

# DATA SHEET

## **P83C562; P80C562** 8-bit microcontroller

Product specification  
File under Integrated Circuits, IC20

1997 Apr 16

**8-bit microcontroller****P83C562; P80C562**

|                 |        |   |
|-----------------|--------|---|
| <b>CONTENTS</b> | 13     | SERIAL I/O                              |
| 1               | 14     | INTERRUPT SYSTEM                        |
| 1               | 14.1   | Interrupt Vectors                       |
| 2               | 14.2   | Interrupt priority                      |
| 3               | 14.3   | Interrupt Enable and Priority Registers |
| 4               | 14.3.1 | Interrupt Enable Register 0 (IEN0)      |
| 5               | 14.3.2 | Interrupt Enable register 1 (IEN1)      |
| 6               | 14.3.3 | Interrupt priority register 0 (IP0)     |
| 6.1             | 14.3.4 | Interrupt Priority Register 1 (IP1)     |
| 6.2             | 15     | REDUCED POWER MODES                     |
| 7               | 15.1   | Idle and Power-down operation           |
| 8               | 15.1.1 | Idle mode                               |
| 8.1             | 15.1.2 | Power-down mode                         |
| 8.2             | 15.2   | Power Control Register (PCON)           |
| 9               | 16     | OSCILLATOR CIRCUITRY                    |
| 10              | 17     | RESET CIRCUITRY                         |
| 10.1            | 17.1   | Power-on-reset                          |
| 10.2            | 18     | INSTRUCTION SET                         |
| 10.3            | 19     | LIMITING VALUES                         |
| 11              | 20     | DC CHARACTERISTICS                      |
| 11.1            | 21     | AC CHARACTERISTICS                      |
| 11.2            | 22     | PACKAGE OUTLINES                        |
| 12              | 23     | SOLDERING                               |
| 12.1            | 23.1   | Introduction                            |
| 12.2            | 23.2   | Reflow soldering                        |
| 12.2.1          | 23.3   | Wave soldering                          |
| 12.2.2          | 23.4   | Repairing soldered joints               |
| 12.2.3          | 24     | DEFINITIONS                             |
| 12.2.4          | 25     | LIFE SUPPORT APPLICATIONS               |
| 12.2.5          |        |   |
| 12.3            |        |   |

## 8-bit microcontroller

## P83C562; P80C562

**1 FEATURES**

- 80C51 Central Processing Unit
- 8 kbytes ROM, expandable externally to 64 kbytes
- 256 bytes RAM, expandable externally to 64 kbytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- An 8-bit ADC with 8 multiplexed analog inputs
- Two 8-bit resolution, Pulse Width Modulated outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analog inputs
- Full-duplex UART compatible with the standard 80C51
- On-chip Watchdog Timer
- Oscillator frequency: 3.5 to 16 MHz.

**2 GENERAL DESCRIPTION**

The P80C562/P83C562 (hereafter generally referred to as P8xC562) single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the 80C51 microcontroller family.

The P8xC562 has the same instruction set as the 80C51. Two versions of the derivative exist:

- With 8 kbytes mask-programmable ROM
- ROMless version of the P8xC562.

**3 ORDERING INFORMATION**

| TYPE NUMBER                   | PACKAGE |                                       |          | FREQUENCY RANGE (MHz) | TEMPERATURE RANGE (°C) |
|-------------------------------|---------|---------------------------------------|----------|-----------------------|------------------------|
|                               | NAME    | DESCRIPTION                           | VERSION  |                       |                        |
| P80CE562EHA <sup>(1)</sup>    | PLCC68  | plastic leaded chip carrier; 68 leads | SOT188-2 | 3.5 to 16             | -40 to +125            |
| P80C562EBA <sup>(1)</sup>     |         |                                       |          |                       | 0 to +70               |
| P80C562EFA <sup>(1)</sup>     |         |                                       |          |                       | -40 to +85             |
| P83C562EHA/nnn <sup>(2)</sup> |         |                                       |          |                       | -40 to +125            |
| P83C562EBA/nnn <sup>(2)</sup> |         |                                       |          |                       | 0 to +70               |
| P83C562EFA/nnn <sup>(2)</sup> |         |                                       |          |                       | -40 to +85             |

**Notes**

1. ROMless type.
2. ROM coded type; nnn denotes the ROM code number.

This I/O intensive device provides architectural enhancements to function as a controller in the field of automotive electronics, specifically engine management and gear box control.

The P8xC562 contains a non-volatile 8 kbyte read only program memory, a volatile 256 byte read/write data memory, six 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a fourteen-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC with pulse width modulated outputs, a serial interface (UART), a Watchdog Timer and on-chip oscillator and timing circuits. For systems that require extra capability, the P8xC562 can be expanded using standard TTL compatible memories and logic.

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  $\mu$ s and 40% in 1.5  $\mu$ s. Multiply and divide instructions require 3  $\mu$ s.

8-bit microcontroller

P83C562; P80C562

4 BLOCK DIAGRAM

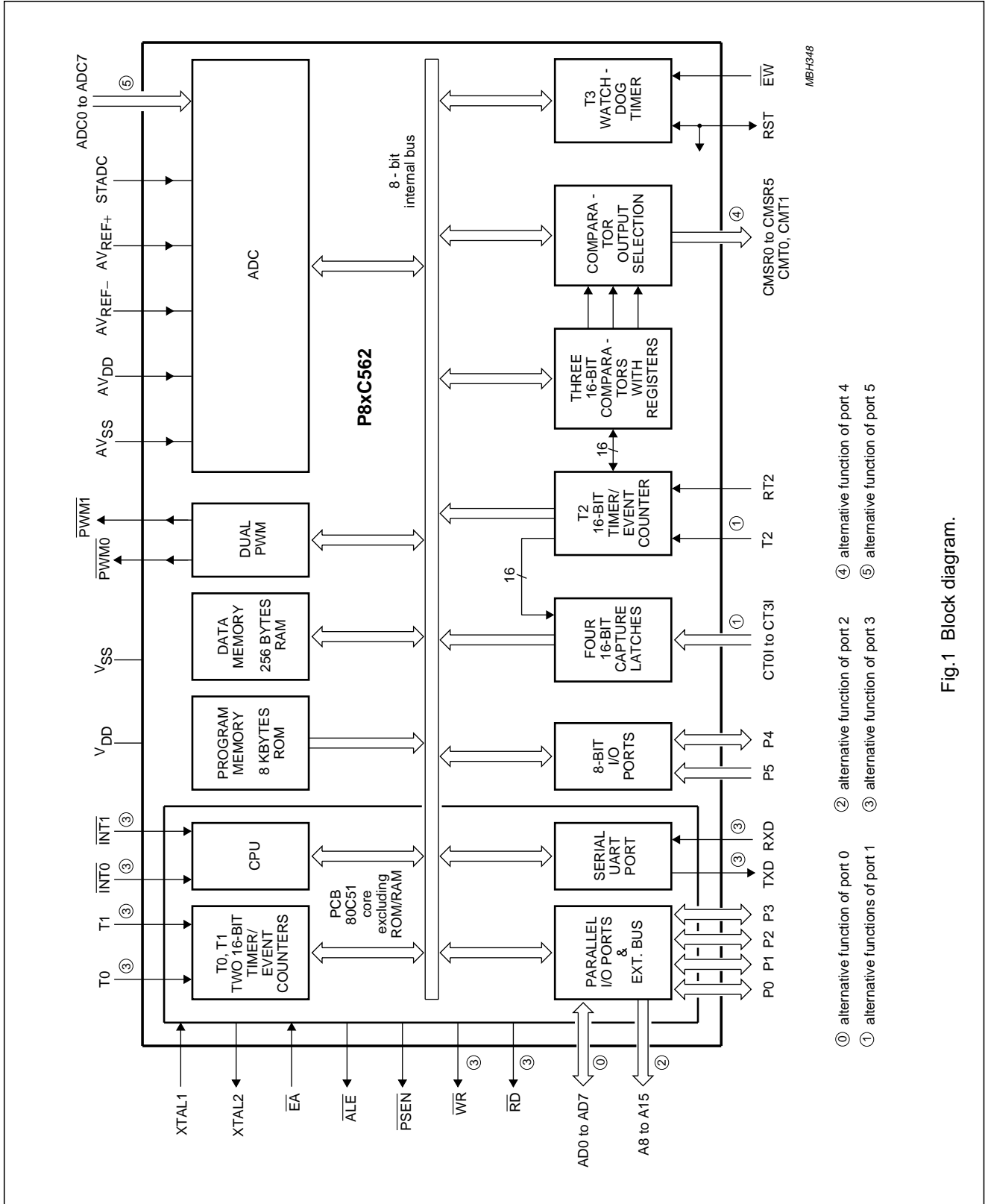


Fig.1 Block diagram.

8-bit microcontroller

P83C562; P80C562

5 FUNCTIONAL DIAGRAM

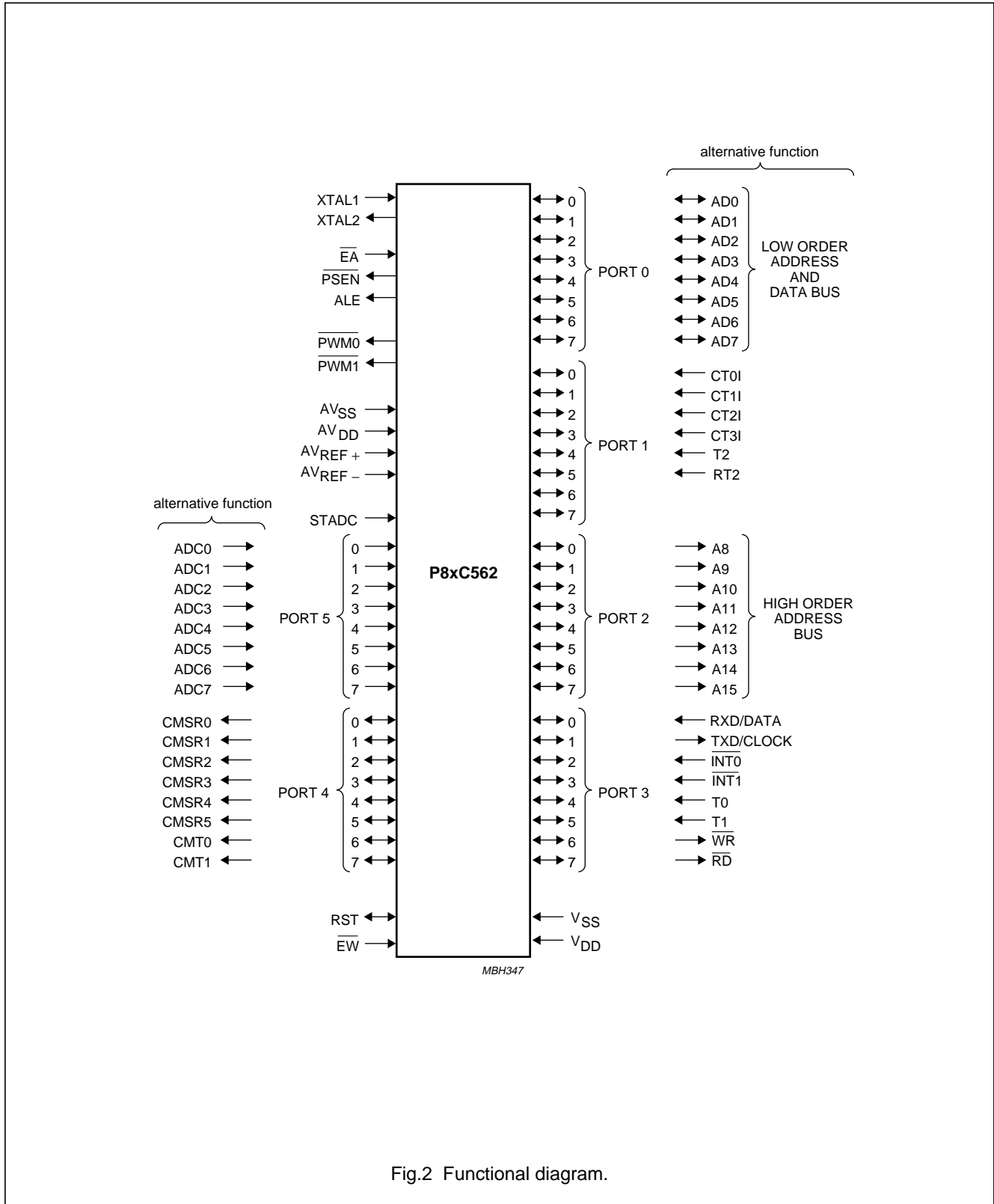


Fig.2 Functional diagram.

8-bit microcontroller

P83C562; P80C562

6 PINNING INFORMATION

6.1 Pinning

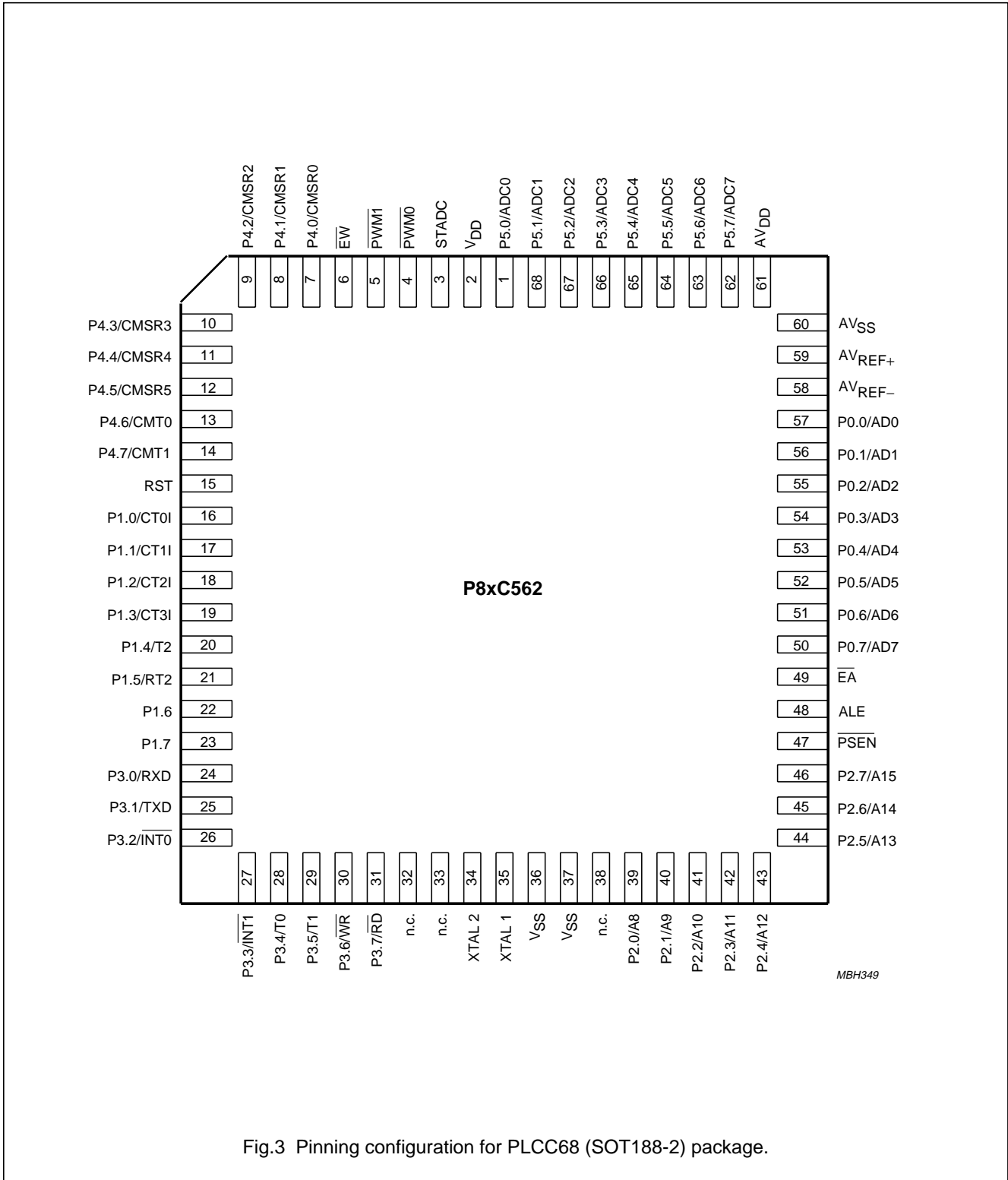


Fig.3 Pinning configuration for PLCC68 (SOT188-2) package.

## 8-bit microcontroller

## P83C562; P80C562

## 6.2 Pin description

Table 1 PLCC68 (SOT188-2)

To avoid latch-up at Power-on, the voltage at any pin at any time must lie within the range  $V_{DD} + 0.5\text{ V}$  to  $V_{SS} - 0.5\text{ V}$ .

| SYMBOL                         | PIN      | DESCRIPTION  |
|--------------------------------|----------|--|
| $V_{DD}$                       | 2        | <b>Power supply</b> , digital part (+5 V). Power supply pins during normal operation and power reduction modes.  |
| STADC                          | 3        | <b>Start ADC operation</b> : Input starting analog-to-digital conversion (ADC operation can also be started by software). This pin must not float.           |
| $\overline{\text{PWM0}}$       | 4        | <b>Pulse Width Modulation output 0</b> .   |
| $\overline{\text{PWM1}}$       | 5        | <b>Pulse Width Modulation output 1</b> .   |
| $\overline{\text{EW}}$         | 6        | <b>Enable Watchdog Timer</b> : enable for Watchdog Timer and disable Power-down mode. This pin must not float.   |
| P4.0/CMSR0<br>to<br>P4.5/CMSR5 | 7 to 12  | <b>P4.0 to P4.5</b> : 8-bit quasi-bidirectional I/O port lines;<br><b>CMSR0 to CMSR5</b> : Compare and Set/Reset outputs for Timer T2.                       |
| P4.6/CMT0                      | 13       | <b>P4.6 to P4.7</b> : 8-bit quasi-bidirectional I/O port lines;<br><b>CMT0 to CMT1</b> : Compare and toggle outputs for Timer T2.                            |
| P4.7/CMT1                      | 14       |  |
| RST                            | 15       | <b>Reset</b> : Input to reset the P8x562; also generated when the Watchdog Timer overflows.  |
| P1.0/CT0I<br>to<br>P1.3/CT3I   | 16 to 19 | <b>P1.0 to P1.3</b> : 8-bit quasi-bidirectional I/O port lines;<br><b>CT0I to CT3I</b> : Capture timer inputs for Timer 2.                                   |
| P1.4/T2                        | 20       | <b>P1.4</b> : 8-bit quasi-bidirectional I/O port line;<br><b>T2</b> : T2 event input (rising edge triggered).  |
| P1.5/RT2                       | 21       | <b>P1.5</b> : 8-bit quasi-bidirectional I/O port line;<br><b>RT2</b> : T2 timer reset input (rising edge triggered)  |
| P1.6 to P1.7                   | 22 to 23 | <b>P1.6 to P1.7</b> : 8-bit quasi-bidirectional I/O port lines, open-drain.  |
| P3.0/RXD                       | 24       | <b>P3.0</b> : 8-bit quasi-bidirectional I/O port line;<br><b>RXD</b> : Serial input port.  |
| P3.1/TXD                       | 25       | <b>P3.1</b> : 8-bit quasi-bidirectional I/O port line;<br><b>TXD</b> : Serial output port.   |
| P3.2/ $\overline{\text{INT0}}$ | 26       | <b>P3.2</b> : 8-bit quasi-bidirectional I/O port line;<br><b>INT0</b> : External interrupt input 0.  |
| P3.3/ $\overline{\text{INT1}}$ | 27       | <b>P3.3</b> : 8-bit quasi-bidirectional I/O port line;<br><b>INT1</b> : External interrupt input 1.  |
| P3.4/T0                        | 28       | <b>P3.4</b> : 8-bit quasi-bidirectional I/O port line;<br><b>T0</b> : Timer 0 external input.  |
| P3.5/T1                        | 29       | <b>P3.5</b> : 8-bit quasi-bidirectional I/O port line;<br><b>T1</b> : Timer 1 external input.  |
| P3.6/ $\overline{\text{WR}}$   | 30       | <b>P3.6</b> : 8-bit quasi-bidirectional I/O port line;<br><b>WR</b> : External Data Memory Write strobe.   |
| P3.7/ $\overline{\text{RD}}$   | 31       | <b>P3.7</b> : 8-bit quasi-bidirectional I/O port line;<br><b>RD</b> : External Data Memory Read strobe.  |
| n.c.                           | 32, 33   | Not connected.   |
| XTAL2                          | 34       | <b>Crystal Oscillator Output</b> : output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used. |

