 Addressing modes
8 FLASH EEPROM
8.1 General
8.2 FEEPROM Features
8.3 Memory Map
8.3.1 User Program Memory Selection
8.3.2 Boot ROM Selection
8.3.3 Switching between User Program Memory and Boot ROM
8.3.4 Description
8.3.5 FMCON: Control Register
8.3.6 User software example of FEEPROM page erase routine
8.3.7 User software example of FEEPROM byte write routine
8.4 Security Feature
8.5 Parallel Programming
8.5.1 89CE528 signature bytes
8.5.2 89CE528 security byte
9 FEEPROM PROGRAMMING/ERASE AND VERIFICATION CHARACTERISTICS
9.1 Serial Programming of FEEPROM
9.2 Boot Routine
9.3 Boot Routine Definitions
10 LIMITING VALUES
11 DC CHARACTERISTICS (EBx)
12 DC CHARACTERISTICS (EFx)
13 AC CHARACTERISTICS
14 I²C-bus CHARACTERISTICS (bit-level)
14.1 Timing symbol definitions
15 PACKAGE OUTLINES
16 SOLDERING
16.1 Plastic leaded chip carrier (PLCC) and quad flat-pack 10mm square; plastic (QFP44S10)
16.1.1 By wave
16.1.2 By solder paste reflow
16.1.3 Repairing soldered joints (by hand-held soldering iron or pulse-heated solder tool)
17 DEFINITIONS
18 LIFE SUPPORT APPLICATIONS
19 PURCHASE OF PHILIPS I²C COMPONENTS



8-bit microcontroller with EMC and FEEPROM

P8xCE528

1 FEATURES

- 80C51 central processing unit
- 32K x 8 ROM resp. FEEPROM, expandable externally to 64 kbytes
- ROM/FEEPROM code protection
- 512 x 8 RAM, expandable externally to 64 kbytes
- Four 8-bit I/O ports
- Full-duplex UART compatible with the standard 80C51 and the 8052
- Two standard 16-bit timer/counters
- An additional 16-bit timer (functionally equivalent to the timer 2 of the 8052)
- On-chip Watchdog Timer (WDT) with an on-chip oscillator
- Bit-level I²C-bus hardware serial I/O Port
- 7-source and 7-vector interrupt structure with 2 priority levels
- Up to 3 external interrupt request inputs
- Two programmable power reduction modes (Idle and Power-down)
- Termination of Idle mode by any interrupt, external or WDT (watchdog) reset
- Wake-up from Power-down by external interrupt, external or WDT reset
- Software enable/disable of ALE output pulse
- Electro-Magnetic Compatibility (EMC) improvements
- XTAL frequency range: 3.5 MHz to 16 MHz
- 4.5 to 5.5 V supply voltage range
- Extended Temperature range (–40 to +85 °C)

2 GENERAL DESCRIPTION

The P83CE528; P80CE528; P89CE528 (hereafter generically referred to as P8xCE528) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family.

The P8xCE528 has the same instruction set as the 80C51. Three versions of the derivative exist:

- P83CE528: 32 kbytes mask programable ROM
- P80CE528: ROMless version of the P83CE528
- P89CE528: 32 kbytes FEEPROM (Flash Electrically Erasable Program Memory).

This device provides architectural enhancements that make it applicable in a variety of applications in general control systems, especially in those systems which need a large ROM and RAM capacity on chip.

The P8xCE528 contains a non-volatile 32K x 8 read-only program memory (P83CE528) or FEEPROM (P89CE528), a volatile 512 x 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a 16-bit timer (identical to the timer 2 of the 8052), a multi-source, two-priority-level, nested interrupt structure, two serial interfaces (UART and bit-level I²C-bus), an on-chip oscillator and timing circuits, a watchdog timer (WDT) with a separate on-chip oscillator. For systems that require extra capability, the P8xCE528 can be expanded using standard TTL compatible memories and logic.

In addition, the P8xCE528 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 plus 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 crystal, 58% of the instructions are executed in 0.75 µs and 40% in 1.5 µs. Multiply and divide instructions require 3 µs.

3 ELECTROMAGNETIC COMPATIBILITY (EMC)

Primary attention is paid to the reduction of electro-magnetic emission of the microcontroller P8xCE528. The following features reduce the electro-magnetic emission and additionally improve the electromagnetic susceptibility:

- Two supply voltage pins ($V_{DD1;2}$) and four ground pins ($V_{SS1;2;3;4}$) are provided on the package as follows:
  - one V_{DD} and one V_{SS} as a pair of pins placed mid-centre on one side of the package
  - a second pair of V_{DD} and V_{SS} pins placed mid-centre on the opposite side of the package
  - two more V_{SS} pins, one placed on each of the other two sides of the package.
- Separated V_{DD} pins for the internal logic and the port buffers.
- Internal decoupling capacitance improves the EMC radiation behaviour and the EMC immunity.
- External capacitors are to be located as close as possible between pins V_{DD1} and V_{SS1} as well as V_{DD2} and V_{SS3}; ceramic chip capacitors are recommended (100 nF).

3.1 Recommendation on ALE

For applications that require no external memory or temporarily no external memory: the ALE output signal (pulses at a frequency of $f_{osc}/6$) can be disabled under software control (bit 5 in the PCON SFR: "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 as a normal MOVX. 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.

4 ORDERING INFORMATION

EXTENDED TYPE NUMBER	PACKAGE				TEMPERATURE RANGE (°C)	FREQUENCY (MHz)
	PINS	PIN POSITION	MATERIAL	CODE		
ROMless						
P80CE528EBB	44	QFP	plastic	SOT307-2	0 to +70	3.5 to 16
P80CE528EFB	44	QFP	plastic	SOT307-2	−40 to +85	3.5 to 16
P80CE528EBA	44	PLCC	plastic	SOT187	0 to +70	3.5 to 16
P80CE528EFA	44	PLCC	plastic	SOT187	−40 to +85	3.5 to 16
ROM						
P83CE528EBB	44	QFP	plastic	SOT307-2	0 to +70	3.5 to 16
P83CE528EFB	44	QFP	plastic	SOT307-2	−40 to +85	3.5 to 16
P83CE528EBA	44	PLCC	plastic	SOT187	0 to +70	3.5 to 16
P83CE528EFA	44	PLCC	plastic	SOT187	−40 to +85	3.5 to 16
FEEPROM						
P89CE528EBA	44	PLCC	plastic	SOT187	0 to +70	3.5 to 16
P89CE528EFA	44	PLCC	plastic	SOT187	−40 to +85	3.5 to 16

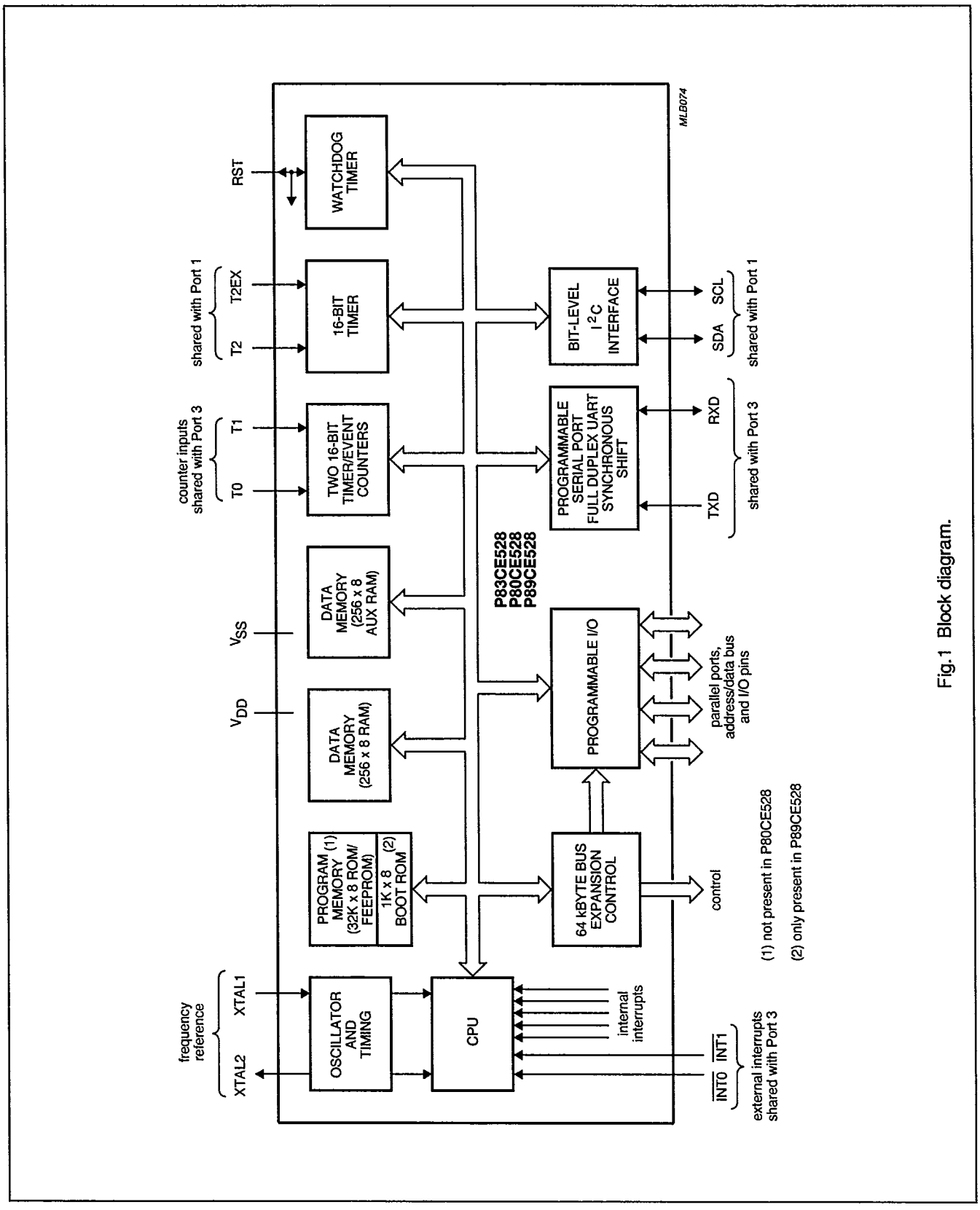
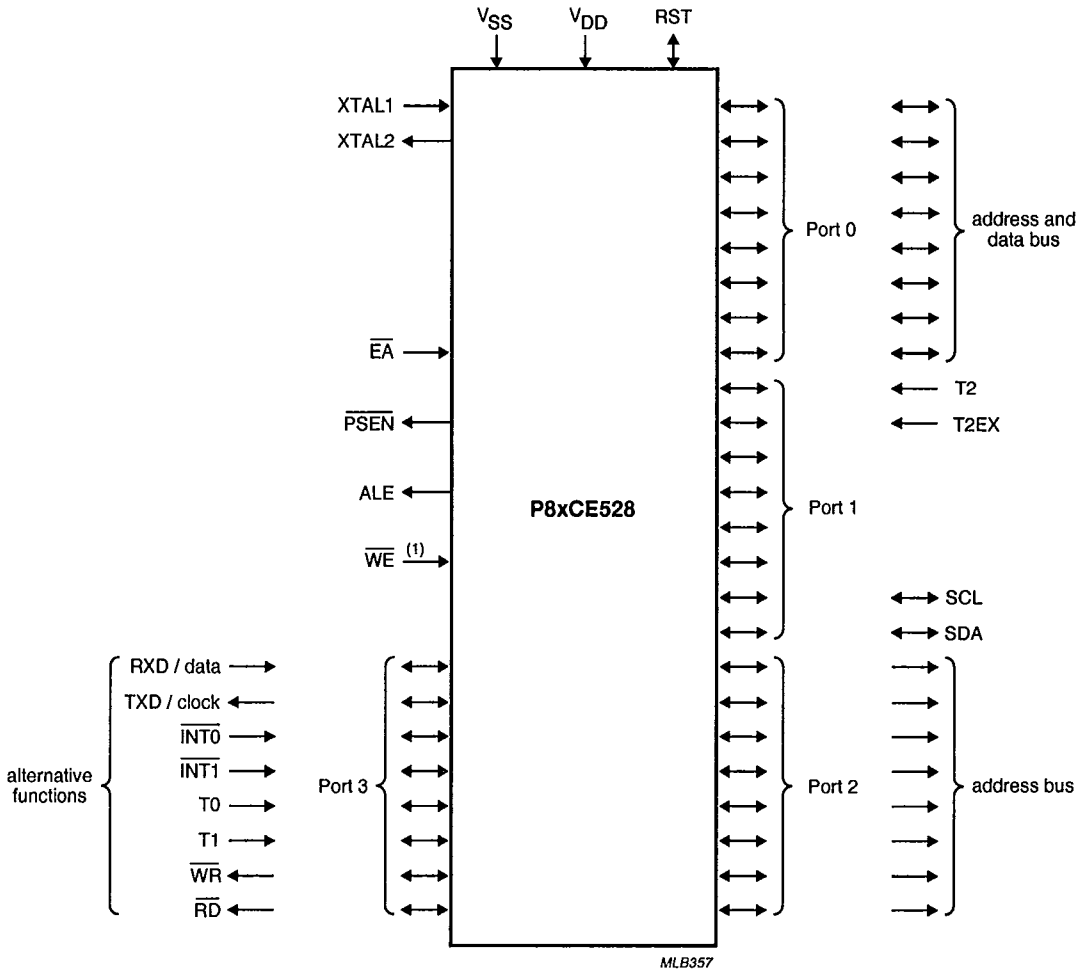


Fig.1 Block diagram.



(1) Only P89CE528 with alternative function.

Fig.2 Functional diagram.

5 PINNING INFORMATION

5.1 Pinning

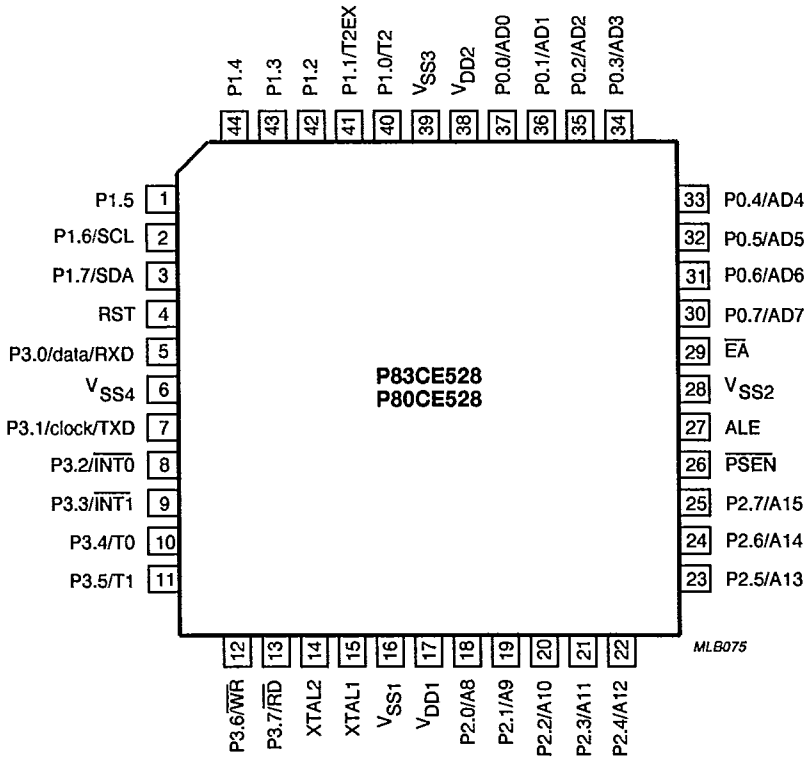
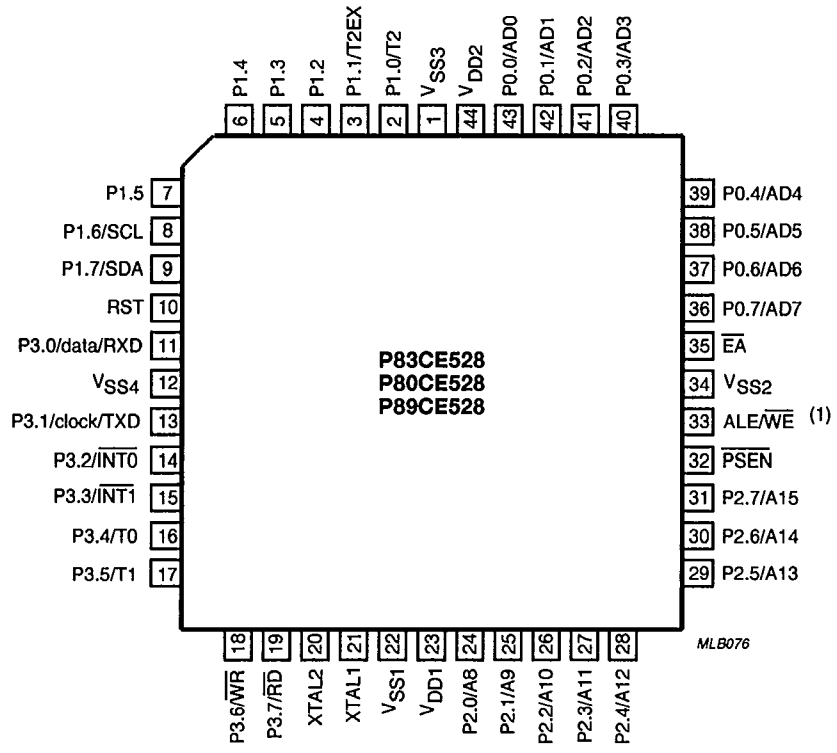


Fig.3 Pin configuration: 44-lead QFP; (SOT307-2).



(1) Only P89CE528 with alternative function.

Fig.4 Pin configuration: 44-lead PLCC; (SOT187CG).

Pin description: 44-lead QFP; (SOT307-2).

SYMBOL	PIN	DESCRIPTION
P1.0 to P1.7	40 to 44 1 to 3	**Port 1:** 8-bit quasi-bidirectional I/O Port. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7 which have open drain outputs.
		Port 1 alternative functions:
P1.0/T2	40	Timer/event counter 2 external event counter input (falling edge triggered).
P1.1/T2EX	41	Timer/event counter 2 capture/reload trigger or external interrupt 2 input (falling edge triggered).
P1.6/SCL	2	I²C-bus Serial Port clock line.
P1.7/SDA	3	I²C-bus Serial Port data line.
RST	4	**RESET:** a HIGH level on this pin for two machine cycles while the oscillator is running, resets the device. An internal pull-down resistor permits power-on reset using only a capacitor connected to V_{DD}. After a WDT overflow this pin is pulled HIGH while the internal reset signal is active.
P3.0 to P3.7	5, 7 to 13	**Port 3:** 8-bit quasi-bidirectional I/O Port with internal pull-ups. Port 3 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
		Port 3 alternative functions:
P3.0/RXD/data	5	Serial Port data input (asynchronous) or data input/output (synchronous).
P3.1/TXD/clock	7	Serial Port data output (asynchronous) or clock output (synchronous).
P3.2/$\overline{INT0}$	8	External interrupt 0 or gate control input for timer/event counter 0.
P3.3/$\overline{INT1}$	9	External interrupt 1 or gate control input for timer/event counter 1.
P3.4/T0	10	External input for timer/event counter 0.
P3.5/T1	11	External input for timer/event counter 1.
P3.6/$\overline{WR}$	12	External data memory write strobe.
P3.7/$\overline{RD}$	13	External data memory read strobe. The generation or use of a Port 3 pin as an alternative function is carried out automatically by the P8xCE528 provided the associated Special Function Register (SFR) bit is set HIGH.
XTAL2	14	**Crystal pin 2:** output of the inverting amplifier that forms the oscillator. This pin left open-circuit when an external oscillator clock is used (see Figures 15 and 16).
XTAL1	15	**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 (see Figures 15 and 16).
$V_{SS1;2;3;4}$	16, 28, 39, 6	**Ground:** circuit ground potential. All pins must be connected.

