

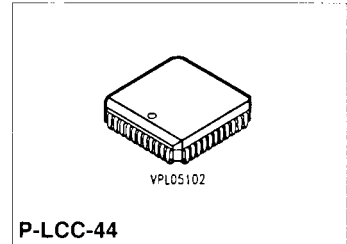
CMOS Dual-Port RAM

SAE 81C80 A

CMOS IC

Features

- Processor interface with address and data bus plus signals ALE, WR, RD
- 8051-, 8096-compatible timing
- Memory capacity 504 bytes
- All functions fully static (excl. oscillator watchdog)
- Standby operation
- On-chip oscillator with separate clock output
- Eight scheduling registers
- Three loadable timers for processor monitoring or applicable as longterm timers
- Monitoring of internal oscillator (hardware watchdog)
- Three outputs for interrupt triggering (can be set on the bus)
- Fully asynchronous operation of two processors possible
- Data retention down to 1 V
- PL-CC-44 (SMD) package
- Extended temperature range from – 40 through 110 °C
- CMOS technology

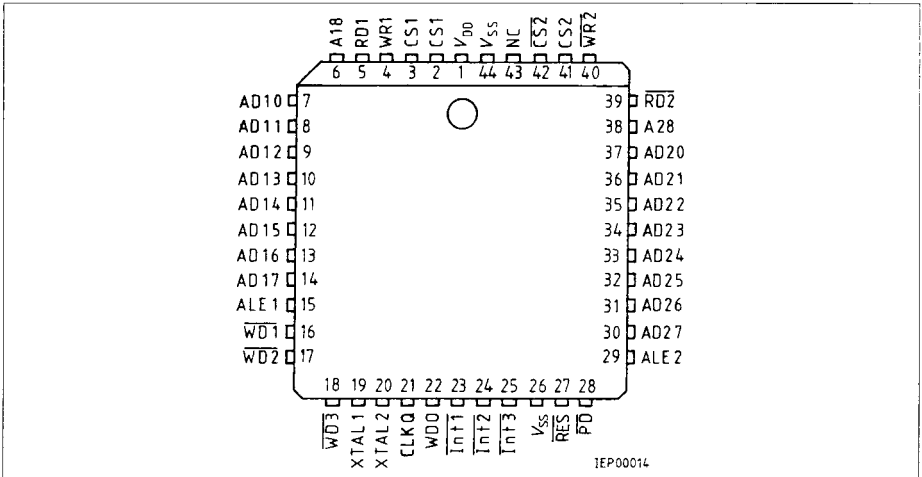


Type	Ordering Code	Package
SAE 81C80 A	Q67100-H8706	P-LCC-44 (SMD)

The SAE 81C80 A dual-port RAM (DPR) is a CMOS memory IC with a capacity of 504 bytes (**figure 1**).

A very notable feature of this DPR is that it can be used by two microcontrollers (MCs) simultaneously and fully asynchronously. Each microcontroller uses the DPR like a normal static RAM. Thus, when comparing the circuit development of this DPR with that of standard memory, no extra effort is required. Access collisions are excluded, which is the pre-requisite for fast communication between the two MCs.

The SAE 81C80 A DPR is ideally suited for multi-processor/multi-controller applications like master/slave configurations or controls where one controller acquires measured data and a second one controls the actuators (e.g. in motors, etc.). (See **figures 2 and 3**).



Pin Configuration

(top view)

Pin Definitions and Function

Pin	Symbol	Function
7	AD10	Data and address bus port 1
8	AD11	
9	AD12	
10	AD13	
11	AD14	
12	AD15	
13	AD16	
14	AD17	
06	A18	Address 8 port 1
37	AD20	Data and address bus port 2
36	AD21	
35	AD22	
34	AD23	
33	AD24	
32	AD25	
31	AD26	
30	AD27	
38	A28	Address 8 port 2
15	ALE1	Address latch enable port 1
29	ALE2	Address latch enable port 2
		These signals are for separating data and addresses on the bus. The address is stored on the falling edge of the signal.

Pin Definitions and Function (cont'd)

Pin	Symbol	Function
5 39	RD1 RD2	Read signal port 1 (active low) Read signal port 2 (active low)
4 40	$\overline{WR1}$ WR2	Write signal port 1 (active low) Write signal port 2 (active low)
3 2	CS1 CS1	Chip select port 1 Chip select port 1 (active low)
41 42	CS2 CS2	Chip select port 2 Chip select port 2 (active low) The chip-select inputs select a port when the two associated inputs are on active level.
27	RES	Reset input Resets the IC to a defined initial state when RES is low. At the same time the outputs WD1, WD2, WD3 are switched low for the duration of the reset pulse. The oscillator continues to operate.
28	\overline{PD}	Power-down Disables all other inputs and the oscillator when \overline{PD} is low.
44 1	V _{SS} V _{DD}	Negative supply voltage Positive supply voltage
43	NC	Not connected
19	XTAL1	Pin for crystal (must remain open for external clock supply).
20	XTAL2	Pin for crystal or applying external clock
21	CLKQ	Clock output
22	WDO	Oscillator watchdog (open-drain output) High indicates that the oscillator is working.
16 17 18	WD1 WD2 WD3	Open-drain outputs of three timers
26	V _{SS}	No function (must be connected)
23 24 25	Int1 Int2 Int3	Open-drain outputs Outputs that can be controlled via the port for triggering an interrupt on a processor for example.

