DALLAS SEMICONDUCTOR

DS80C320
High–Speed Micro

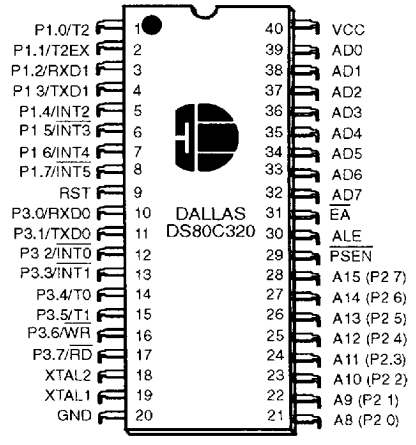
FEATURES

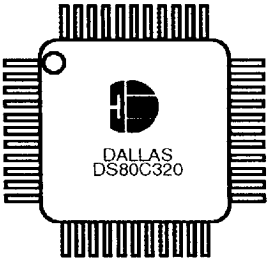
- 80C32–Compatible
  - Pin–compatible
  - Standard 8051 instruction set
  - Four 8–bit I/O ports
  - Three 16–bit timer/counters
  - 256 bytes scratchpad RAM
  - Multiplexed address/data bus
  - Addresses 64K bytes ROM and 64K bytes RAM
- ***New*** High–speed architecture
  - 4 clocks/machine cycle (8032=12)
  - Wasted cycles removed
  - Runs DC to 25 MHz clock rates
  - Single–cycle instruction in 160 ns
  - Uses less power for equivalent work
  - Dual data pointer
  - Optional variable length MOVX to access fast/slow RAM /peripherals
- ***New*** High integration controller includes:
  - Power–fail reset
  - Programmable Watchdog timer
  - Early–warning power–fail interrupt
- ***New*** Two full–duplex hardware serial ports
- ***New*** 13 total interrupt sources with six external
- Available in 40–pin DIP, 44–pin PLCC and QFP

DESCRIPTION

The DS80C320 is a fast 80C31/80C32–compatible microcontroller. Wasted clock and memory cycles have been removed using a redesigned processor core. As a result, every 8051 instruction is executed between 1.5 and 3 times faster than the original for the same crystal speed. Typical applications will see a speed improvement of 2.5 times using the same code and same crystal. The DS80C320 offers a maximum crystal rate of 25 MHz, resulting in apparent execution speeds of 62.5 MHz (approximately 2.5X).

PIN ASSIGNMENT





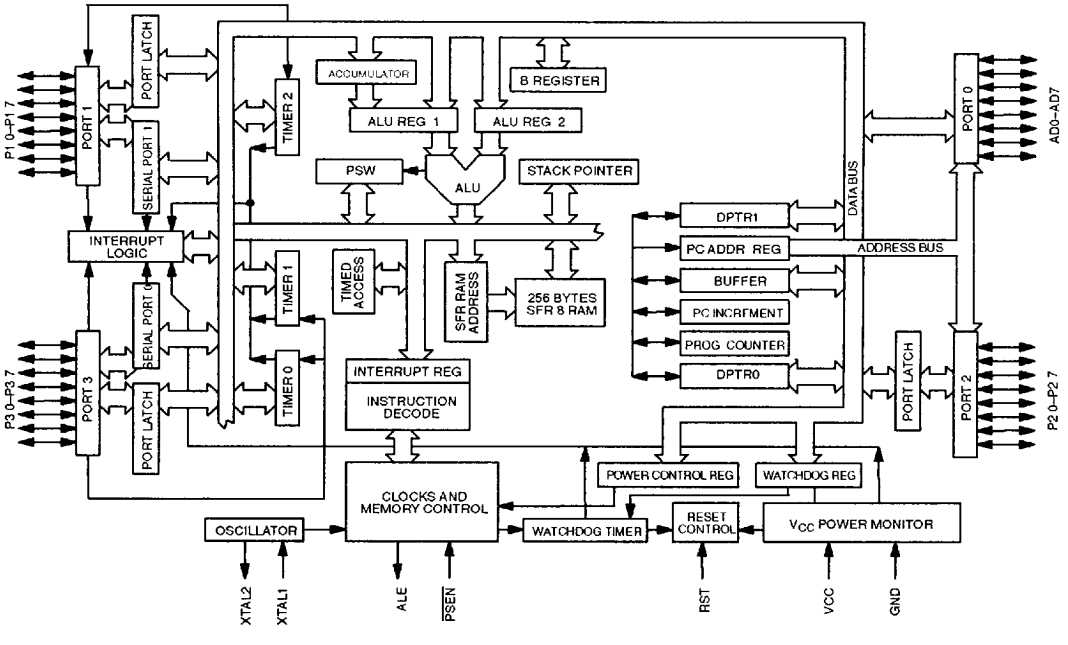
The DS80C320 is pin compatible with all three packages of the standard 80C32 and offers the same timer/counters, serial port, and I/O ports. In short, the DS80C320 is extremely familiar to 8051 users but provides the speed of a 16–bit processor.