## 8-bit microcontroller

## P83C562; P80C562

| SYMBOL                       | PIN            | DESCRIPTION  |
|------------------------------|----------------|--|
| XTAL1                        | 35             | <b>Crystal Oscillator Input:</b> 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.   |
| V <sub>SS</sub>              | 36, 37         | <b>Digital ground pins.</b>  |
| n.c.                         | 38             | Not connected.   |
| P2.0/A08<br>to<br>P2.7/A15   | 39 to 46       | <b>P2.0 to P2.7:</b> 8-bit quasi-bidirectional I/O port lines;<br><b>A08 to A15:</b> High-order address byte for external memory.  |
| PSEN                         | 47             | <b>Program Store Enable:</b> read strobe to the external program memory via Port 0 and 2. Is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of PSEN are skipped during each access to external data memory. PSEN is not activated (remains HIGH) during no fetches from external program memory. PSEN can sink/source 8 LSTTL inputs and can drive CMOS inputs without external pull-ups. |
| ALE                          | 48             | <b>Address Latch Enable:</b> latches the low byte of the address during access of external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LSTTL inputs and can drive CMOS inputs without an external pull-up. To prohibit the toggling of the ALE pin (RFI noise reduction) the RFI bit in the Power Control Register must be set by software.   |
| $\overline{EA}$              | 49             | <b>External Access:</b> if, during RESET, $\overline{EA}$ is HIGH the CPU executes out of the internal program memory provided the program Counter is less than 8192. If, during RESET, $\overline{EA}$ is LOW the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float. $\overline{EA}$ is latched during RESET and don't care after RESET.   |
| P0.7/AD7<br>to<br>P0.0/AD0   | 50 to 57       | <b>P0.7 to P0.0:</b> 8-bit open drain bidirectional I/O port lines;<br><b>AD7 to AD0:</b> Multiplexed Low-order address and Data bus for external memory.  |
| AV <sub>REF-</sub>           | 58             | <b>Low-end of ADC</b> (analog-to-digital conversion) reference resistor.   |
| AV <sub>REF+</sub>           | 59             | <b>High-end of ADC</b> (analog-to-digital conversion) reference resistor.  |
| AV <sub>SS</sub>             | 60             | <b>Ground</b> , analog part. For ADC receiver and reference voltage.   |
| AV <sub>DD</sub>             | 61             | <b>Power supply</b> , analog part (+5 V). For ADC receiver and reference voltage.  |
| P5.7/ADC7<br>to<br>P5.0/ADC0 | 62 to 68,<br>1 | <b>P5.7 to P5.0:</b> 8-bit input port lines;<br><b>ADC7 to ADC0:</b> eight analog ADC inputs   |



## 8-bit microcontroller

## P83C562; P80C562

### 7 FUNCTIONAL DESCRIPTION

The P8xC562 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control 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 P8xC562 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 P8xC562 can be expanded using standard memories and peripherals.

The P8xC562 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.

### 8 MEMORY ORGANIZATION

The Central Processing Unit (CPU) manipulates operands in three memory spaces; these are the 64 kbyte external data memory, 256 byte internal data memory and the 64 kbyte internal and external program memory. The internal data memory is divided into 3 sections: the lower 128 bytes of RAM, the upper 128 bytes of RAM and the 128 byte Special Function Register memory (see Fig.4). Figure 5 shows the Special Function Registers memory map. Internal RAM locations 0 to 127 are directly and indirectly addressable. Internal RAM locations 128 to 155 are only indirectly addressable. The Special Function Register locations 128 to 255 are only directly addressable.

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, a scratch pad area and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit Stack Pointer. All registers except the Program Counter and the four 8-register banks reside in the Special Function Register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, ADC and PWM registers, timers and serial port registers. There are 120 addressable bit locations in the SFR address space.

The P8xC562 contains 256 bytes of internal data RAM and 52 Special Function Registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the Program Counter.

The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 kbyte program memory address space.

#### 8.1 Program Memory

The program memory address space of the P83C562 consists of internal and external memory. The P83C562 has 8 kbytes of program memory on-chip. The program memory can be externally expanded up to 64 kbytes. If the  $\overline{EA}$  pin is held HIGH, the P83C562 executes out of the internal program memory unless the address exceeds 1FFFH then locations 2000H through to 0FFFFH are fetched from the external program memory. If the  $\overline{EA}$  pin is held LOW, the P83C562 fetches all instructions from the external memory. Figure 4 illustrates the program memory address space.

By setting a mask programmable security bit (i.e. user dependent) the ROM content is protected i.e. it cannot be read at any time by any test mode or by any instruction in the external program memory space. The MOVC instructions are the only ones which have access to program code in the internal or external program memory. The  $\overline{EA}$  input is latched during reset and is 'don't care' after reset. This implementation prevents from reading internal program code by switching from the external program memory to internal program memory during MOVC instruction or an instruction that handles immediate data. Table 2 lists the access to internal and external program memory by the MOVC instructions when the security bit has been set to a logic 1. If the security bit has been set to a logic 0 there are no restrictions for the MOVC instructions.

**Table 2** Memory access by the MOVC instruction

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

# 8-bit microcontroller

# P83C562; P80C562

## 8.2 Addressing

The P8xC562 has five methods for addressing source operands:

- 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 addressing is as follows:

- Registers in one of the four 8-register banks through Register, Direct or Register-Indirect

- 256 bytes of internal data RAM through Direct or Register-Indirect. Bytes 0 to 127 may be addressed directly/indirectly. Bytes 128 to 155 share their address locations with the SFR registers and so may only be addressed indirectly as data RAM
- Special Function Registers through Direct at address locations 128 to 255
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect.

The P8xC562 is classified as an 8-bit device since the internal ROM, RAM, Special Function Registers, Arithmetic Logic Unit and external data bus are all 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

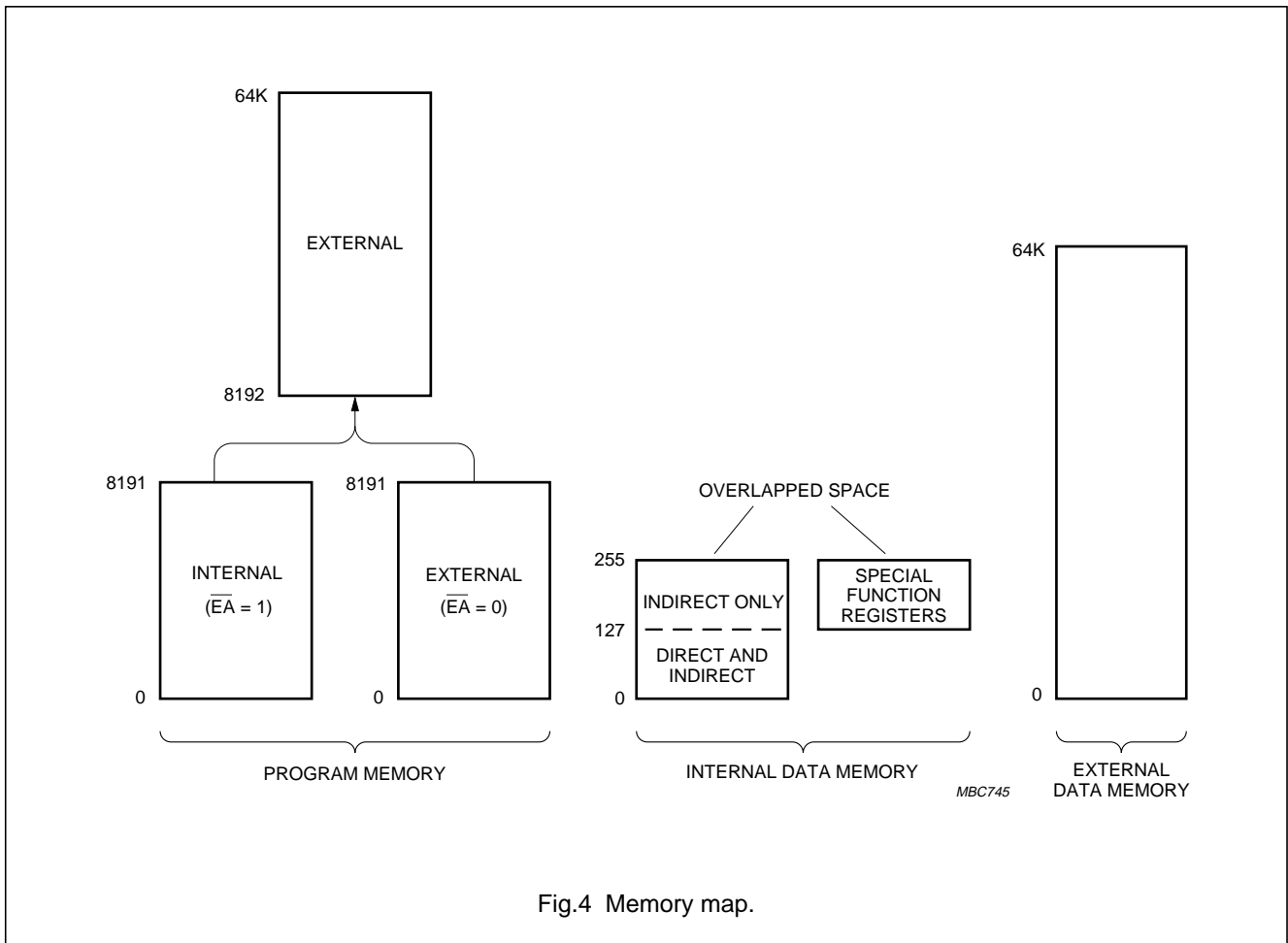


Fig.4 Memory map.

8-bit microcontroller

P83C562; P80C562

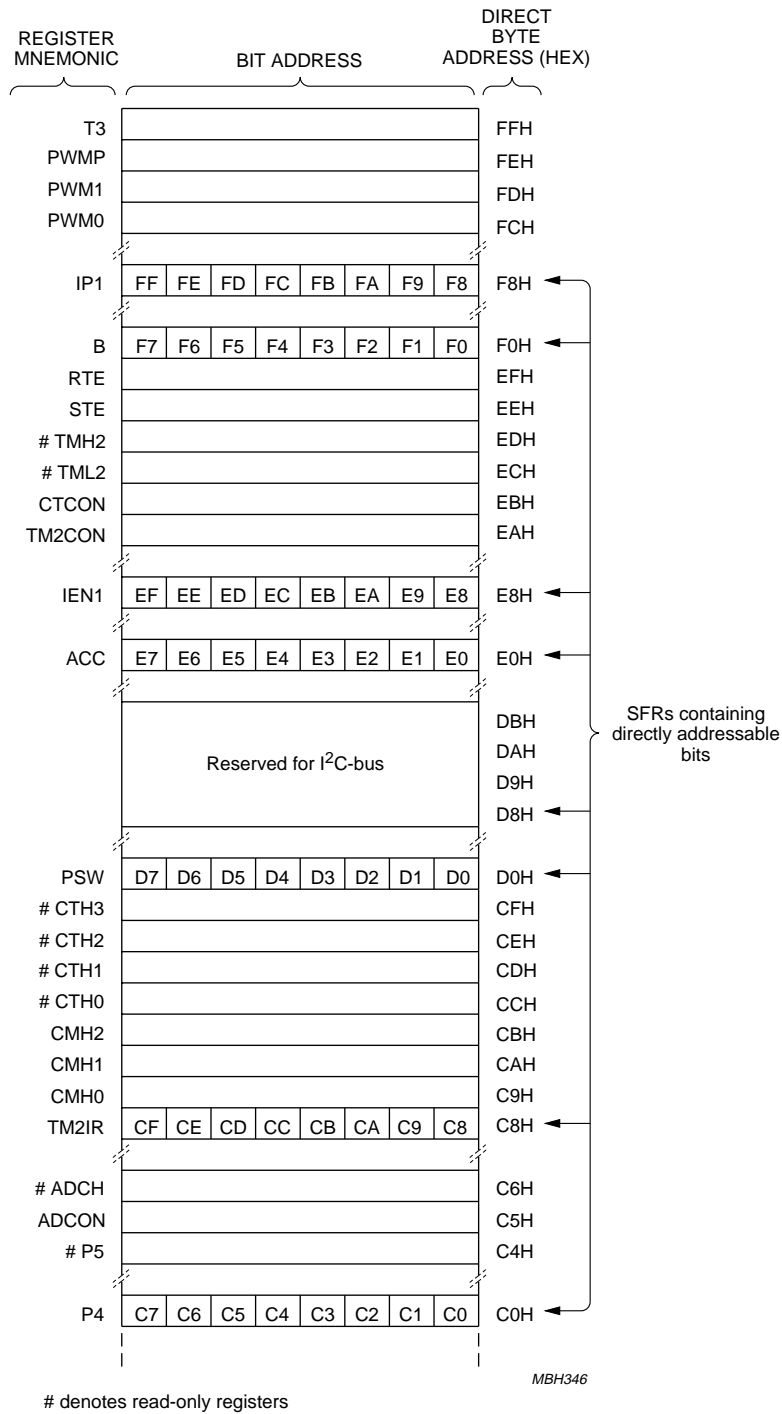


Fig.5 Special Function Register memory map.

8-bit microcontroller

P83C562; P80C562

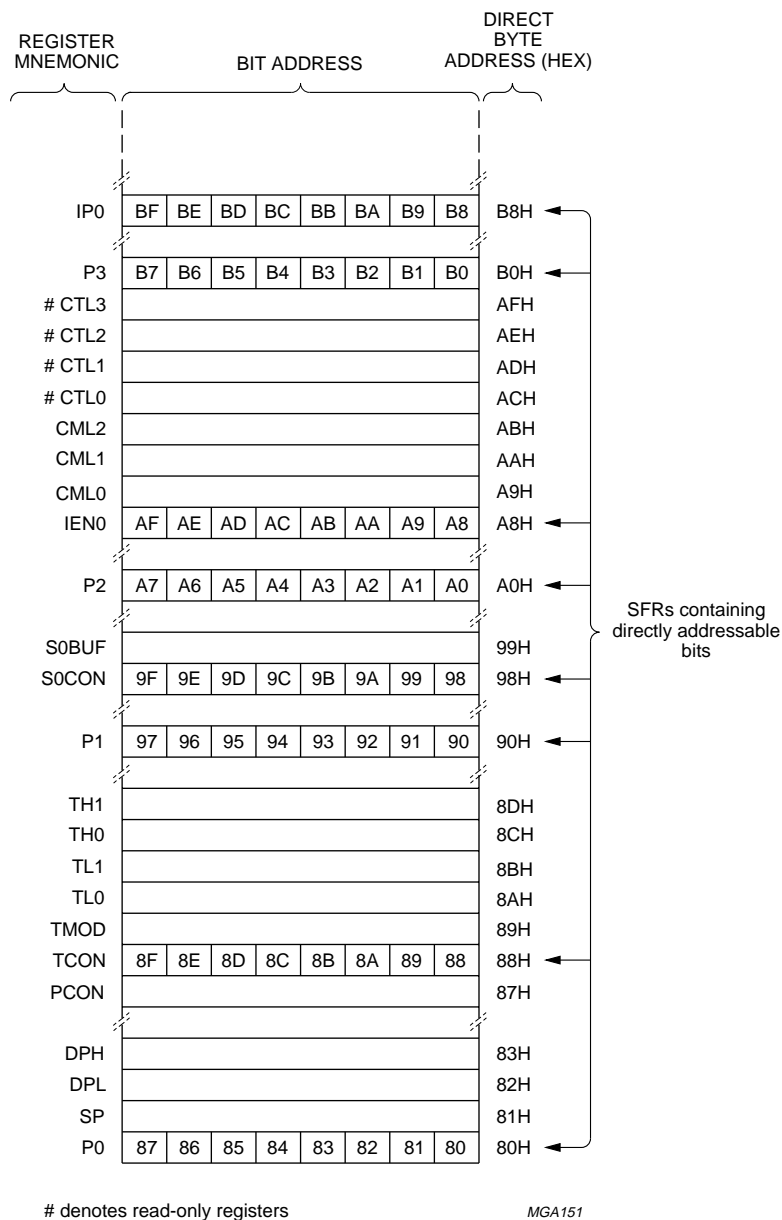


Fig.6 Special Function Register memory map (continued).

8-bit microcontroller

P83C562; P80C562

9 I/O FACILITIES

The P8xC562 has six 8-bit ports. Ports 0 to 3 are the same as in the 80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port.

Ports 0 to 5 perform the following alternative functions:

Port 0 Provides the multiplexed low-order address and data bus used for expanding the P8xC562 with standard memories and peripherals.

Port 1 is used for a number of special functions:

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

Port 2 Provides the high-order address bus when expanding the P8xC562 with external program memory and/or external data memory.

Port 3 Pins can be configured individually to provide:

- External interrupt request inputs
- Counter inputs
- Serial port receiver input and transmitter output
- Control signals to READ and WRITE external data memory.

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

Port 5 May be used in conjunction with the ADC interface. Unused analog inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analog signals. Channel-to-channel crosstalk should be taken into consideration when both digital and analog signals are simultaneously input to Port 5 (see Chapter 20).

All ports are bidirectional with the exception of Port 5 which is an input port. Alternative function bits which are not used may be used as normal bidirectional I/O pins.

The generation or use of a Port 1, Port 3 or Port 4 pin as an alternative function is carried out automatically by the P8xC562 provided the associated Special Function Register bit is set HIGH.

In addition to the standard 8-bit ports, the I/O facilities of the P8xC562 also include a number of special I/O lines.

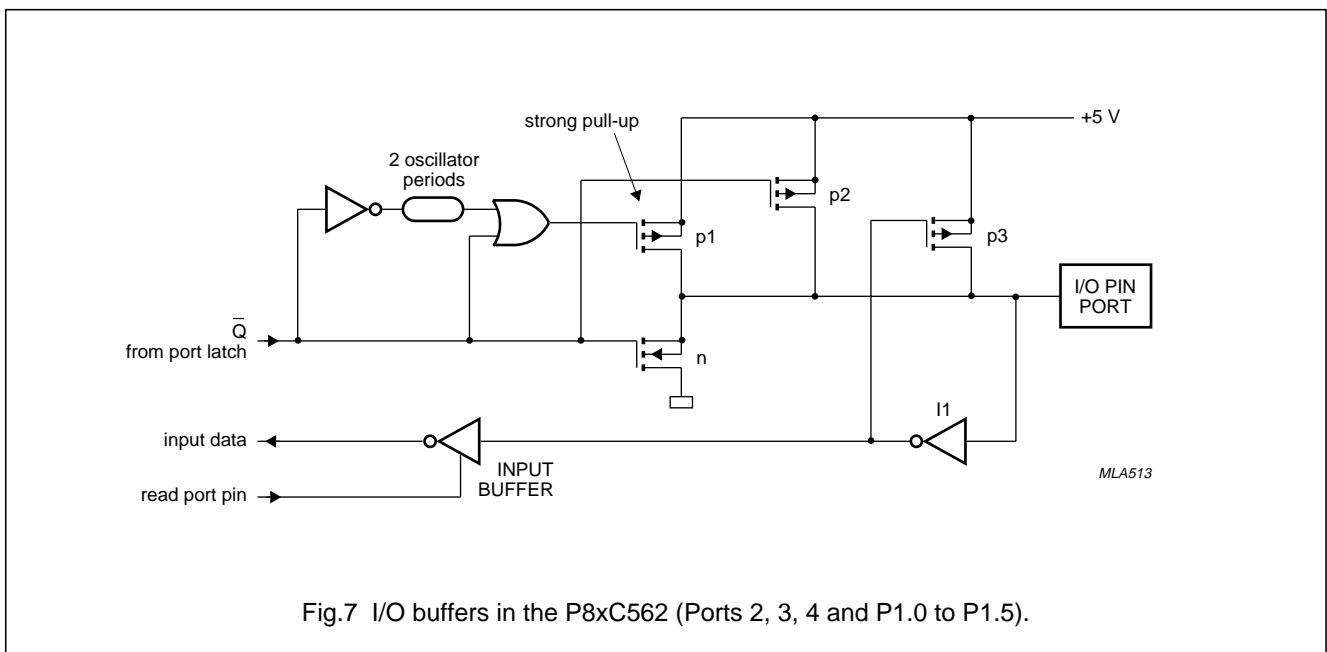


Fig.7 I/O buffers in the P8xC562 (Ports 2, 3, 4 and P1.0 to P1.5).