SYMBOL	PIN	DESCRIPTION
P2.0 to P2.7	18 to 25	**Port 2:** 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 (A8 to A15). Port 2 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
$\overline{PSEN}$	26	**Program Store Enable output:** read strobe to the external program memory via Port 0 and Port 2. It 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/$\overline{WE}$	27	**Address Latch Enable output:** latches the LOW byte of the address during access to 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 (note 1).
$\overline{EA}$	29	**External Access input:** when during RESET, $\overline{EA}$ is held at a TTL HIGH level, the CPU executes out of the internal program ROM, provided the program counter is less than 32768. When $\overline{EA}$ is held at a TTL LOW level during RESET, 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.
P0.0 to P0.7	30 to 37	**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 (AD0 to AD7). During these accesses internal pull-ups are activated. Port 0 can sink/source 8 LSTTL inputs.
$V_{DD1;2}$		**Power supply:** +5 V power supply pin during normal operation and power reduction modes. Both pins must be connected.
V_{DD1}	17	Power supply pin for ports, ALE, $\overline{PSEN}$ and on-chip oscillator.
V_{DD2}	38	Power supply pin for internal logic.
		To avoid a latch-up effect at power-on, the voltage on any pin (at any time) must not be higher than V_{DD} +0.5 V or lower than V_{SS} −0.5 V respectively.

Note

1. To prohibit the toggling of ALE/$\overline{WE}$ 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 cleared by software. When set, ALE/$\overline{WE}$ pin will be pulled down internally, switching an external address latch to a quiet state. The MOVX instruction will still toggle ALE/$\overline{WE}$ as a normal MOVX. ALE/$\overline{WE}$ 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/$\overline{WE}$ will toggle normally when the address exceeds the internal program memory size. During external access ($\overline{EA}$ = 0) ALE/$\overline{WE}$ will always toggle normally, whether the flag "RFI" is set or not.

6 FUNCTIONAL DESCRIPTION

6.1 General

The P8xCE528 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 P8xCE528 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 P8xCE528 can be expanded using standard memories and peripherals.

The P8xCE528 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 except the WDT if it is enabled. The Power-down mode can be terminated by an external reset, a WDT overflow, and in addition, by either of the two external interrupts.

6.2 Memory Organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 kbyte external data memory (of which the lower 256 bytes reside in the internal AUX-RAM), 512 byte internal data memory (consisting of 256 bytes standard RAM and 256 bytes AUX-RAM) and the 64 kbyte internal and external program memory.

6.2.1 Program Memory

The program memory of the P8xCE528 consists of 32 kbyte of ROM respectively FEEPROM ("Flash Memory") on-chip, externally expandable up to 64 kbyte. If the $\overline{EA}$ pin was HIGH during RESET, the P8xCE528 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 LOW during RESET, the P8xCE528 fetches all instructions from the external program memory. The $\overline{EA}$ input level is latched during RESET and is don't care after RESET. Figure 5 illustrates the program memory address space.

By setting a mask programmable security bit (ROM) respectively software programmable security byte (FEEPROM) the internal memory content is protected i.e. it cannot be read out 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 reading from 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 by the MOVC instructions when the security feature has been activated. If the security feature is not activated, there are no restrictions for the MOVC instructions.

Table 1 Internal and external program memory access with activated security feature.

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

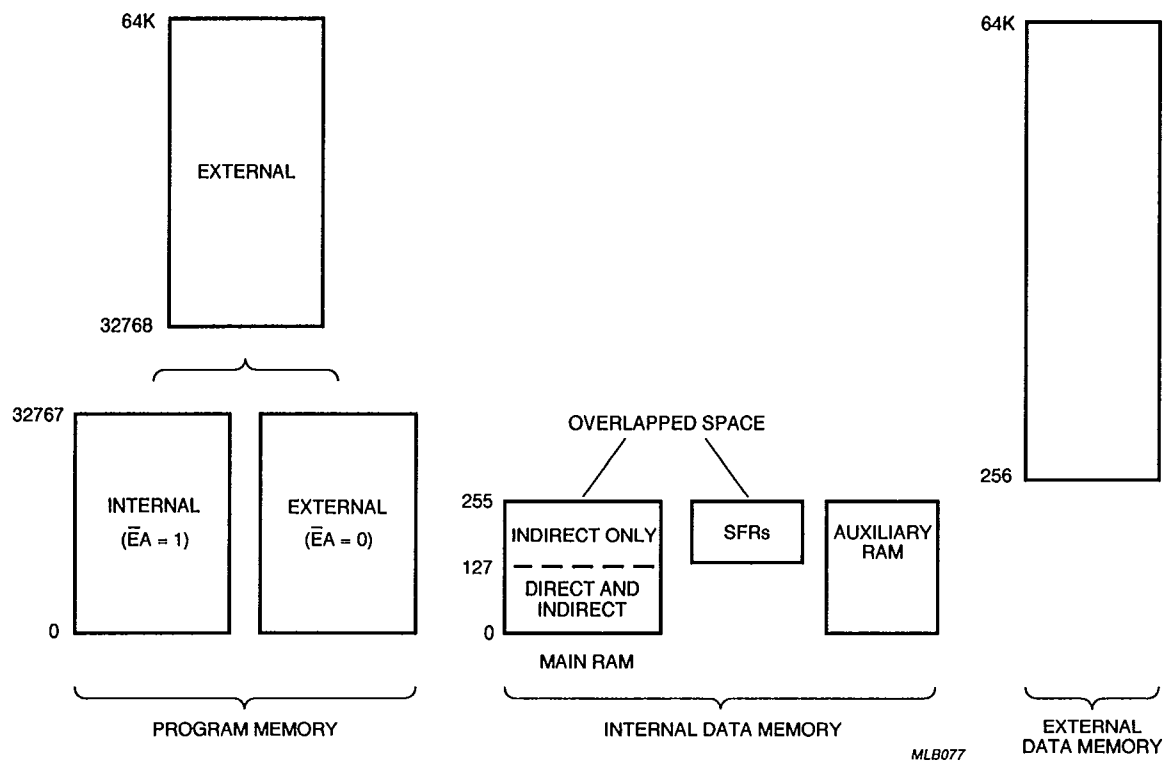


Fig.5 Memory map and address Space.

6.2.2 Internal Data Memory

The internal data memory is divided into three physically separated parts: 256 byte of Main RAM, 256 byte of AUX-RAM, and a 128 byte special function area (SFR) see Table 2. These parts can be addressed as follows:

- Main RAM 0 to 127 can be addressed directly and indirectly as in the 80C51. Address pointers are R0 and R1 of the selected register bank.
- Main RAM 128 to 255 can only be addressed indirectly. Address pointers are R0 and R1 of the selected register bank.
- AUX-RAM 0 to 255 is indirectly addressable in the same way as the external Data Memory locations 0 to 255 with the MOVX instructions. Address pointers are R0 and R1 of the selected register bank and DPTR. When executing from internal program memory, an access to AUX-RAM 0 to 255 will not affect the Ports P0, P2, P3.6 and P3.7.
- An access to external Data Memory locations higher than 255 will be performed with the MOVX DPTR instructions in the same way as in the 80C51 structure, i.e. with P0 and P2 as data/address bus and P3.6 and P3.7 as write and read strobe signals. Note that it is impossible to access the external Data Memory with R0, R1 or DPTR < 256 as address pointer.
- The SFRs can only be addressed directly in the address range from 128 to 255 (Figure 6 illustrates the SFRs memory map).

Four 8-bit register banks 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 byte RAM. The stack depth is only limited by the available internal RAM space of 256 bytes. All registers except the Program Counter and the four 8-bit register banks reside in the SFR address space.

Table 2 Internal data memory access.

LOCATION	ADDRESSED
Main RAM 0 to 127	DIRECT and INDIRECT
AUX-RAM 0 to 255	INDIRECT only with MOVX
Main RAM 128 to 255	INDIRECT only
SFR 128 to 255	DIRECT only

6.3 Addressing

The P8xCE528 has five modes 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 8-bit register banks through Register, Direct or Register-Indirect addressing.
- 512 bytes of internal RAM through Direct or Register-Indirect addressing. Bytes 0 to 127 of internal RAM may be addressed directly/indirectly. Bytes 128 to 255 of internal RAM share their address location with the SFRs and so may only be addressed indirectly as data RAM. Bytes 0 to 255 of AUX-RAM can only be addressed indirectly via MOVX.
- SFR through Direct addressing at address locations 128 to 255 (see Figure 6).
- External data memory through Register-Indirect addressing.
- Program memory look-up tables through Base-Register plus Index-Register-Indirect addressing.

6.4 I/O Facilities

The P8xCE528 has four 8-bit ports. Ports 0-3 are the same as in the 80C51, with the exception of the additional function of Port 1. Port lines P1.0 and P1.1 may be used as inputs for Timer 2, P1.1 may also be used as an additional (third) external interrupt request input. Port lines P1.6 and P1.7 may be selected as the SCL and SDA lines of Serial Port SIO1 (I^2C-bus). Because the I^2C-bus may be active while the device is disconnected from V_{DD}, these pins are provided with open drain drivers. Pins P1.6 and P1.7 do not have pull-up devices when used as ports.

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

- Port 0: provides the multiplexed low-order address and data bus used for expanding the P8xCE528 with standard memories and peripherals.
- Port 1: pins can be configured individually to provide: external interrupt request input (external interrupt 2); external inputs for Timer/counter 2; SCL and SDA for the I^2C-bus interface.
- Port 2: provides the high-order address bus when expanding the P8xCE528 with external program memory and/or external data memory.
- Port 3: pins can be configured individually to provide: external interrupt request inputs (external interrupt 0/1); external inputs for Timer/counter 0 and Timer/counter 1; Serial Port receiver input and transmitter output; control-signals to read and write external data memory.

Bits which are not used for the alternative functions may be used as normal bidirectional I/O pins. The generation or use of a Port 1 or Port 3 pin as an alternative function is carried out automatically by the P8xCE528 provided the associated SFR bit is HIGH. Otherwise the port pin is held at a logical LOW level.

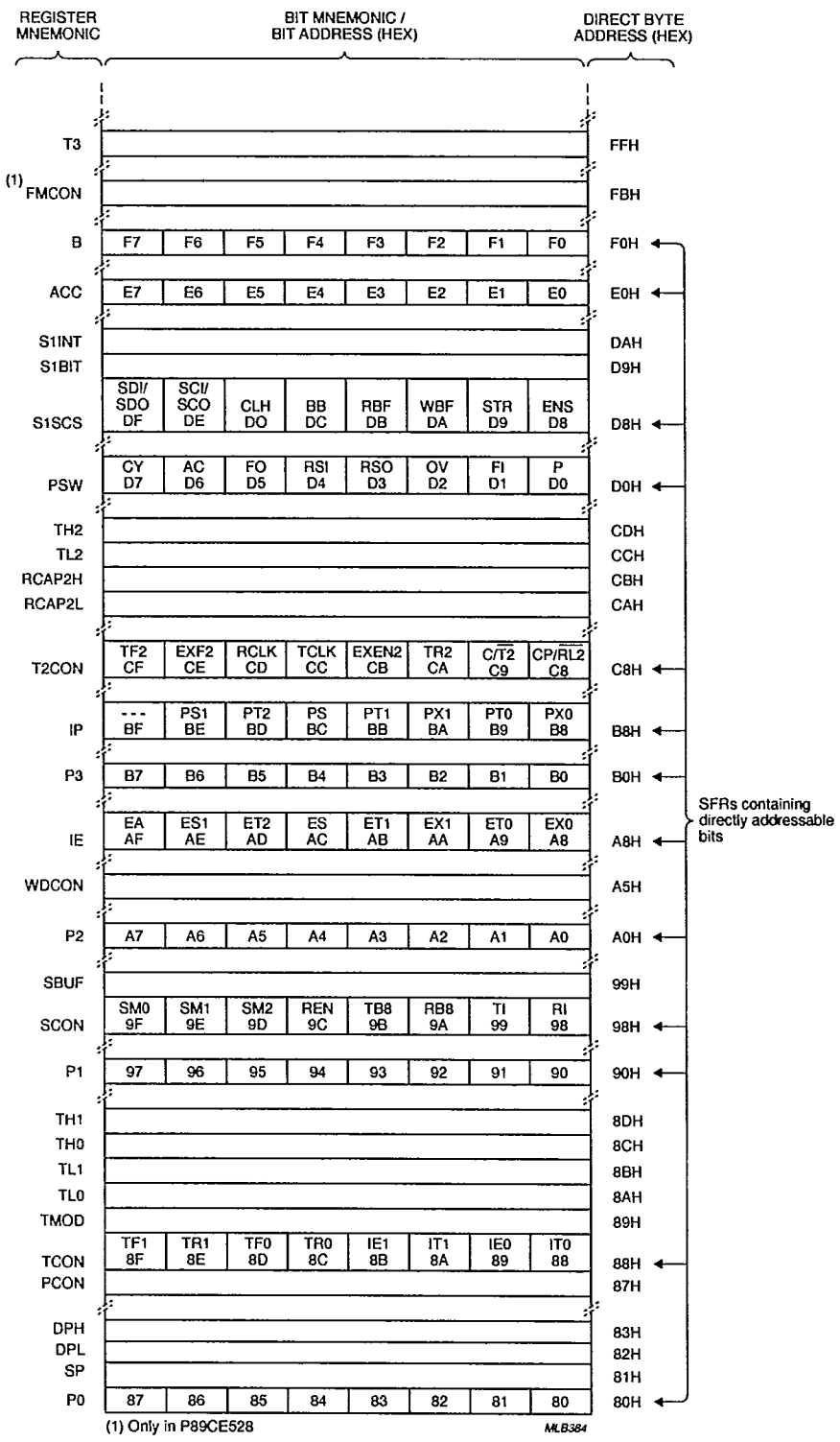


Fig.6 Special Function Registers (SFR) memory map.

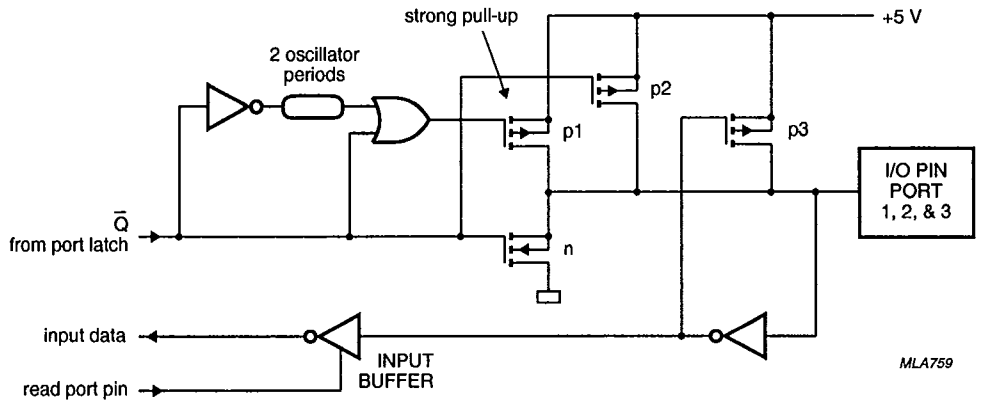


Fig.7 I/O buffers in the P8xCE528 (Ports 1, 2 and 3 except P1.6 and P1.7).

6.5 Timers/Counters

The P8xCE528 contains three 16-bit timer/counters, Timer 0, Timer 1 and Timer 2, and one 8-bit timer, the Watchdog Timer T3. Timer 0, Timer 1 and Timer 2 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- generate interrupt requests.

6.5.1 Timer 0 and Timer 1

Timers 0 and 1 each have a control bit in TMOD SFR 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 HIGH-to-LOW 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 HIGH-to-LOW transition. There are no restrictions on the duty cycle of the external input signal, but to ensure 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 and 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 (oscillator frequency divided by 12).
- In the counter function, the frequency handling range for external inputs is 0 Hz to 0.66 MHz ($f_{CLK}/24$).

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

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, in which the incrementing pulse occurred.

The timers are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (respectively, the automatic reload value), with the exception of Mode 3 as previously described.

6.5.2 TMOD: Timer/counter Mode Control Register

This register is located at address 89H.

TMOD SFR (89H).

bit-7	6	5	4	3	2	1	0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
TIMER 1				**TIMER 0**			

Table 3 Description of TMOD bits.

MNEMONIC	BIT	FUNCTION
TIMER 1		
GATE	TMOD.7	**Timer 1 gating control:** when set, Timer/counter 1 is enabled only while $\overline{INT1}$ pin is HIGH and TR1 control bit is set. When cleared, Timer/counter 1 is enabled whenever TR1 control bit is set.
C/$\overline{T}$	TMOD.6	**Timer or counter selector:** cleared for timer operation (input from internal system clock). Set for counter operation (input from T1 input pin).
M1	TMOD.5	**Operating mode:** see Table 4.
M0	TMOD.4	**Operating mode:** see Table 4.
TIMER 0		
GATE	TMOD.3	**Timer 0 gating control:** when set, Timer/counter 0 is enabled only while $\overline{INT0}$ pin is HIGH and TR0 control bit is set. When cleared, Timer/counter 0 is enabled whenever TR0 control bit is set.
C/$\overline{T}$	TMOD.2	**Timer or counter selector:** cleared for timer operation (input from internal system clock). Set for counter operation (input from T0 input pin).
M1	TMOD.1	**Operating mode:** see Table 4.
M0	TMOD.0	**Operating mode:** see Table 4.

Table 4 TMOD M1 and M0 operating modes.

M1	M0	FUNCTION
0	0	**8-bit timer/counter:** "THX" with "TLX" as 5-bit prescaler.
0	1	**16-bit timer/counter:** "THX" and "TLX" are cascaded, there is no prescaler.
1	0	**8-bit autoload 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 controlled by Timer 1 control bits.
1	1	**Timer 1:** Timer/counter 1 stopped.

6.5.3 TCON: Timer/counter Control Register

This register is located at address 88H.

TCON SFR (88H).

bit-7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Table 5 Description of TCON bits.

MNEMONIC	BIT	FUNCTION
TF1	TCON.7	**Timer 1 overflow flag**: set by hardware on timer/counter overflow. Cleared when interrupt is processed.
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 when interrupt is processed.
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 is detected. Cleared when interrupt is processed.
IT1	TCON.2	**Interrupt 1 type control bit**: set/cleared by software to specify falling edge/LOW level triggered external interrupt.
IE0	TCON.1	**Interrupt 0 edge flag**: set by hardware when external interrupt is detected. Cleared when interrupt is processed.
IT0	TCON.0	**Interrupt 0 type control bit**: set/cleared by software to specify falling edge/LOW level triggered external interrupt.