The DS80C320 provides several extras in addition to greater speed. These include a second full hardware serial port, seven additional interrupts, programmable watchdog timer, power–fail interrupt and reset. The DS80C320 also provides dual data pointers (DPTRs) to speed block data memory moves. It can also adjust the speed of off–chip data memory access to between two and nine machine cycles for flexibility in selecting memory and peripherals.

ORDERING INFORMATION

PART NUMBER	PACKAGE	MAX CLOCK SPEED	TEMPERATURE RANGE
DS80C320–MCG	40–pin plastic DIP	25 MHz	0°C to +70°C
DS80C320–QCG	44–pin PLCC	25 MHz	0°C to +70°C
DS80C320–FCG	44–pin PQFP	25 MHz	0°C to +70°C
DS80C320–MNG	40–pin plastic DIP	25 MHz	–40°C to +85°C
DS80C320–QNG	44–pin PLCC	25 MHz	–40°C to +85°C
DS80C320–FNG	44–pin PQFP	25 MHz	–40°C to +85°C

DS80C320 BLOCK DIAGRAM Figure 1



PIN DESCRIPTION Table 1

DIP	PLCC	QFP	SIGNAL NAME	DESCRIPTION
40	44	38	V_{CC}	**V_{CC}** – +5V.
20	22, 23	16, 17	GND	**GND** – Digital circuit ground.
9	10	4	RST	**RST – Input**. The RST input pin contains a schmitt voltage input to recognize external active high Reset inputs. The pin also employs an internal pull–down resistor to allow for a combination of wired OR external Reset sources. An RC is not required for power–up, as the DS80C320 provides this function internally.
18 19	20 21	14 15	XTAL2 XTAL1	**XTAL1, XTAL2** – The crystal oscillator pins XTAL1 and XTAL2 provide support for parallel resonant, AT cut crystals. XTAL1 acts also as an input in the event that an external clock source is used in place of a crystal. XTAL2 serves as the output of the crystal amplifier.
29	32	26	$\overline{\text{PSEN}}$	**$\overline{\text{PSEN}}$ – Output**. The Program Store Enable output. This signal is commonly connected to external ROM memory as a chip enable. $\overline{\text{PSEN}}$ will provide an active low pulse width of 1.75 XTAL1 cycles with a period of four XTAL1 cycles. $\overline{\text{PSEN}}$ is driven high when data memory (RAM) is being accessed through the bus and during a reset condition.
30	33	27	ALE	**ALE – Output**. The Address Latch Enable output functions as a clock to latch the external address LSB from the multiplexed address/data bus. This signal is commonly connected to the latch enable of an external 373 family transparent latch. ALE has a pulse width of 1.5 XTAL1 cycles and a period of four XTAL1 cycles. ALE is forced high when the DS80C320 is in a Reset condition.
39 38 37 36 35 34 33 32	43 42 41 40 39 38 37 36	37 36 35 34 33 32 31 30	AD0 AD1 AD2 AD3 AD4 AD5 AD6 AD7	**AD0–7 (Port 0) – I/O.** Port 0 is the multiplexed address/data bus. During the time when ALE is high, the LSB of a memory address is presented. When ALE falls, the port transitions to a bidirectional data bus. This bus is used to read external ROM and read/write external RAM memory or peripherals. The Port 0 has no true port latch and can not be written directly by software. The reset condition of Port 0 is high. No pullup resistors are needed.
1–8	2–9	40–44 1–3	P1.0–P1.7	**Port 1 – I/O**. Port 1 functions as both a 8–bit bidirectional I/O port and a alternate functional interface for Timer 2 I/O, new External Interrupts, and new Serial Port 1. The reset condition of Port 1 is with all bits at a logic 1. In this state, a weak pull–up holds the port high. This condition also serves as an input mode, since any external circuit that writes to the port will overcome the weak pull–up. When software writes a 0 to any port pin, the DS80C320 will activate a strong pull–down that remains on until either a 1 is written or a reset occurs. Writing a 1 after the port has been at 0 will cause a strong transition driver to turn on, followed by a weaker sustaining pull–up. Once the momentary strong driver turns off, the port once again becomes the output high (and input) state. The alternate modes of Port 1 are outlined as follows. **Port** **Alternate** **Function**