# 8-bit microcontroller

# P83C562; P80C562

## 10 PULSE WIDTH MODULATED OUTPUTS

Two pulse width modulated output channels are provided with the P8x562. These channels output pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP which generates the clock for the counter. Both the prescaler and counter are common to both PWM channels. The 8-bit counter counts modulo 255 i.e. from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1.

Provided the contents of either of these registers is greater than the counter value, the output of  $\overline{PWM0}$  or  $\overline{PWM1}$  is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse width ratio is therefore defined by the contents of the registers PWM0 and PWM1.

The pulse width ratio is in the range of 0 to 255/255 and may be programmed in increments of 1/255.

The repetition frequency  $f_{PWM}$ , at the  $\overline{PWMn}$  outputs is

$$f_{PWM} = \frac{f_{osc}}{2 \times (1 + PWMP) \times 255}$$

When using an oscillator frequency of 16 MHz for example, the above formula would give a repetition frequency range of 123 Hz to 31.4 kHz.

By loading the PWM registers with either 00H or FFH, the PWM outputs can be retained at a constant HIGH or LOW level respectively. When loading FFH to the PWM registers, the 8-bit counter will never actually reach this value. Both  $\overline{PWMn}$  output pins are driven by push-pull drivers, and are not shared with any other function.

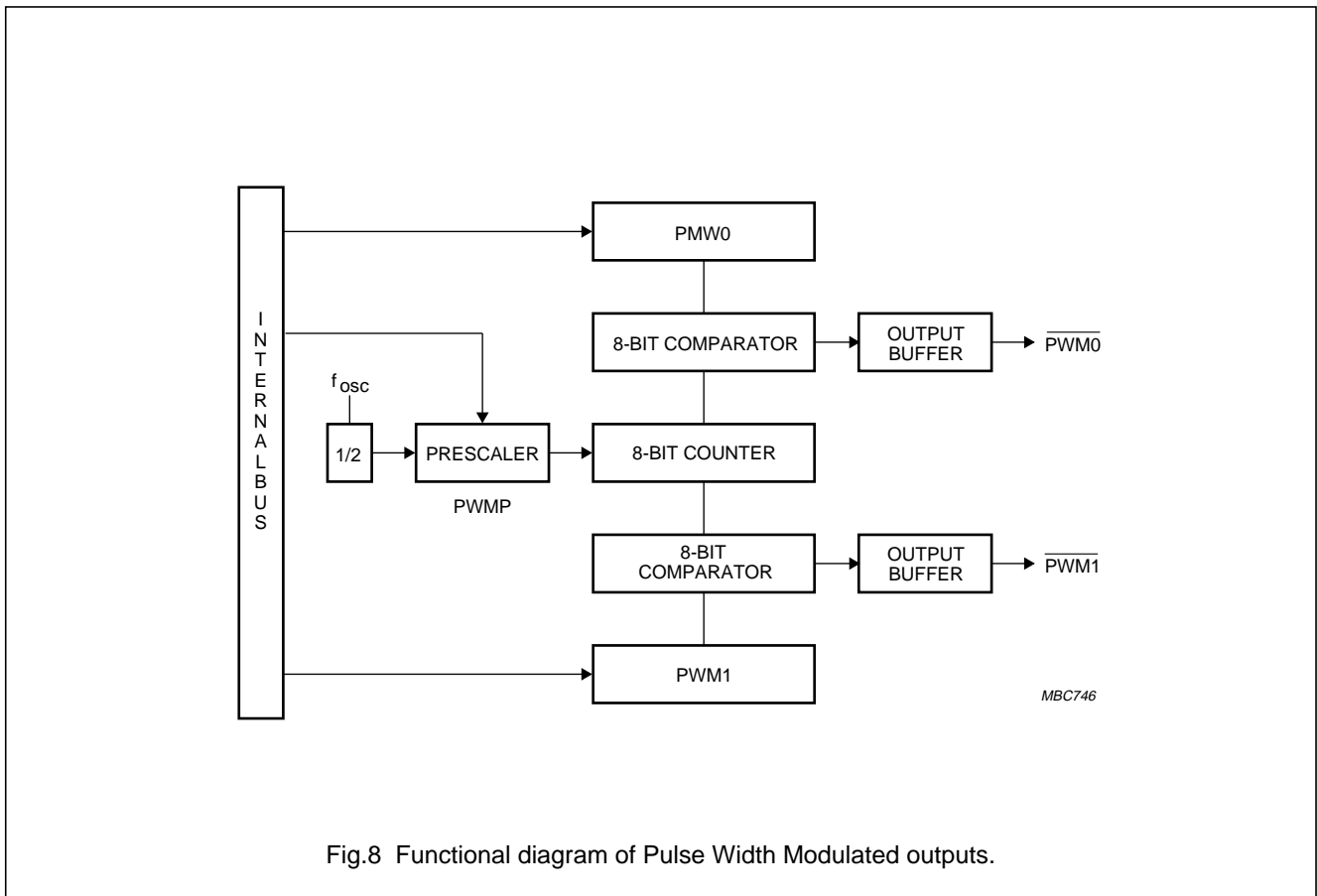


Fig.8 Functional diagram of Pulse Width Modulated outputs.

## 8-bit microcontroller

## P83C562; P80C562

**10.1 Prescaler Frequency Control Register (PWMP)****Table 3** Prescaler Frequency Control Register (SFR address FEH)

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b> | <b>3</b> | <b>2</b> | <b>1</b> | <b>0</b> |
| PWMP.7   | PWMP.6   | PWMP.5   | PWMP.4   | PWMP.3   | PWMP.2   | PWMP.1   | PWMP.0   |

**Table 4** Description of PWMP bits

| <b>BIT</b>   | <b>SYMBOL</b>          | <b>DESCRIPTION</b>   |
|--------------|------------------------|--|
| 7<br>to<br>0 | PWMP.7<br>to<br>PWMP.0 | <b>Prescaler division factor.</b><br>The prescaler division factor = (PWMP) + 1. |

**10.2 Pulse Width Register 0 (PWM0)****Table 5** Pulse Width Register 0 (SFR address FCH)

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b> | <b>3</b> | <b>2</b> | <b>1</b> | <b>0</b> |
| PWM0.7   | PWM0.6   | PWM0.5   | PWM0.4   | PWM0.3   | PWM0.2   | PWM0.1   | PWM0.0   |

**Table 6** Description of PWM0 bits

| <b>BIT</b>   | <b>SYMBOL</b>          | <b>DESCRIPTION</b>  |
|--------------|------------------------|---|
| 7<br>to<br>0 | PWM0.7<br>to<br>PWM0.0 | <b>Pulse width ratio.</b><br>LOW/HIGH ratio of $\overline{\text{PWMn}}$ signals = $\frac{(\text{PWMn})}{255 - (\text{PWMn})}$ |

**10.3 Pulse Width Register 1 (PWM1)****Table 7** Pulse Width Register 1 (SFR address FDH)

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b> | <b>3</b> | <b>2</b> | <b>1</b> | <b>0</b> |
| PWM1.7   | PWM1.6   | PWM1.5   | PWM1.4   | PWM1.3   | PWM1.2   | PWM1.1   | PWM1.0   |

**Table 8** Description of PWM1 bits

| <b>BIT</b>   | <b>SYMBOL</b>          | <b>DESCRIPTION</b>  |
|--------------|------------------------|---|
| 7<br>to<br>0 | PWM1.7<br>to<br>PWM1.0 | <b>Pulse width ratio.</b><br>LOW/HIGH ratio of $\overline{\text{PWMn}}$ signals = $\frac{(\text{PWMn})}{255 - (\text{PWMn})}$ |

# 8-bit microcontroller

# P83C562; P80C562

## 11 ANALOG-TO-DIGITAL CONVERTER (ADC)

The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register and the result is stored in Special Function Register ADCH.

An ADC conversion in progress is unaffected by an external or software ADC start. The result of a completed conversion remains unaffected provided ADCI = 1. While ADCS = 1 or ADCI = 1, a new ADC start will be blocked and consequently lost.

An ADC conversion already in progress is aborted when the Idle or Power-down mode is entered. The result of a completed conversion (ADCI = 1) remains unaffected when entering the Idle mode.

If ADCI is cleared by software and ADCS is set at the same time, a new analog-to-digital conversion with the same channel number, may be started. However, it is recommended to reset ADCI before ADCS is set.

## 11.1 Analog input pins

The analog input circuitry consists of an 8-input analog multiplexer and an ADC with 8-bit resolution. The analog reference voltage and analog power supplies are connected via separate input pins. The conversion takes 24 machine cycles i.e. 18  $\mu$ s at an oscillator frequency of 16 MHz.

The ADC is controlled using the ADC Control Register (ADCON). Input channels are selected by the analog multiplexer, using bits AADR.0 to AADR.2 in ADCON.

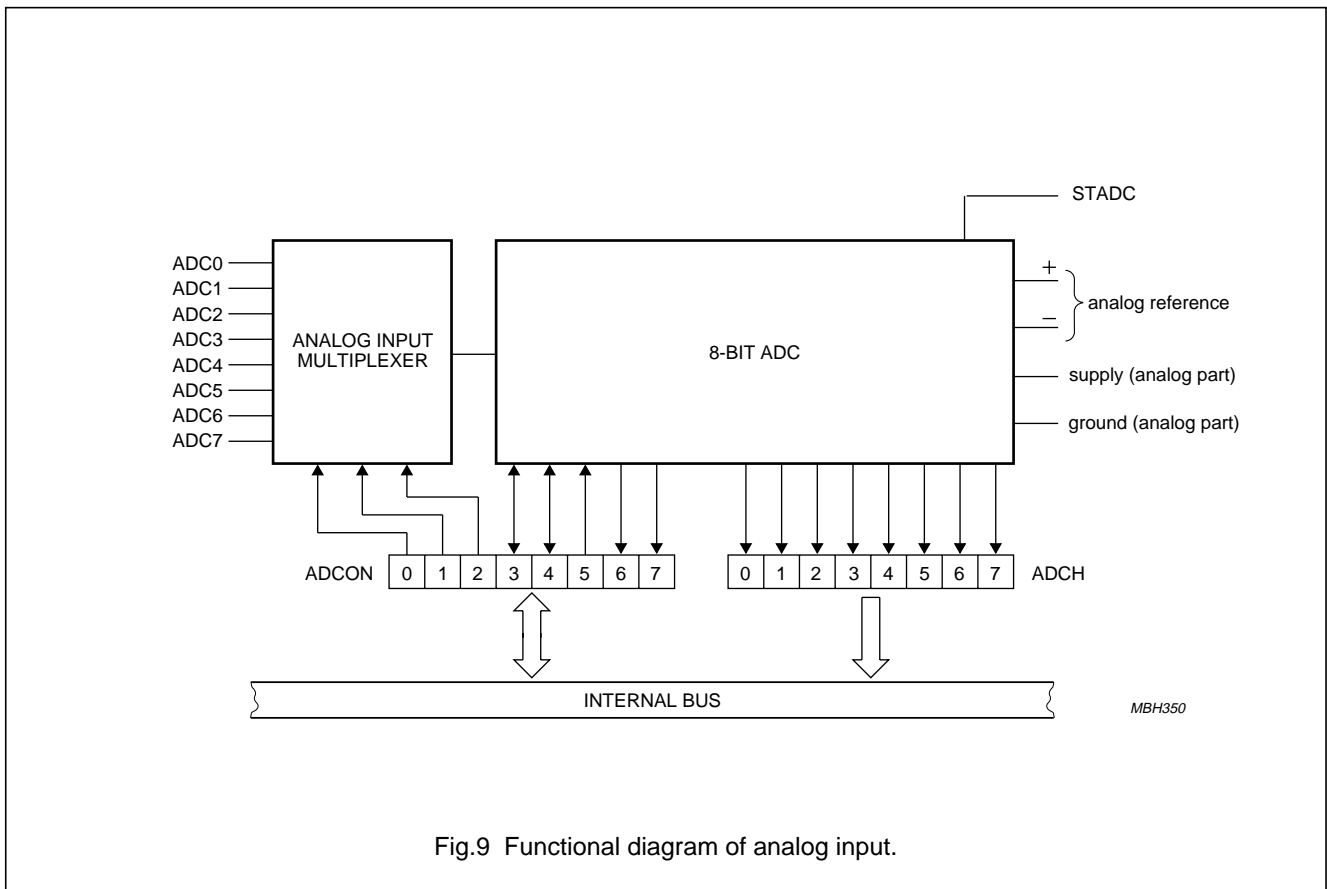


Fig.9 Functional diagram of analog input.



## 8-bit microcontroller

## P83C562; P80C562

## 11.2 ADC Control Register (ADCON)

Table 9 ADC Control Register (SFR address C5H)

|   |   |      |      |      |       |       |       |
|---|---|------|------|------|-------|-------|-------|
| 7 | 6 | 5    | 4    | 3    | 2     | 1     | 0     |
| – | – | ADEX | ADCI | ADCS | AADR2 | AADR1 | AADR0 |

Table 10 Description of ADCON bits

| BIT | SYMBOL | DESCRIPTION  |
|-----|--------|--|
| 7   | –      | These two bits are reserved.   |
| 6   | –      |  |
| 5   | ADEX   | <b>Enable external start:</b> start of conversion by STADC. If ADEX = 0, then conversion can not be started externally by STADC (only by software by setting ADCS). If ADEX = 1, then conversion can be started externally by a rising edge on STADC or by software.   |
| 4   | ADCI   | <b>ADC interrupt flag:</b> this flag is set when an analog-to-digital conversion result is ready to be read. An interrupt is invoked if it is enabled. The flag must be cleared by the interrupt service routine. While this flag is set, the ADC cannot start a new conversion. ADCI cannot be set by software.   |
| 3   | ADCS   | <b>ADC start and status:</b> setting this bit starts an ADC conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset immediately after the interrupt flag has been set. ADCS can not be reset by software nor can a new conversion be started if either ADCS or ADCI is HIGH. |
| 2   | AADR.2 | <b>Analog input select:</b> these three bits are used to select one of the eight analog inputs of Port 5, for conversion. A selection can only be made when ADCI and ADCS are both LOW. AADR2 is the most significant bit (e.g. 100 selects the ADC4 analog input channel).  |
| 1   | AADR.1 |  |
| 0   | AADR.0 |  |

Table 11 Function of ADCI and ADCS bits

| ADCI | ADCS | OPERATION  |
|------|------|--|
| 0    | 0    | ADC not busy, a conversion can be started.   |
| 0    | 1    | ADC busy, start of a new conversion is blocked.  |
| 1    | 0    | Conversion completed; start of a new conversion is blocked.                            |
| 1    | 1    | Intermediate status for a maximum of one machine cycle before conversion is completed. |

## 8-bit microcontroller

## P83C562; P80C562

**12 TIMER/ COUNTERS**

The P8xC562 contains:

- Three 16-bit timer/event counters: Timer 0, Timer 1 and Timer 2
- One 8-bit Watchdog Timer.

**12.1 Timer 0 and Timer 1**

Timer 0 and Timer 1 may be programmed to carry out the following operations:

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

Timer 0 and Timer 1 can also be programmed independently to operate in three 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.

Timer 0 can be programmed to operate in an additional mode as follows:

Mode 3 one 8-bit time-interval or event counter and one 8-bit time-interval counter.

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

The frequency handling range of these counters with a 16 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1.33 MHz; a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0.66 MHz when programmed for external inputs.

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

The counters 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 (or automatic reload value), with the exception of Mode 3 as previously described.

**12.2 Timer T2 Capture and Compare Logic**

Timer T2 is a 16-bit timer/counter which has, coupled to it, capture and compare facilities. The operational diagram is shown in Fig.10.

The 16-bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with  $\frac{1}{12}$  of the oscillator frequency, or with positive edges on the T2 input, or it is switched to the off position. The prescaler is cleared if its division factor or its input source is changed, or if the timer/counter is reset. T2 is readable on-the-fly, but possesses no extra read latches; this means that software precautions have to be taken against misinterpretation on overflow from least to most significant byte during a read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit Capture Registers: CT0, CT1, CT2 and CT3. These registers are loaded with the contents of T2 and an interrupt is requested upon receipt of the input signals CT0I, CT1I, CT2I or CT3I. These input signals are shared with Port 1. Using the Capture Register (CTCON), these inputs may invoke capture and interrupt request on a positive or negative edge or on both edges. If neither a positive nor a negative edge is selected for a capture input, no capture or interrupt request can be generated by this input.

The contents of the Compare Registers CM0, CM1 and CM2 are continually compared with the counter value of Timer 2. When a match is found an interrupt may be invoked. Using the match signal of CM0, the controller sets bits 0 to 5 of Port 4, if the corresponding bits of the Set Enable Register are logic 1s.

Considering a match with CM1, if the corresponding bits of the Reset/toggle Enable Register (RTE) are logic 1, then the controller will use the match signal to reset bits 0 to 5 of Port 4. Bits 6 and 7 of Port 4 may be toggled by the signal that indicates a match of Timer T2 and CM2 if the corresponding bits of RTE are logic 1. CM0, CM1 and CM2 are reset by the RST signal.

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

All interrupt flags must be reset by software.