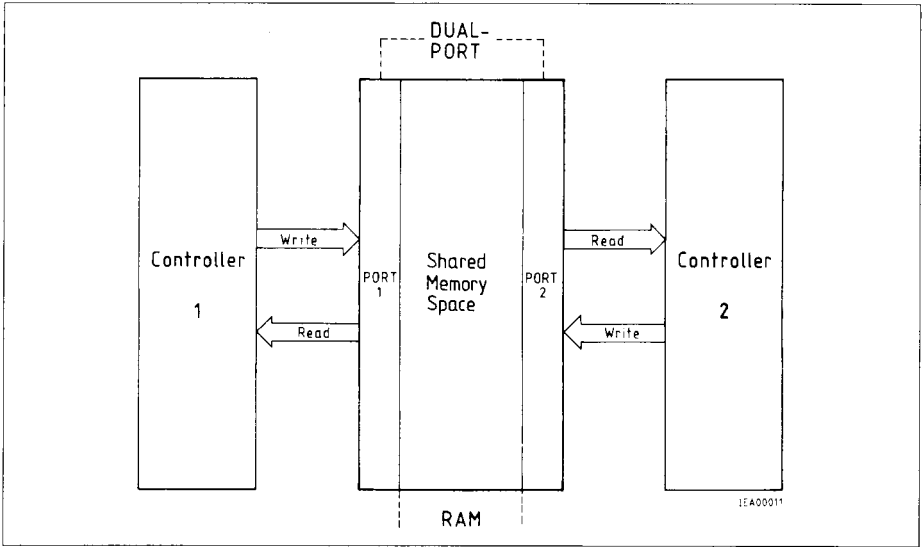


Figure 1
Principle of the Dual-Port RAM (DPR)

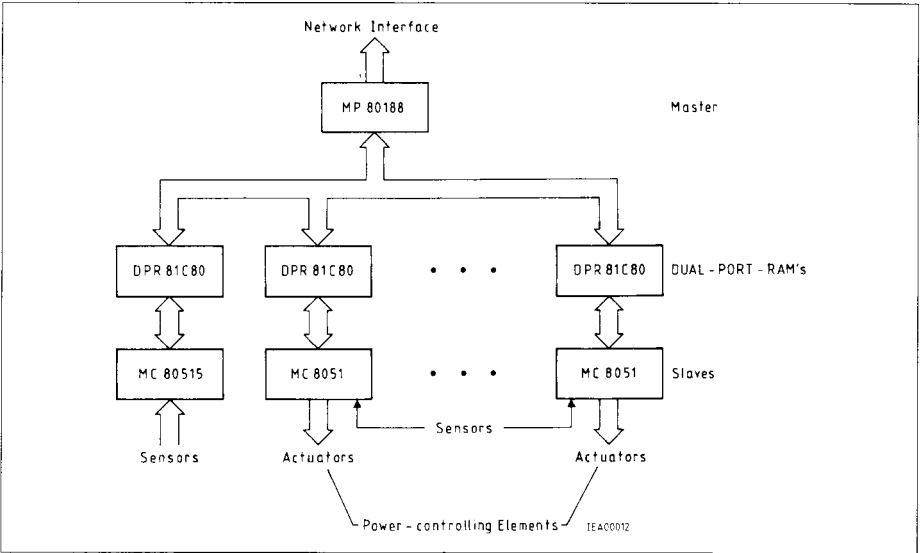


Figure 2
Interfacing Master and Slave Processors by DPRs

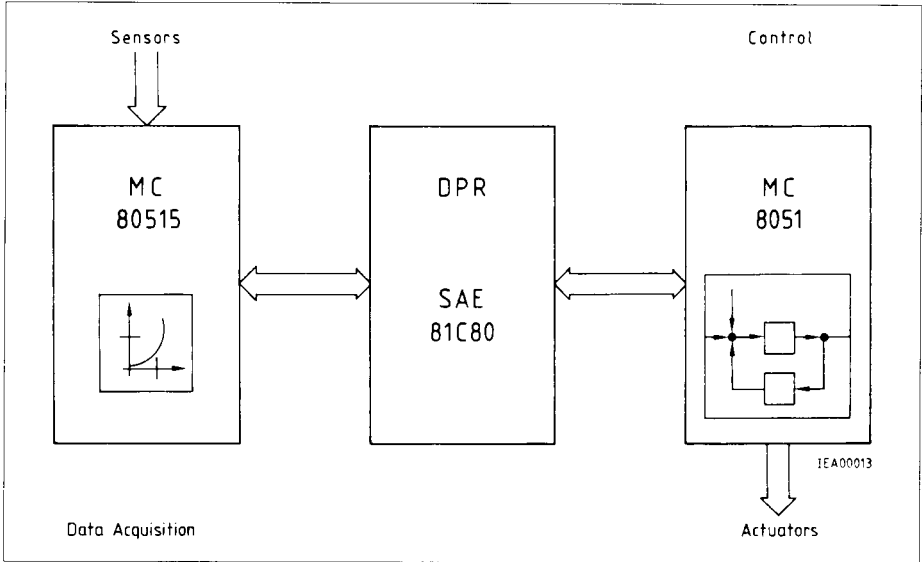


Figure 3
Dual-Port RAM used between Measured-Data Acquisition and the Actuators

Functional Description

Dual-Port RAM

The SAE 81C80A is a 504-byte static RAM simultaneously accessible by two microcontrollers. The memory locations are selected via a multiplexed address/data bus and two chip-select inputs. The direction of data transfer is determined by the RD and WR inputs.