1	2	40		**P1.0** T2 External I/O for Timer/Counter 2
2	3	41		**P1.1** T2EX Timer/Counter 2 Capture/Reload Trigger
3	4	42		**P1.2** RXD1 Serial Port 1 Input
4	5	43		**P1.3** TXD1 Serial Port 1 Output
5	6	44		**P1.4** $\overline{\text{INT2}}$ External Interrupt 2 (Positive Edge Detect)
6	7	1		**P1.5** $\overline{\text{INT3}}$ External Interrupt 3 (Negative Edge Detect)
7	8	2		**P1.6** $\overline{\text{INT4}}$ External Interrupt 4 (Positive Edge Detect)
8	9	3		**P1.7** $\overline{\text{INT5}}$ External Interrupt 5 (Negative Edge Detect)

DIP	PLCC	QFP	SIGNAL NAME	DESCRIPTION
21 22 23 24 25 26 27 28	24 25 26 27 28 29 30 31	18 19 20 21 22 23 24 25	A8 (P2.0) A9 (P2.1) A10 (P2.2) A11 (P2.3) A12 (P2.4) A13 (P2.5) A14 (P2.6) A15 (P2.7)	**A15–A8 (Port 2) – Output.** Port 2 serves as the MSB for external addressing. P2.7 is A15 and P2.0 is A8. The DS80C320 will automatically place the MSB of an address on P2 for external ROM and RAM access. Although Port 2 can be accessed like an ordinary I/O port, the value stored on the Port 2 latch will never be seen on the pins (due to memory access). Therefore writing to Port 2, in software is only useful for the instructions MOVX A, @Ri or MOVX @Ri, A. These instructions use the Port 2 internal latch to supply the external address MSB. In this case, the Port 2 latch value will be supplied as the address information.
10–17	11, 13–19	5, 7–13	P3.0–P3.7	**Port 3 – I/O.** Port 3 functions as both a 8–bit bidirectional I/O port and an alternate functional interface for External Interrupts, Serial Port 0, Timer 0 & 1 Inputs, $\overline{RD}$ and $\overline{WR}$ strobes. The reset condition of Port 3 is with all bits at a logic 1. In this state, a weak pull–up holds the port high. This condition also serves as an input mode, since any external circuit that writes to the port will overcome the weak pull–up. When software writes a 0 to any port pin, the DS80C320 will activate a strong pull–down that remains on until either a 1 is written or a reset occurs. Writing a 1 after the port has been at 0 will cause a strong transition driver to turn on, followed by a weaker sustaining pull–up. Once the momentary strong driver turns off, the port once again becomes both the output high and input state. The alternate modes of Port 3 are outlined below. **Port — Alternate Mode** **P3.0** RXD0 — Serial Port 0 Input **P3.1** $\overline{TXD0}$ — Serial Port 0 Output **P3.2** $\overline{INT0}$ — External Interrupt 0 **P3.3** $\overline{INT1}$ — External Interrupt 1 **P3.4** T0 — Timer 0 External Input **P3.5** T1 — Timer 1 External Input **P3.6** $\overline{WR}$ — External Data Memory Write Strobe **P3.7** $\overline{RD}$ — External Data Memory Read Strobe
31	35	29	$\overline{EA}$	**$\overline{EA}$ – Input.** Connect to ground for compatibility with other 8051 family parts and future versions.
–	12 34	6 28	NC	**NC – Reserved.** These pins should not be connected. They are reserved for use with future devices in this family.
–	1	39		**NC – Reserved.** These pins are reserved for additional ground pins on future products.