8-bit microcontroller

P83C562; P80C562

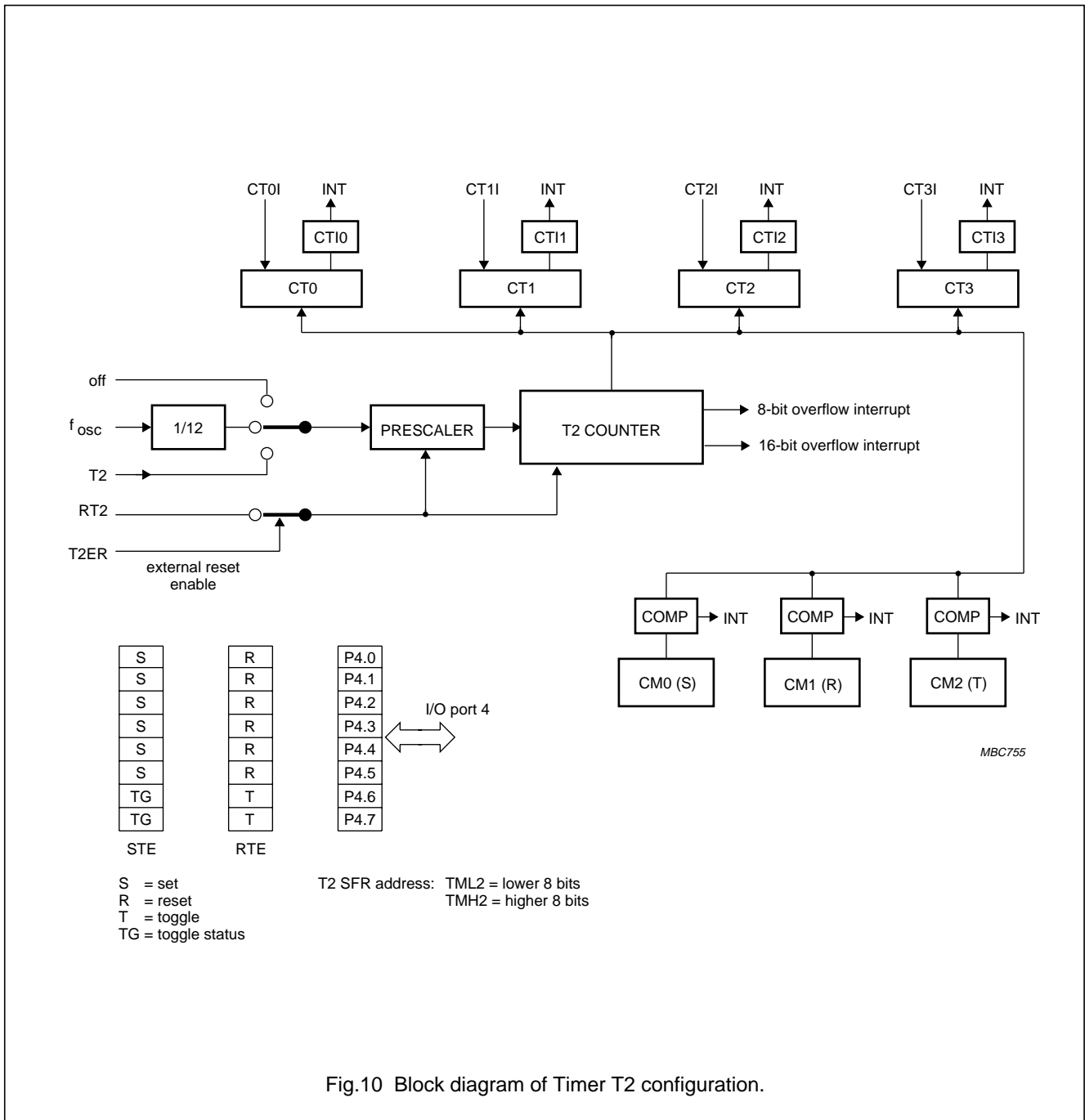


Fig.10 Block diagram of Timer T2 configuration.

## 8-bit microcontroller

## P83C562; P80C562

## 12.2.1 T2 CONTROL REGISTER (TM2CON)

**Table 12** T2 Control Register (SFR address EAH)

| 7     | 6     | 5    | 4    | 3    | 2    | 1     | 0     |
|-------|-------|------|------|------|------|-------|-------|
| T2IS1 | T2IS0 | T2ER | T2B0 | T2P1 | T2P0 | T2MS1 | T2MS0 |

**Table 13** Description of TM2CON bits

| BIT | SYMBOL | DESCRIPTION                               |
|-----|--------|---|
| 7   | T2IS1  | Timer 2 16-bit overflow interrupt select. |
| 6   | T2IS0  | Timer 2 byte overflow interrupt select.   |
| 5   | T2ER   | Timer 2 external reset enable.            |
| 4   | T2B0   | Timer 2 byte overflow interrupt flag.     |
| 3   | T2P1   | Timer 2 prescaler select (see Table 14).  |
| 2   | T2P0   |   |
| 1   | T2MS1  | Timer 2 mode select (see Table 15).       |
| 0   | T2MS0  |   |

**Table 14** Timer 2 prescaler select

| T2P1 | T2P0 | T2 CLOCK                   |
|------|------|----------------------------|
| 0    | 0    | Clock source               |
| 0    | 1    | $\frac{1}{2}$ clock source |
| 1    | 0    | $\frac{1}{4}$ clock source |
| 1    | 1    | $\frac{1}{8}$ clock source |

**Table 15** Timer 2 mode select

| T2MS1 | T2MS0 | MODE  |
|-------|-------|---|
| 0     | 0     | Timer T2 is halted                              |
| 0     | 1     | T2 clock source = $\frac{1}{12} \times f_{osc}$ |
| 1     | 0     | Test mode; do not use                           |
| 1     | 1     | T2 clock source = pin T2                        |

## 8-bit microcontroller

## P83C562; P80C562

## 12.2.2 CAPTURE CONTROL REGISTER (CTCON)

**Table 16** Capture Control Register (SFR address EBH)

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b> | <b>3</b> | <b>2</b> | <b>1</b> | <b>0</b> |
| CTN3     | CTP3     | CTN2     | CTP2     | CTN1     | CTP1     | CTN0     | CTP0     |

**Table 17** Description of CTCON bits

| <b>BIT</b> | <b>SYMBOL</b> | <b>DESCRIPTION</b>                            |
|------------|---------------|---|
| 7          | CTN3          | Interrupt triggered on negative edge of CT3I. |
| 6          | CTP3          | Interrupt triggered on positive edge of CT3I. |
| 5          | CTN2          | Interrupt triggered on negative edge of CT2I. |
| 4          | CTP2          | Interrupt triggered on positive edge of CT2I. |
| 3          | CTN1          | Interrupt triggered on negative edge of CT1I. |
| 2          | CTP1          | Interrupt triggered on positive edge of CT1I. |
| 1          | CTN0          | Interrupt triggered on negative edge of CT0I. |
| 0          | CTP0          | Interrupt triggered on positive edge of CT0I. |

## 12.2.3 INTERRUPT FLAG REGISTER (TM2IR)

**Table 18** Interrupt Flag Register (SFR address C8H)

|          |          |          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|----------|----------|
| <b>7</b> | <b>6</b> | <b>5</b> | <b>4</b> | <b>3</b> | <b>2</b> | <b>1</b> | <b>0</b> |
| T2OV     | CMI2     | CMI1     | CMI0     | CTI3     | CTI2     | CTI1     | CTI0     |

**Table 19** Description of TM2IR bits (see notes 1 and 2)

| <b>BIT</b> | <b>SYMBOL</b> | <b>DESCRIPTION</b>                  |
|------------|---------------|-------------------------------------|
| 7          | T2OV          | T2: 16-bit overflow interrupt flag. |
| 6          | CMI2          | CM2: interrupt flag.                |
| 5          | CMI1          | CM1: interrupt flag.                |
| 4          | CMI0          | CM0: interrupt flag.                |
| 3          | CTI3          | CT3: interrupt flag.                |
| 2          | CTI2          | CT2: interrupt flag.                |
| 1          | CTI1          | CT1: interrupt flag.                |
| 0          | CTI0          | CT0: interrupt flag.                |

**Notes**

1. Interrupt Enable Register 1 (IEN1) is used to enable/disable Timer 2 interrupts.
2. Interrupt Priority Register 1 (IP1) is used to determine the Timer 2 interrupt priority.

## 8-bit microcontroller

## P83C562; P80C562

## 12.2.4 SET ENABLE REGISTER (STE)

**Table 20** Set Enable Register (SFR address EEH)

| 7    | 6    | 5    | 4    | 3    | 2    | 1    | 0    |
|------|------|------|------|------|------|------|------|
| TG47 | TG46 | SP45 | SP44 | SP43 | SP42 | SP41 | SP40 |

**Table 21** Description of STE bits (see notes 1 and 2)

| BIT | SYMBOL | DESCRIPTION   |
|-----|--------|---|
| 7   | TG47   | If HIGH then P4.7 is reset on the next toggle, if LOW P4.7 is set on the next toggle. |
| 6   | TG46   | If HIGH then P4.6 is reset on the next toggle, if LOW P4.6 is set on the next toggle. |
| 5   | SP45   | If HIGH the P4.5 is set on a match of CM0 and T2.                                     |
| 4   | SP44   | If HIGH the P4.4 is set on a match of CM0 and T2.                                     |
| 3   | SP43   | If HIGH the P4.3 is set on a match of CM0 and T2.                                     |
| 2   | SP42   | If HIGH the P4.2 is set on a match of CM0 and T2.                                     |
| 1   | SP41   | If HIGH the P4.1 is set on a match of CM0 and T2.                                     |
| 0   | SP40   | If HIGH the P4.0 is set on a match of CM0 and T2.                                     |

**Notes**

1. If STE.n is LOW then P4.n is not affected by a match of CM0 and T2 (n = 0 to 5).
2. STE.6 and STE.7 are read only.

## 12.2.5 RESET/TOGGLE ENABLE REGISTER (RTE)

**Table 22** Reset/toggle enable register (SFR address EFH)

| 7    | 6    | 5    | 4    | 3    | 2    | 1    | 0    |
|------|------|------|------|------|------|------|------|
| TP47 | TP46 | RP45 | RP44 | RP43 | RP42 | RP41 | RP40 |

**Table 23** Description of RTE bits (note 1)

| BIT | SYMBOL | DESCRIPTION  |
|-----|--------|--|
| 7   | TP47   | If HIGH then P4.7 toggles on a match of CM2 and T2.  |
| 6   | TP46   | If HIGH then P4.6 toggles on a match of CM2 and T2.  |
| 5   | RP45   | If HIGH then P4.5 is reset on a match of CM1 and T2. |
| 4   | RP44   | If HIGH then P4.4 is reset on a match of CM1 and T2. |
| 3   | RP43   | If HIGH then P4.3 is reset on a match of CM1 and T2. |
| 2   | RP42   | If HIGH then P4.2 is reset on a match of CM1 and T2. |
| 1   | RP41   | If HIGH then P4.1 is reset on a match of CM1 and T2. |
| 0   | RP40   | If HIGH then P4.0 is reset on a match of CM1 and T2. |

**Note**

1. If RTE.n is LOW then P4.n is not affected by a match of CM1 and T2 or CM2 and T2. For more information, refer to the 8051-based "8-bit Microcontrollers Data Handbook IC20".

# 8-bit microcontroller

# P83C562; P80C562

### 12.3 Watchdog Timer (T3)

In addition to Timer T2 and the standard timers, a Watchdog Timer is also available, consisting of an 11-bit prescaler and an 8-bit timer. The functional diagram of the Watchdog Timer is shown in Fig.11. The timer is incremented every t seconds,

$$\text{where: } t = \frac{12 \times 2048}{f_{\text{OSC}}}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at the RST pin.

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

The Watchdog Timer can only be reloaded if the condition flag WLE in the Power Control Register has been previously set by software. At the moment the counter is loaded the condition flag is automatically cleared.

The timer interval between the timer's reloading and occurrence of a reset, is dependent upon the reloaded value. For example, this may range from 2 ms to 0.5 s when using an oscillator frequency of 12 MHz. In the Idle state the Watchdog Timer and reset circuitry remain active.

The Watchdog Timer is controlled by the Enable Watchdog pin ( $\bar{E}W$ ). A logic 0 enables the Watchdog Timer and disables the Power-down mode. A logic 1 disables the Watchdog Timer and enables the Power-down mode.

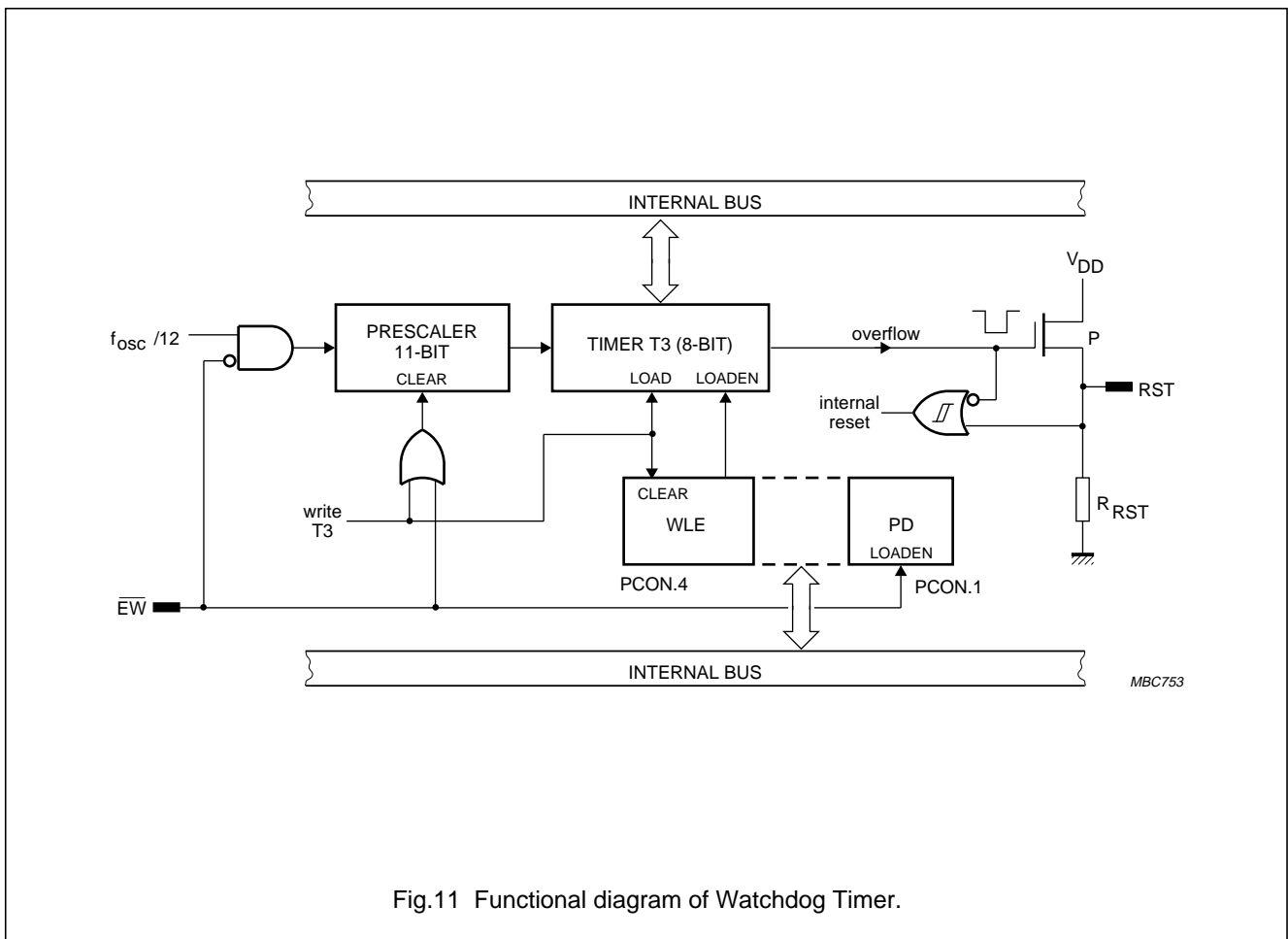


Fig.11 Functional diagram of Watchdog Timer.

## 8-bit microcontroller

## P83C562; P80C562

**13 SERIAL I/O**

The P8x562 is equipped with a full duplex UART port and is identical to the serial port of the 80C51 (see 'Single-chip 8-bit Microcontrollers User Manual' .

**14 INTERRUPT SYSTEM**

External events and the real-time driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. The interrupt system is shown in Fig.12. Interrupt response latency is from 2.25  $\mu$ s to 6  $\mu$ s when using a 16 MHz crystal.

The P8x562 acknowledges interrupt requests from 14 sources as follows:

- $\overline{\text{INT0}}$  and  $\overline{\text{INT1}}$ : externally via pins P3.2/ $\overline{\text{INT0}}$  and P3.3/ $\overline{\text{INT1}}$  respectively
- Timer 0 and Timer 1: from the two internal counters
- Timer T2 (8 separate interrupts): 4 capture interrupts, 3 compare interrupts and an overflow interrupt. If the Capture Register remains unused and its contents are 'don't care', then the corresponding input pin CTnI may be used as a positive and/or negative edge triggered external interrupt.
- ADC conversion completed interrupt
- UART serial I/O port interrupt.

Each interrupt vectors to a separate location in program memory for its service routine. Each source can be individually enabled or disabled by a corresponding bit in the IEN0 or IEN1 registers, in addition each interrupt may be programmed to a high or low priority level using the corresponding bit in the IP0 or IP1 registers. All enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated; an active LOW level allows 'wire-ORing' of several interrupt sources to the input pin.

**14.1 Interrupt Vectors**

Table 24 gives the vector address in Program Memory where the appropriate interrupt service routine is located.

**Table 24** Interrupt vectors

| SOURCE              | SYMBOL | VECTOR |
|---------------------|--------|--------|
| External 0          | X0     | 0003H  |
| Timer 0 overflow    | T0     | 000BH  |
| External 1          | X1     | 0013H  |
| Timer 1 overflow    | T1     | 001BH  |
| Serial I/O 0 (UART) | S0     | 0023H  |
| T2 capture 0        | CT0    | 0033H  |
| T2 capture 1        | CT1    | 003BH  |
| T2 capture 2        | CT2    | 0043H  |
| T2 capture 3        | CT3    | 004BH  |
| ADC completion      | ADC    | 0053H  |
| T2 compare 0        | CM0    | 005BH  |
| T2 compare 1        | CM1    | 0063H  |
| T2 compare 2        | CM2    | 006BH  |
| T2 overflow         | T2     | 0073H  |

**14.2 Interrupt priority**

Each interrupt source can be either high priority or low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, ADC, T0, CT0, CM0, X1, CT1, CM1, T1, CT2, CM2, S0, CT3, T2.

A low priority interrupt routine can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.



8-bit microcontroller

P83C562; P80C562

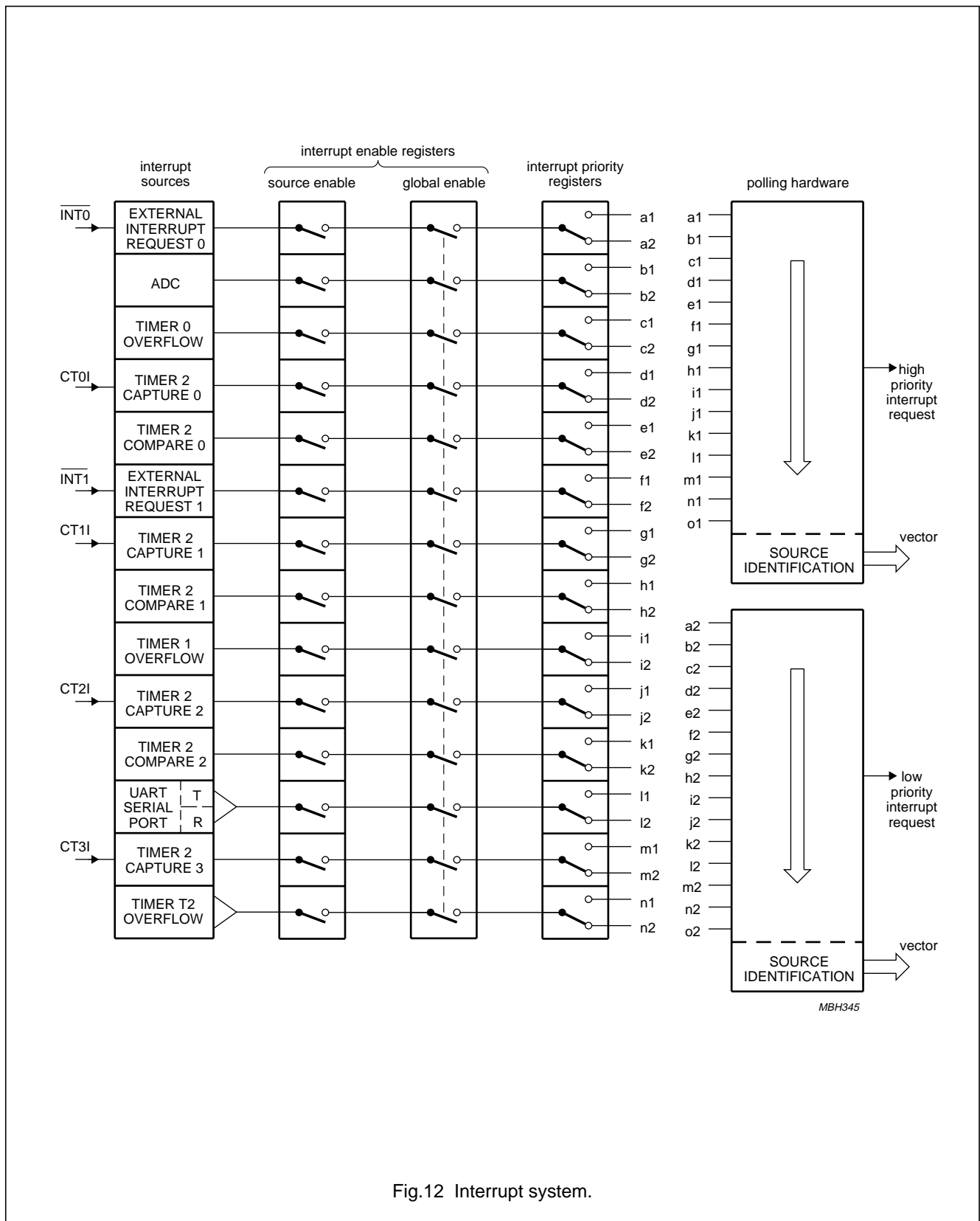


Fig.12 Interrupt system.