6.5.4 Timer 2

Timer 2 is functionally similar to the Timer 2 of the 8052AH. Timer 2 is a 16-bit timer/counter which is formed by two SFRs, TL2 and TH2. Another pair of SFRs, RCAP2L and RCAP2H, form a 16-bit capture register or a 16-bit reload register. Like timers 0 and 1, Timer 2 can operate either as timer or as event counter. This is selected by bit C/$\overline{T2}$ in the T2CON SFR. The timer has three operating modes: capture, autoload and baud rate generator, which are selected by bits in the T2CON SFR (see Table 6).

Table 6 Timer 2 operating modes.

RCLK + TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit automatic reload
0	1	1	16-bit capture
1	X	1	Baud rate generator
X	X	0	OFF

6.5.5 T2CON: Timer 2 Control Register

This register is located at address C8H.

T2CON SFR (C8H).

bit-7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

Table 7 Description of T2CON bits.

MNEMONIC	BIT	FUNCTION
TF2	T2CON.7	**Timer 2 overflow flag:** set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	T2CON.6	**Timer 2 external flag:** set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine.
RCLK	T2CON.5	**Receive clock flag:** when set, causes the Serial Port to use Timer 2 overflow pulses for its receive clock in Modes 1 and 3. RCLK = 0 causes Timer 1 overflows to be used for the receive clock.
TCLK	T2CON.4	**Transmit clock flag:** when set, causes the Serial Port to use Timer 2 overflow pulses for its transmit clock in Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	T2CON.3	**Timer 2 external enable flag:** when set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the Serial Port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	T2CON.2	**Start/stop control:** a logic 1 starts Timer 2. A logic 0 stops Timer 2.
C/$\overline{T2}$	T2CON.1	**Timer/counter select:** 0 = internal timer (OSC/12). 1 = external event counter (falling edge triggered).
CP/$\overline{RL2}$	T2CON.0	**Capture/reload flag:** when set, capture will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, reloads will occur upon either Timer 2 overflows or negative transitions at T2EX if EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to reload upon overflow.

6.5.6 Capture Mode

In the capture mode (see Figure 8) there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer/counter which on overflow sets bit TF2 (Timer 2 overflow bit). TF2 can be used to generate an interrupt. If EXEN2 = 1, Timer 2 operates as above, with the added feature that a HIGH-to-LOW transition at the external input T2EX causes the current value in Timer 2 registers (TL2 and TH2) to be captured into registers RCAP2L and RCAP2H, respectively. The HIGH-to-LOW transition of T2EX also causes bit EXF2 in T2CON to be set. EXF2 can be used to generate an interrupt.

6.5.7 Automatic Reload Mode

In the automatic reload mode (see Figure 9) there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then a Timer 2 overflow sets TF2 and causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software.

If EXEN2 = 1, Timer 2 operates as above, with the added feature that a HIGH-to-LOW transition at the external input T2EX triggers the 16-bit reload and sets EXF2.

6.5.8 Baud Rate Generator Mode

The baud rate generator mode (see Figure 10) is selected by RCLK = 1 and/or TCLK = 1 in T2CON. Overflows of either Timer 2 or Timer 1 can be used independently for generating baud rates for transmit and receive. The baud rate generation by Timer 1 and/or Timer 2 is used for the Serial Port in Mode 1 and Mode 3. The baud rate generation mode is similar to the automatic reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software. The baud rate for the Serial Port in Modes 1 and 3 are determined by Timer 2 overflow rate as follows:

$$Baud\ rate = \frac{Timer\ 2\ overflow\ rate}{16}$$

Timer 2 can be configured for either timer or counter operation. In timer operation a prescaler divides the oscillator frequency by 2 (by 12 in the previous modes) and the baud rate is given by the formula:

$$Baud\ Rate = \frac{oscillator\ frequency}{32 \times [\,65536 - (\,RCAP2H, RCAP2L\,)\,]}$$

In this mode an overflow of Timer 2 does not set TF2. If EXEN2 = 1, a HIGH-to-LOW transition at pin T2EX sets EXF2 and can be used to generate an interrupt.

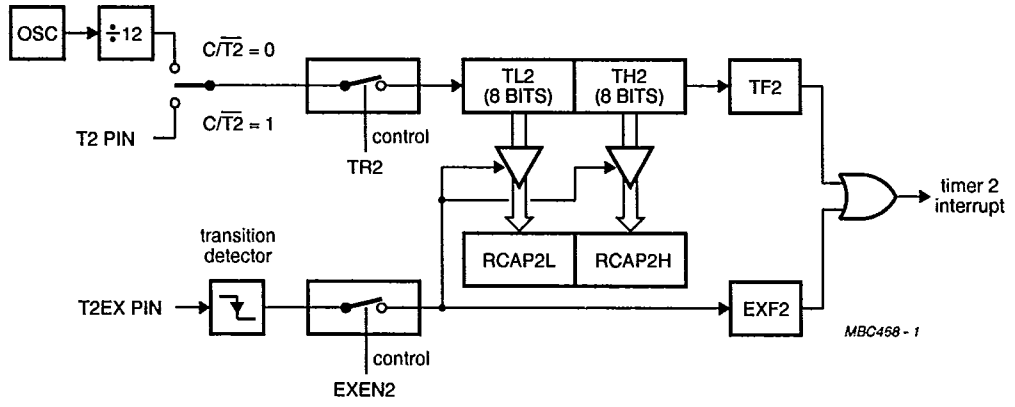


Fig.8 Timer 2 in capture mode.

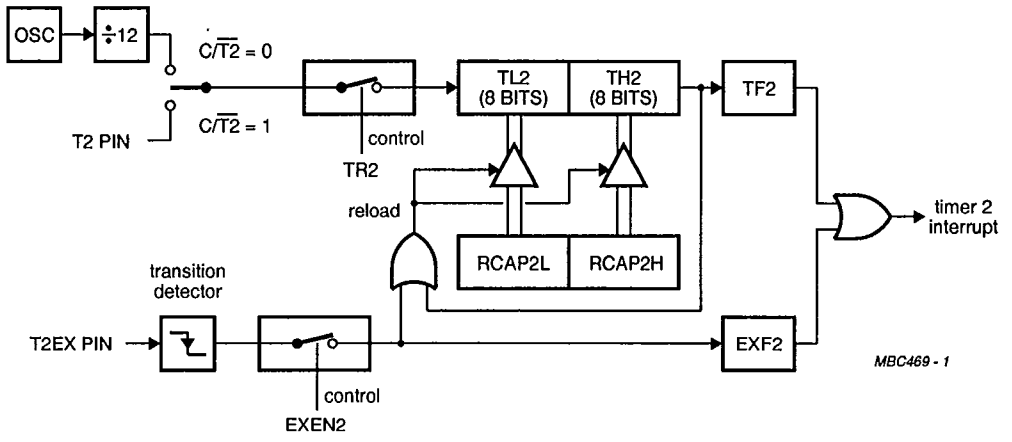


Fig.9 Timer 2 in automatic reload mode.

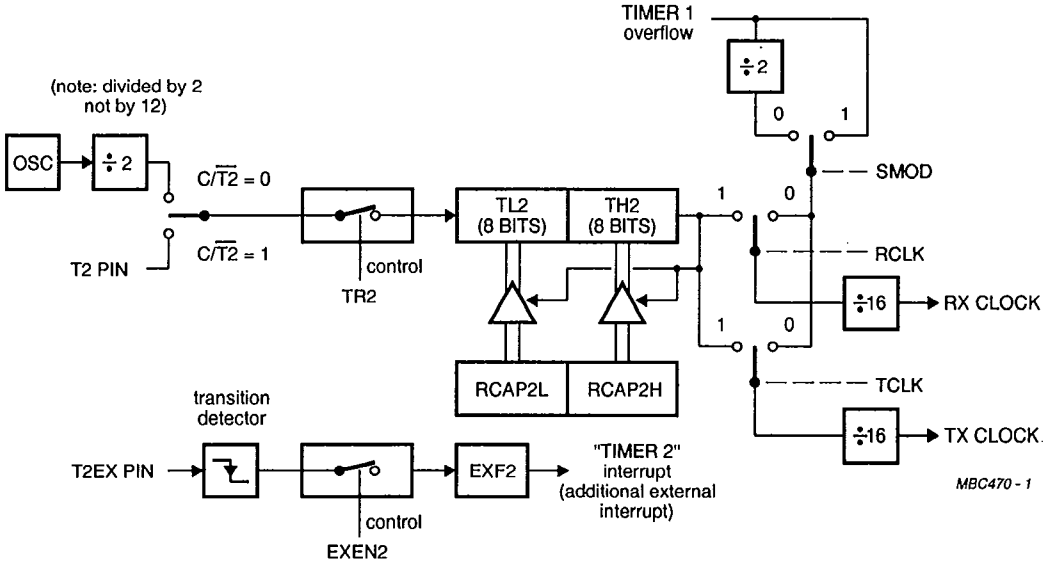


Fig.10 Timer 2 in baud rate generator mode.

6.6 Watchdog Timer T3

The Watchdog Timer (WDT) see Figure 11, consists of an 11-bit prescaler and an 8-bit timer formed by SFR T3. The prescaler is incremented by an on-chip oscillator with a fixed frequency of 1 MHz. The maximum tolerance of this frequency is –50% and +100%. The 8-bit timer increments every 2048 cycles of the on-chip oscillator. When a timer overflow occurs, the microcontroller is reset and a reset-output-pulse of 16 x 2048 cycles of the on-chip oscillator is generated at pin RST. The internal RESET signal is not inhibited when the external RST pin is kept LOW by e.g. an external reset circuit. The RESET signal drives Ports 1, 2 and 3 outputs into the HIGH state and Port 0 into high impedance, no matter if the XTAL-clock is running or not.

The WDT is controlled by WDCON SFR with the direct address location A5H. WDCON can be read and written by software. A value of A5H in WDCON halts the on-chip oscillator and clears both the prescaler and Timer T3. After RESET, WDCON contains A5H. Every value other than A5H in WDCON enables the WDT. When the WDT is enabled it runs independent of the XTAL-clock.

Timer T3 can be read on the fly. Timer T3 can be written only if WDCON has previously been loaded with 5AH, otherwise T3 and the prescaler are not affected. A successful write operation to T3 also clears the prescaler and clears WDCON.

To prevent an overflow of the WDT, the user program has to reload T3 within periods that are shorter than the programmed Watchdog time interval. This time interval is determined by the 8-bit reload value that is written into register T3.

$$\textit{Watchdog time interval} = \frac{[256 - (T3)] \times 2048}{\textit{on-chip oscillator frequency}}$$

The advantages of this implementation are:

- Only an internal reset connection to the microcontroller core.
- The Power-down mode and the Watchdog (WDT) function can be used concurrently.

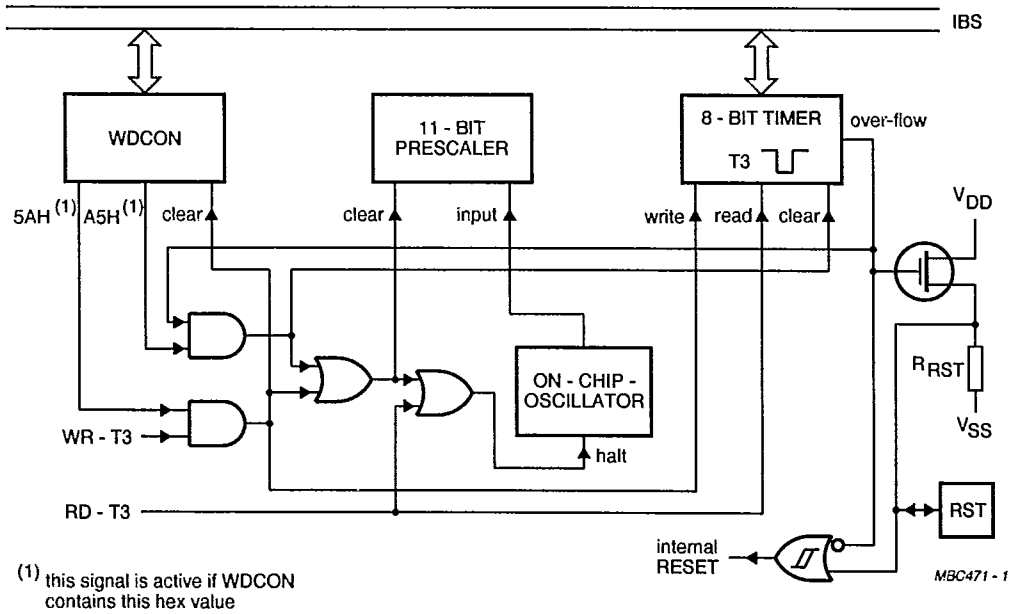


Fig.11 Watchdog Timer T3.

- The WDT also monitors the XTAL oscillator. In case of a failure the port outputs are forced to a defined HIGH state.
- Interference will not disable the WDT because it is unlikely that it will force WDCON to A5H.
- Tolerances of the on-chip oscillator can be adjusted by testing the T3 value and adapting the reload value.
- The WDT can be enabled and disabled under control of the user software. This gives the possibility to use both the Watchdog function and the Power-down function.
- The direct address A5H of WDCON and its disable value A5H will not unintentionally be present at a random location in the field of program code, except for immediate data, because the opcode A5H is not used in the instruction set.

6.7 Full Duplex Serial Port (UART)

The Serial Port is functionally similar to the implementation in the 8052AH, with the possibility of two different baud rates for receive and transmit with Timer 1 and Timer 2 as baud rate generators. It is full duplex, meaning it can receive and transmit simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the receive register. However, if the first byte still has not been read by the time the reception of the second byte is complete, one of the bytes will be lost. The Serial Port receive and transmit registers are both accessed as SBUF SFR. Writing to SBUF loads the transmit register, and reading SBUF accesses the physically separate receive register.

6.7.1 The Serial Port operating modes

The Serial Port can operate in one of four modes (see Table 8):

Table 8 The Serial Port operating modes.

MODE	DESCRIPTION
Mode 0	Serial data enters and exits through RXD. TXD outputs the shift clock. 8 bits are transmitted/received: 8 data bits (LSB first). The baud rate is fixed at 1/12 the oscillator frequency.
Mode 1	10 bits are transmitted (through 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 stored in RB8 in SCON SFR. 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 SCON) usually represents the parity bit (P, in the PSW). On receive, the 9th data bit is stored in RB8 (SCON), while 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). In fact, Mode 3 is the same as Mode 2 in all respects except the baud rate. The baud rate in Mode 3 is variable.
Note: in all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. In Mode 0, reception is initiated by the condition RI = 0 and REN = 1. Reception is initiated by incoming start bit if REN = 1 in the other modes.	

6.7.2 SCON: Serial Port Control Register

This register is located at address 98H.

SCON SFR (98H).

bit-7	6	5	4	3	2	1	0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table 9 Description of SCON bits.

MNEMONIC	BIT	FUNCTION
SM0	SCON.7	**Operating modes**, see Table 10.
SM1	SCON.6	**Operating modes**, see Table 10.
SM2	SCON.5	**Enables the multiprocessor communication feature in Modes 2 and 3.** In Modes two and three if SM2 is set to logic 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 is not received. In Mode 0, SM2 should be logic 0.
REN	SCON.4	**Enables serial reception.** Set and cleared by software as required.
TB8	SCON.3	**9th data bit that will be transmitted in Modes 2 and 3.** Set and cleared by software as required.
RB8	SCON.2	**In Modes 2 and 3, RB8 is the 9th data bit that is received.** In Mode 1, if SM2 = 0, RB8 is the stop bit that is received. In Mode 0, RB8 is not used.
TI	SCON.1	**Transmit interrupt flag.** It is 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. TI must be cleared by software.
RI	SCON.0	**Receive interrupt flag.** It is 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 (except: see SM2). RI must be cleared by software.

6.7.3 SCON: SM0 and SM1 operating modes

SCON bits SM0 and SM1 can operate in the following modes (see Table 10):

Table 10 SCON: SM0 and SM1 operating modes.

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

6.8 Bit-Level I²C-bus Interface

This bit-level serial I/O interface supports the I²C-bus. P1.6/SCL and P1.7/SDA are the serial I/O pins. These two pins meet the I²C-bus specification concerning the input levels and output drive capability. Consequently, these pins have an open drain output configuration. All four modes of the I²C-bus are supported:

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

The advantages of the bit-level I²C-bus hardware compared with a full software I²C-bus implementation are:

- The hardware can generate the SCL pulse
- Testing a single bit (RBF respectively, WBF) is a sufficient check for error free transmission.

The bit-level I²C-bus hardware operates on serial bit level and performs the following functions:

- Filtering the incoming serial data and clock signals
- Recognizing the START condition
- Generating a serial interrupt request SI after reception of a START condition and the first falling edge of the serial clock
- Recognizing the STOP condition
- Recognizing a serial clock pulse on the SCL line
- Latching a serial bit on the SDA line (SDI)
- Stretching the SCL LOW period of the serial clock to suspend the transfer of the next serial data bit
- Setting Read Bit Finished (RBF) when the SCL clock pulse has finished and Write Bit Finished (WBF) if there is no arbitration loss detected (i.e. SDA = 0 while SDO = 1)
- Setting a serial clock LOW-to-HIGH detection flag (CLH)
- Setting a Bus Busy (BB) flag on a START condition and clearing this flag on a STOP condition
- Releasing the SCL line and clearing the CLH, RBF and WBF flags to resume transfer of the next serial data bit
- Generating an automatic clock if the single bit data register S1BIT is used in master mode.

The following functions must be done in software:

- Handling the I²C-bus START interrupts
- Converting serial to parallel data when receiving
- Converting parallel to serial data when transmitting
- Comparing the received slave address with its own
- Interpreting the acknowledge information.
- Guarding the I²C-bus status if RBF or WBF = 0.

Additionally, if acting as master:

- Generating START and STOP conditions
- Handling bus arbitration
- Generating serial clock pulses if S1BIT is not used.

The bit-level I²C-bus interface is controlled by three SFRs: S1INT, S1BIT and S1SCS.

6.8.1 S1INT: I²C-BUS INTERRUPT REGISTER

This register is located at address DAH.

S1INT SFR (DAH).

bit-7	6	5	4	3	2	1	0
SI (note 1)	X	X	X	X	X	X	X

Note

1. **SI bit:** writing a logic 0 clears this bit, writing a logic 1 has no effect.

Table 11 Description of S1INT bits.

MNEMONIC	BIT	FUNCTION
SI	S1INT.7	**Serial Interrupt request (SI) flag:** if a START condition occurs the SI flag in the S1INT SFR is set on the falling edge of the filtered serial clock (note 1).
–	S1INT.6 to 0	X = undefined during read, don't care during write.

Note

1. If SI = 1 is detected during a transfer this can be a "spurious START" error condition. If no transfer is taking place the SI = 1 is a START from an external master. Provided the bits EA and ES1 in IE SFR are set, SI then generates an interrupt so that a slave address receive routine can be started. SI can be cleared by accessing the S1BIT register or by writing 00 to S1INT. Also after reception of a START condition, the LOW period of the clock pulse is stretched, suspending the serial transfer to allow the software to take action. This clock stretching is ended by a read or write access to S1BIT.

6.8.2 S1BIT: Single-bit Data Register with I²C-bus Auto-clock

This register is located at address D9H.

S1BIT SFR (D9H) (note 1).

bit-7	6	5	4	3	2	1	0
READ							
SDI	0	0	0	0	0	0	0
WRITE							
SDO	X	X	X	X	X	X	X

Note

1. A read or write access of the **S1BIT SFR** clears SI, CLH, RBF and WBF. It starts the auto-clock if SCO = 0.

Table 12 Description of S1BIT bits.