There will be no undefined states when a memory location is concurrently accessed by two processors, even if they write simultaneously to the same memory location. Depending on the internal state of the access control and the actual physical sequence, the value of one of the two ports will be stored. Also, if one memory location is read and written to at the same time, the data will not be mixed, i.e. either the original data or the new data are read out.

Chip-Select Inputs

The chip-select inputs affect signals WR and RD, but not the ALE input. Therefore, the ALE signal on the DPR (even if the DPR is not selected) must correspond to the specified values. To eliminate selection, it is sufficient if one of the two chip-select inputs becomes inactive when the falling edge of WR or RD appears.

Reset

The reset is necessary for setting the control units of the DPR to a defined initial state. It initializes the timer-mode registers with the values 0000XXX0_b (timers 1 and 2) and 00000XX0_b (timer 3). The INT outputs are set to 0.

The reset input is a TTL input without Schmitt-trigger response. For this reason, neither an ALE nor a WR signal must be applied to the DPR if the voltage on the reset input is below V_{IH}. The length of the Reset pulse must be greater than six clock (oscillator) cycles and the clock must be active.

When the reset input is low, outputs $\overline{WD1}$, $\overline{WD2}$ and $\overline{WD3}$ are set to low. After a reset these outputs are high. The scheduling registers are set to state 1 by reset.

A reset is also necessary if the DPR is reactivated from power-down, while the contents of the RAM and oscillator remain unaffected.

Power-Down Mode

When the power-down mode (\overline{PD}) is activated, all inputs (except \overline{PD} and XTAL1, XTAL2) plus the oscillator are disabled. This means that any levels are possible on the remaining inputs.

An active level on \overline{PD} also produces an internal reset. Nevertheless, to ensure proper operation after deactivation of the power-down mode, an external reset should be made to bridge the time required by the oscillator for buildup. The outputs of the ports go high-impedance, while outputs CLKO, \overline{WDO} , $\overline{WD1}$, $\overline{WD2}$, $\overline{WD3}$, $\overline{INT1}$, INT2 and $\overline{INT3}$ are set to low. The \overline{PD} input shows a Schmitt-trigger response. This allows V_{DD} to be evaluated directly, for example (see application circuit).

Interrupt Outputs

The DPR has three interrupt outputs that can be set and reset directly by writing to an address (see table 1). The outputs are located in the same address range as the scheduling registers. However, only bits 2 and 3 are relevant for the interrupt outputs. At least one of bits 0 and 1 should be other than 1 to prevent the scheduling registers from being affected. The functions of the outputs are shown in the following table:

RES	Bit 3	Bit 2	Output
1	0	0	no change
1	0	1	1
1	1	0	0
1	1	1	undefined
0	–	–	0

Oscillator Watchdog

This part of the circuit is a fail-save mechanism for the oscillator. If the frequency of the clock is missing, the output switches to \overline{WDO} low. The circuit works like an analog integrator. Below approx. 100 kHz, low pulses are produced on the output. The pulse width depends on the clock frequency. This part of the circuit should not be used at operating frequencies of less than 500 kHz.

Supply Voltage

To prevent any interference, the supply voltage of the DPR should be blocked as close as possible to the pins with a capacitor of approx. 100 nF (**see application circuit**).

Timers

The three timers are 24-bit counters with a clock frequency of $f_{osc}/6$. Each of the counters can be set by writing to three specific RAM addresses. The value is then simultaneously stored in the RAM and a buffer register of the timer. When the low byte is written, all three bytes are parallelly stored in the reload register. The value in the reload register is kept in all operating modes until the associated low byte is written again.

The counters are down-counters. They can be started by setting bit 7 in the associated timer-mode register (TMR). Each counter can be configured by a TMR. The bits of the TMRs have the following function:

Bit 0: This bit provides overwrite protection for the reload register.

Use: After writing to the reload registers and starting of the timer – by writing to the associated protection bit – the adjacent RAM area can be used without affecting the reload register (reset state = 0).

Bit 4: It serves for switching the polarity of the output signal (reset state = 0).

Bit 4 = 0; idle state 1, active 0

Bit 4 = 1; idle state 0, active 1

Bit 5: This bit switches the operating mode (reset state = 0).

Bit 5 = 0 single-shot, i.e. when the counter is started, the output signal becomes active. After reaching zero, the output signal is reset. The timer has to be restarted to trigger another count cycle. The values from the reload register are then loaded into the counter.

Bit 5 = 1 auto reload, i.e. when the counter is started, the value of the reload register is loaded into it. When zero is reached, the counter issues a pulse ($\approx 4 \mu\text{s}$ at 12 MHz), automatically reloads the original value and the entire operation starts again. In this way a frequency can be set with a resolution of 24 bits. Because of the pulse width of eight timer clock pulses, however, the shortest period is limited to nine timer clock pulses ($t_{OSX} \times 6$). If a new start pulse appears in the count cycle (even without "STOP"), no pulse is issued and the counter is reloaded.