## 8-bit microcontroller

## P83C562; P80C562

**14.3 Interrupt Enable and Priority Registers**

## 14.3.1 INTERRUPT ENABLE REGISTER 0 (IEN0)

**Table 25** Interrupt Enable Register 0 (SFR address A8H)

| 7  | 6   | 5 | 4   | 3   | 2   | 1   | 0   |
|----|-----|---|-----|-----|-----|-----|-----|
| EA | EAD | – | ES0 | ET1 | EX1 | ET0 | EX0 |

**Table 26** Description of IEN0 bits (note 1)

| BIT | SYMBOL | DESCRIPTION   |
|-----|--------|---|
| 7   | EA     | <b>General enable/disable control.</b> If EA = 0, then no interrupt is enabled. If EA =1, then any individually enabled interrupt will be accepted. |
| 6   | EAD    | Enable ADC interrupt.   |
| 5   | –      | Reserved.   |
| 4   | ES0    | Enable SIO (UART) interrupt.  |
| 3   | ET1    | Enable Timer 1 interrupt.   |
| 2   | EX1    | Enable External interrupt.  |
| 1   | ET0    | Enable Timer 0 interrupt.   |
| 0   | EX0    | Enable External 0 interrupt.  |

**Note**

- Logic 0 = interrupt disabled; Logic 1 = interrupt enabled.

## 14.3.2 INTERRUPT ENABLE REGISTER 1 (IEN1)

**Table 27** Interrupt Enable Register 1 (SFR address E8H)

| 7   | 6    | 5    | 4    | 3    | 2    | 1    | 0    |
|-----|------|------|------|------|------|------|------|
| ET2 | ECM2 | ECM1 | ECM0 | ECT3 | ECT2 | ECT1 | ECT0 |

**Table 28** Description of IEN1 bits (note 1)

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

**Note**

- Logic 0 = interrupt disabled; Logic 1 = interrupt enabled.

## 8-bit microcontroller

## P83C562; P80C562

## 14.3.3 INTERRUPT PRIORITY REGISTER 0 (IP0)

**Table 29** Interrupt Priority Register 0 (SFR address B8H)

| 7 | 6   | 5 | 4   | 3   | 2   | 1   | 0   |
|---|-----|---|-----|-----|-----|-----|-----|
| – | PAD | – | PS0 | PT1 | PX1 | PT0 | PX0 |

**Table 30** Description of IP0 bits (note 1)

| BIT | SYMBOL | DESCRIPTION                           |
|-----|--------|---------------------------------------|
| 7   | –      | Reserved.                             |
| 6   | PAD    | ADC interrupt priority level.         |
| 5   | –      | Reserved.                             |
| 4   | PS0    | SIO0 (UART) interrupt priority level. |
| 3   | PT1    | Timer 1 interrupt priority level.     |
| 2   | PX1    | External interrupt 1 priority level.  |
| 1   | PT0    | Timer 0 interrupt priority level.     |
| 0   | PX0    | External interrupt 0 priority level.  |

**Note**

1. A logic 0 = low priority; a logic 1 = high priority.

## 14.3.4 INTERRUPT PRIORITY REGISTER 1 (IP1)

**Table 31** Interrupt Priority Register 1 (SFR address F8H)

| 7   | 6    | 5    | 4    | 3    | 2    | 1    | 0    |
|-----|------|------|------|------|------|------|------|
| PT2 | PCM2 | PCM1 | PCM0 | PCT3 | PCT2 | PCT1 | PCT0 |

**Table 32** Description of IP1 bits (note 1)

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

**Note**

1. A logic 0 = low priority; a logic 1 = high priority.

## 8-bit microcontroller

## P83C562; P80C562

**15 REDUCED POWER MODES****15.1 Idle and Power-down operation**

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the CPU is halted. The Idle and Power-down clock configuration is shown in Fig.13. The following functions are switched off when the processor enters the Idle mode.

- Timer T2 - stopped and reset
- PWM0 and PWM1 - reset, output HIGH
- ADC - aborted if in progress.

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 T3
- SIO
- 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. The PD bit can only be set if the  $\overline{EW}$  input is HIGH.

**15.1.1 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 the external pins during Idle mode is shown in Table 33.

There are two ways to terminate the Idle mode:

- Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating the Idle mode.

The interrupt is serviced, and following the 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.

- The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of the Watchdog Timer (T3). Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods but at least 2  $\mu$ s) to complete the reset operation.

**15.1.2 POWER-DOWN MODE**

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The Special Function Registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode,  $V_{DD}$  may be reduced to minimize circuit power consumption. The supply voltage must not be reduced until the Power-down mode is entered, and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 33. If the Power-down mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig.7).

**Table 33** Status of external pins during Idle and Power-down modes

| MODE       | MEMORY   | ALE | $\overline{PSEN}$ | PORT 0    | PORT 1    | PORT 2    | PORT 3    | PORT 4    | $\overline{PWM0}$ |
|------------|----------|-----|-------------------|-----------|-----------|-----------|-----------|-----------|-------------------|
| Idle       | internal | 1   | 1                 | port data | port data | port data | port data | port data | 1                 |
|            | external | 1   | 1                 | floating  | port data | port data | port data | port data | 1                 |
| Power-down | internal | 0   | 0                 | port data | port data | port data | port data | port data | 1                 |
|            | external | 0   | 0                 | floating  | port data | port data | port data | port data | 1                 |

8-bit microcontroller

P83C562; P80C562

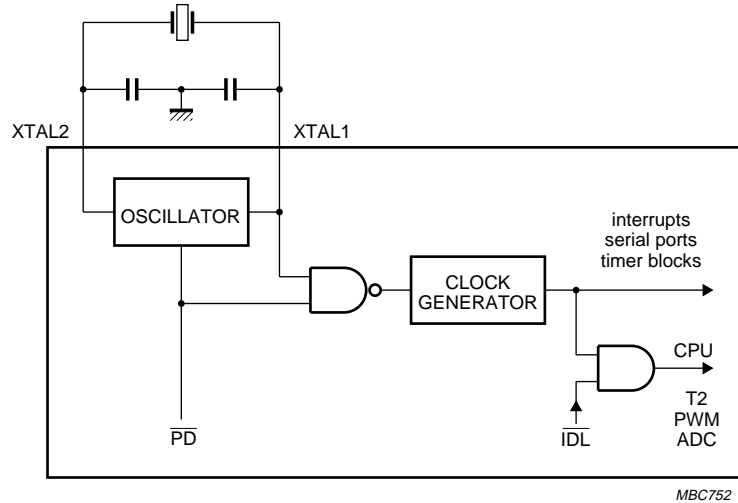


Fig.13 Internal Idle and Power-down clock configuration.

15.2 Power Control Register (PCON)

The reduced power modes are activated by software using this register. PCON is not bit addressable.

Table 34 Power Control Register (SFR address 87H)

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

Table 35 Description of PCON bits (note 1)

| BIT | SYMBOL | DESCRIPTION  |
|-----|--------|--|
| 7   | SMOD   | <b>Double Baud rate.</b> When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3.                        |
| 6   | –      | Reserved.  |
| 5   | RFI    | <b>Reduced radio frequency interference.</b> When set to logic 1, the toggling of the ALE pin is prohibited; this bit is cleared on RESET (see Table 1). |
| 4   | WLE    | <b>Watchdog Load Enable.</b> This flag must be set by software prior to loading the Watchdog Timer. It is cleared when the timer is loaded.              |
| 3   | GF1    | <b>General-purpose flag bits.</b>  |
| 2   | GF0    |  |
| 1   | PD     | <b>Power-down bit.</b> Setting this bit activates the Power-down mode. It can only be set if input $\overline{EW}$ is HIGH.                              |
| 0   | IDL    | <b>Idle mode.</b> Setting this bit activates the Idle mode.  |

Note

1. If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0X000000).

# 8-bit microcontroller

# P83C562; P80C562

## 16 OSCILLATOR CIRCUITRY

The oscillator circuitry of the P8x562 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 (pin 35) is the high gain amplifier input, and XTAL2 (pin 34) is the output (see Fig.14). To drive the P8x562 externally, XTAL1 is driven from an external source and XTAL2 left open-circuit (see Fig.15).

## 17 RESET CIRCUITRY

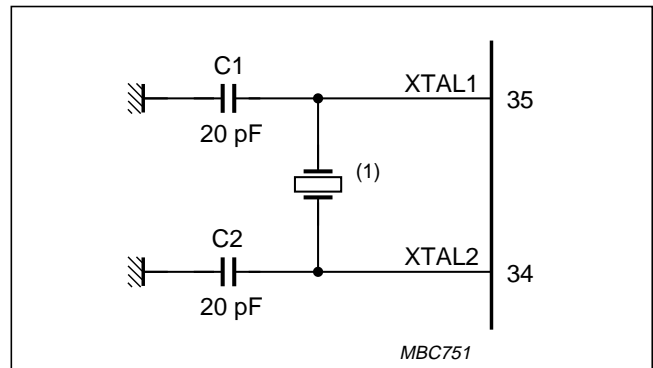
The reset circuitry for the P8x562 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle. The on-chip Reset circuit is shown in Fig.16.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods but at least 2  $\mu$ s). The CPU responds by executing an internal reset. During reset both ALE and PSEN output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

Also with the P8x562, the RST line can be pulled HIGH internally by a pull-up transistor activated by the Watchdog Timer (T3). The length of the output pulse from the Watchdog Timer is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

It can be seen that the short reset pulse from T3 cannot discharge the Power-on reset capacitor (see Fig.17). Consequently, when the Watchdog Timer is also used to reset external devices this capacitor arrangement should not be connected to the RST pin, and an extra circuit should be used to perform the Power-on-reset operation. It should be remembered that a T3 overflow, if enabled, will force a reset condition to the P8x562 by an internal connection, whether the output RST is tied LOW or not.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. The internal RAM is not affected by reset. When  $V_{DD}$  is turned on, the RAM content is indeterminate. An internal reset leaves the internal registers as shown in Table 36.



1) Use fundamental crystals only.

Fig.14 P8x562P8x562 oscillator circuit.

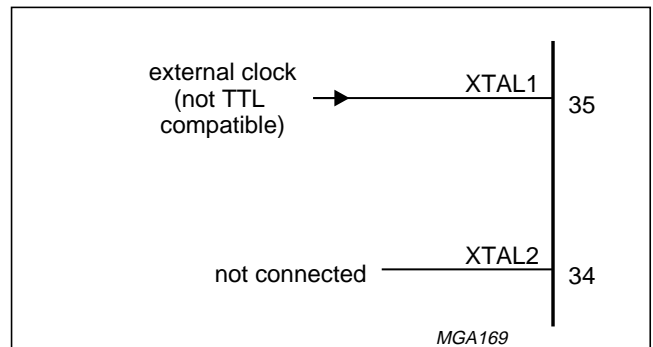


Fig.15 Driving the P8x562 from an external source.

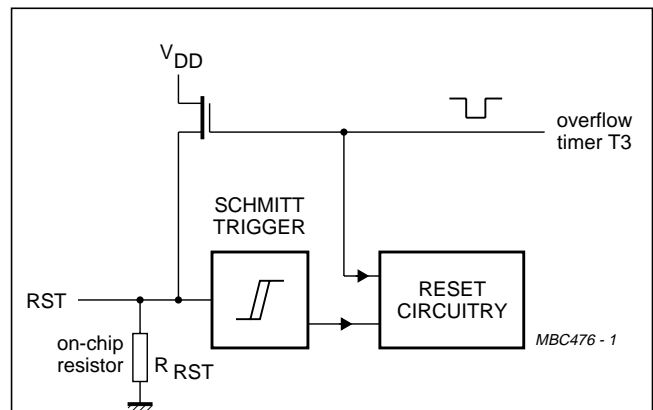


Fig.16 On-chip reset configuration.

8-bit microcontroller

P83C562; P80C562

**Table 36** State of internal registers after an internal reset  
X = undefined state.

| REGISTER     | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------|---|---|---|---|---|---|---|---|
| ACC          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ADCON        | X | X | 0 | 0 | 0 | 0 | 0 | 0 |
| ADCH         | X | X | X | X | X | X | X | X |
| B            | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CML0 to CML2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CMH0 to CMH2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CTCON        | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CTL0 to CTL3 | X | X | X | X | X | X | X | X |
| CTH0 to CTH3 | X | X | X | X | X | X | X | X |
| DPL          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DPH          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IEN0         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IEN1         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IP0          | X | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| IP1          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCH          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCL          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCON         | 0 | X | 0 | 0 | 0 | 0 | 0 | 0 |
| PSW          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PWM0         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PWM1         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PWMP         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| P0 to P4     | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| P5           | X | X | X | X | X | X | X | X |
| RTE          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SBUF         | X | X | X | X | X | X | X | X |
| SCON         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SP           | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| STE          | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| TCON         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TH0, TH1     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TMH2         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TL0, TL1     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TML2         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TMOD         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TM2CON       | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TM2IR        | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| T3           | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**17.1 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  $\mu$ F 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. The port pins will be in a random state until the oscillator has started and the internal reset algorithm has written logic 1s to the port pins. The Power-on-reset circuitry is shown in Fig.17.

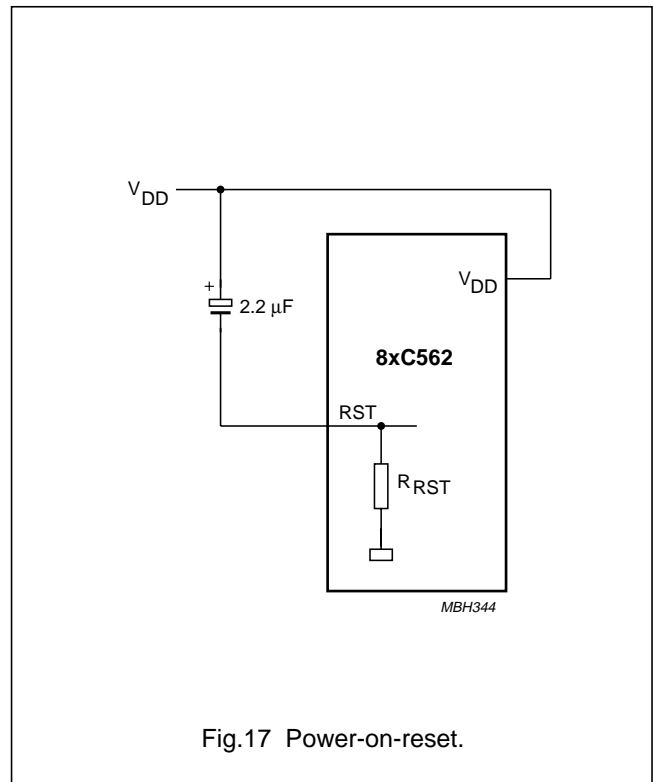


Fig.17 Power-on-reset.

## 8-bit microcontroller

## P83C562; P80C562

**18 INSTRUCTION SET**

The P8x562 uses the powerful instruction set of the 80C51. Additional Special Function Registers are incorporated to control the on-chip peripherals. 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 0.75  $\mu$ s and 45 instructions execute in 1.5  $\mu$ s. Multiply and divide instructions execute in 3  $\mu$ s.

Tables 37 to 41 describe the Instruction set; Table 42 explains the Data addressing modes and the Hexadecimal opcodes.

**Table 37** Instruction set descriptions: Arithmetic operations

| MNEMONIC                     | DESCRIPTION                                | BYTES | CYCLES | OPCODE (HEX) |
|------------------------------|--|-------|--------|--------------|
| <b>Arithmetic operations</b> |  |       |        |              |
| 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 & B                             | 1     | 4      | A4           |
| DIV AB                       | Divide A by B                              | 1     | 4      | 84           |
| DA A                         | Decimal adjust A                           | 1     | 1      | D4           |



## 8-bit microcontroller

## P83C562; P80C562

**Table 38** Instruction set description: Logic operations

| MNEMONIC                |              | DESCRIPTION                                | BYTES | CYCLES | OPCODE (HEX) |
|-------------------------|--------------|--|-------|--------|--------------|
| <b>Logic operations</b> |              |  |       |        |              |
| 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           |

## 8-bit microcontroller

## P83C562; P80C562

**Table 39** Instruction set description: Data transfer

| MNEMONIC             | DESCRIPTION                                   | BYTES | CYCLES | OPCODE (HEX) |
|----------------------|---|-------|--------|--------------|
| <b>Data transfer</b> |   |       |        |              |
| 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 byte               | 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,#data16     | 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 external RAM (8-bit address) to A        | 1     | 2      | E2, E3       |
| MOVX A,@DPTR         | Move external RAM (16-bit address) to A       | 1     | 2      | E0           |
| MOVX @Ri,A           | Move A to external RAM (8-bit address)        | 1     | 2      | F2, F3       |
| MOVX @DPTR,A         | Move A to 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 nibble indirect RAM with A | 1     | 1      | D6, D7       |

## 8-bit microcontroller

## P83C562; P80C562

**Table 40** Instruction set description: Program and machine control

| MNEMONIC                           | DESCRIPTION                                     | BYTES | CYCLES | OPCODE (HEX) |
|------------------------------------|---|-------|--------|--------------|
| <b>Program and machine control</b> |   |       |        |              |
| ACALL addr11                       | Absolute subroutine call                        | 2     | 2      | •1           |
| 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      | ♦1           |
| 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 reg. and jump if not equal | 3     | 2      | B*           |
| CJNE @Ri,#data,rel                 | Compare immediate to ind. 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           |

## 8-bit microcontroller

## P83C562; P80C562

**Table 41** Instruction set description: Boolean variable manipulation

| MNEMONIC                             | DESCRIPTION                                | BYTES | CYCLES | OPCODE (HEX) |
|--------------------------------------|--|-------|--------|--------------|
| <b>Boolean variable manipulation</b> |  |       |        |              |
| 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           |

**Table 42** Description of the mnemonics in the instruction set

| MNEMONIC                                  | DESCRIPTION  |
|---|--|
| <b>Data addressing modes</b>              |  |
| 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 actual 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 kbytes program memory address space.                                     |
| addr11                                    | 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 kbytes page of program memory as the first byte of the following instruction. |
| rel                                       | Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is -128 to +127 bytes relative to first byte of the following instruction.  |
| <b>Hexadecimal opcode cross-reference</b> |  |
| *   | 8, 9, A, B, C, D, E, F.  |
| •   | 1, 3, 5, 7, 9, B, D, F.  |
| ♦   | 0, 2, 4, 6, 8, A, C, E.  |