MNEMONIC	BIT	FUNCTION
SDO/SDI	S1BIT.7	**Serial Data Output (SDO) and the filtered Serial Data Input (SDI).** SDI data is latched on the rising edge of the filtered serial clock. The physical memory of S1BIT.7 and S1SCS.7 is identical, but S1BIT is not bit-addressable.
–	S1BIT.6 to 0	X = don't care.

6.8.3 Reading or Writing the S1BIT SFR

Reading or writing the S1BIT SFR starts an I²C-bus bit I/O sequence: some flags are cleared (SI, CLH, RBF, WBF), clock stretching is finished and the auto-clock is started. An auto-clock pulse is "OR-ed" with SCO and thus will be output only if the SCO flag has been set to logic 0. SCO = 1 inhibits the auto-clock start, so a dummy read or write of S1BIT SFR can be used to finish clock stretching and clear SI, CLH, RBF and WBF if the auto-clock is not used.

The auto-clock is an active HIGH SCL pulse that starts 28 XTAL clock periods after the SDI read or SDO write via S1BIT. The duration of the auto-clock pulse is 100 XTAL clock periods. If the SCL line is kept LOW by any device that wants to hold up the bus transfer, the auto-clock counter waits after 20 XTAL clock periods so that the auto-clock pulse length will be at least 80 XTAL clock periods (5 µs at f_{osc} = 16 MHz).

Every bit I/O should be followed by a RBF or WBF bit test. A bit transfer has been finished successfully if after reading a bit the RBF flag is 1 or after writing a bit the WBF flag is 1. When after reading a bit the RBF flag is still logic 0, the bus status just before the S1SCS status read can be determined as follows:

- If CLH = 0 then a bus device is still stretching the clock.
- If SCI = 1 while CLH = 1 then the SCL pulse is not finished.
- If BB = 0 there has been a STOP condition.

When after writing a bit the WBF flag is still 0 and none of the 3 status conditions mentioned for RBF are found then a "bus arbitration lost" condition will be the cause. This can be determined also from the states of the received bit and the last transmitted bit: "arbitration loss" if SDO = 1 and SDI = 0.

6.8.4 S1SCS: Control and Status Register for the I²C-bus

This register is located at address D8H.

S1SCS SFR (D8H).

bit-7	6	5	4	3	2	1	0
READ (note 1)							
SDI	SCI	CLH	BB	RBF	WBF	STR	ENS
WRITE (notes 2, 3 and 4)							
SDO	SCO	CLH	X	X	X	STR	ENS

Notes

1. **SDI and SCI bits:** read-modify-write operations like "SETB bit" or "CLR bit" access SDO and SCO for reading and writing.
2. **CLH bit:** writing a logic 0 clears this bit, writing a logic 1 has no effect.
3. **RBF and WBF bits:** writing a logic 0 to CLH also clears these bits.
4. X = don't care.

Table 13 Description of S1SCS bits.

MNEMONIC	BIT	FUNCTION
SDO/SDI	S1SCS.7	**Serial Data Output and the filtered Serial Data Input.** SDI data is latched on the rising edge of the filtered serial clock. S1SCS.7 accesses the same memory locations as S1BIT.7. Access of the data bit via S1SCS will not start an auto-clock pulse.
SCO/SCI	S1SCS.6	**Serial Clock Output and the filtered Serial Clock Input.** Serial clock output SCO is "OR-ed" with the auto-clock. If SCO = 1 the auto-clock output is inhibited. The internal clock stretching logic and external devices can pull the SCL line LOW. If the auto-clock is not used, the SCL line has to be controlled by setting SCO = 1, waiting for CLH = 1 and setting SCO = 0 after the specified SCL HIGH time. (Because of the input filter, CLH will be set at least 8 XTAL clock periods after the SCL LOW-to-HIGH transition.)
CLH	S1SCS.5	**Serial Clock LOW-to-HIGH transition flag:** set on a rising edge of the filtered serial clock. CLH = 1 indicates that, since the last CLH reset, a new valid data bit has been latched in SDI. CLH can be reset by writing a logic 0 to S1SCS.5 or by a read/write of S1BIT. Clearing CLH also clears RBF and WBF.
BB	S1SCS.4	**Bus Busy flag:** indicating that there has been a START condition that was not yet followed by a STOP condition.
RBF	S1SCS.3	**Read Bit Finished flag:** indicating a successful bit read. RBF = 1 implies the following conditions: – CLH = 1: SCL had a rising edge – SCI = 0: the SCL pulse has finished – SI = 0: no START condition occurred – BB = 1: no STOP condition occurred The RBF flag can be cleared by clearing the CLH flag.

MNEMONIC	BIT	FUNCTION
WBF	S1SCS.2	**Write Bit Finished flag:** indicating a successful bit write. The same conditions as for RBF are true and also no "arbitration loss" condition occurred. Arbitration is lost if a HIGH level transmitted (SDO = 1) is overruled by an external I²C-bus device applying a LOW level to the SDA line. The WBF flag can be cleared by clearing the CLH flag.
STR	S1SCS.1	**STRetch control flag.** STR = 1 enables stretching of all SCL LOW periods. This allows the processor in I²C-bus slave mode to react on a fast master. The STR flag remains set until cleared by writing a 0 to S1SCS.1. **The STretch (ST)** flag (not readable) pulls the serial clock LOW while ST = 1. The ST flag is set on the falling edge of the filtered serial clock if STR = 1. It is also set after reception of a START condition, regardless of the STR contents. ST is cleared with a read or write of S1BIT.
ENS	S1SCS.0	**ENable Serial I/O flag.** ENS = 1 enables the START detection and clock stretching logic. ENS = 0 can be used to switch off the I²C-bus hardware. Note that the SDO and SCO control flags must be set to logic 1 before ENS is set to avoid pulling SCL or SDA lines to logic 0.

6.9 Idle and Power-down Operation

Idle mode operation permits the interrupt, serial ports and timer blocks to function while the CPU is halted. The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode:

- Timer 0, Timer 1, Timer 2, Watchdog Timer
- UART, I²C-bus Interface
- External interrupt.

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register (see Figure 12).

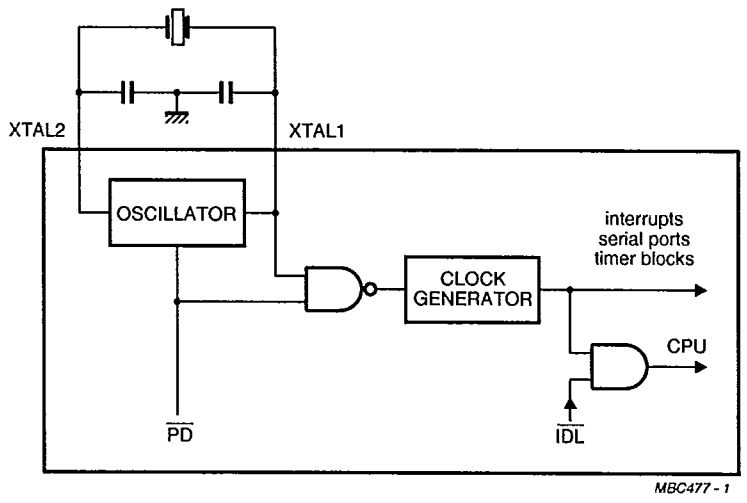


Fig.12 Internal Idle and Power-down clock configuration.

6.10 PCON: Power Control Register

Special modes are activated by software via the PCON SFR. This register is located at address 87H. PCON is not bit addressable. The reset value of PCON is 0X0X0000B.

PCON SFR (87H).

bit-7	6	5	4	3	2	1	0
SMOD	–	RFI	–	GF1	GF0	PD	IDL

Table 14 Description of PCON bits.

MNEMONIC	BIT	FUNCTION
SMOD	PCON.7	**Double Baud rate bit**: when set to logic 1 the baud rate is doubled when Timer 1 is used to generate baud rate, and the Serial Port is used in modes 1, 2 or 3.
–	PCON.6	**Reserved** for future use (note 1).
RFI	PCON.5	**Reduced Radio Frequency Interference** bit. When set to logic 1 the toggling of ALE pin is suppressed. Then the pin output level is LOW. This bit is cleared on RESET (see also section 5).
–	PCON.4	**Reserved** for future use.
GF1	PCON.3	**General-purpose** flag bit.
GF0	PCON.2	**General-purpose** flag bit.
PD	PCON.1	**Power-down bit**: setting this bit activates Power-down mode (note 2).
IDL	PCON.0	**Idle mode bit**: setting this bit activates the Idle mode.

Notes

1. User software should not write 1s to reserved bits. These bits may be used in future 80C51 family products to invoke new features. The reset or inactive value of the new feature bit will be logic 0 and the active value of the new feature bit will be logic 1. The value read from a reserved bit is indeterminate.
2. If logic 1s are written to PD and IDL at the same time, PD takes precedence.

6.11 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 15.

There are three ways to terminate the Idle mode:

1. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. 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 the 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.
2. 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 oscillator periods) to complete the reset operation.
3. The third way of terminating the Idle mode is by internal watchdog reset.

6.12 Power-down Mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. The oscillator is stopped. Only the content of the on-chip RAM is preserved. The Special Function Registers (SFRs) are not saved.

Note that the Power-down mode also can 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 or in addition by any one of the two external interrupts, $\overline{INT0}$ and $\overline{INT1}$ (see Section 6.13). A reset generated by the WDT terminates the Power-down mode in the same way as an external RESET.

The status of the external pins during Power-down mode is shown in Table 15. If the Power-down mode is activated executing external Program Memory, the port data that is held in the P2 SFR is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Figure 7).

Table 15 Status of the external pins during Idle and Power-down modes (note 1).

MODE	MEMORY	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3
Idle	internal	HIGH	HIGH	port data	port data	port data	port data
Idle	external	HIGH	HIGH	floating	port data	address	port data
Power-down	internal	LOW	LOW	port data	port data	port data	port data
Power-down	external	LOW	LOW	floating	port data	port data	port data

Note

1. Ports P1.7 and P1.6 if selected, function as SDA and SCL respectively in the Idle mode.

6.13 Wake-up from Power-down Mode

The Power-down mode of the P8xCE528 can also be terminated by any one of the two external interrupts, $\overline{INT0}$ and $\overline{INT1}$. A termination with an external interrupt does not affect the internal data memory and does not affect the Special Function Registers (SFRs). This gives the possibility to exit Power-down without changing the port output levels. To terminate the Power-down mode with an external interrupt, $\overline{INT0}$ and $\overline{INT1}$ must be switched to be level-sensitive and must be enabled. The external interrupt input signal $\overline{INT0}$ and $\overline{INT1}$ must be kept LOW until the oscillator has restarted and stabilized (see Figure 13).

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 wake-up and the completion of the interrupt routine.

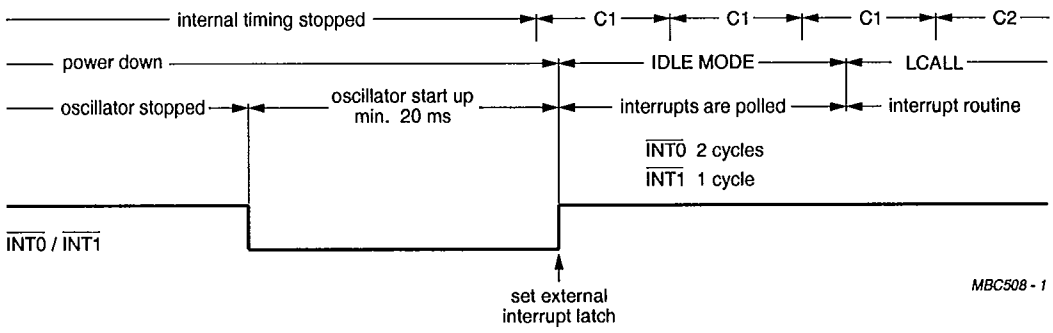


Fig.13 Wake up by external interrupt input.

6.14 Interrupt System

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, a multi-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is in the range from 2.25 µs to 6 µs when using a 16 MHz crystal. The latency time strongly depends on the sequence of instructions executed directly after an interrupt request (see Figure 14).

The P8xCE528 acknowledges interrupt requests from 7 sources as follows:

- $\overline{\text{INT0}}$ external interrupt
- $\overline{\text{INT1}}$ external interrupt
- Timer 0 internal counter
- Timer 1 internal counter
- Timer 2 internal counter / EXF2 external interrupt
- I²C-bus serial I/O interrupt
- UART serial I/O port interrupt.

The External Interrupts $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$ can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in TCON SFR. 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 $\overline{\text{INTX}}$ 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 "wired-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 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 Timer 2 Interrupt is generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware. In fact the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and the bit will have to be cleared by software.

An additional (third) external interrupt is available, if Timer 2 is not used as timer/counter or if Timer 2 is used in baud rate generator mode. That external interrupt 2 is falling edge triggered. It shares the Timer 2 interrupt vector, interrupt enable and interrupt priority bits. If bit T2CON.3/EXEN2 = 1, a HIGH-to-LOW transition at pin P1.1/T2EX sets the interrupt request flag T2CON.6/EXF2 and can be used to generate an external interrupt.

The I²C-bus interrupt is generated by SI in S1INT. This flag has to be cleared by software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though they had been set or cleared by hardware, with the exception of the I²C-bus interrupt request flag SI, which cannot be set by software. Thus, interrupts can be generated or pending interrupts can be cancelled in software.

6.14.1 Setting or clearing the Interrupt Enable Register

Each interrupt source can be individually enabled or disabled by setting or clearing a corresponding bit in the interrupt enable SFR IE. All interrupt sources can also be globally enabled or disabled by setting or clearing bit EA in the IE register.

6.14.2 Interrupt priority structure

Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined in the SFR IP.

Interrupt priority levels are as follows:

- logic 0 = low priority
- logic 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 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 18.

6.14.3 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 a 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 IP or IE until at least one other instruction has been subsequently executed.)

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 hard-ware 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 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 software. The LCALL pushes the contents of the program counter on to the stack (but 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 18.

Execution proceeds from vector address until the RETI instruction is encountered. The RETI instruction clears the "priority level active" flip-flop that was set when the 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.

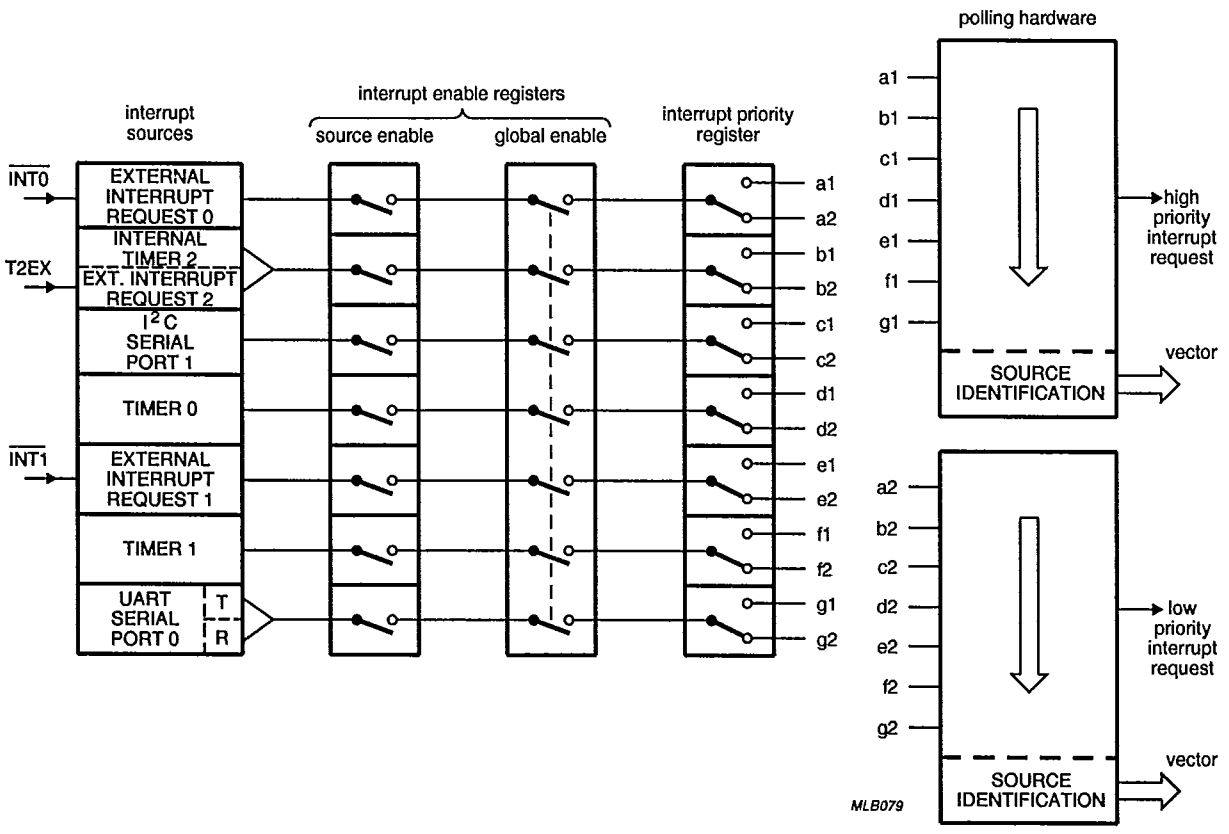


Fig.14 Interrupt System.

6.14.4 IE: Interrupt Enable Register

This register is located at address A8H.

IE SFR (A8H).

bit-7	6	5	4	3	2	1	0
EA	ES1	ET2	ES	ET1	EX1	ET0	EX0

Table 16 Description of IE bits.

MNEMONIC	BIT	FUNCTION
EA	IE.7	General enable/disable control: 0 = NO interrupt is enabled. 1 = ANY individually enabled interrupt will be accepted.
ES1	IE.6	Enable bit-level I²C-bus I/O interrupt.
ET2	IE.5	Enable Timer 2 interrupt.
ES	IE.4	Enable Serial Port interrupt.
ET1	IE.3	Enable Timer 1 interrupt.
EX1	IE.2	Enable External interrupt 1.
ET0	IE.1	Enable Timer 0 interrupt.
EX0	IE.0	Enable External interrupt 0.

6.14.5 IP: Interrupt Priority Register

This register is located at address B8H.

IP SFR (B8H).

bit-7	6	5	4	3	2	1	0
–	PS1	PT2	PS	PT1	PX1	PT0	PX0

Table 17 Description of IP bits.

MNEMONIC	BIT	FUNCTION
–	IP.7	Reserved for future use.
PS1	IP.6	Bit-level I²C-bus interrupt priority level.
PT2	IP.5	Timer 2 interrupt priority level.
PS	IP.4	Serial Port interrupt priority level.
PT1	IP.3	Timer 1 interrupt priority level.
PX1	IP.2	External interrupt 1 priority level.
PT0	IP.1	Timer 0 interrupt priority level.
PX0	IP.0	External interrupt 0 priority level.

6.14.6 Interrupt Vectors

The interrupt vectors are listed in Table 18.

Table 18 Interrupt vectors.