Bit 6: In the reload mode the timer can be halted by setting this bit and resetting bit 5. (In a new start the contents of the counter are lost and that of the reload registers remain unaffected).

Bit 7: Setting this bit starts the counter.

Only for the registers of timers 1 and 2

Bit 1-3: These are used together with bit 0 for switching the watchdog mode ON and OFF.

Only for the register of timer 3

Bit 1-2: Reserved (should always be 0 for correct operation).

Bit 3: Switches **all three timers** to test mode, i.e. only the upper twelve bits are used to generate the output signal (reset state = 0).

Watchdog Mode

For timers 1 and 2a special mode was implemented which can be used to monitor the two processors. In this mode there is a control register (CR) for each timer (**see table 1** for addresses). The watchdog mode is set by loading the TMR with the value 101X1111B, the polarity of the output signal being freely selectable with bit 4. This mode works similarly to the auto-reload mode, but neither the reload register nor the TMR can be altered.

In the watchdog mode, the timer can only be restarted (and the output pulse suppressed) if the values 055H and 0AAH are successively written into the control register. The time between these two write operations is random, but the sequence must be completed before the timer has run down, i.e. the output pulse is generated. No value may be written into either the TMR or CR between the two write operations, otherwise the sequence has to be started again.

To reset the timer to the normal mode, first the value 055H has to be written into the CR, then the value 010X0000B into the TMR, and finally the value 0AAH into the CR. Here, too, if any other value is written into either of the two registers during the sequence, the entire operation has to be started again. The time between the accesses is random.

The timer operation in watchdog mode is illustrated in the appendix in an 8051 example program.

Note: The relevant bits for changing the timer state to watchdog mode are bit 0 - bit 3; the shown pattern is the only one, which makes sense for this mode.

Figure 4
Bit Assignment of Timer-Mode Registers for Timer 1 and 2

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Software start (=1)	Timer stop (=1) for auto-reload	Mode (auto-reload=1, single-shot = 0)	Polarity of output pulse (High = 0)	Only for watchdog mode (normal mode = 0)	Only for watchdog mode (normal mode = 0)	Only for watchdog mode (normal mode = 0)	Protection (=1) against overwriting of reload register

Figure 5
Bit Assignment of Timer-Mode Register for Timer 3

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Software start (=1)	Timer stop (=1) for auto-reload	Mode (auto-reload =1, single-shot = 0)	Polarity of output pulse (High = 0)	Test (=1) switches timer to test mode	Reserved (normal mode = 0)	Reserved (normal mode = 0)	Protection (=1) against overwriting of reload register

Access Collisions

With a RAM which can be written to or read simultaneously by two controllers, different kinds of access collision are possible:

1. Simultaneous read access to the same memory location from both ports;
2. One port reads the same memory location which the other port writes to concurrently
3. Concurrent write access to the same memory location from both ports;
4. Read access to a logically linked data block by one port, while the other port modifies the same data block.

The SAE 81C80A dual-port RAM avoids the first three types of access collision by hardware. The fourth problem can be solved by user software.

The standard solution for the access collisions described above would be as follows: **before** accessing the memory area, an additional memory location must be established by setting an access flag (semaphore). This would necessitate three memory operations:

- First access: read the flag and check whether the data area is free.
- Second access: write the flag with the data for reservation.
- Third access: read the flag and make sure that your own reservation has not been overwritten by the other port.

Only after this sequence would a microcontroller be privileged for access and could write or read to the data area without the risk of contention.

With the SAE 81C80A dual-port RAM this access routine is simplified using scheduling registers.

Scheduling Registers

Note: The assignment of a memory area to a scheduling register is defined by the user software of both controllers

With the scheduling registers synchronization can be done with only one access because the reservation is performed during reading. The other port cannot overwrite it.

This means that a scheduling register is **written by reading**, unless it was occupied.

The description above shows that these registers are no ordinary RAM locations. They are formed by a finite state machine (FSM), which can assume the following four states (see figure 6):

- State 1: port 1 was the previous owner and the register is free.
- State 2: port 1 occupies the register.
- State 3: port 2 was the previous owner and the register is free.
- State 4: port 2 occupies the register.

The state of a register can be read out from the particular address, but causes also a change in the state of the FSM (arrows in figure 6). Reading produces 2-bit information:

- Bit 0 is the owner bit. It is set when the reading port is or was the owner of a register.
- Bit 1 is the occupied bit. It is set when a register has been reserved by a port.
- Bit 2 through 7 are always 0.

Reserving is done by reading a register and enabling by writing to it XXXXXX11_B (pay attention to the interrupt outputs of bits 2 and 3!). Thus a correct protocol using the scheduling registers takes the following form:

1. Read the scheduling register.
2. Check whether the occupied bit is set and the owner bit is not set (i.e. the other port has reserved). If so, go back to 1, otherwise continue.
3. Process the data area.
4. Enable the scheduling register by writing 03_H to the address of the register
5. End

In cases where accessing of a data area requires prior reading of or writing to this data area by the second processor, a separate evaluation of the occupied bit and owner bit can be done in step 2:

- 2a. Owner bit self? If so, continue to 2c, otherwise to 2b.
- 2b. Occupied bit self? If so, continue to 2c, otherwise to 3.
- 2c. Enable the scheduling register by writing 03_H to the address of the register (continue with 1).