80C32 COMPATIBILITY

The DS80C320 is a CMOS 80C32 compatible microcontroller designed for high performance. In most cases the DS80C320 can drop into an existing 80C32 design to significantly improve the operation. Every effort has been made to keep the device familiar to 8032 users, yet it has many new features. In general, software written for existing 80C32 based systems will work on the DS80C320 The exception is critical timing since the High Speed Micro performs its instructions much faster than the original. It may be necessary to use memories with faster access times if the same crystal frequency is used.

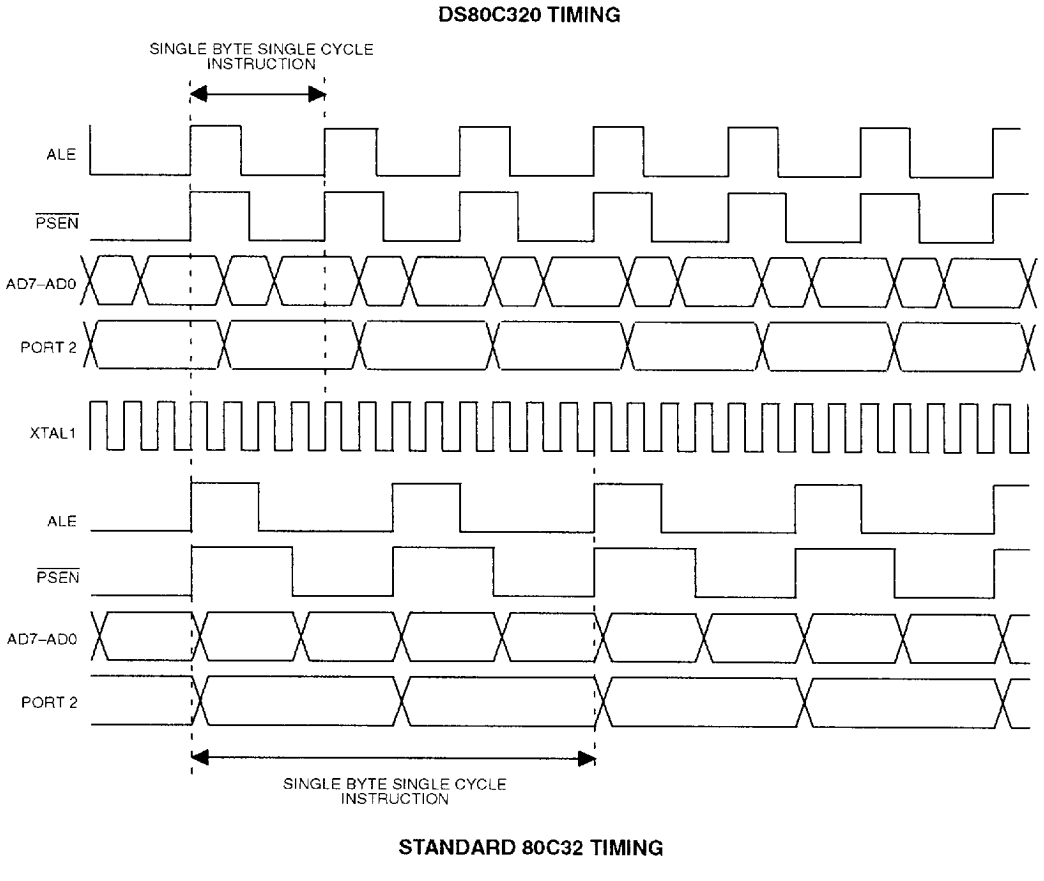
The DS80C32 runs the standard 8051 instruction set and is pin compatible with an 80C32 in any of three standard packages. The DS80C320 also provides the same timer/counter resources, full–duplex serial port, 256 bytes of scratchpad RAM and I/O ports as the standard 80C32. Timers will default to a 12 clock per cycle operation to keep timing compatible with original 8051 systems. However, they can be programmed to run at the new 4 clocks per cycle if desired.