NUMBER	SOURCE	PRIORITY WITHIN LEVEL	VECTOR ADDRESS
1	IE0	(highest)	0003H
2	TF2 + EXF2	–	002BH
3	SI (I^2C-bus)	–	0053H
4	TF0	–	000BH
5	IE1	–	0013H
6	TF1	–	001BH
7	RI + TI	(lowest)	0023H

6.15 Oscillator Circuit

The oscillator circuit of the P8xCE528 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between the XTAL 1 and XTAL 2 is basically an invertor 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. XTAL 1 is the high gain amplifier input, and XTAL 2 is the output (see Figure 15). If XTAL 1 is driven from an external source, XTAL 2 must be left open-circuit (see Figure 16).

6.16 Reset Circuit

The reset circuitry for the P8xCE528 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. 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.

With the P8xCE528, the RST line can also be pulled HIGH internally by a pull-up transistor activated by the WatchDog Timer (WDT) T3. The length of the reset pulse from T3 is 16 x 2048 cycles of the on-chip watchdog oscillator. If the WDT is also used to reset external devices, the usual capacitor arrangement should not be connected to RST pin. Instead, an extra circuit should be used to perform the Power-on Reset operation. However, the internal reset is forced after a T3 overflow, independent of the external level of the RST pin (see Figure 17).

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as shown by Table 19.

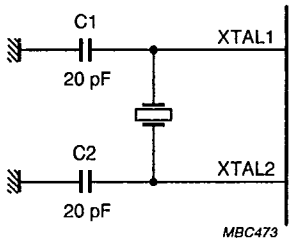


Fig.15 P8xCE528 oscillator circuit.

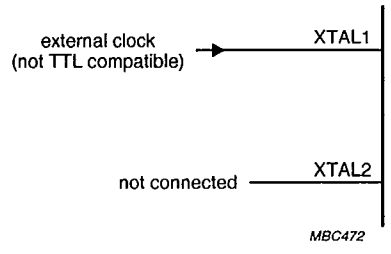


Fig.16 Driving the P8xCE528 from an external source.

Table 19 Internal registers status after a RESET (note 1).

REGISTER	CONTENTS
ACC	00H
B	00H
DPH, DPL	00H
IE	0000 0000B
IP	X000 0000B
PCH, PCL	00H
PCON	0X0X 0000B
PSW	00H
P0 to P3	FFH
SBUF	XXH
SCON	00H
SP	07H
TCON	00H
TMOD	00H
TH0, TL0	00H
TH1, TL1	00H
T2CON	00H
TH2, TL2	00H
RCAP2H, RCAP2L	00H
S1BIT	X000 0000B
S1INT	0XXX XXXXB
S1SCS	XXX0 0000B
T3	00H
WDCON	A5H
FMCON	000X 0000B

Note

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

6.17 Power-on Reset

When V_{DD} is turned on, and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2.2 µF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles, or 16 x 2048 cycles of the on-chip watchdog oscillator if it is running, whichever is longer (see Figure 18).

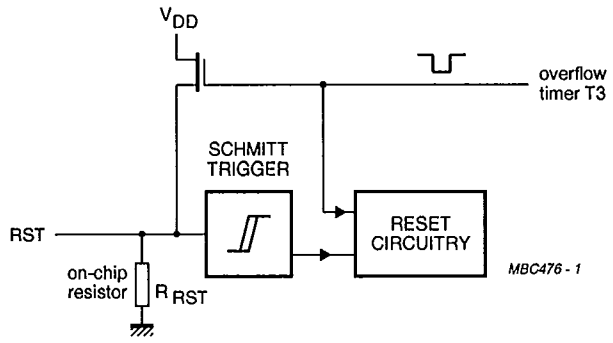


Fig.17 On-chip reset configuration.

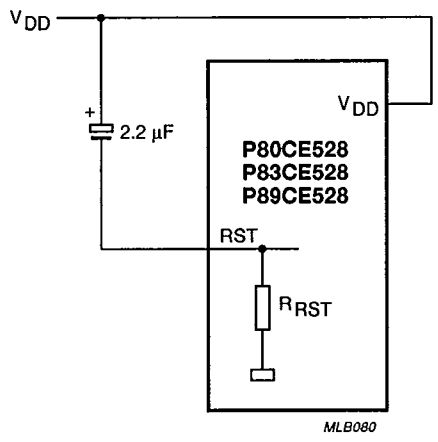


Fig.18 Power-on Reset.

7 INSTRUCTION SET

The P8xCE528 uses the powerful instruction set of the 80C51. Additional Special Function Registers are incorporated to control the on-chip peripherals. A summary of the instruction set is given in Table 20.

The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 16 MHz oscillator, 64 instructions execute in 750 ns and 45 instructions execute in 1.5 µs. Multiply and divide instructions execute in 3 µs.

7.1 Addressing modes

Most instructions have a "destination source" field that specifies the data type, addressing modes and operantes 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:

1. Register Addressing
   - R0 to R7 (4 banks)
   - A,B,C (bit), AB (2 bytes), DPTR (double byte)
2. Direct Addressing
   - lower 128 bytes of internal Main RAM (including the 4 R0 to 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
3. Register-Indirect Addressing
   - internal Main RAM (@R0, @R1, @SP, [PUSH/POP])
   - internal Auxiliary RAM (@R0, @R1, @DPTR)
   - external Data Memory (@DPTR)
4. Immediate Addressing
   - Program Memory (in-code 8 bit or 16 bit constant)
5. 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.

Table 20 Instruction set summary

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Arithmetic operation					
ADD	A,Rr	Add register to A	1	1	2*
ADD	A,direct	Add direct byte to A	2	1	25
ADD	A,@Ri	Add indirect RAM to A	1	1	26, 27
ADD	A,#data	Add immediate data to A	2	1	24
ADDC	A,Rr	Add register to A with carry flag	1	1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2	1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2	1	34
SUBB	A,Rr	Subtract register from A with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2	1	94
INC	A	Increment A	1	1	04
INC	Rr	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 A	1	1	14
DEC	Rr	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 A	1	1	D4

Instruction set description (continued).

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Logic operations					
ANL	A,Rr	AND register to A	1	1	5*
ANL	A,direct	AND direct byte to A	2	1	55
ANL	A,@Ri	AND indirect RAM to A	1	1	56, 57
ANL	A,#data	AND immediate data to A	2	1	54
ANL	direct,A	AND A to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rr	OR register to A	1	1	4*
ORL	A,direct	OR direct byte to A	2	1	45
ORL	A,@Ri	OR indirect RAM to A	1	1	46, 47
ORL	A,#data	OR immediate data to A	2	1	44
ORL	direct,A	OR A to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rr	Exclusive-OR register to A	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2	1	64
XRL	direct,A	Exclusive-OR A to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear A	1	1	E4
CPL	A	Complement A	1	1	F4
RL	A	Rotate A left	1	1	23
RLC	A	Rotate A left through the carry flag	1	1	33
RR	A	Rotate A right	1	1	03
RRC	A	Rotate A right through the carry flag	1	1	13
SWAP	A	Swap nibbles within A	1	1	C4

Instruction set description (continued).

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Data transfer					
MOV	A,Rr	Move register to A	1	1	E*
MOV	A,direct**	Move direct byte to A	2	1	E5
MOV	A,@Ri	Move indirect RAM to A	1	1	E6, E7
MOV	A,#data	Move immediate data to A	2	1	74
MOV	Rr,A	Move A to register	1	1	F*
MOV	Rr,direct	Move direct byte to register	2	2	A*
MOV	Rr,#data	Move immediate data to register	2	1	7*
MOV	direct,A	Move A to direct byte	2	1	F5
MOV	direct,Rr	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 A 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,#data 16	Load data pointer with a 16-bit constant	3	2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1	2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1	2	83
MOVX	A,@Ri	Move AUX-RAM (8-bit address) to A	1	2	E2, E3
MOVX	A,@DPTR	Move AUX-RAM or external RAM (16-bit address) to A	1	2	E0
MOVX	@Ri,A	Move A to AUX-RAM (8-bit address)	1	2	F2, F3
MOVX	@DPTR,A	Move A to AUX-RAM or external RAM (16-bit address)	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,Rr	Exchange register with A	1	1	C*
XCH	A,direct	Exchange direct byte with A	2	1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1	1	C6, C7
XCHD	A,@Ri	Exchange LOW-order digit indirect RAM with A	1	1	D6, D7

Note: ** MOV A,ACC is not permitted.

Instruction set description (continued).

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
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 address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if carry flag is not set	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#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	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

Notes to Instruction Set.

MNEMONIC	DESCRIPTION
Data addressing modes	
Rr	working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	indirect internal RAM location addressed by register R0 or R1 of the selected register bank.
#data	8-bit constant included in instruction.
#data 16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 kbyte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 kbyte 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.
Hexadecimal opcode cross-reference	
*	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 21 Instruction map.

first hexadecimal character of opcode ↓ / second hexadecimal character of opcode →	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr 11	LJMP addr 16	RR A	INC A	INC dir	INC @ Ri		0	1	2	3	4	5	6	7
1	JBC bit, rel	ACALL addr 11	LCALL addr 16	RRC A	DEC A	DEC dir	DEC @ Ri		0	1	2	3	4	5	6	7
2	JB bit, rel	AJMP addr 11	RET	RL A	ADD A, # data	ADD A, dir	ADD A, @ Ri		0	1	2	3	4	5	6	7
3	JNB bit, rel	ACALL addr 11	RETI	RLC A	ADDC A, # data	ADDC A, dir	ADDC A, @ Ri		0	1	2	3	4	5	6	7
4	JC rel	AJMP addr 11	ORL dir, A	ORL dir, # data	ORL A, # data	ORL A, dir	ORL A, @ Ri		0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr 11	ANL dir, A	ANL dir, # data	ANL A, # data	ANL A, dir	ANL A, @ Ri		0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr 11	XRL dir, A	XRL dir, # data	XRL A, # data	XRL A, dir	XRL A, @ Ri		0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr 11	ORL C, bit	JMP @ A+DPTR	MOV A, # data	MOV dir, # data	MOV @ Ri, # data		0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr 11	ANL C, bit	MOVC A, @ A+PC	DIV AB	MOV dir, dir	MOV dir, @ Ri		0	1	2	3	4	5	6	7
9	MOV DPTR, # data 16	ACALL addr 11	MOV bit, C	MOVC A, @ A+DPTR	SUBB A, # data	SUBB A, dir	SUBB A, @ Ri		0	1	2	3	4	5	6	7
A	ORL C, / bit	AJMP addr 11	MOV bit, C	INC DPTR	MUL AB		MOV @ Ri, dir		0	1	2	3	4	5	6	7
B	ANL C, / bit	ACALL addr 11	CPL bit	CPL C	CJNE A, # data, rel	CJNE A, dir, rel	CJNE @ Ri, # data, rel		0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr 11	CLR bit	CLR C	SWAP A	XCH A, dir	XCH A, @ Ri		0	1	2	3	4	5	6	7
D	POP dir	ACALL addr 11	SETB bit	SETB C	DA A	DJNZ dir, rel	XCHD A, @ Ri		0	1	2	3	4	5	6	7
E	MOVX A, @ DPTR	AJMP addr 11	MOVX A, @ Ri		CLR A	MOV A, dir	MOV A, @ Ri		0	1	2	3	4	5	6	7
F	MOVX @ DPTR, A	ACALL addr 11	MOVX @ Ri, A		CPL A	MOV dir, A	MOV @ Ri, A		0	1	2	3	4	5	6	7

* MOV A, ACC is not a valid instruction.

MLA632 - 1

8 FLASH EEPROM

8.1 General

The FEEPROM (Flash Memory) can be read and written byte-wise. Full erase, block erase, and page erase will erase 32 kbyte, 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 provide the following user friendly interface:

- 32 kbyte electrically erasable program memory with block and page-erase option
- Internal fixed Boot ROM
- Up to 32 kbyte external program memory in combination with the internal FEEPROM ($\overline{EA}$ = 1)
- Up to 64 kbyte external program memory if the internal program memory is switched off ($\overline{EA}$ = 0).

8.2 FEEPROM 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 kbyte) 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 all 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

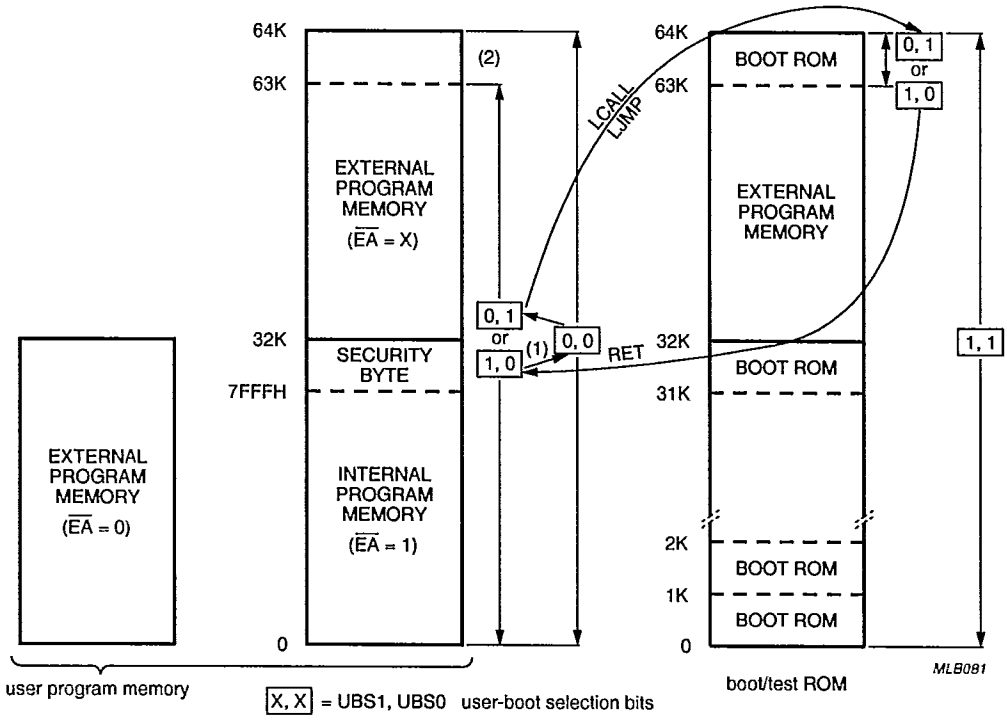
Figure 19 illustrates the memory map of the user program memory and the boot ROM. They are located in the same program memory address space. Two bits UBS1 and UBS0 of the FEEPROM control special function register FMCON select between the two memory blocks.

8.3.1 User Program Memory Selection

If UBS1 and UBS0 are both logic 0, then the user program memory is mapped into the 64K 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.

8.3.2 Boot ROM Selection

If UBS1 and UBS0 are both logic 1 (boot mode) the boot ROM is mapped into the 64K program memory space and the user program memory cannot be selected. This is the situation after a reset when during reset $\overline{PSEN}$ and $\overline{EA}$ are pulled down while ALE stays HIGH. Program execution starts at 0000H of the boot ROM. The boot ROM size is 1 kbyte. Besides the serial in-circuit programming routines 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 63K and 64K).



(1) UBS1 is cleared by hardware if program address is < 63K and UBS0 = 0.

(2) With the program address between 63K and 64K in the user program no call or jump to the boot ROM is allowed.

Fig.19 FEEPROM Program memory map.

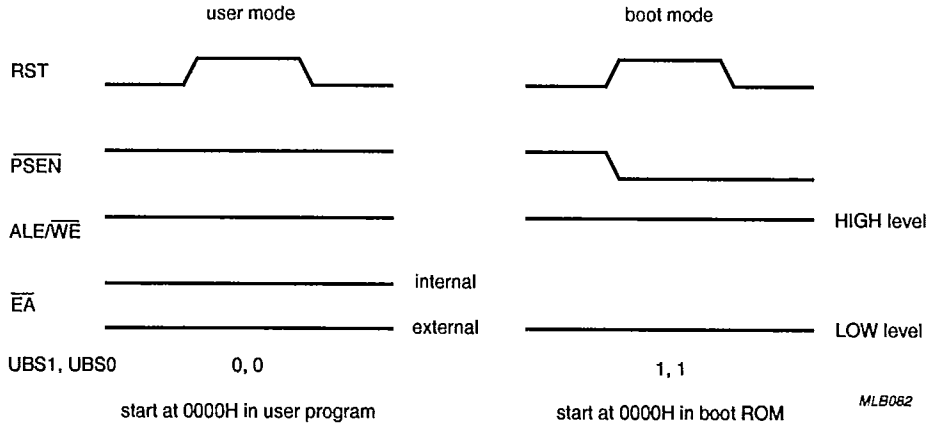


Fig.20 FEEPROM operational modes.

8.3.3 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 or 1 0. Then in the program memory address space between 0 and 63K the user program memory is selected and in the memory space between 63K and 64K the boot ROM is selected.

To switch from user program memory to boot ROM first UBS0 must be set (UBS1 stays 0) and a jump or call instruction to a location > 63K must be executed. At the moment of crossing the 63K address border by a call or jump instruction the switching between user and boot memory is performed without timing problems.

Switching from boot ROM to user program memory is performed as follows: UBS1 is set, UBS0 is cleared and a return instruction to a location < 63K is executed. At the moment of crossing the 63K address border by the return instruction the switching from boot ROM to user memory (internal or external) is performed. After crossing the 63K address border UBS1 is immediately cleared by hardware and the total 64K memory space is mapped as user program memory. By clearing UBS1 by hardware, no special requirements to the user program are necessary to do that after a boot routine 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 63K and 64K of the user program memory because the UBS bits must stay logic 0 in this range and no 63K address crossing would take place. This restriction can be avoided if the memory switching is always done by a subroutine in the address range between 0 and 63K.

8.3.4 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 63K and 64K. 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, blank-check, verify, high voltage control, error message, interrupt disabling/enabling and return to the user program memory. It also contains the serial communication routine.

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

8.3.5 FMCON: Control Register

FMCON SFR (FBH).

bit-7	6	5	4	3	2	1	0
UBS1	UBS0	HV	–	FCB3	FCB2	FCB1	FCB0

Table 22 Description of FMCON bits.

MNEMONIC	BIT	FUNCTION
UBS1	FMCON.7	User/boot selection bit.
UBS0	FMCON.6	User/boot selection bit.
HV	FMCON.5	HIGH voltage indication bit. Read only, is logic 1 as long as the HIGH voltage for an erase or write operation is present.
–	FMCON.4	Reserved for future use. A write operation must write 0 to this bit location.
FCB3	FMCON.3	Function code bit.
FCB2	FMCON.2	Function code bit.
FCB1	FMCON.1	Function code bit.
FCB0	FMCON.0	Function code bit.

Table 23 FMCON UBS1 and UBS0 bit modes.

UBS1	UBS0	MODE
0	0	User memory mapped from 0 to 64K.
0	1	User memory mapped from 0 to 63K. Boot ROM mapped from 63K to 64K.
1	0	User memory mapped from 0 to 63K, but UBS1 bit cleared by hardware in this user address range. Boot ROM mapped from 63K to 64K.
1	1	Boot ROM mapped from 0 to 64K.

Table 24 FMCON FCB-bit function codes.

FCB3	FCB2	FCB1	FCB0	FUNCTION CODE
0	0	0	0	Value after reset.
0	1	0	1	Byte write or byte read (verify).
1	1	0	0	Page erase (32 byte boundaries).
0	0	1	1	Block erase (256 byte boundaries).
1	0	1	0	Full erase (32 kbyte).