## 8-bit microcontroller

## P83C562; P80C562

Table 43. 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<br>addr11  | LJMP<br>addr16      | RR<br>A             | INC<br>A                                   | INC<br>direct                  | INC @Ri<br>0   1            |   | INC Rr<br>0   1   2   3   4   5   6   7            |   |   |   |   |   |   |   |
| 1 | JBC<br>bit,rel                        | ACALL<br>addr11 | LCALL<br>addr16     | RRC<br>A            | DEC<br>A                                   | DEC<br>direct                  | DEC @Ri<br>0   1            |   | DEC Rr<br>0   1   2   3   4   5   6   7            |   |   |   |   |   |   |   |
| 2 | JB<br>bit,rel                         | AJMP<br>addr11  | RET                 | RL<br>A             | ADD<br>A,#data                             | ADD<br>A,direct                | ADD A,@Ri<br>0   1          |   | ADD A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| 3 | JNB<br>bit,rel                        | ACALL<br>addr11 | RETI                | RLC<br>A            | ADDC<br>A,#data                            | ADDC<br>A,direct               | ADDC A,@Ri<br>0   1         |   | ADDC A,Rr<br>0   1   2   3   4   5   6   7         |   |   |   |   |   |   |   |
| 4 | JC<br>rel                             | AJMP<br>addr11  | ORL<br>direct,A     | ORL<br>direct,#data | ORL<br>A,#data                             | ORL<br>A,direct                | ORL A,@Ri<br>0   1          |   | ORL A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| 5 | JNC<br>rel                            | ACALL<br>addr11 | ANL<br>direct,A     | ANL<br>direct,#data | ANL<br>A,#data                             | ANL<br>A,direct                | ANL A,@Ri<br>0   1          |   | ANL A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| 6 | JZ<br>rel                             | AJMP<br>addr11  | XRL<br>direct,A     | XRL<br>direct,#data | XRL<br>A,#data                             | XRL<br>A,direct                | XRL A,@Ri<br>0   1          |   | XRL A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| 7 | JNZ<br>rel                            | ACALL<br>addr11 | ORL<br>C,bit        | JMP<br>@A+DPTR      | MOV<br>A,#data                             | MOV<br>direct,#data            | MOV @Ri,#data<br>0   1      |   | MOV Rr,#data<br>0   1   2   3   4   5   6   7      |   |   |   |   |   |   |   |
| 8 | SJMP<br>rel                           | AJMP<br>addr11  | ANL<br>C,bit        | MOVC<br>A,@A+PC     | DIV<br>AB                                  | MOV<br>direct,direct           | MOV direct,@Ri<br>0   1     |   | MOV direct,Rr<br>0   1   2   3   4   5   6   7     |   |   |   |   |   |   |   |
| 9 | MOV<br>DTPR,#data16                   | ACALL<br>addr11 | MOV<br>bit,C        | MOVC<br>A,@A+DPTR   | SUBB<br>A,#data                            | SUBB<br>A,direct               | SUBB A,@Ri<br>0   1         |   | SUB A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| A | ORL<br>C,/bit                         | AJMP<br>addr11  | MOV<br>bit,C        | INC<br>DPTR         | MUL<br>AB                                  |                                | MOV @Ri,direct<br>0   1     |   | MOV Rr,direct<br>0   1   2   3   4   5   6   7     |   |   |   |   |   |   |   |
| B | ANL<br>C,/bit                         | ACALL<br>addr11 | CPL<br>bit          | CPL<br>C            | CJNE<br>A,#data,rel                        | CJNE<br>A,direct,rel           | CJNE @Ri,#data,rel<br>0   1 |   | CJNE Rr,#data,rel<br>0   1   2   3   4   5   6   7 |   |   |   |   |   |   |   |
| C | PUSH<br>direct                        | AJMP<br>addr11  | CLR<br>bit          | CLR<br>C            | SWAP<br>A                                  | XCH<br>A,direct                | XCH A,@Ri<br>0   1          |   | XCH A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| D | POP<br>direct                         | ACALL<br>addr11 | SETB<br>bit         | SETB<br>C           | DA<br>A                                    | DJNZ<br>direct,rel             | XCHD A,@Ri<br>0   1         |   | DJNZ Rr,rel<br>0   1   2   3   4   5   6   7       |   |   |   |   |   |   |   |
| E | MOVX<br>A,@DTPR                       | AJMP<br>addr11  | MOVX A,@Ri<br>0   1 |                     | CLR<br>A                                   | MOV<br>A,direct <sup>(1)</sup> | MOV A,@Ri<br>0   1          |   | MOV A,Rr<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |
| F | MOVX<br>@DTPR,A                       | ACALL<br>addr11 | MOVX @Ri,A<br>0   1 |                     | CPL<br>A                                   | MOV<br>direct,A                | MOV @Ri,A<br>0   1          |   | MOV Rr,A<br>0   1   2   3   4   5   6   7          |   |   |   |   |   |   |   |

## Note

1. MOV A, ACC is not a valid instruction.

## 8-bit microcontroller

## P83C562; P80C562

**19 LIMITING VALUES**

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

| SYMBOL     | PARAMETER  | MIN. | MAX. | UNIT |
|------------|--|------|------|------|
| $V_I$      | input voltage on any pin with respect to ground ( $V_{SS}$ ) | -0.5 | +6.5 | V    |
| $I_I, I_O$ | input, output DC current on any single I/O pin               | -    | 5.0  | mA   |
| $P_{tot}$  | total power dissipation                                      | -    | 1.0  | W    |
| $T_{stg}$  | storage temperature range                                    | -65  | +150 | °C   |
| $T_{amb}$  | operating ambient temperature range                          |      |      |      |
|            | P8xC562EBx   | 0    | +70  | °C   |
|            | P8xC562EFx   | -40  | +85  | °C   |
|            | P8xC562EHx   | -40  | +125 | °C   |

**20 DC CHARACTERISTICS**

$V_{DD} = 5\text{ V} \pm 10\%$ ;  $V_{SS} = 0\text{ V}$ ; all voltages with respect to  $V_{SS}$  unless otherwise specified;  $f_{OSC} = 16\text{ MHz}$ .

$T_{amb} = -40\text{ to }+85\text{ °C}$  for the **P8xC562EFx**.

| SYMBOL                       | PARAMETER  | CONDITIONS                              | MIN.              | MAX.              | UNIT          |
|------------------------------|--|---|-------------------|-------------------|---------------|
| <b>Supply (digital part)</b> |  |   |                   |                   |               |
| $V_{DD}$                     | supply voltage P8xC562Exx  |   | 4.5               | 5.5               | V             |
| $I_{DD}$                     | operating supply current<br>P8xC562Exx                                       | note 1                                  | -                 | 40                | mA            |
| $I_{DD(ID)}$                 | supply current Idle mode<br>P8xC562Exx                                       | note 2                                  | -                 | 9                 | mA            |
| $I_{DD(PD)}$                 | supply current Power-down mode   | note 3                                  |                   |                   |               |
|                              | P8X562EBx  | $2\text{ V} < V_{DD(PD)} < V_{DD(max)}$ | -                 | 50                | $\mu\text{A}$ |
|                              | P8X562EFx  | $2\text{ V} < V_{DD(PD)} < V_{DD(max)}$ | -                 | 50                | $\mu\text{A}$ |
|                              | P8X562EHx  | $2\text{ V} < V_{DD(PD)} < V_{DD(max)}$ | -                 | 150               | $\mu\text{A}$ |
| <b>Inputs</b>                |  |   |                   |                   |               |
| $V_{IL}$                     | LOW level input voltage (except $\overline{EA}$ )                            |   | -0.5              | $0.2V_{DD} - 0.1$ | V             |
| $V_{IL1}$                    | LOW level input voltage ( $\overline{EA}$ )                                  |   | -0.5              | $0.2V_{DD} - 0.3$ | V             |
| $V_{IH}$                     | HIGH level input voltage<br>(except RST, XTAL1)                              |   | $0.2V_{DD} + 0.9$ | $V_{DD} + 0.5$    | V             |
| $V_{IH1}$                    | HIGH level input voltage<br>(RST and XTAL1)                                  |   | $0.7V_{DD}$       | $V_{DD} + 0.5$    | V             |
| $I_{IL}$                     | input current logic 0<br>Ports 1, 2, 3 and 4;<br>(except P1.6/SCL, P1.7/SDA) | $V_I = 0.45\text{ V}$                   | -                 | -50               | $\mu\text{A}$ |
| $I_{TL}$                     | input current HIGH-to-LOW transition<br>(Ports 1, 2, 3 and 4)                | $V_I = 2.0\text{ V}$                    | -                 | -650              | $\mu\text{A}$ |
| $I_{LI1}$                    | input leakage current<br>(Port 0, $\overline{EA}$ , STADC, $\overline{EW}$ ) | $0.45\text{ V} < V_I < V_{DD}$          | -                 | $\pm 10$          | $\mu\text{A}$ |
| $I_{LI3}$                    | input leakage current (Port 5)   | $0.45\text{ V} < V_I < V_{DD}$          | -                 | $\pm 1$           | $\mu\text{A}$ |

## 8-bit microcontroller

## P83C562; P80C562

| SYMBOL                      | PARAMETER  | CONDITIONS  | MIN.                | MAX. | UNIT |
|-----------------------------|--|---|---------------------|------|------|
| <b>Outputs</b>              |  |   |                     |      |      |
| V <sub>OL</sub>             | LOW level output voltage<br>(Ports 1, 2, 3 and 4)  | I <sub>OL</sub> = 1.6 mA; note 4                    | –                   | 0.45 | V    |
| V <sub>OL1</sub>            | LOW level output voltage<br>(Port 0, ALE, PSEN, PWM0, PWM1)                                | I <sub>OL</sub> = 3.2 mA; note 4                    | –                   | 0.45 | V    |
| V <sub>OH</sub>             | HIGH level output voltage<br>Ports 1, 2, 3 and 4   | I <sub>OH</sub> = –60 µA                            | 2.4                 | –    | V    |
|                             |  | I <sub>OH</sub> = –25 µA                            | 0.75V <sub>DD</sub> | –    | V    |
|                             |  | I <sub>OH</sub> = –10 µA                            | 0.9V <sub>DD</sub>  | –    | V    |
| V <sub>OH1</sub>            | HIGH level output voltage<br>Port 0 in external bus mode,<br>ALE, PSEN, PWM0, PWM1; note 5 | I <sub>OH</sub> = –800 µA                           | 2.4                 | –    | V    |
|                             |  | I <sub>OH</sub> = –300 µA                           | 0.75V <sub>DD</sub> | –    | V    |
|                             |  | I <sub>OH</sub> = –80 µA                            | 0.9V <sub>DD</sub>  | –    | V    |
| V <sub>OH2</sub>            | HIGH level output voltage (RST)  | I <sub>OH</sub> = –400 µA                           | 2.4                 | –    | V    |
|                             |  | I <sub>OH</sub> = –120 µA                           | 0.8V <sub>DD</sub>  | –    | V    |
| R <sub>RST</sub>            | RST pull-down resistor   |   | 50                  | 150  | kΩ   |
| C <sub>I/O</sub>            | capacitance of I/O buffer  | test frequency = 1 MHz;<br>T <sub>amb</sub> = 25 °C | –                   | 10   | pF   |
| <b>Supply (analog part)</b> |  |   |                     |      |      |
| V <sub>DDA</sub>            | supply voltage<br>P8X562Exx  | V <sub>DDA</sub> = V <sub>DD</sub> ±0.2 V           | 4.5                 | 5.5  | V    |
| I <sub>DDA</sub>            | supply current operating   | Port 5 = 0 to V <sub>DDA</sub>                      |                     | 1.2  | mA   |
| I <sub>DDA(ID)</sub>        | supply current Idle mode<br>P8X562EBx<br>P8X562EFx<br>P8X562EHx                            |   | –                   | 50   | µA   |
|                             |  |   | –                   | 50   | µA   |
|                             |  |   | –                   | 100  | µA   |
| I <sub>DDA(PD)</sub>        | supply current Power-down mode<br>P8X562EBx<br>P8X562EFx<br>P8X562EHx                      | 2 V < V <sub>DDA(PD)</sub> < V <sub>DDA(max)</sub>  | –                   | 50   | µA   |
|                             |  |   | –                   | 50   | µA   |
|                             |  |   | –                   | 100  | µA   |

## 8-bit microcontroller

## P83C562; P80C562

| SYMBOL               | PARAMETER                                    | CONDITIONS     | MIN.            | MAX.            | UNIT       |
|----------------------|--|----------------|-----------------|-----------------|------------|
| <b>Analog inputs</b> |  |                |                 |                 |            |
| $V_{IN}$             | analog input voltage                         |                | $AV_{SS} - 0.2$ | $AV_{DD} + 0.2$ | V          |
| $V_{REF+}$           | reference voltage (+)                        |                | –               | $AV_{DD} + 0.2$ | V          |
| $V_{REF-}$           | reference voltage (–)                        |                | $AV_{SS} - 0.2$ | –               | V          |
| $R_{REF}$            | resistance between $V_{REF+}$ and $V_{REF-}$ |                | 5               | 25              | k $\Omega$ |
| $C_{IA}$             | analog input capacitance                     |                | –               | 15              | pF         |
| $t_{ADS}$            | sampling time                                |                | –               | $6t_{CY}$       | $\mu$ s    |
| $t_{ADC}$            | conversion time<br>(including sample time)   |                | –               | $24t_{CY}$      | $\mu$ s    |
| $DL_e$               | differential non-linearity                   | notes 7 and 11 | –               | $\pm 1$         | LSB        |
| $IL_e$               | integral non-linearity                       | notes 6 and 8  | –               | $\pm 1$         | LSB        |
| $OS_e$               | offset error                                 | notes 6 and 10 | –               | $\pm 1$         | LSB        |
| $G_e$                | gain error                                   | notes 6 and 9  | –               | $\pm 0.4$       | %          |
| $M_{ctc}$            | channel-to-channel matching                  |                | –               | $\pm 1$         | LSB        |
| $C_t$                | crosstalk between P5 inputs                  | 0 to 100 kHz   | –               | –60             | dB         |

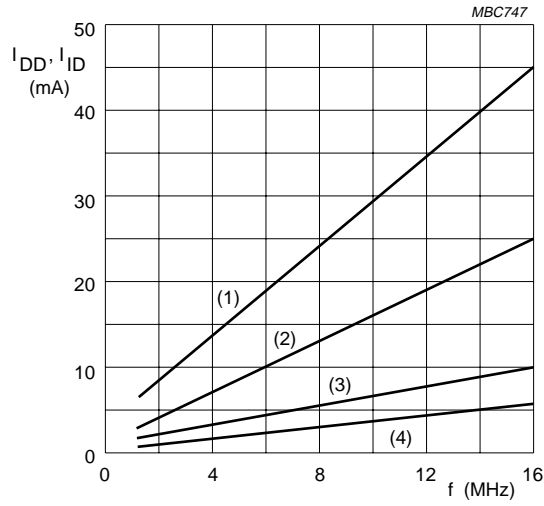
**Notes to the DC characteristics**

- The operating supply current is measured with all output pins disconnected; XTAL1 driven with  $t_r = t_f = 10$  ns;  $V_{IL} = V_{SS} + 0.5$  V;  $V_{IH} = V_{DD} - 0.5$  V; XTAL2 not connected;  $\overline{EA} = RST = Port\ 0 = \overline{EW} = V_{DD}$ ;  $STADC = V_{SS}$ .
- The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with  $t_r = t_f = 10$  ns;  $V_{IL} = V_{SS} + 0.5$  V;  $V_{IH} = V_{DD} - 0.5$  V; XTAL2 not connected;  $\overline{EA} = Port\ 0 = \overline{EW} = V_{DD}$ ;  $RST = STADC = V_{SS}$ .
- The Power-down current is measured with all output pins disconnected; XTAL2 not connected;  $\overline{EA} = Port\ 0 = \overline{EW} = V_{DD}$ ;  $RST = STADC = XTAL1 = V_{SS}$ .
- Capacitive loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the low level output voltage of ALE and Ports 1, 3 and 4. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make HIGH-to-LOW transitions during bus operations. In the most adverse condition (capacitive loading > 100 pF), the noise pulse on the ALE line may exceed 0.8 V. In such events it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger strobe input.
- Capacitive loading on Ports 0 and 2 may cause the high level output voltage on ALE and  $\overline{PSEN}$  to momentarily fall below to  $0.9V_{DD}$  specification when the address bits are stabilizing.
- $V_{REF+} = 5.12$  V;  $V_{REF-} = 0$  V;  $V_{DDA} = 5.0$  V.
- The differential non-linearity ( $DL_e$ ) is the difference between the actual step width and the ideal step width.
- The integral non-linearity ( $IL_e$ ) is the peak difference between the centre of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset error.
- The gain error ( $G_e$ ) is the relative difference in percent between the straight line fitting the actual transfer curve after removing offset error, and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve.
- The offset error ( $OS_e$ ) is the absolute difference between the straight line which fits the actual transfer curve after removing gain error, and a straight line which fits the ideal transfer curve. The offset error is constant at every point of the actual transfer curve.
- $V_{REF-} = 0$  V;  $V_{DDA} = 5$  V;  $V_{REF+} = 5.12$  V. The ADC is monotonic with no missing codes. Measurement by continuously increasing  $V_{IN}$  from –20 mV to 5.12 V in increments of 2 mV.



8-bit microcontroller

P83C562; P80C562

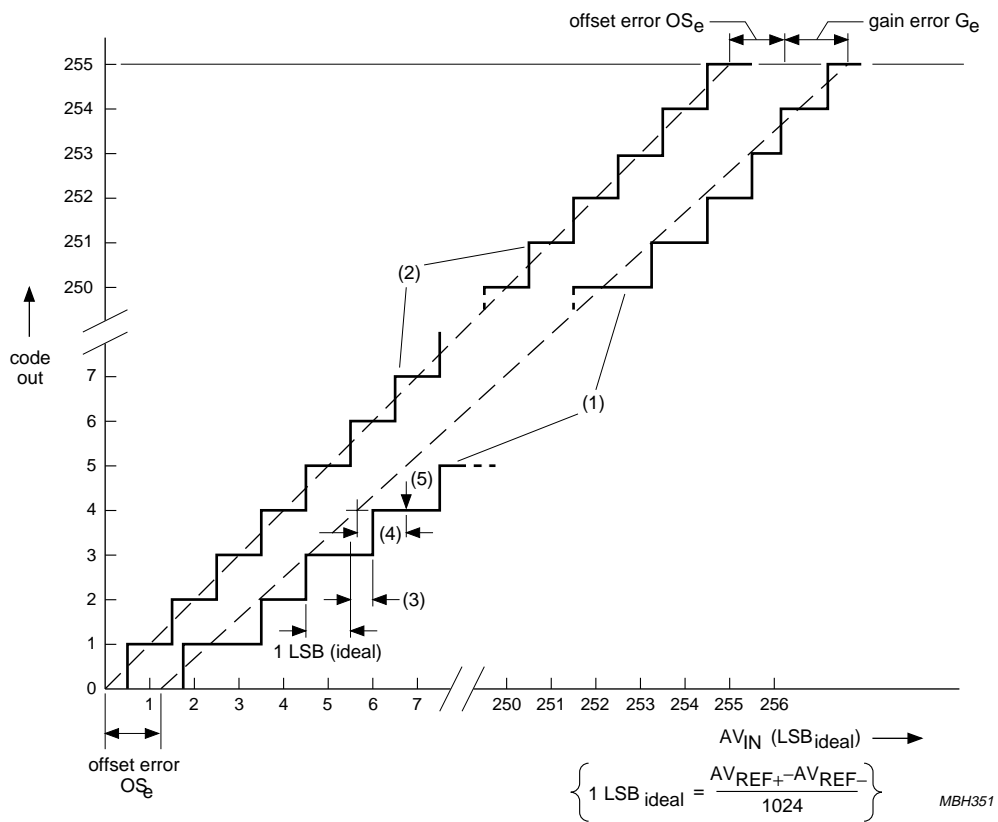


- (1)  $I_{DD(max)}$  operating mode;  $V_{DD} = 5.5$  V.
  - (2)  $I_{DD(max)}$  operating mode;  $V_{DD} = 4.5$  V.
  - (3)  $I_{ID(max)}$  Idle mode;  $V_{DD} = 5.5$  V.
  - (4)  $I_{ID(max)}$  Idle mode;  $V_{DD} = 4.5$  V.
- These values are valid within the specified frequency range.

Fig.18 Supply current ( $I_{DD}$ ,  $I_{ID}$ ) as a function of frequency at XTAL1 ( $f_{osc}$ ).

8-bit microcontroller

P83C562; P80C562



- (1) Example of an actual transfer curve.
- (2) The ideal transfer curve.
- (3) Differential non-linearity (DL<sub>e</sub>).
- (4) Integral non-linearity (IL<sub>e</sub>).
- (5) Centre of a step of the actual transfer curve.