The following applies only to the scheduling registers:

Usually, in the case of a concurrent access by both processors, writing has priority over reading. However, a simultaneous read or write access from the two ports means that port 1 has priority over port 2.

The addresses of the scheduling registers are listed in **table 1**.

The assignment of scheduling registers to specific data areas is made by the user. The software (of both controllers) should be configured so that, prior to accessing a logically related data area, the associated scheduling register is accessed first (according to the above sequence).

So the assignment of the various dual-port RAM address spaces to scheduling registers will depend solely on the structure of the user software.

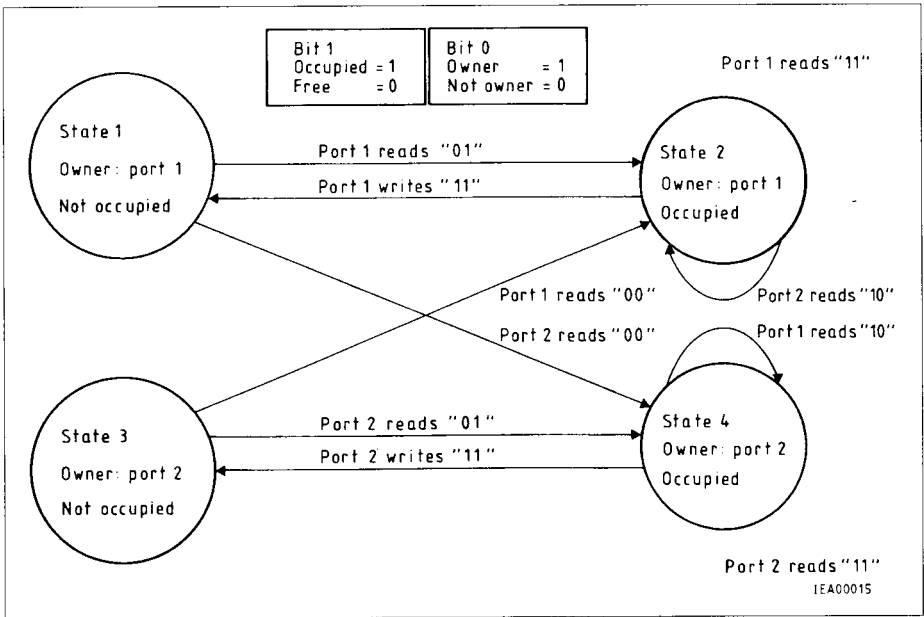


Figure 6
Diagram Showing the Various States of the Scheduling Registers

Only the two least significant bits of the data are shown (in converted commas).

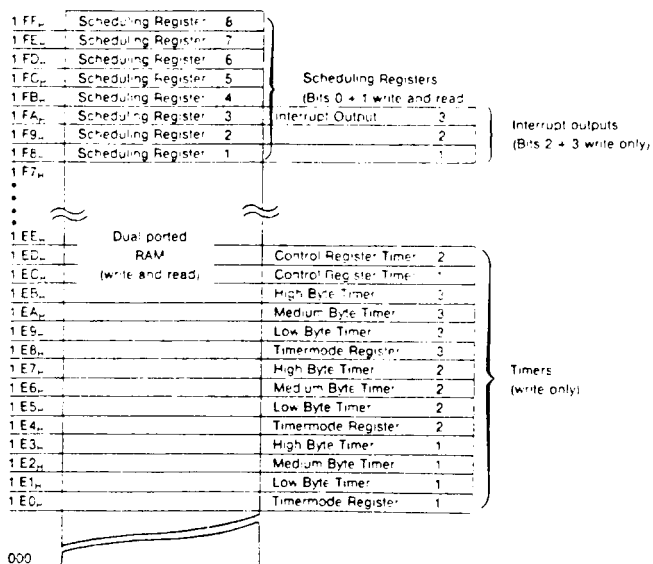
Notes:

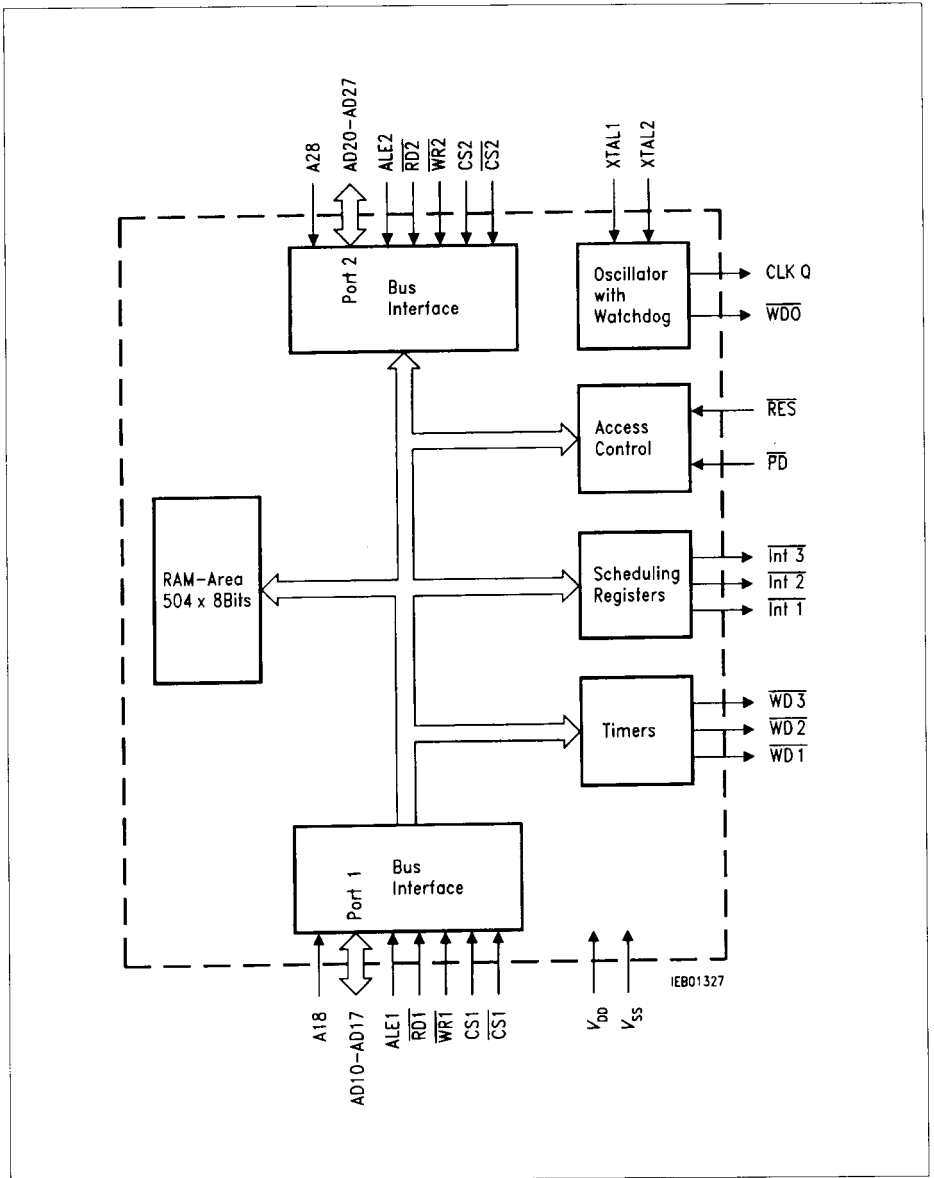
- 1) The owner bit indicates the last owner of a register
- 2) Only if the port is owner of the register will writing change the state.
- 3) The reset state is state 1.
- 4) The FSM is symmetrical. Therefore, the two processors can use the same program.

Table 1
Address Assignment of DPR Registers

Register	Address	Register	Address
Scheduling register 1	1F8H	High-byte timer 2	1E7H
Scheduling register 2	1F9H	Medium-byte timer 2	1E6H
Scheduling register 3	1FAA	Low-byte timer 2	1E5H
Scheduling register 4	1FBH		
Scheduling register 5	1FC H	High-byte timer 3	1EB H
Scheduling register 6	1FDH	Medium-byte timer 3	1EA H
Scheduling register 7	1FEH	Low-byte timer 3	1E9H
Scheduling register 8	1FFH		
Timer-mode register 1	1E0H	Control-register timer 1	1EC H
Timer-mode register 2	1E4H	Control-register timer 2	1EDH
Timer-mode register 3	1E8H		
High-byte timer 1	1E3H	Interrupt output 1	1F8H
Medium-byte timer 1	1E2H	Interrupt output 2	1F9H
Low-byte timer 1	1E1H	Interrupt output 3	1FA A

Figure 7
Memory Map





Block Diagram

Absolute Maximum Ratings

$T_A = -40$ to 110 °C; all voltages referred to V_{SS}

Parameter	Symbol	Limit Values			Unit
		min.	typ.	max.	
Storage temperature	T_{stg}	- 50	-	125	°C
Total power dissipation	P_{tot}	-	-	500	mW
Power dissipation per output	P_Q	-	-	50	mW
Input voltage	V_i	- 0.5	-	$V_{DD} + 0.5$	V
Supply voltage	V_{DD}	- 0.5	-	6	V

Operating Range

Supply voltage	V_{DD}	4.5	5	5.5	V
Supply current (w/o loading of outputs)	I_{DD}	-	-	20	mA
Operating frequency	f_s	-	-	12	MHz
Ambient temperature *)	T_A	- 40	-	110	°C
Standby current	I_{DD}	-	-	1	μA
Data-retention voltage	V_{DH}	1	-	-	V

DC Characteristics

$T_A = 25$ °C; all voltages referred to V_{SS}

Parameter	Symbol	Limit Values			Test Condition
		min.	max.	Unit	

All Input Signals Except XTAL2 and PD

H-input voltage	V_{IH}	2.2	V_{DD}	V	-
L-input voltage	V_{IL}	0	0.8	V	-
Input capacitance	C_i	-	10	pF	-
Input current	I_i	-	1	μA	-