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 FCB3-0. Boot ROM and external program memory instructions cannot change FCB 3 to 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 can not 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.

Table 25 lists the boot ROM user routines, which can be called by the user program. The contents of FMCON, A and DPTR before calling is described by (IN) and by (OUT) after return. The boot ROM user routines do not change other registers or data memory.

Table 25 Boot ROM user routines (notes 1 and 2).

BOOT ROM ROUTINE	CALL ADDRESS	FMCON (IN)	FMCON (OUT)	A (IN)	A (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 (note 3)	PAGE ADDRESS (note 4)
BLOCK_ERASE	FFA5H	43H	13H	XXH	02H	BLOCK ADDRESS (note 5)	BLOCK ADDRESS (note 6)
FULL_ERASE	FFA0H	4AH	1AH	XXH	0AH	XXXXH	0018H

Notes

1. X = don't care or not defined.
2. V = verified byte (read back).
3. = 5 LSB's of DPTR are don't care.
4. = 5 LSB's of DPTR are 0.
5. = 8 LSB's of DPTR are don't care.
6. = 8 LSB's of DPTR contain 08H.

8.3.6 User software example of FEEPROM page erase routine

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 (5 ms)
SETB  EA ; enable interrupts again
```

8.3.7 User software example of FEEPROM byte write routine

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 (2.5 ms)
SETB  EA ; enable interrupts again
XRL  A, R5 ; compare the "read back" byte
JNZ  . . .; jump if verify error
```

8.4 Security Feature

The security feature (security mode) protects against software piracy by limiting the read/write access to the contents of the FEEPROM. The security feature is activated or de-activated by a security code written into a security byte. If the security feature is de-activated, then there are no access restrictions to the FEEPROM.

The security byte is located in the highest address location 7FFFH of the FEEPROM. The security byte is functionally the same as all other bytes of the FEEPROM.

The security byte should be written 01010000B to activate and 00H or FFH to de-activate the security feature. The security code is chosen in such a way that single bit failures will not change the security mode.

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 to 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 to 3 of FMCON. At the end of a full erase operation the security feature is de-activated. Parallel programming and verify is also 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 only 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, $\overline{PSEN}$ = 0, $\overline{EA}$ = 1. The XTAL clock must have a frequency between 4 MHz and 6 MHz. Table 26 shows the logic levels for programming, erasing, verifying and read signature.

Table 26 Parallel programming modes.
(notes 1 and 2)

MODE	ALE/$\overline{WE}$	P2.7	P2.6	P3.7	P3.6
Full Erase	note 3	1	1	0	1
Program FEEPROM	note 4	1	0	1	1
Verify FEEPROM	1	0	0	1	1
Read signature	1	0	0	0	0

Notes

1. ALE/$\overline{WE}$ Write enable signal (program/erase), active LOW.
2. P2.7, P2.6, P3.7, P3.6 Output enable signals for verify/read modes, active LOW.
3. One 5 ms LOW pulse.
4. One 2.5 ms LOW pulse.

Table 27 Data and address bits.

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

8.5.1 89CE528 SIGNATURE BYTES

The 89CE528 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. The values are given in Table 28:

Table 28 Signature bytes.

ADDRESS	CONTENT	MEANING
30H	15H	Philips
31H	9FH	89CE528

8.5.2 89CE528 SECURITY BYTE

The 89CE528 has a security byte in location 7FFFH (the highest address) of the FEEPROM, which should be programmed to 01010000B to activate the security feature.

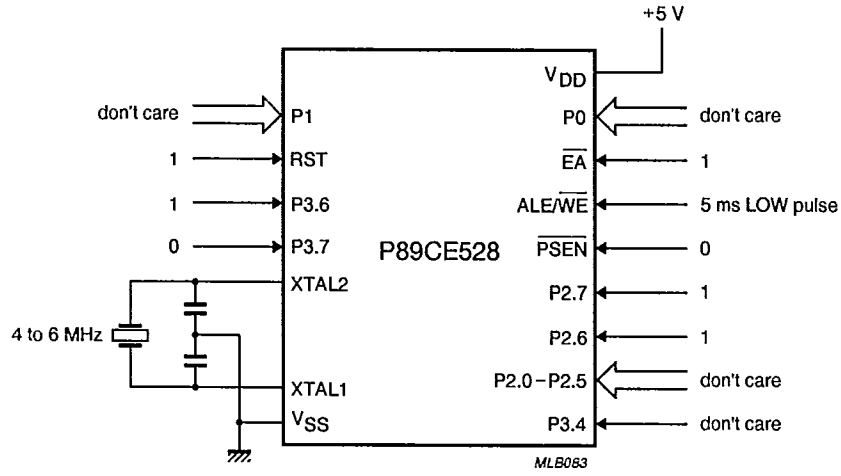


Fig.21 Erase configuration.

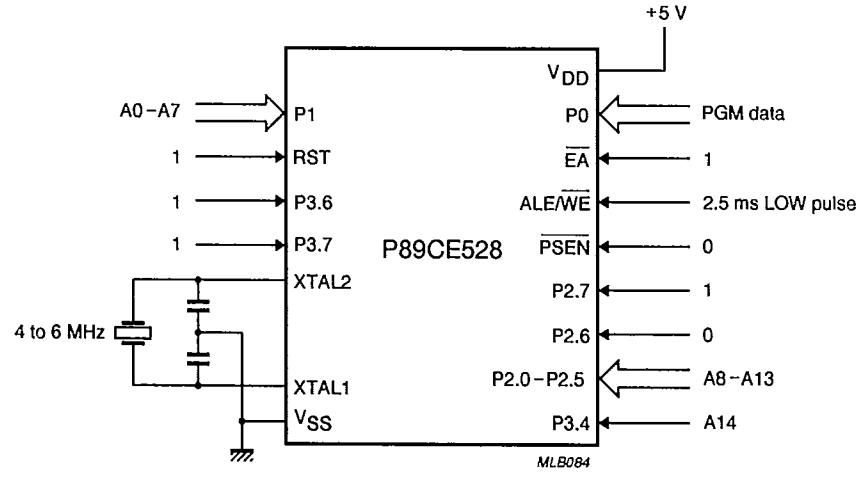


Fig.22 Programming configuration.

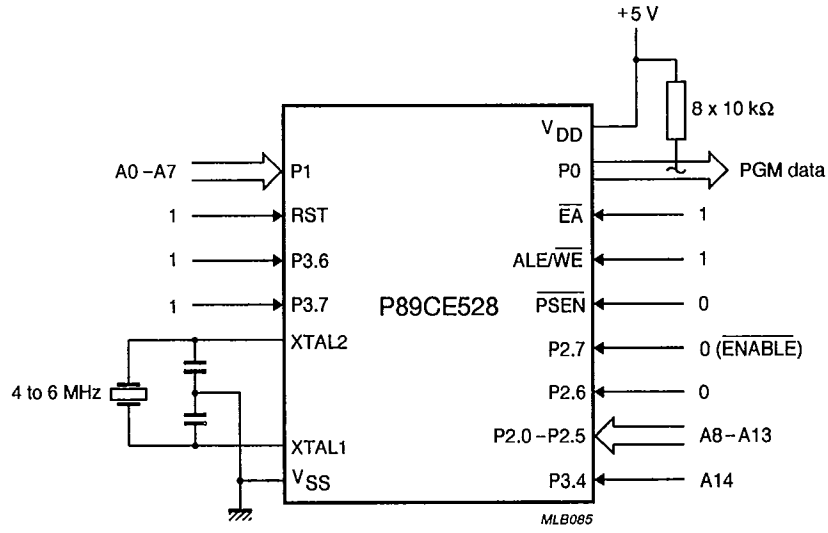
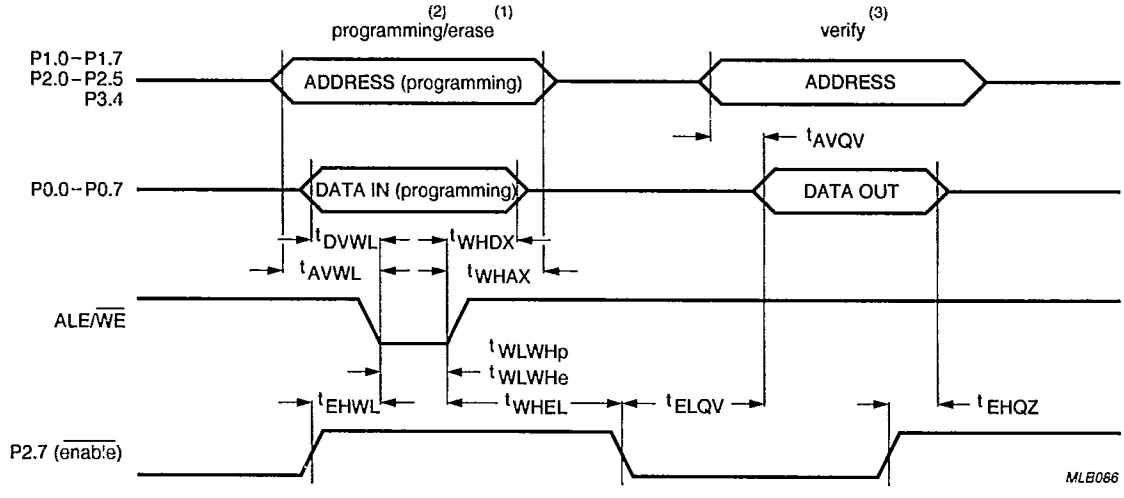


Fig.23 Programming verification.

9 FEEPROM PROGRAMMING/ERASE AND VERIFICATION CHARACTERISTICS

T_{amb} = −40 °C to +85 °C; V_{DD} = 5 V ±10%; V_{SS} = 0 V (see Figure 24).

SYMBOL	PARAMETER	MIN.	MAX.	UNITS
$1/t_{CLCL}$	oscillator frequency	4	6	MHz
t_{AVWL}	address setup to $\overline{WE}$ LOW	48 t_{CLCL}	–	
t_{WHAX}	address hold after $\overline{WE}$ HIGH	48 t_{CLCL}	–	
t_{DVWL}	data setup to $\overline{WE}$ LOW	48 t_{CLCL}	–	
t_{WHDX}	data hold after $\overline{WE}$ HIGH	48 t_{CLCL}	–	
t_{EHWL}	P2.7 ($\overline{Enable}$) HIGH to $\overline{WE}$ LOW	48 t_{CLCL}	–	
t_{WHEL}	$\overline{WE}$ HIGH to P2.7 ($\overline{Enable}$) LOW	48 t_{CLCL}	–	
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	–	48 t_{CLCL}	
t_{ELQV}	P2.7 ($\overline{Enable}$) LOW to data valid	–	48 t_{CLCL}	
t_{EHQZ}	Data float after P2.7 ($\overline{Enable}$) HIGH	0	48 t_{CLCL}	



(1) For erase conditions see Figure 21.

(2) For program conditions see Figure 22.

(3) For verify conditions see Figure 23.

Fig.24 FEEPROM Program/Erase and Verification waveforms.

9.1 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 2 kΩ to V_{SS}). The two UBS bits are set to logic 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.

The receive and transmit channel have the same baud rate. The boot routine is totally in software without the use of timers or UART. It 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 precede the first (:) character. Programming is only started if the first received record has the right type indication ($\overline{TT}$). If the security feature is activated 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 the 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.

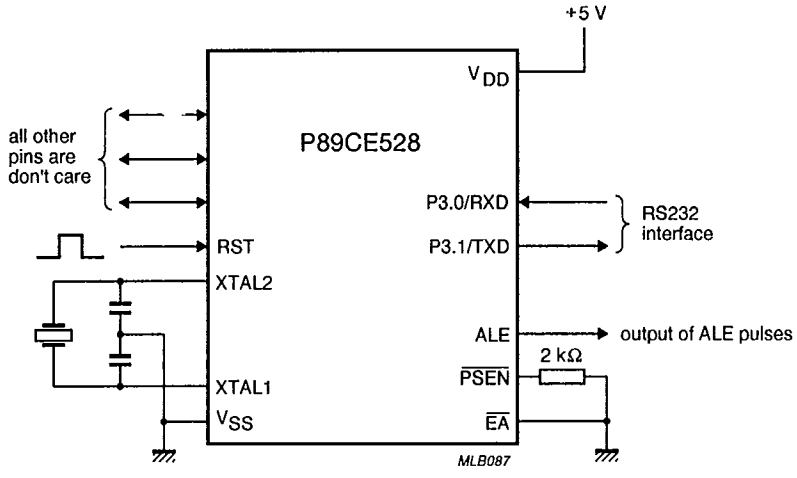


Fig.25 Serial programming (Boot Mode) configuration.

9.2 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.

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. The next baud rates will be detected automatically within the specified µC clock range in MHz.

BAUD RATE	f_{CLK} MIN.	f_{CLK} MAX.
1200	1 (note 1)	3.6
2400	2 (note 1)	7.3
4800	4	19.7 (note 1)
9600	7.9	29.5 (note 1)
19200	15.7	59 (note 1)

Note

1. value outside the specified clock range.

9.3 Boot Routine Definitions

In the following description, each letter corresponds to one hexadecimal digit in ASCII representation (the digits 0 through 9, and the letters A through F).

Construction of data records (using the notation defined in Table 29) is as follows:

:BCAAAATTHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
HHHHCC

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 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 (H) 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 ":") should be ignored, if present.

An example of a valid HEX file follows:

:10010000C2F0E53030E704F404D2F08531F030F786
:100110000763F0FF05F0B2F0A430F00A63F0FFF4DB
:0C012000240150020 5F085F032F5332276
:00000001FF

Table 29 Boot routine definitions.

MNEMONIC	FUNCTION
:	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 prefer to deal with 16 data bytes per record (as shown in the example above).
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 two's complement of the summation of all of the bytes in the record from the byte count through the last data byte. While 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.

10 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{DD}	supply voltage range	–	−0.5	+6.5	V
V_I	all input voltages	–	−0.5	V_{DD} + 0.5	V
P_{tot}	total power dissipation	note 1	–	1	W
T_{stg}	storage temperature range		−65	+150	°C
T_{amb}	operating ambient temperature range:				
	version EBx		0	+70	°C
	version EFx		−40	+85	°C

Note

1. This value is based on the maximum allowable die temperature and the thermal resistance of the package, not on the device power consumption.

11 DC CHARACTERISTICS (EBx)

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

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
Supply					
V_{DD}	supply voltage range		4.5	5.5	V
Supply current					
I_{DD}	operating modes	V_{DD} = 5.5 V; f_{CLK} = 16 MHz notes 1 and 8			
	version P89CE528		–	50	mA
	version P80CE528		–	40	mA
	version P83CE528		–	40	mA
I_{ID}	idle mode	V_{DD} = 5.5 V; f_{CLK} = 16 MHz notes 2 and 8			
	version P89CE528		–	10	mA
	version P80CE528		–	8	mA
	version P83CE528		–	8	mA
I_{PD}	power-down mode	2 V ≤ V_{PD} ≤ $V_{DD\ max.}$ note 3	–	100	µA
Inputs					
V_{IL}	LOW level input voltage (except $\overline{EA}$, P1.6, P1.7)		−0.5	0.2 V_{DD} − 0.1	V
V_{IL1}	LOW level input voltage $\overline{EA}$		−0.5	0.2 V_{DD} − 0.3	V
V_{IL2}	LOW level input voltage P1.6, P1.7	note 6	−0.5	0.3 V_{DD}	V
V_{IH}	HIGH level input voltage (except RST, XTAL1, P1.6, P1.7)		0.2 V_{DD} +0.9	V_{DD} + 0.5	V

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
Inputs					
V_{IH1}	HIGH level input voltage RST, XTAL1		0.7 V_{DD}	V_{DD} + 0.5	V
V_{IH2}	HIGH level input voltage P1.6, P1.7	note 6	0.7V_{DD}	6.0	V
I_{IL}	input current logic 0 Ports 1, 2 and 3 (except P1.6 and P1.7)	V_I = 0.45 V	–	–50	µA
I_{TL}	input current HIGH-to-LOW transition Ports 1, 2 and 3 (except P1.6 and P1.7)	V_I = 2.0 V	–	–650	µA
I_{LI1}	input leakage current Port 0, $\overline{EA}$	0.45 < V_I < V_{DD}	–	±10	µA
I_{LI2}	input leakage current P1.6 and P1.7	0 V < V_I < 6 V 0 V < V_{DD} < 5.5 V	–	±10	µA
Outputs					
V_{OL}	LOW level output voltage Ports 1, 2 and 3 (except P1.6 and P1.7)	I_{OL} = 1.6 mA; notes 4 and 7	–	0.45	V
V_{OL1}	LOW level output voltage Port 0, ALE, $\overline{PSEN}$	I_{OL} = 3.2 mA notes 4 and 7	–	0.45	V
V_{OL2}	LOW level output voltage P1.6 and P1.7	I_{OL} = 3.0 mA; note 7	–	0.40	V
V_{OH}	HIGH level output voltage Ports 1, 2, 3	I_{OH} = –60 µA; V_{DD} = 5 V ±10% I_{OH} = –25 µA I_{OH} = –10 µA	2.4 0.75V_{DD} 0.9V_{DD}	– – –	V
V_{OH1}	HIGH level output voltage Port 0 in external Bus mode, ALE, $\overline{PSEN}$, RST	I_{OH} = –800 µA; V_{DD} = 5 V ±10% I_{OH} = –300 µA; I_{OH} = –80 µA; note 5	2.4 0.75V_{DD} 0.9V_{DD}	– – –	V
R_{RST}	RST pull–down resistor		50	150	kΩ
$C_{I/O}$	capacitance of input buffer	test frequency = 1 MHz; T_{amb} = 25 °C	–	10	pF

Notes to DC Characteristics (EBx)

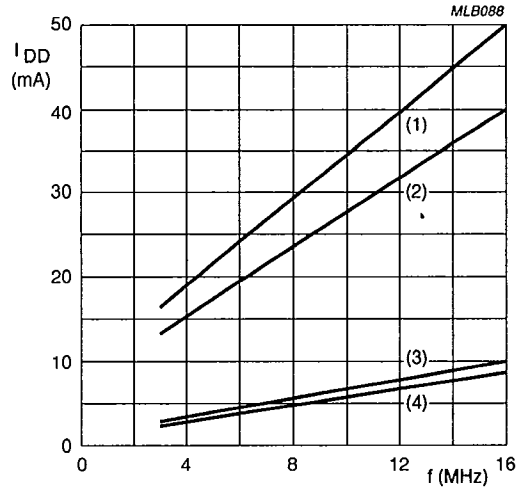
1. The operating supply current is measured with all output pins disconnected;
   XTAL1 driven with $t_r = t_f$ = 5 ns;
   $V_{IL} = V_{SS}$ +0.5 V;
   $V_{IH} = V_{DD}$ –0.5 V;
   XTAL2 not connected;
   $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = V_{DD};
   the WDT is disabled (by the external RESET).