Fig.19 ADC conversion characteristic.

## 8-bit microcontroller

## P83C562; P80C562

## 21 AC CHARACTERISTICS

Parameters are valid over operating temperature range and operating supply voltage range unless otherwise specified.  $C_L = 100$  pF for Port 0, ALE and  $\overline{\text{PSEN}}$ ;  $C_L = 80$  pF for all other outputs unless specified. See Figs 23 to 25.

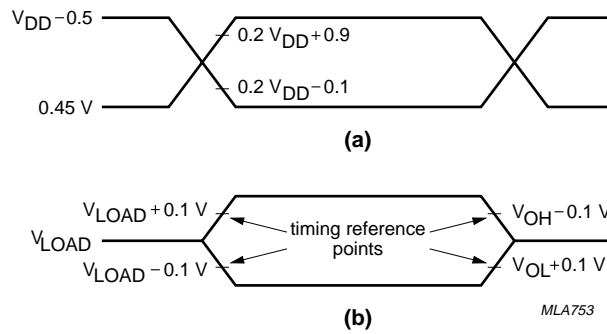
| SYMBOL                      | PARAMETER   | $f_{\text{osc}} = 16 \text{ MHz}$ |      | $f_{\text{osc}} = \text{VARIABLE}$ |                         | UNIT |
|-----------------------------|---|-----------------------------------|------|------------------------------------|-------------------------|------|
|                             |   | MIN.                              | MAX. | MIN.                               | MAX.                    |      |
| <b>Program memory</b>       |   |                                   |      |                                    |                         |      |
| $t_{\text{LL}}$             | ALE pulse duration  | 85                                | –    | $2t_{\text{CLK}} - 40$             | –                       | ns   |
| $t_{\text{AL}}$             | address set-up time to ALE  | 8                                 | –    | $t_{\text{CLK}} - 55$              | –                       | ns   |
| $t_{\text{LA}}$             | address hold time after ALE   | 28                                | –    | $t_{\text{CLK}} - 35$              | –                       | ns   |
| $t_{\text{LIV}}$            | time from ALE to valid instruction input                                    | –                                 | 150  | –                                  | $4t_{\text{CLK}} - 100$ | ns   |
| $t_{\text{LC}}$             | time from ALE to control pulse $\overline{\text{PSEN}}$                     | 23                                | –    | $t_{\text{CLK}} - 40$              | –                       | ns   |
| $t_{\text{CC}}$             | control pulse duration $\overline{\text{PSEN}}$                             | 143                               | –    | $3t_{\text{CLK}} - 45$             | –                       | ns   |
| $t_{\text{CIV}}$            | time from $\overline{\text{PSEN}}$ to valid instruction input               | –                                 | 83   | –                                  | $3t_{\text{CLK}} - 105$ | ns   |
| $t_{\text{CI}}$             | input instruction hold time after $\overline{\text{PSEN}}$                  | 0                                 | –    | 0                                  | –                       | ns   |
| $t_{\text{CIF}}$            | input instruction float delay after $\overline{\text{PSEN}}$                | –                                 | 38   | –                                  | $t_{\text{CLK}} - 25$   | ns   |
| $t_{\text{AIV}}$            | address to valid instruction input  | –                                 | 208  | –                                  | $5t_{\text{CLK}} - 105$ | ns   |
| $t_{\text{AFC}}$            | address float delay after $\overline{\text{PSEN}}$                          | –                                 | 10   | –                                  | 10                      | ns   |
| <b>External data memory</b> |   |                                   |      |                                    |                         |      |
| $t_{\text{RR}}$             | RD pulse duration   | 275                               | –    | $6t_{\text{CLK}} - 100$            | –                       | ns   |
| $t_{\text{WW}}$             | WR pulse duration   | 275                               | –    | $6t_{\text{CLK}} - 100$            | –                       | ns   |
| $t_{\text{AL}}$             | address set-up time to ALE  | 8                                 | –    | $t_{\text{CLK}} - 55$              | –                       | ns   |
| $t_{\text{LA}}$             | address hold time after ALE   | 28                                | –    | $t_{\text{CLK}} - 35$              | –                       | ns   |
| $t_{\text{RD}}$             | RD to valid data input  | –                                 | 148  | –                                  | $5t_{\text{CLK}} - 165$ | ns   |
| $t_{\text{DR}}$             | data hold time after $\overline{\text{RD}}$                                 | 0                                 | –    | 0                                  | –                       | ns   |
| $t_{\text{DFR}}$            | data float delay after $\overline{\text{RD}}$                               | –                                 | 55   | –                                  | $2t_{\text{CLK}} - 70$  | ns   |
| $t_{\text{LD}}$             | time from ALE to valid data input   | –                                 | 350  | –                                  | $8t_{\text{CLK}} - 150$ | ns   |
| $t_{\text{AD}}$             | address to valid data input   | –                                 | 398  | –                                  | $9t_{\text{CLK}} - 165$ | ns   |
| $t_{\text{LW}}$             | time from ALE to $\overline{\text{RD}}$ or $\overline{\text{WR}}$           | 138                               | 238  | $3t_{\text{CLK}} - 50$             | $3t_{\text{CLK}} + 50$  | ns   |
| $t_{\text{AW}}$             | time from address to $\overline{\text{RD}}$ or $\overline{\text{WR}}$       | 120                               | –    | $4t_{\text{CLK}} - 130$            | –                       | ns   |
| $t_{\text{WHLH}}$           | time from $\overline{\text{RD}}$ or $\overline{\text{WR}}$ HIGH to ALE HIGH | 23                                | 103  | $t_{\text{CLK}} - 40$              | $t_{\text{CLK}} + 40$   | ns   |
| $t_{\text{DWX}}$            | data valid to $\overline{\text{WR}}$ transition                             | 3                                 | –    | $t_{\text{CLK}} - 60$              | –                       | ns   |
| $t_{\text{DW}}$             | data set-up time before $\overline{\text{WR}}$                              | 288                               | –    | $7t_{\text{CLK}} - 150$            | –                       | ns   |
| $t_{\text{WD}}$             | data hold time after $\overline{\text{WR}}$                                 | 13                                | –    | $t_{\text{CLK}} - 50$              | –                       | ns   |
| $t_{\text{AFR}}$            | address float delay after $\overline{\text{RD}}$                            | –                                 | 0    | –                                  | 0                       | ns   |

**Note**

- $t_{\text{CLK}} = 1/f_{\text{osc}} =$  one oscillator clock period. If  $f_{\text{osc}} = 16 \text{ MHz}$  then  $t_{\text{CLK}} = 62.5 \text{ ns}$ .

8-bit microcontroller

P83C562; P80C562



AC inputs during testing are driven at  $V_{DD} - 0.5$  V for a logic 1, and 0.45 V for a logic 0.

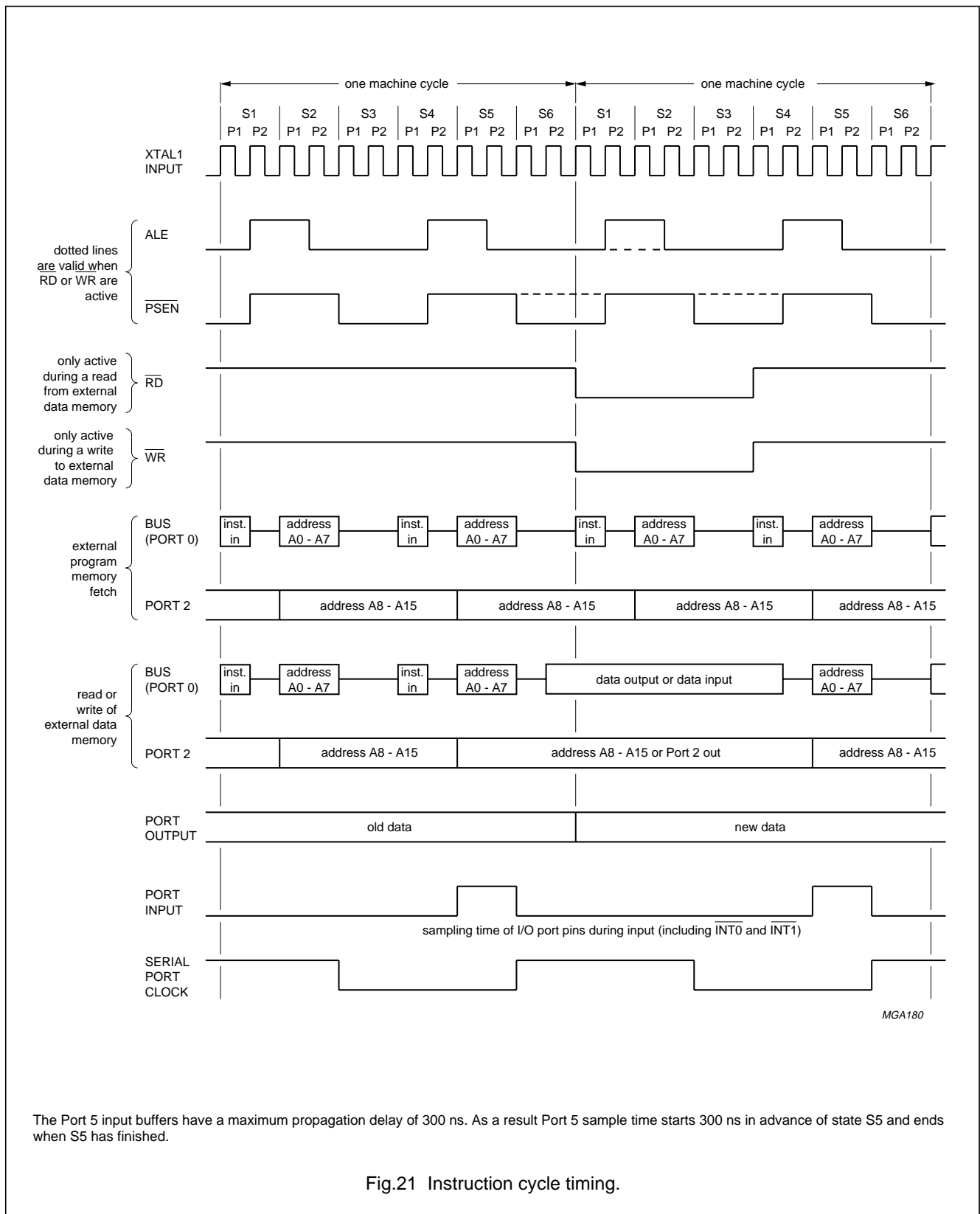
Timing measurements are made at  $V_{IH(min)}$  for a logic 1, and  $V_{IL(max)}$  for a logic 0. See Fig.25 (a).

For timing purposes a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded  $V_{OH}/V_{OL}$  level occurs.  $I_{OL}/I_{OH} \geq \pm 20$  mA (for testing purposes only). See Fig.25 (b).

Fig.20 AC inputs test conditions.

8-bit microcontroller

P83C562; P80C562



The Port 5 input buffers have a maximum propagation delay of 300 ns. As a result Port 5 sample time starts 300 ns in advance of state S5 and ends when S5 has finished.

Fig.21 Instruction cycle timing.

8-bit microcontroller

P83C562; P80C562

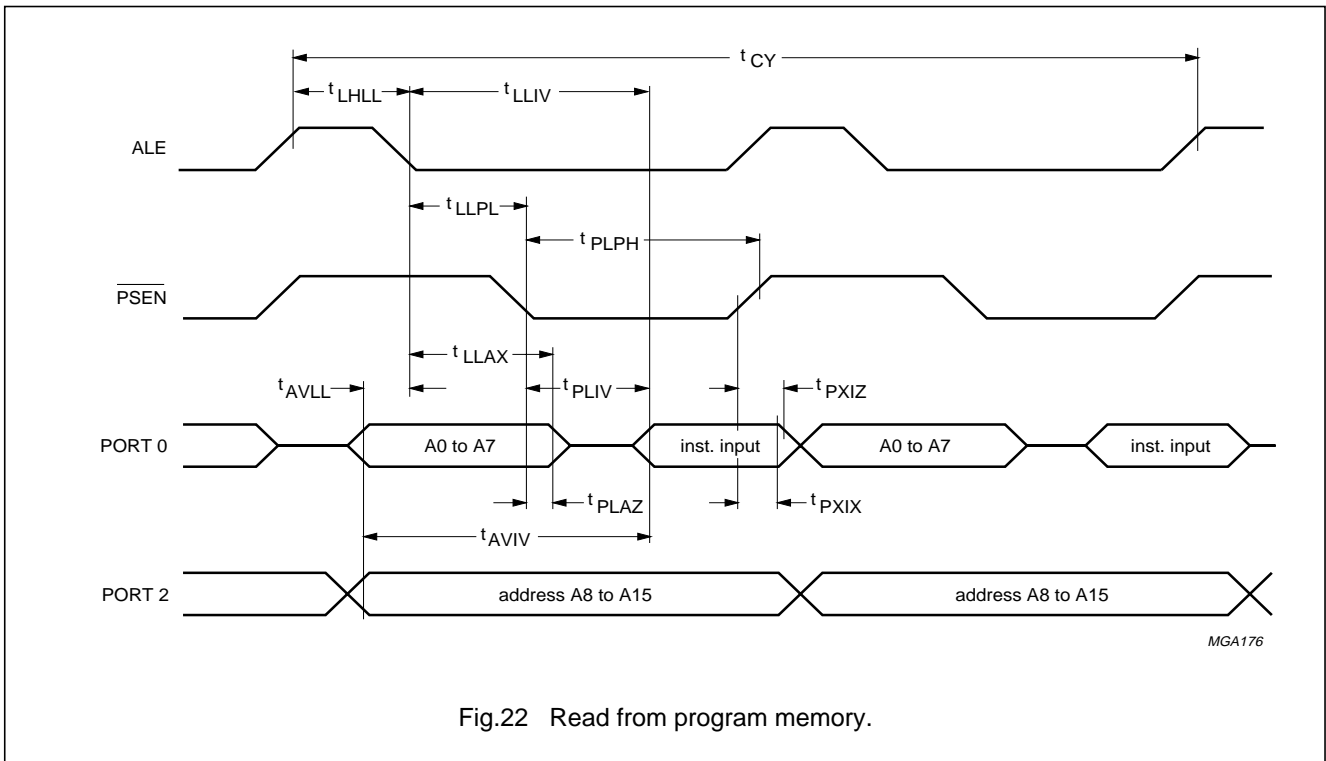


Fig.22 Read from program memory.

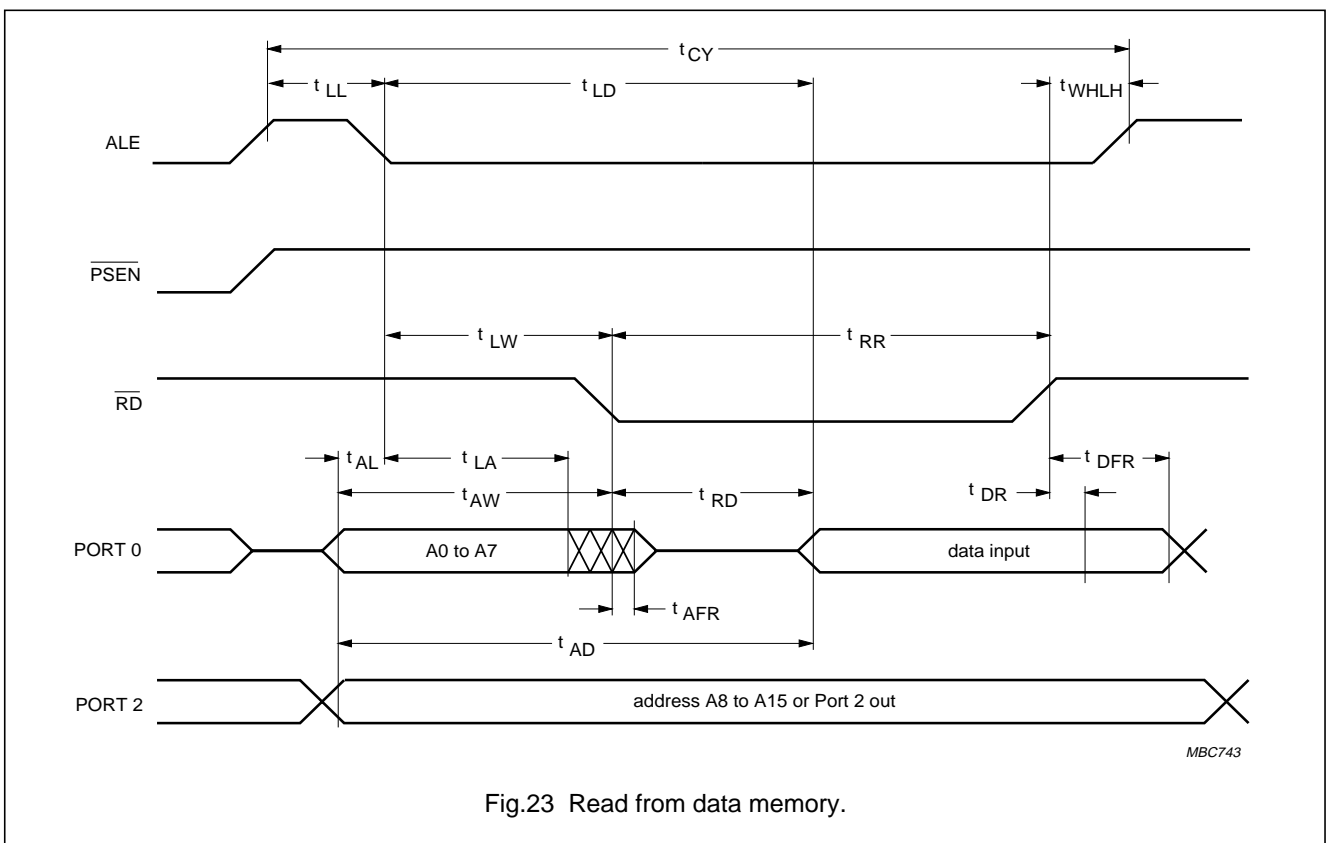


Fig.23 Read from data memory.