XTAL2 (as external clock input)

H-input voltage	V_{IH}	3.5	V_{DD}	V	-
L-input voltage	V_{IL}	0	0.5	V	-
Input capacitance	C_i	-	10	pF	-

PD (Schmitt-trigger characteristics)

H-input voltage	V_{IH}	$V_{DD} - 1$	V_{DD}	V	-
L-input voltage	V_{IL}	0	1.0	V	-
Input capacitance	C_i	-	10	pF	-

Output Signals AD10-17, AD20-27

H-output voltage	V_{OH}	2.4	V_{DD}	V	$I_Q = 0.5$ mA
L-output voltage	V_{OL}	-	0.4	V	$I_Q = 1.6$ mA

DC Characteristics (cont'd)

$T_A = 25\text{ }^\circ\text{C}$; all voltages referred to V_{SS}

Parameter	Symbol	Limit Values			Test Condition
		min.	max.	Unit	

Output Signals WD1, WD2, WD3, WD0 (open drain, weak pull-up)

L-output voltage	V_{OL}	–	0.4	V	$I_O = 1.6\text{ mA}$
------------------	----------	---	-----	---	-----------------------

Output Signal Clock-Out

H-output voltage	V_{OH}	2.4	–	V	$I_O = 0.5\text{ mA}$
L-output voltage	V_{OL}	–	0.4	V	$I_O = 1.6\text{ mA}$
Load capacitance	C_L	–	80	pF	–

AC Characteristics

The AC characteristics apply throughout the operating range $T_A = 25\text{ }^\circ\text{C}$.

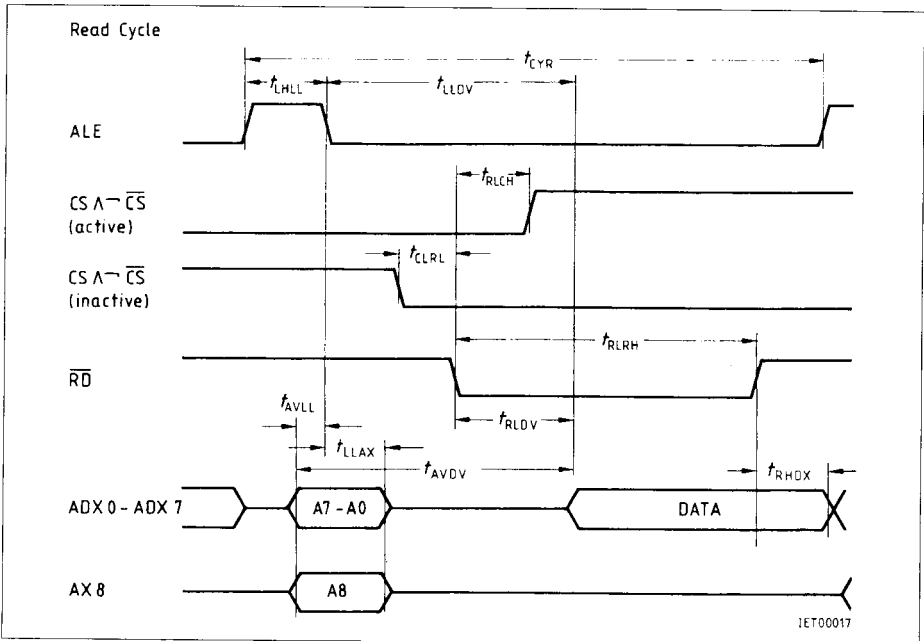
Parameter	Symbol	Limit Values		Unit
		min.	max.	
Read cycle time	t_{CYR}	$300 + t_{LHLL}$	–	ns
Write cycle time	t_{CYW}	$440 + t_{LHLL}$	–	ns
ALE pulse width	t_{LHLL}	40	–	ns
Address setup to ALE low	t_{AVLL}	30	–	ns
Address hold after ALE low	t_{LLAX}	40	–	ns
RD pulse width	t_{RLRH}	120	–	ns
WR pulse width	t_{WLWH}	120	–	ns
ALE low to RD or WR active	t_{LLWL}	30	–	ns
Data hold after RD high	t_{RHDX}	0	30	ns
ALE low to valid data out	t_{LLDV}	–	290	ns
RD low to data valid (only scheduling registers)	t_{RLDV}	–	$2 t_{OSC} + 20$	ns
Valid data in after WR low	t_{DVWL}	–	30	ns
WR low to ALE high	t_{WLLH}	150	–	ns
Data setup before WR high	t_{QVWH}	30	–	ns
Data hold after WR high	t_{WHQX}	30	–	ns
Delay RD low to both chip select active	t_{RLCH}	–	20	ns
Delay WR low to both chip select active	t_{WLCH}	–	20	ns

AC Characteristics (cont'd)