2. The Idle mode supply current is measured with all output pins disconnected;
   XTAL1 driven with $t_r = t_f = 5$ ns;
   $V_{IL} = V_{SS} + 0.5$ V;
   $V_{IH} = V_{DD} - 0.5$ V;
   XTAL2 not connected;
   the WDT is disabled;
   $\overline{EA}$ = RST = V_{SS};
   Port 0 = P1.6 = P1.7 = V_{DD}.
3. The Power-down current is measured with all output pins disconnected;
   XTAL2 not connected;
   WDT is disabled;
   $\overline{EA}$ = RST = XTAL1 = V_{SS};
   Port 0 = P1.6 = P1.7 = V_{DD}.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a HIGH-to-LOW transition during bus operations. In the worst cases (capacitive loading > 100pF), the noise pulse on the ALE line may exceed 0.8 V. 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.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9 V_{DD} specification when the address bits are stabilizing.
6. The input threshold voltage of P1.6 and P1.7 (SIO1) meets the I²C-bus specification, so a voltage below 0.3 V_{DD} will be recognized as a logic 0 while an input above 0.7 V_{DD} will be recognized as a logic 1.
7. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin:10 mA.
   Maximum I_{OL} per 8-bit port:- Port 0: 26 mA; Ports 1, 2 and 3: 15 mA.
   Maximum total I_{OL} for all output pins: 71 mA.
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
8. I_{DD} max. at other frequencies can be derived from Figure 26, where f is the external oscillator frequency in MHz; I_{DD} max. is given in mA.

12 DC CHARACTERISTICS (EFx)

V_{DD} = 5 V ±10%; V_{SS} = 0 V; T_{amb} = −40 to +85°C (extended temperature range). All voltages with respect to V_{SS} unless otherwise specified. DC parameters not included here are the same as for the EBx temperature range data sheets.

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
Inputs					
V_{IL}	LOW level input voltage (except $\overline{EA}$, P1.6, P1.7)		−0.5	0.2 V_{DD}−0.15	V
V_{IL1}	LOW level input voltage $\overline{EA}$		−0.5	0.2 V_{DD}−0.35	V
V_{IH}	HIGH level input voltage (except RST, XTAL1, P1.6, P1.7)		0.2 V_{DD} +1.0	V_{DD} +0.5	V
V_{IH1}	HIGH level input voltage RST, XTAL1		0.7 V_{DD} +0.1	V_{DD} +0.5	V
I_{IL}	input current logic 0 Ports 1, 2 and 3 (except P1.6 and P1.7)	V_I = 0.45 V	–	−75	µA
I_{TL}	input current HIGH-to-LOW transition Ports 1, 2 and 3 (except P1.6 and P1.7)	V_I = 2.0 V	–	−750	µA



V_{DD} = 5.5 V

Valid only within frequency specifications of device under test.

(1) Mximum operating mode P89CE528.

(2) Maximum operating mode P83CE528 and P80CE528.

(3) Maximum idle mode P89CE528.

(4) Maximum idle mode P83CE528 and P80CE528.

Fig.26 Supply current I_{DD} as a function of frequency.

13 AC CHARACTERISTICS

EBX: V_{DD} = 5 V ±10%; V_{SS} = 0 V; T_{amb} = 0 to +70 °C; t_{CLCL} min. = 63 ns.
EFX: V_{DD} = 5 V ±10%; V_{SS} = 0 V; T_{amb} = −40 to +85 °C; t_{CLCL} min. = 63 ns.
All versions EXX: Cl = 100 pF for Port 0, ALE and $\overline{PSEN}$; Cl = 80 pF for all other outputs unless otherwise specified.
t_{CLCL} min. = 1/f max. (maximum operating frequency).

SYMBOL	PARAMETER	16 MHz		12 MHz		VARIABLE CLOCK		UNIT
		MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
$1/t_{CLCL}$	oscillator frequency	–	–	–	–	3.5	16	MHz
t_{LHLL}	ALE pulse duration	85	–	127	–	$2\,t_{CLCL}-40$	–	ns
t_{AVLL}	address set-up time to ALE	8	–	28	–	$t_{CLCL}-55$	–	ns
t_{LLAX}	address hold time after ALE	28	–	48	–	$t_{CLCL}-35$	–	ns
t_{LLIV}	time from ALE to valid instruction input	–	150	–	233	–	$4\,t_{CLCL}-100$	ns
t_{LLPL}	time from ALE to control pulse $\overline{PSEN}$	23	–	43	–	$t_{CLCL}-40$	–	ns
t_{PLPH}	control pulse duration $\overline{PSEN}$	143	–	205	–	$3\,t_{CLCL}-45$	–	ns
t_{PLIV}	time from $\overline{PSEN}$ to valid instruction input	–	83	–	145	–	$3\,t_{CLCL}-105$	ns
t_{PXIX}	input instruction hold time after $\overline{PSEN}$	0	–	0	–	0	–	ns
t_{PXIZ}	input instruction float delay after $\overline{PSEN}$	–	38	–	59	–	$t_{CLCL}-25$	ns
t_{AVIV}	address to valid instruction input	–	208	–	312	–	$5\,t_{CLCL}-105$	ns
t_{PLAZ}	address float time to $\overline{PSEN}$	–	10	–	10	–	10	ns
Data memory								
t_{RLRH}	$\overline{RD}$ pulse duration	275	–	400	–	$6\,t_{CLCL}-100$	–	ns
t_{WLWH}	$\overline{WR}$ pulse duration	275	–	400	–	$6\,t_{CLCL}-100$	–	ns
t_{RLDV}	$\overline{RD}$ to valid data input	–	148	–	252	–	$5\,t_{CLCL}-165$	ns
t_{RHDX}	data hold time after $\overline{RD}$	0	–	0	–	0	–	ns
t_{RHDZ}	data float delay after $\overline{RD}$	–	55	–	97	–	$2\,t_{CLCL}-70$	ns
t_{LLDV}	time from ALE to valid data input	–	350	–	517	–	$8\,t_{CLCL}-150$	ns
t_{AVDV}	address to valid data input	–	398	–	585	–	$9\,t_{CLCL}-165$	ns
t_{LLWL}	time from ALE to $\overline{RD}$ or $\overline{WR}$	138	238	200	300	$3\,t_{CLCL}-50$	$3\,t_{CLCL}+50$	ns
t_{AVWL}	time from address to $\overline{RD}$ or $\overline{WR}$	120	–	203	–	$4\,t_{CLCL}-130$	–	ns
t_{QVWX}	data valid to $\overline{WR}$ transition	3	–	23	–	$t_{CLCL}-60$	–	ns
t_{WHQX}	data hold time after $\overline{WR}$	13	–	33	–	$t_{CLCL}-50$	–	ns
t_{RLAZ}	address float delay after $\overline{RD}$	–	0	–	0	–	0	ns
t_{WHLH}	time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	23	103	43	123	$t_{CLCL}-40$	$t_{CLCL}+40$	ns

SYMBOL	PARAMETER	16 MHz		12 MHz		VARIABLE CLOCK		UNIT
		MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
External clock								
t_{CHCX}	HIGH time	20	–	–	–	20	–	ns
t_{CLCX}	LOW time	20	–	–	–	20	–	ns
t_{CLCH}	rise time	–	20	–	–	–	20	ns
t_{CHCL}	fall time	–	20	–	–	–	20	ns
UART timing shift register mode								
t_{XLXL}	serial port clock cycle time	750	–	1000	–	$12t_{CLCL}$	–	ns
t_{QVXH}	output data setup to clock rising edge	492	–	700	–	$10t_{CLCL}-133$	–	ns
t_{XHQX}	output data hold after clock rising edge	8	–	50	–	$2t_{CLCL}-117$	–	ns
t_{XHDX}	input data hold after clock rising edge	0	–	0	–	0	–	ns
t_{XHDV}	clock rising edge to input data valid	–	492	–	700	–	$10t_{CLCL}-133$	ns

14 I²C-bus CHARACTERISTICS (bit-level)

$t_{CLCL} = 1/f_{CLK}$ = one oscillator period at pin XTAL1. For 63 ns < t_{CLCL} < 286 ns (16 MHz > f_{OSC} > 3.5 MHz) the I²C interface meets the I²C-bus specification for bit rates up to 100 kbit/s.

SYMBOL	PARAMETER	INPUT	OUTPUT	I²C-bus Spec.	UNIT
SCL timing					
$t_{HD;STA}$	START condition hold time	≥ 14 t_{CLCL}; note 1	note 2	≥ 4.0	µs
t_{LOW}	SCL LOW time	≥ 16 t_{CLCL}	note 2	≥ 4.7	µs
t_{HIGH}	SCL HIGH time	≥ 14 t_{CLCL}; note 1	≥ 80 t_{CLCL}; note 3	≥ 4.0	µs
t_{RC}	SCL rise time	≤ 1 µs; note 4	note 5	≤ 1.0	µs
t_{FC}	SCL fall time	≤ 0.3 µs; note 4	≤ 0.3 µs; note 6	≤ 0.3	µs
SDA timing					
$t_{SU;DAT}$	data set-up time	≥ 250 ns	note 2	≥ 250	ns
$t_{HD;DAT}$	data hold time	≥ 0 ns	note 2	≥ 0	ns
$t_{SU;STA}$	repeated START set-up time	≥ 14 t_{CLCL}; note 1	note 2	≥ 4.7	µs
$t_{SU;STO}$	STOP condition set–up time	≥ 14 t_{CLCL}; note 1	note 2	≥ 4.0	µs
t_{BUF}	bus free time	≥ 14 t_{CLCL}; note 1	note 2	≥ 4.7	µs
t_{RD}	SDA rise time	≤ 1 µs; note 4	note 5	≤ 1.0	µs
t_{FD}	SDA fall time	≤ 0.3 µs; note 4	≤ 0.3 µs; note 6	≤ 0.3	µs

Notes to I²C-bus Characteristics (bit-level)

1. At f_{CLK} = 3.5 MHz, this evaluates to 14 x 286 ns = 4 µs, i.e. the bit-level I²C-bus interface can respond to the I²C-bus protocol for f_{CLK} ≥ 3.5 MHz.
2. This parameter is determined by the user software, it has to comply with the I²C-bus specification.
3. This value gives the auto-clock pulse length which meets the I²C-bus specification for the specified XTAL1 clock frequency range. Alternatively, the SCL pulse may be timed by software.
4. Spikes on SDA and SCL lines with a duration of less than 4 x t_{CLCL} will be filtered out.
5. The rise time is determined by the external bus line capacitance and pull-up resistor, it must be ≤ 1 µs.
6. The maximum capacitance on bus lines SDA and SCL is 400 pF.

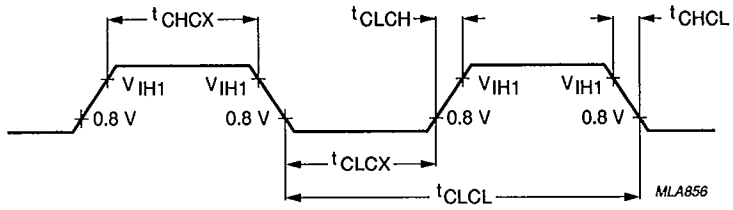
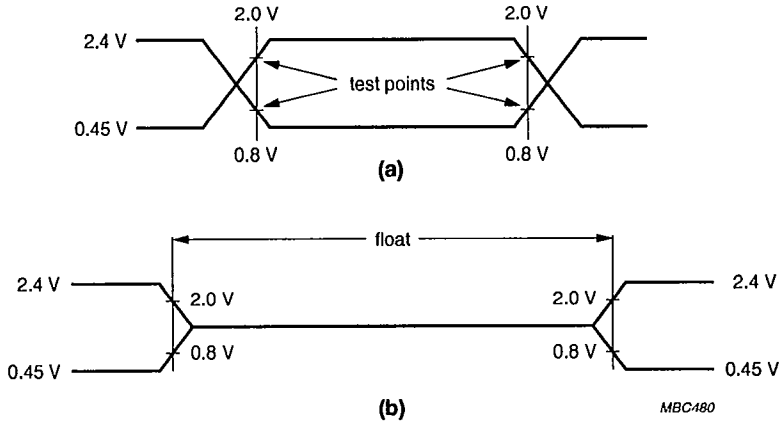


Fig.27 External clock drive XTAL1.



AC testing inputs are driven at 2.4 V for a logic 1 and 0.45 V for a logic 0. Timing measurements are taken at 2.0 V for a logic 1 and 0.8 V for logic 0 see (a). The float state is defined as the point at which a Port 0 pin sinks 3.2 mA or sources 400 µA at the voltage test levels see (b).

Fig.28 AC testing input, output waveform (a) and float waveform (b).

14.1 Timing symbol definitions

Oscillator:

f_{CLK} = clock frequency

t_{CLCL} = clock period

Timing symbols (acronyms):

Each timing symbol has five characters. The first character is always a "t" (= time). the remaining four characters of the symbol (typed in subscript), depending on their relative positions, indicate the name of a signal or the logical status of that signal. the designations are as follows:

A = address
C = clock
D = input data
H = logic level HIGH
I = instruction (program memory contents)
L = Logic level LOW or ALE
P = $\overline{\text{PSEN}}$
Q = output data
R = $\overline{\text{RD}}$ signal
t = time
V = valid
W = $\overline{\text{WR}}$ signal
X = no longer a valid logic level
Z = float

Examples:

t_{AVLL} = time for address valid to ALE LOW
t_{LLPL} = time for ALE LOW to $\overline{\text{PSEN}}$ LOW

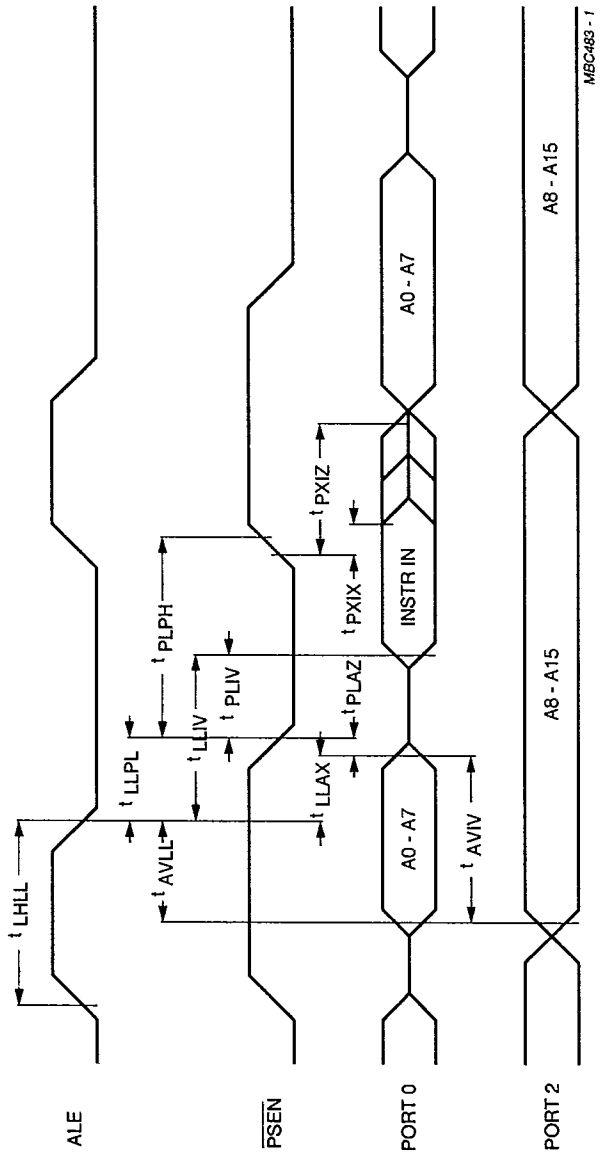


Fig.29 External program memory read cycle.

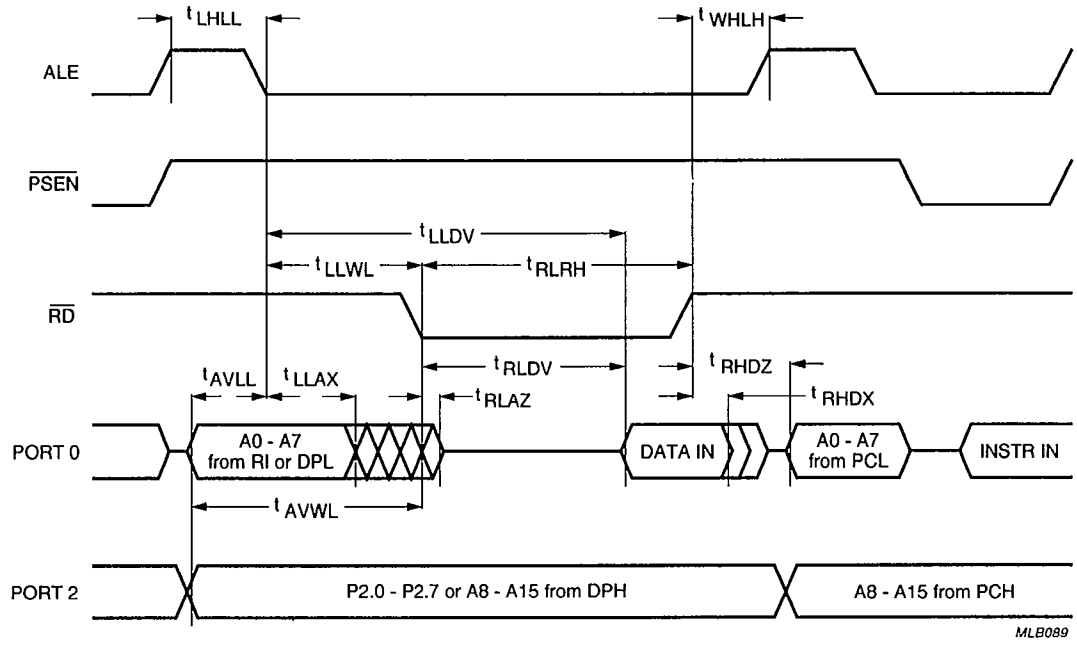


Fig.30 External data memory read cycle.

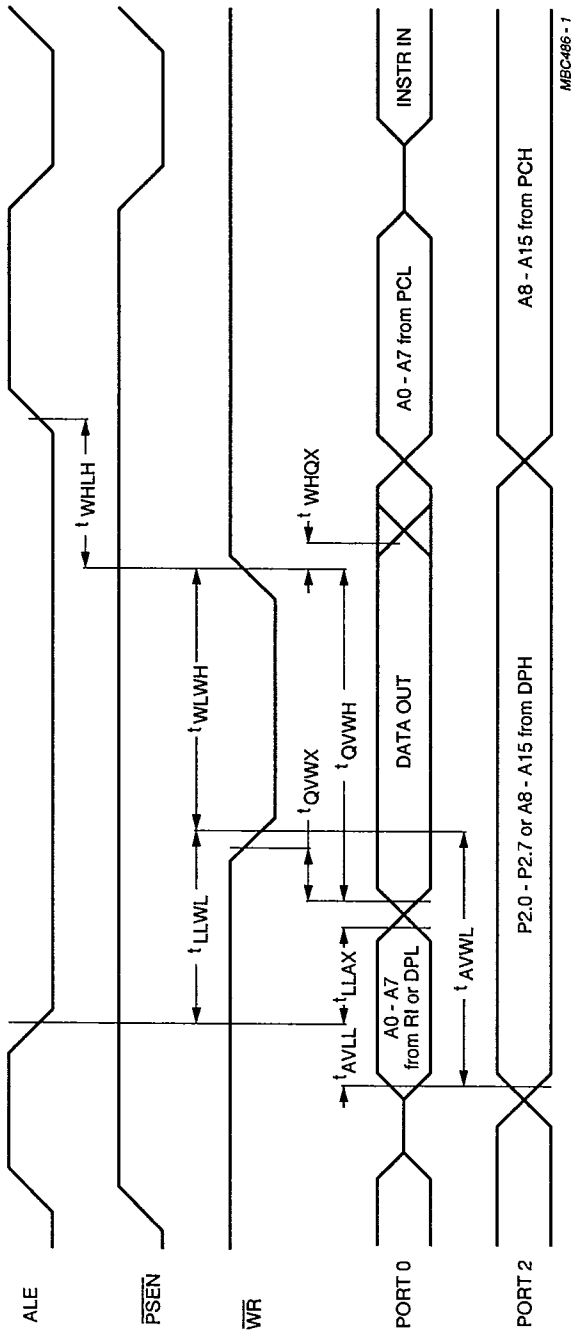


Fig.31 External data memory write cycle.