New hardware features are accessed using Special Function Registers that do not overlap with standard 80C32 locations. A summary of these SFRs is provided below.

The DS80C320 addresses memory in an identical fashion to the standard 80C32 Electrical timing will appear different due to the high speed nature of the product. However, the signals are essentially the same Detailed timing diagrams are provided below in the electrical specifications.

This data sheet assumes the user is familiar with the basic features of the standard 80C32. In addition to these standard features, the DS80C320 includes many new functions This data sheet provides only a summary and overview. Detailed descriptions are available in the High Speed Micro User's Guide.

COMPARATIVE TIMING OF THE DS80C320 AND 80C32 Figure 2



HIGH–SPEED OPERATION

The DS80C320 is built around a high speed 80C32 compatible core. Higher speed comes not just from increasing the clock frequency, but from a newer, more efficient design.

In this updated core, dummy memory cycles have been eliminated. In a conventional 80C32, machine cycles are generated by dividing the clock frequency by 12. In the DS80C320, the same machine cycle is performed in 4 clocks. Thus the fastest instruction, 1 machine cycle, is executed three times faster for the same crystal frequency. Note that these are identical instructions. A comparison of the timing differences is shown in Figure 2. The majority of instructions on the DS80C320 will see the full 3 to 1 speed improvement. Some instructions will get between 1.5 and 2.4 X improvement. Note that all instructions are faster than the original 80C51. Table 2 below shows a summary of the instruction set including the speed.

The numerical average of all opcodes is approximately a 2.5 to 1 speed improvement. Individual programs will be affected differently, depending on the actual instructions used. Speed sensitive applications would make the most use of instructions that are three times faster. However, the sheer number of 3 to 1 improved opcodes makes dramatic speed improvements likely for any code. When these architecture improvements are combined with 0.8 μ CMOS, the result is a single cycle instruction execution in 160 ns. The Dual Data Pointer feature also allows the user to eliminate wasted instructions when moving blocks of memory.

INSTRUCTION SET SUMMARY

All instructions in the DS80C320 perform the same functions as their 80C32 counterparts. Their affect on bits, flags, and other status functions is identical. However, the timing of each instruction is different. This applies both in absolute and relative number of clocks.

For absolute timing of real–time events, the timing of software loops will need to be calculated using the table below. However, counter/timers default to run at the older 12 clocks per increment. Therefore, while software runs at higher speed, timer–based events need no modification to operate as before. Timers can be set to run at 4 clocks per increment cycle to take advantage of higher speed operation.

The relative time of two instructions might be different in the new architecture than it was previously. For example, in the original architecture, the "MOVX A, @DPTR" instruction and the "MOV direct, direct" instruction used two machine cycles or 24 oscillator cycles. Therefore, they required the same amount of time. In the DS80C320, the MOVX instruction can be done in two machine cycles or eight oscillator cycles but the "MOV direct, direct" uses three machine cycles or 12 oscillator cycles. While both are faster than their original counterparts, they now have different execution times from each other. This is because in most cases, the DS80C320 uses one cycle for each byte. The timing of each instruction should be examined for familiarity with the changes. Note that a machine cycle now requires just four clocks, and provides one ALE pulse per cycle. Many instructions require only one cycle, but some require five. In the original architecture, all were one or two cycles except for MUL and DIV.