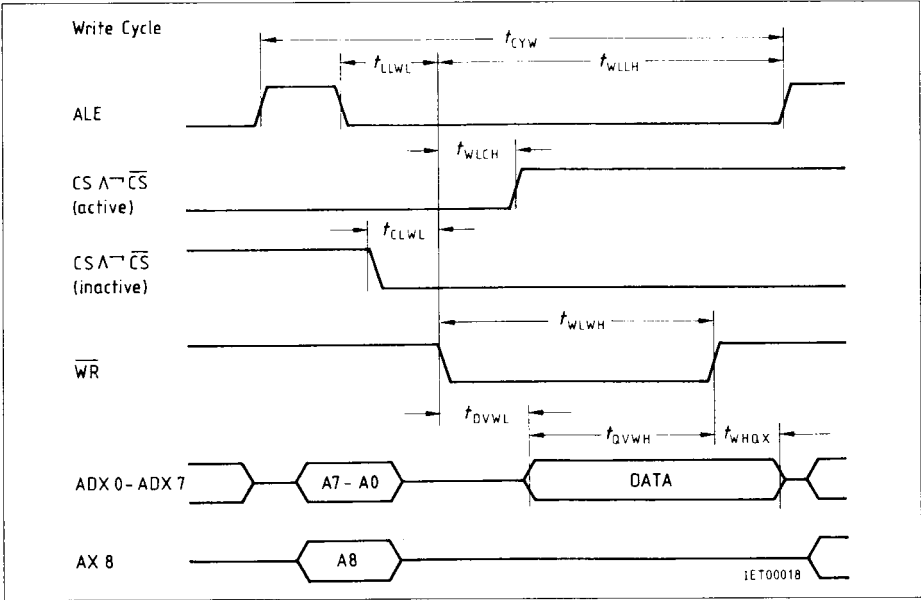
The AC characteristics apply throughout the operating range $T_A = 25\text{ }^\circ\text{C}$.

Parameter	Symbol	Limit Values		Unit
		min.	max.	
Set-up of chip select to RD ^{*)}	t_{CLRL}	0	—	ns
Set-up of chip select to WR ^{*)}	t_{CLWL}	0	—	ns
Active pulse length of timer outputs	t_{ACT}	48 t_{OSC}	48 t_{OSC}	ns
Oscillator period	t_{OSC}	83	—	ns
High time	t_{OSCH}	35	—	ns
Low time	t_{OSCL}	35	—	ns

*) For deselection



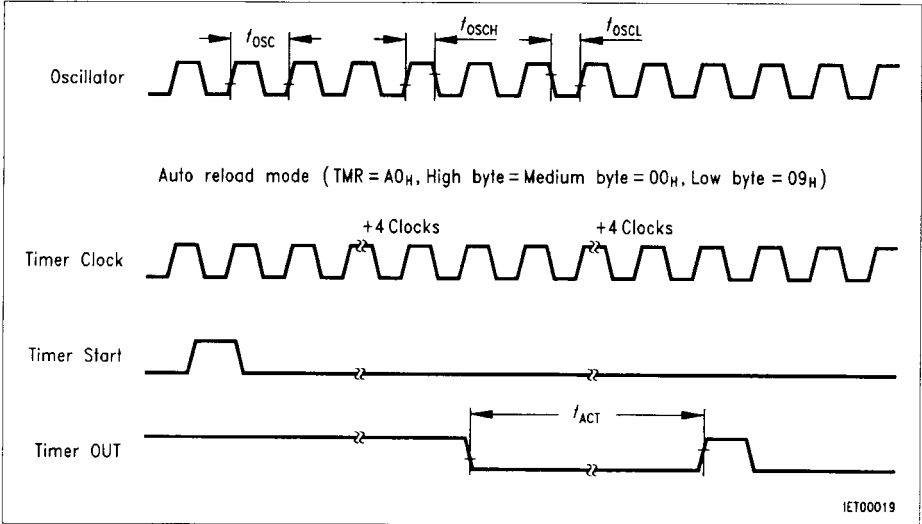
Pulse Diagram 1



Pulse Diagram 2

Note to Chip Select Timing:

The shown timing is not necessary, if the device is always activated or deactivated. This means either of CS or CS or both may be constant "high" or "low".



Pulse Diagram 3

Appendix

8051 Program for Timer Operation in Watchdog Mode

HBYTE	EQU	1E3 _H	; Address high byte reload register
TMR	EQU	1E0 _H	; Address timer-mode register
CR	EQU	1EC _H	; Address control register
REST1	EQU	055 _H	; 1st value to restart timer
REST2	EQU	0AA _H	; 2nd value to restart timer
WDOFF	EQU	040 _H	; Value to switch off watchdog mode

; Load reload register

```

MOV     DPTR, #HBYTE
CLR     A
MOVX    @DPTR,A
DEC     DPL
MOV     A, #0FFH
MOVX    @DPTR,A
DEC     DPL
MOVX    @DPTR,A

```

; Set watchdog mode and start timer

```

MOV     A, #0AFH
DEC     DPL
MOVX    @DPTR,A

```

; Reset timer

```

MOV     DPTR, #KR
MOV     A, #REST1
MOVX    @DPTR,A
MOV     A, #REST2
MOVX    @DPTR,A

```

; Switch off watchdog mode and halt timer

```

MOV     DPTR, #KR
MOV     A, #REST1
MOVX    @DPTR,A
MOV     A, #WDOFF
MOV     DPTR, #TMR
MOVX    @DPTR,A
MOV     A, #REST2
MOV     DPTR, #KR
MOVX    @DPTR,A

```

;

END