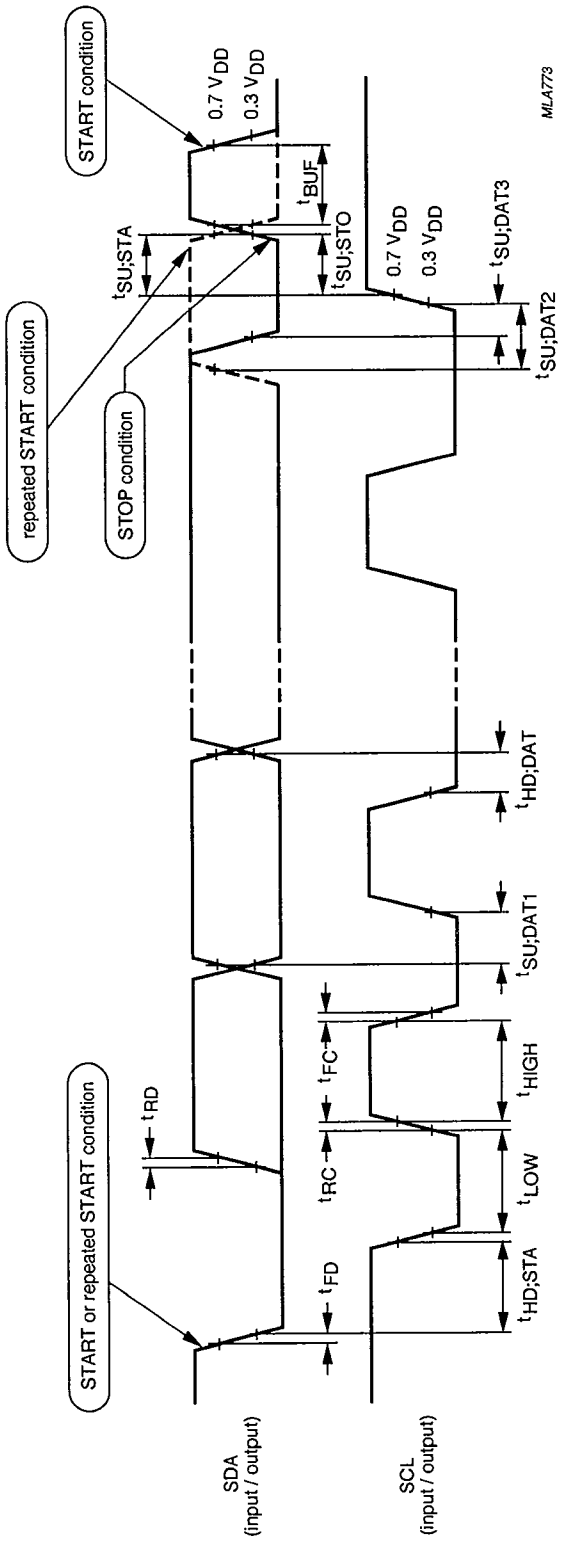


Fig.32 I²C-bus interface timing.

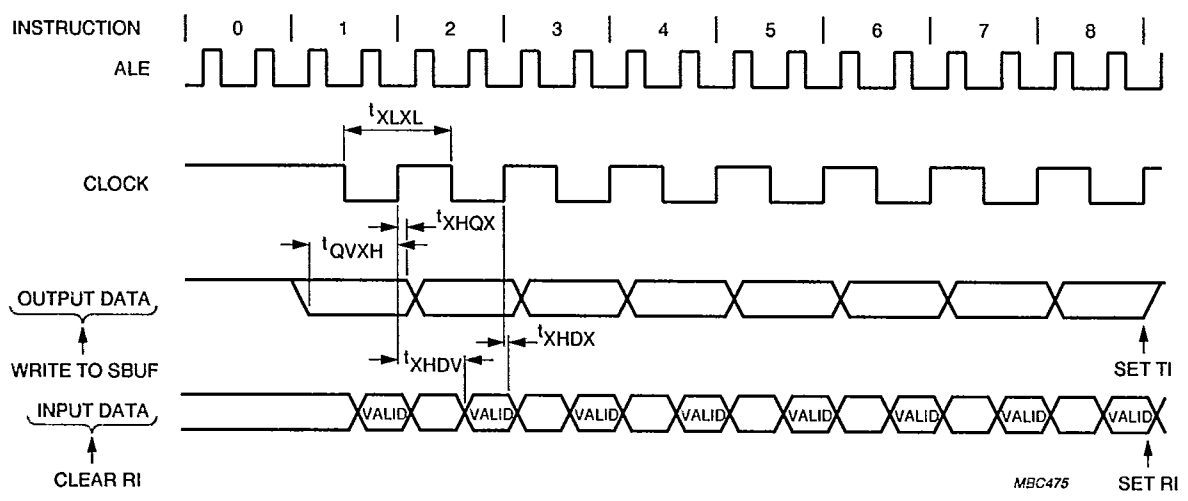


Fig.33 UART waveforms in Shift Register mode.

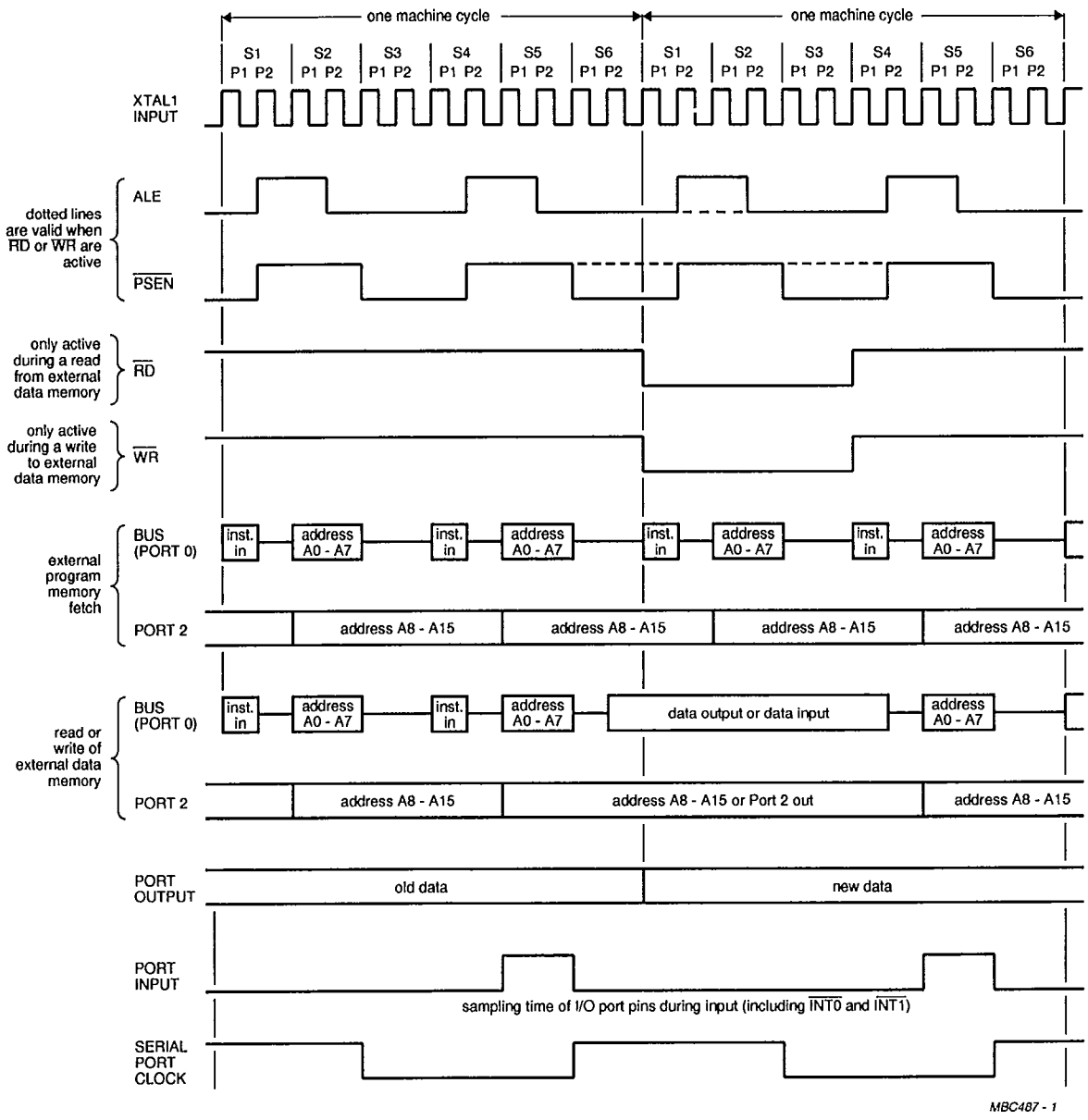
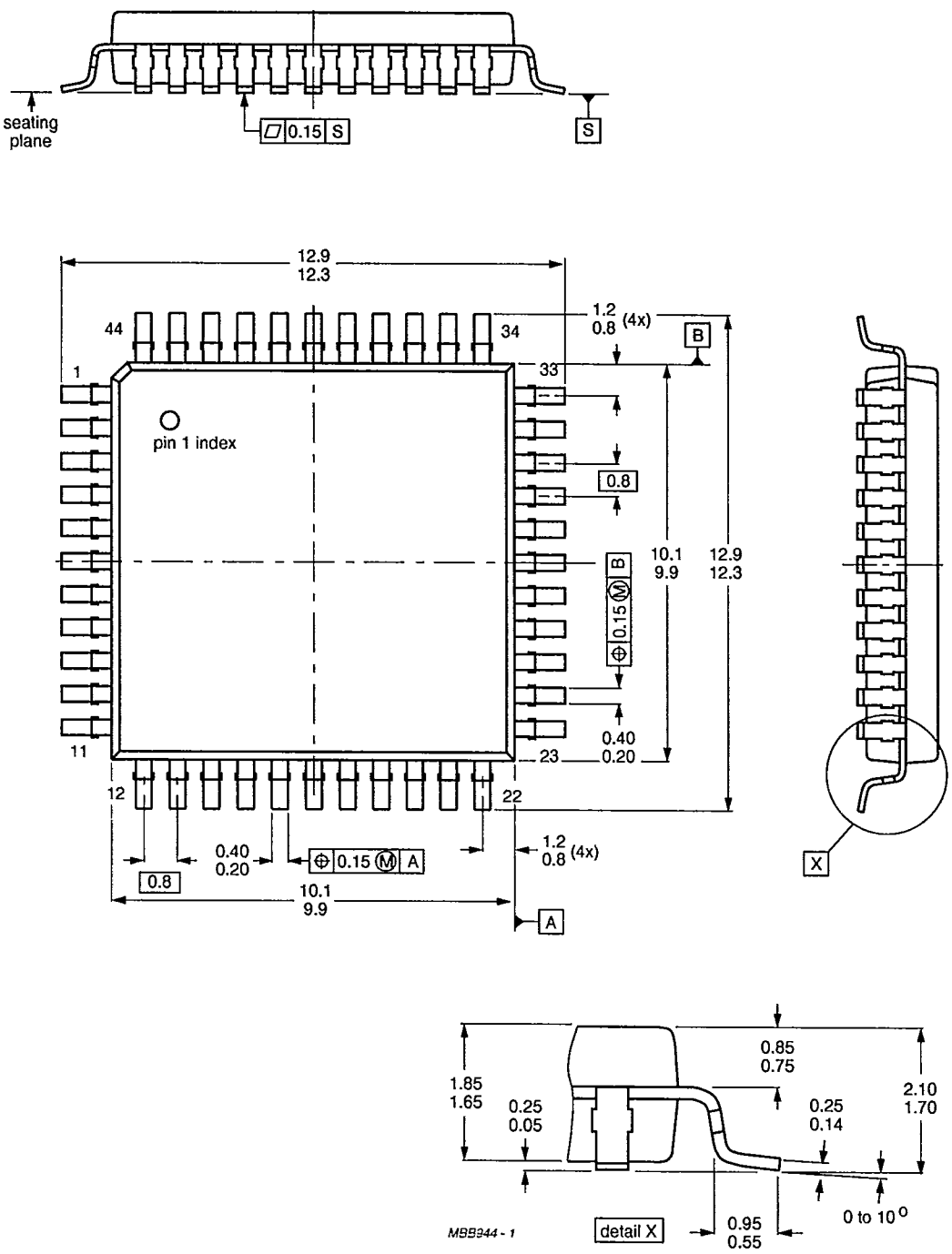


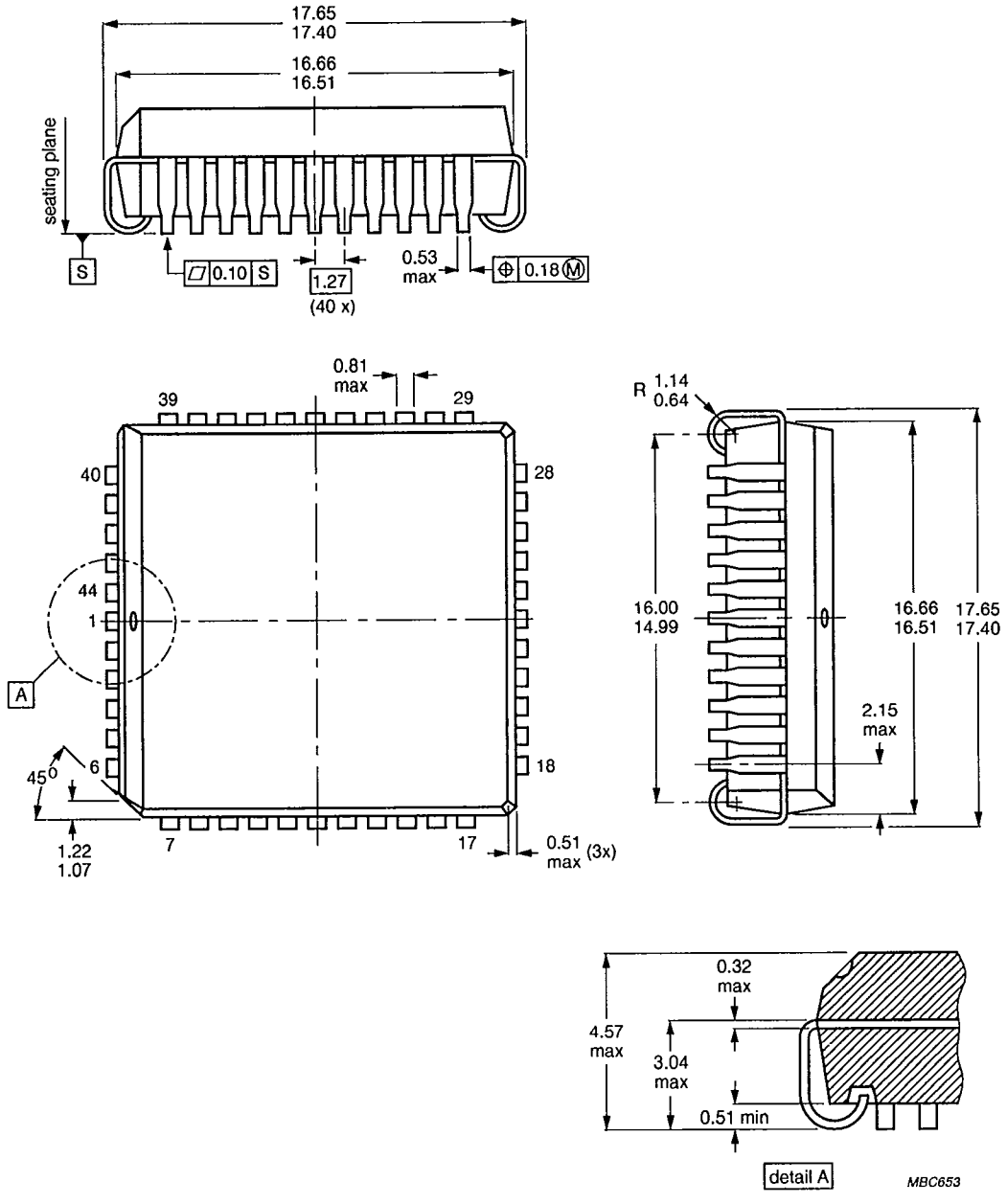
Fig.34 Instruction cycle timing.

15 PACKAGE OUTLINES



Dimensions in mm.

Fig.35 44-lead quad flat-pack 10mm square; plastic (QFP44S10); SOT307-2).



Dimensions in mm.

Fig.36 44-lead plastic leaded chip carrier (PLCC); (SOT187CG).

16 SOLDERING

16.1 Plastic leaded chip carrier (PLCC) and quad flat-pack 10mm square; plastic (QFP44S10)

16.1.1 By wave

During placement and before soldering, the component must be fixed with a droplet of adhesive. After curing the adhesive, the component can be soldered. The adhesive can be applied by screen printing, pin transfer or syringe dispensing.

Maximum permissible solder temperature is 260 °C, and maximum duration of package immersion in solder bath is 10 s, if allowed to cool to less than 150 °C within 6 s. Typical dwell time is 4 s at 250 °C.

A modified wave soldering technique is recommended using two solder waves (dual-wave), in which a turbulent wave with high upward pressure is followed by a smooth laminar wave. Using a mildly-activated flux eliminates the need for removal of corrosive residues in most applications.

16.1.2 By solder paste reflow

Reflow soldering requires the solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the substrate by screen printing, stencilling or pressure-syringe dispensing before device placement.

Several techniques exist for reflowing; for example, thermal conduction by heated belt, infrared, and vapour-phase reflow. Dwell times vary between 50 and 300 s according to method. Typical reflow temperatures range from 215 to 250 °C.

Preheating is necessary to dry the paste and evaporate the binding agent. Preheating duration: 45 min at 45 °C.

16.1.3 Repairing soldered joints (by hand-held soldering iron or pulse-heated solder tool)

Fix the component by first soldering two, diagonally opposite, end pins. Apply the heating tool to the flat part of the pin only. Contact time must be limited to 10 s at up to 300 °C. When using proper tools, all other pins can be soldered in one operation within 2 to 5 s at between 270 and 320 °C. (Pulse-heated soldering is not recommended for SO packages.)

For pulse-heated solder tool (resistance) soldering of VSO packages, solder is applied to the substrate by dipping or by an extra thick tin/lead plating before package placement.

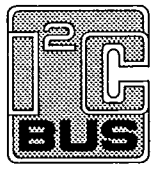
17 DEFINITIONS

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
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	
Where application information is given, it is advisory and does not form part of the specification.	

18 LIFE SUPPORT APPLICATIONS

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 customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips for any damages resulting from such improper use or sale.

19 PURCHASE OF PHILIPS I²C COMPONENTS



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 specification defined by Philips. This specification can be ordered using the code 9398 393 40011.