INSTRUCTION SET SUMMARY Table 2

Legends:

A	–	Accumulator
Rn	–	Register R7–R0
direct	–	Internal Register address
@Ri	–	Internal Register pointed–to by R0 or R1 (except MOVX)
rel	–	2's complement offset byte
bit	–	direct bit–address
#data	–	8–bit constant
#data 16	–	16–bit constant
addr 16	–	16–bit destination address
addr 11	–	11–bit destination address

INSTRUCTION	BYTE	OSCILLATOR CYCLES	INSTRUCTION	BYTE	OSCILLATOR CYCLES
Arithmetic Instructions:					
ADD A, Rn	1	4	INC A	1	4
ADD A, direct	2	8	INC Rn	1	4
ADD A, @Ri	1	4	INC direct	2	8
ADD A, #data	2	8	INC @Ri	1	4
ADDC A, Rn	1	4	INC DPTR	1	12
ADDC A, direct	2	8	DEC A	1	4
ADDC A, @Ri	1	4	DEC Rn	1	4
ADDC A, #data	2	8	DEC direct	2	8
SUBB A, Rn	1	4	DEC @Ri	1	4
SUBB A, direct	2	8	MUL AB	1	20
SUBB A, @Ri	1	4	DIV AB	1	20
SUBB A, #data	2	8	DA A	1	4
Logical Instructions:					
ANL A, Rn	1	4	XRL A, Rn	1	4
ANL A, direct	2	8	XRL A, direct	2	8
ANL A, @Ri	1	4	XRL A, @Ri	1	4
ANL A, #data	2	8	XRL A, #data	2	8
ANL direct, A	2	8	XRL direct, A	2	8
ANL direct, #data	3	12	XRL direct, #data	3	12
ORL A, Rn	1	4	CLR A	1	4
ORL A, direct	2	8	CPL A	1	4
ORL A, @Ri	1	4	RL A	1	4
ORL A, #data	2	8	RLC A	1	4
ORL direct, A	2	8	RR A	1	4
ORL direct, #data	3	12	RRC A	1	4
			SWAP A	1	4
Data Transfer Instructions:					
MOV A, Rn	1	4	MOVC A, @A+DPTR	1	12
MOV A, direct	2	8	MOVC A, @A+PC	1	12
MOV A, @Ri	1	4	MOVX A, @Ri	1	8–36 *
MOV A, #data	2	8	MOVX A, @DPTR	1	8–36 *
MOV Rn, A	1	4	MOVX @Ri, A	1	8–36 *
MOV Rn, direct	2	8	MOVX @DPTR, A	1	8–36 *
MOV Rn, #data	2	8	PUSH direct	2	8
MOV direct, A	2	8	POP direct	2	8
MOV direct, Rn	2	8	XCH A, Rn	1	4
MOV direct1, direct2	3	12	XCH A, direct	2	8
MOV direct, @Ri	2	8	XCH A, @Ri	1	4
MOV direct, #data	3	12	XCHD A, @Ri	1	4
MOV @Ri, A	1	4			
MOV @Ri, direct	2	8			
MOV @Ri, #data	2	8			
MOV DPTR, #data 16	3	12			

*User Selectable

Bit Manipulation Instructions:					
CLR C	1	4	ANL C, bit	2	8
CLR bit	2	8	ANL C, $\overline{bit}$	2	8
SETB C	1	4	ORL C, bit	2	8
SETB bit	2	8	ORL C, $\overline{bit}$	2	8
CPL C	1	4	MOV C, bit	2	8
CPL bit	2	8	MOV bit, C	2	8
Program Branching Instructions:					
ACALL addr 11	2	12	CJNE A, direct, rel	3	16
LCALL addr 16	3	16	CJNE A, #data, rel	3	16
RET	1	16	CJNE Rn, #data, rel	3	16
RETI	1	16	CJNE Ri, #data, rel	3	16
AJMP addr 11	2	12	NOP	1	4
LJMP addr 16	3	16	JC rel	2	12
SJMP rel	2	12	JNC rel	2	12
JMP @A+DPTR	1	12	JB bit, rel	3	16
JZ rel	2	12	JNB bit, rel	3	16
JNZ rel	2	12	JBC bit, rel	3	16
DJNZ Rn, rel	2	12			
DJNZ direct, rel	3	16			