8-bit microcontroller

P83C562; P80C562

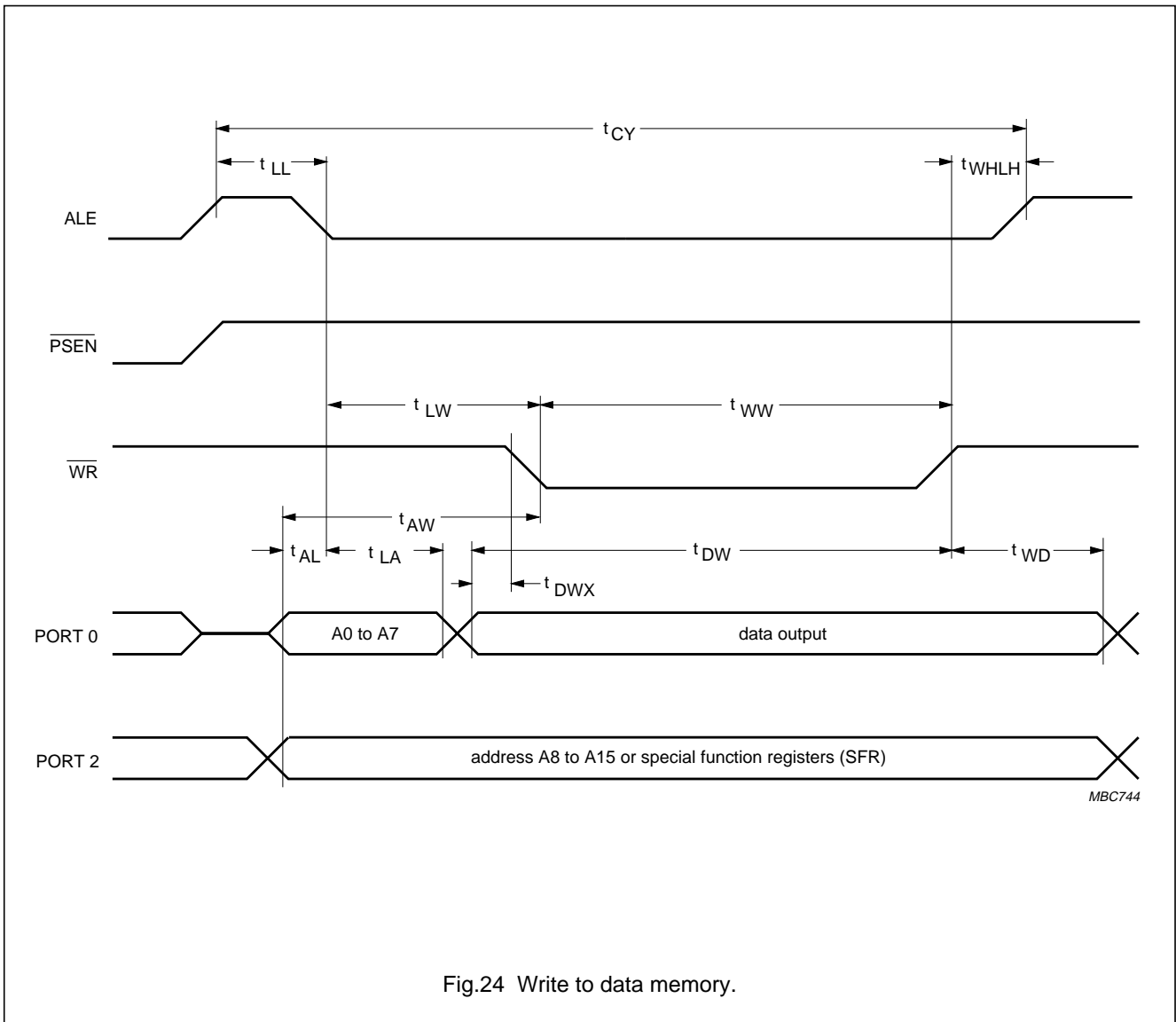


Fig.24 Write to data memory.

8-bit microcontroller

P83C562; P80C562

**Table 44** External clock drive XTAL1

Test conditions: operating temperature and supply voltage ranges; load capacitance = 80 pF.

| SYMBOL     | PARAMETER                  | MIN. | MAX.  | UNIT    |
|------------|----------------------------|------|-------|---------|
| $t_{CLK}$  | clock period               | 62.5 | 285.7 | ns      |
| $t_{HIGH}$ | HIGH time                  | 20   | –     | ns      |
| $t_{LOW}$  | LOW time                   | 20   | –     | ns      |
| $t_r$      | rise time                  | –    | 20    | ns      |
| $t_f$      | fall time                  | –    | 20    | ns      |
| $t_{CY}$   | cycle time (12 $t_{CLK}$ ) | 0.75 | 3.43  | $\mu$ s |

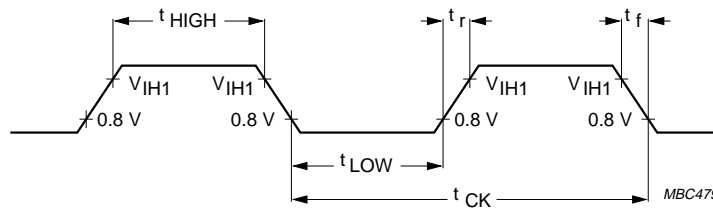


Fig.25 External clock drive XTAL.

**Table 45** Serial timing - shift register mode using 16 MHz oscillator

| SYMBOL     | PARAMETER                                | 16 MHz OSC |      | VARIABLE OSCILLATOR |                   | UNIT    |
|------------|--|------------|------|---------------------|-------------------|---------|
|            |  | MIN.       | MAX. | MIN.                | MAX.              |         |
| $t_{XLXL}$ | serial port clock cycle time             | 0.75       | –    | $12t_{CLK}$         | –                 | $\mu$ s |
| $t_{QVXH}$ | output data set-up to clock rising edge  | 492        | –    | $10t_{CLK} - 133$   | –                 | ns      |
| $t_{XHGX}$ | output data hold after clock rising edge | 8.0        | –    | $2t_{CLK} - 117$    | –                 | ns      |
| $t_{XHDX}$ | input data hold after clock rising edge  | 0          | –    | 0                   | –                 | ns      |
| $t_{XHDV}$ | clock rising edge to input data valid    | –          | 492  | –                   | $10t_{CLK} - 133$ | ns      |



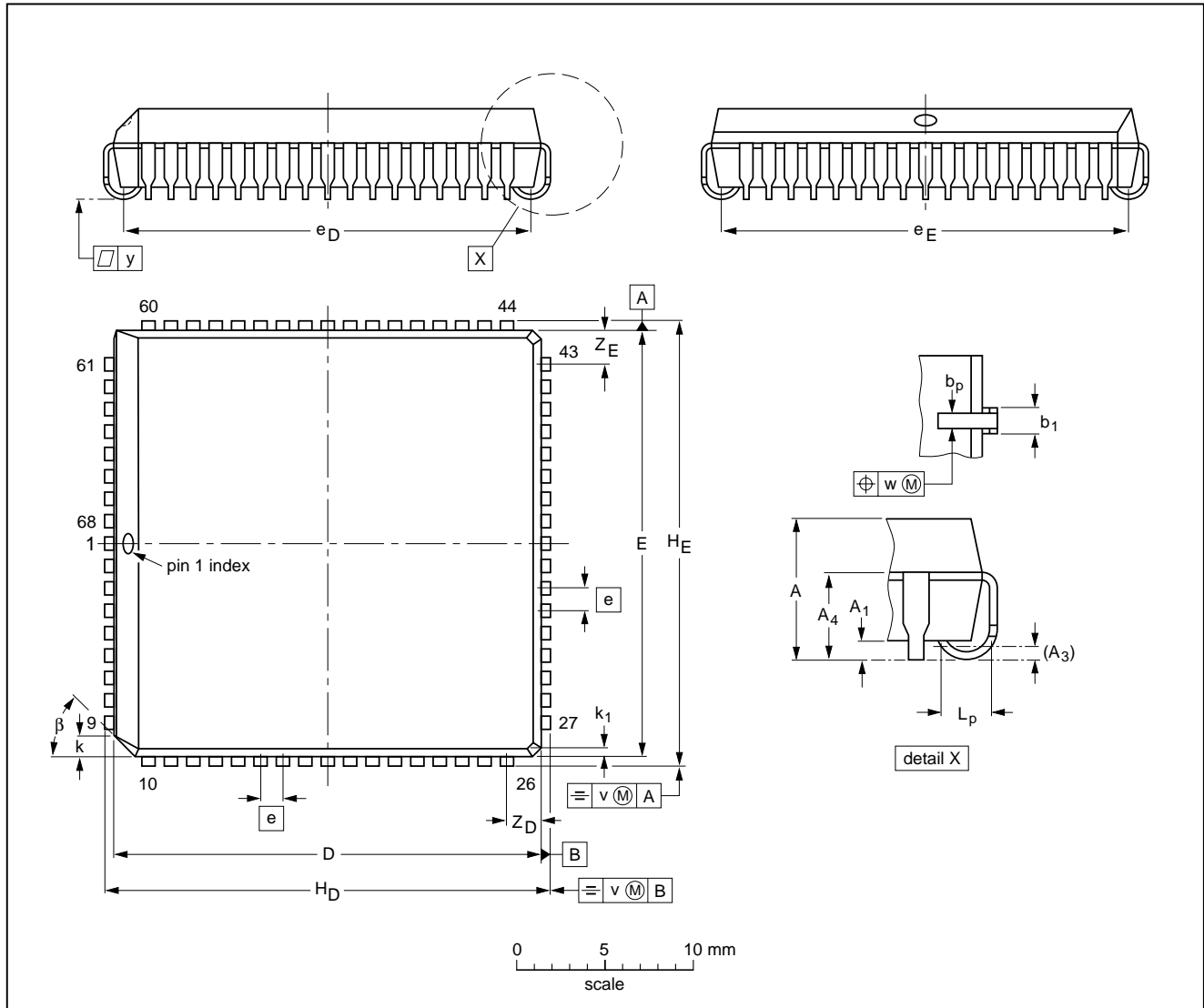
8-bit microcontroller

P83C562; P80C562

22 PACKAGE OUTLINE

PLCC68: plastic leaded chip carrier; 68 leads

SOT188-2



DIMENSIONS (millimetre dimensions are derived from the original inch dimensions)

| UNIT   | A              | A <sub>1</sub> min. | A <sub>3</sub> | A <sub>4</sub> max. | b <sub>p</sub> | b <sub>1</sub> | D <sup>(1)</sup> | E <sup>(1)</sup> | e    | e <sub>D</sub> | e <sub>E</sub> | H <sub>D</sub> | H <sub>E</sub> | k              | k <sub>1</sub> max. | L <sub>p</sub> | v     | w     | y     | Z <sub>D</sub> <sup>(1)</sup> max. | Z <sub>E</sub> <sup>(1)</sup> max. | $\beta$ |
|--------|----------------|---------------------|----------------|---------------------|----------------|----------------|------------------|------------------|------|----------------|----------------|----------------|----------------|----------------|---------------------|----------------|-------|-------|-------|------------------------------------|------------------------------------|---------|
| mm     | 4.57<br>4.19   | 0.51                | 0.25           | 3.30                | 0.53<br>0.33   | 0.81<br>0.66   | 24.33<br>24.13   | 24.33<br>24.13   | 1.27 | 23.62<br>22.61 | 23.62<br>22.61 | 25.27<br>25.02 | 25.27<br>25.02 | 1.22<br>1.07   | 0.51                | 1.44<br>1.02   | 0.18  | 0.18  | 0.10  | 2.16                               | 2.16                               | 45°     |
| inches | 0.180<br>0.165 | 0.020               | 0.01           | 0.13                | 0.021<br>0.013 | 0.032<br>0.026 | 0.958<br>0.950   | 0.958<br>0.950   | 0.05 | 0.930<br>0.890 | 0.930<br>0.890 | 0.995<br>0.985 | 0.995<br>0.985 | 0.048<br>0.042 | 0.020               | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.085                              | 0.085                              |         |

Note  
1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

| OUTLINE VERSION | REFERENCES |          |      |  | EUROPEAN PROJECTION | ISSUE DATE           |
|-----------------|------------|----------|------|--|---------------------|----------------------|
|                 | IEC        | JEDEC    | EIAJ |  |                     |                      |
| SOT188-2        | 112E10     | MO-047AC |      |  |                     | 92-11-17<br>95-03-11 |

---

## 8-bit microcontroller

## P83C562; P80C562

---

### 23 SOLDERING

#### 23.1 Introduction

There is no soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and surface mounted components are mixed on one printed-circuit board. However, wave soldering is not always suitable for surface mounted ICs, or for printed-circuits with high population densities. In these situations reflow soldering is often used.

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our *"IC Package Databook"* (order code 9398 652 90011).

#### 23.2 Reflow soldering

Reflow soldering techniques are suitable for all PLCC packages.

The choice of heating method may be influenced by larger PLCC packages (44 leads, or more). If infrared or vapour phase heating is used and the large packages are not absolutely dry (less than 0.1% moisture content by weight), vaporization of the small amount of moisture in them can cause cracking of the plastic body. For more information, refer to the Drypack chapter in our *"Quality Reference Handbook"* (order code 9397 750 00192).

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

Several techniques exist for reflowing; for example, thermal conduction by heated belt. Dwell times vary between 50 and 300 seconds depending on heating 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 minutes at 45 °C.

#### 23.3 Wave soldering

Wave soldering techniques can be used for all PLCC packages if the following conditions are observed:

- A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used.
- The longitudinal axis of the package footprint must be parallel to the solder flow.
- The package footprint must incorporate solder thieves at the downstream corners.

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

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

A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

#### 23.4 Repairing soldered joints

Fix the component by first soldering two diagonally-opposite end leads. Use only a low voltage soldering iron (less than 24 V) applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300 °C. When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320 °C.

## 8-bit microcontroller

## P83C562; P80C562

**24 DEFINITIONS**

|  |   |
|--|---|
| <b>Data sheet status</b>   |   |
| 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.                                |
| <b>Limiting values</b>   |   |
| 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 this specification is not implied. Exposure to limiting values for extended periods may affect device reliability. |   |
| <b>Application information</b>   |   |
| Where application information is given, it is advisory and does not form part of the specification.  |   |

**25 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.

# Philips Semiconductors – a worldwide company

**Argentina:** see South America

**Australia:** 34 Waterloo Road, NORTH RYDE, NSW 2113,  
Tel. +61 2 9805 4455, Fax. +61 2 9805 4466

**Austria:** Computerstr. 6, A-1101 WIEN, P.O. Box 213,  
Tel. +43 1 60 101, Fax. +43 1 60 101 1210

**Belarus:** Hotel Minsk Business Center, Bld. 3, r. 1211, Volodarski Str. 6,  
220050 MINSK, Tel. +375 172 200 733, Fax. +375 172 200 773

**Belgium:** see The Netherlands

**Brazil:** see South America

**Bulgaria:** Philips Bulgaria Ltd., Energoproject, 15th floor,  
51 James Bourchier Blvd., 1407 SOFIA,  
Tel. +359 2 689 211, Fax. +359 2 689 102

**Canada:** PHILIPS SEMICONDUCTORS/COMPONENTS,  
Tel. +1 800 234 7381

**China/Hong Kong:** 501 Hong Kong Industrial Technology Centre,  
72 Tat Chee Avenue, Kowloon Tong, HONG KONG,  
Tel. +852 2319 7888, Fax. +852 2319 7700

**Colombia:** see South America

**Czech Republic:** see Austria

**Denmark:** Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S,  
Tel. +45 32 88 2636, Fax. +45 31 57 0044

**Finland:** Sinikalliontie 3, FIN-02630 ESPOO,  
Tel. +358 9 615800, Fax. +358 9 61580920

**France:** 4 Rue du Port-aux-Vins, BP317, 92156 SURESNES Cedex,  
Tel. +33 1 40 99 6161, Fax. +33 1 40 99 6427

**Germany:** Hammerbrookstraße 69, D-20097 HAMBURG,  
Tel. +49 40 23 53 60, Fax. +49 40 23 536 300

**Greece:** No. 15, 25th March Street, GR 17778 TAVROS/ATHENS,  
Tel. +30 1 4894 339/239, Fax. +30 1 4814 240

**Hungary:** see Austria

**India:** Philips INDIA Ltd, Shivsagar Estate, A Block, Dr. Annie Besant Rd.  
Worli, MUMBAI 400 018, Tel. +91 22 4938 541, Fax. +91 22 4938 722

**Indonesia:** see Singapore

**Ireland:** Newstead, Clonskeagh, DUBLIN 14,  
Tel. +353 1 7640 000, Fax. +353 1 7640 200

**Israel:** RAPAC Electronics, 7 Kehilat Saloniki St, PO Box 18053,  
TEL AVIV 61180, Tel. +972 3 645 0444, Fax. +972 3 649 1007

**Italy:** PHILIPS SEMICONDUCTORS, Piazza IV Novembre 3,  
20124 MILANO, Tel. +39 2 6752 2531, Fax. +39 2 6752 2557

**Japan:** Philips Bldg 13-37, Kohnan 2-chome, Minato-ku, TOKYO 108,  
Tel. +81 3 3740 5130, Fax. +81 3 3740 5077

**Korea:** Philips House, 260-199 Itaewon-dong, Yongsan-ku, SEOUL,  
Tel. +82 2 709 1412, Fax. +82 2 709 1415

**Malaysia:** No. 76 Jalan Universiti, 46200 PETALING JAYA, SELANGOR,  
Tel. +60 3 750 5214, Fax. +60 3 757 4880

**Mexico:** 5900 Gateway East, Suite 200, EL PASO, TEXAS 79905,  
Tel. +9-5 800 234 7381

**Middle East:** see Italy

**Netherlands:** Postbus 90050, 5600 PB EINDHOVEN, Bldg. VB,  
Tel. +31 40 27 82785, Fax. +31 40 27 88399

**New Zealand:** 2 Wagener Place, C.P.O. Box 1041, AUCKLAND,  
Tel. +64 9 849 4160, Fax. +64 9 849 7811

**Norway:** Box 1, Manglerud 0612, OSLO,  
Tel. +47 22 74 8000, Fax. +47 22 74 8341

**Philippines:** Philips Semiconductors Philippines Inc.,  
106 Valero St. Salcedo Village, P.O. Box 2108 MCC, MAKATI,  
Metro MANILA, Tel. +63 2 816 6380, Fax. +63 2 817 3474

**Poland:** Ul. Lukiska 10, PL 04-123 WARSZAWA,  
Tel. +48 22 612 2831, Fax. +48 22 612 2327

**Portugal:** see Spain

**Romania:** see Italy

**Russia:** Philips Russia, Ul. Usatcheva 35A, 119048 MOSCOW,  
Tel. +7 095 755 6918, Fax. +7 095 755 6919

**Singapore:** Lorong 1, Toa Payoh, SINGAPORE 1231,  
Tel. +65 350 2538, Fax. +65 251 6500

**Slovakia:** see Austria

**Slovenia:** see Italy

**South Africa:** S.A. PHILIPS Pty Ltd., 195-215 Main Road Martindale,  
2092 JOHANNESBURG, P.O. Box 7430 Johannesburg 2000,  
Tel. +27 11 470 5911, Fax. +27 11 470 5494

**South America:** Rua do Rocio 220, 5th floor, Suite 51,  
04552-903 São Paulo, SÃO PAULO - SP, Brazil,  
Tel. +55 11 821 2333, Fax. +55 11 829 1849

**Spain:** Balmes 22, 08007 BARCELONA,  
Tel. +34 3 301 6312, Fax. +34 3 301 4107

**Sweden:** Kottbygatan 7, Akalla, S-16485 STOCKHOLM,  
Tel. +46 8 632 2000, Fax. +46 8 632 2745

**Switzerland:** Allmendstrasse 140, CH-8027 ZÜRICH,  
Tel. +41 1 488 2686, Fax. +41 1 481 7730

**Taiwan:** Philips Semiconductors, 6F, No. 96, Chien Kuo N. Rd., Sec. 1,  
TAIPEI, Taiwan Tel. +886 2 2134 2865, Fax. +886 2 2134 2874

**Thailand:** PHILIPS ELECTRONICS (THAILAND) Ltd.,  
209/2 Sanpavuth-Bangna Road Prakanong, BANGKOK 10260,  
Tel. +66 2 745 4090, Fax. +66 2 398 0793

**Turkey:** Talatpasa Cad. No. 5, 80640 GÜLTEPE/ISTANBUL,  
Tel. +90 212 279 2770, Fax. +90 212 282 6707

**Ukraine:** PHILIPS UKRAINE, 4 Patrice Lumumba str., Building B, Floor 7,  
252042 KIEV, Tel. +380 44 264 2776, Fax. +380 44 268 0461

**United Kingdom:** Philips Semiconductors Ltd., 276 Bath Road, Hayes,  
MIDDLESEX UB3 5BX, Tel. +44 181 730 5000, Fax. +44 181 754 8421

**United States:** 811 East Arques Avenue, SUNNYVALE, CA 94088-3409,  
Tel. +1 800 234 7381

**Uruguay:** see South America

**Vietnam:** see Singapore

**Yugoslavia:** PHILIPS, Trg N. Pasica 5/v, 11000 BEOGRAD,  
Tel. +381 11 625 344, Fax. +381 11 635 777

**For all other countries apply to:** Philips Semiconductors, Marketing & Sales Communications,  
Building BE-p, P.O. Box 218, 5600 MD EINDHOVEN, The Netherlands, Fax. +31 40 27 24825

**Internet:** <http://www.semiconductors.philips.com>

© Philips Electronics N.V. 1997

SCA54

All rights are reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner.

The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent- or other industrial or intellectual property rights.

Printed in The Netherlands

457047/1200/01/pp52

Date of release: 1997 Apr 08

Document order number: 9397 750 02133

*Let's make things better.*

**Philips  
Semiconductors**



**PHILIPS**