The Table above shows the speed for each class of instruction. Note that many of the instructions have multiple opcodes. There are 255 opcodes for 111 instructions. Of the 255 opcodes, 159 are three times faster than the original 80C32. While a system that emphasizes those instructions will see the most improvement, the large total number that receive a 3 to 1 improvement assure a dramatic speed increase for any system. The speed improvement summary is provided below.

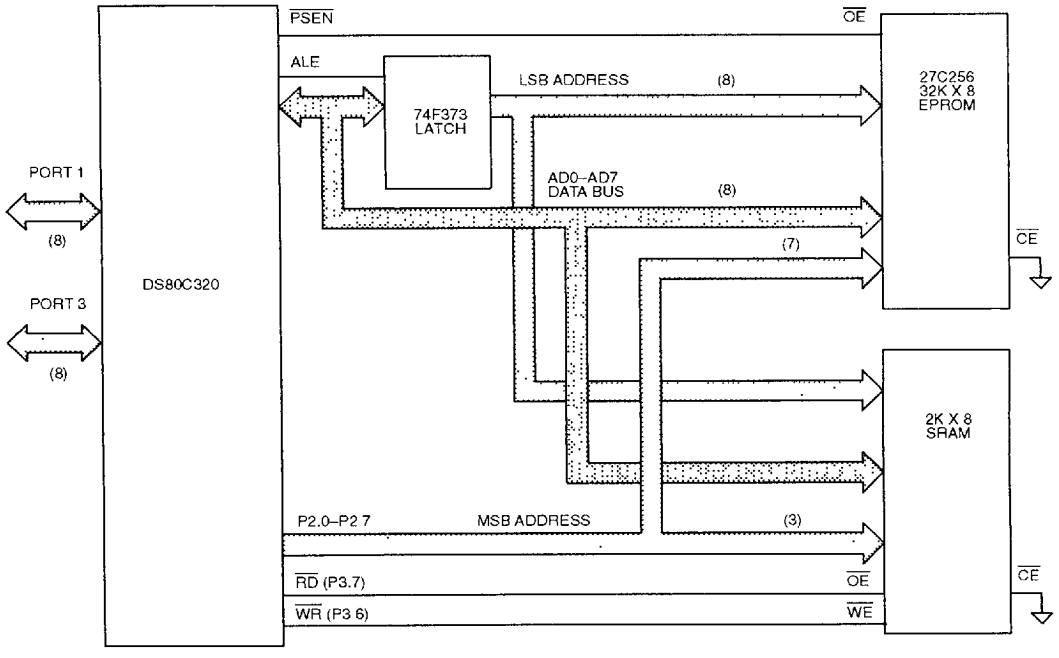
SPEED ADVANTAGE SUMMARY

#Opcodes	Speed Improvement
159	3.0 x
51	1.5 x
43	2.0 x
2	2.4 x
255	Average: 2.5

MEMORY ACCESS

The DS80C320 contains no on–chip ROM, and 256 bytes of scratchpad RAM. Off–chip memory is accessed using the multiplexed address/data bus on P0 and the MSB address on P2. A typical memory connection is shown in Figure 3. Timing diagrams are provided in the Electrical Specifications. Program memory (ROM) is accessed at a fixed rate determined by the crystal frequency and the actual instructions. As mentioned above, an instruction cycle requires four clocks. Data memory (RAM) is accessed according to a variable speed MOVX instruction as described below.

TYPICAL MEMORY CONNECTION Figure 3



STRETCH MEMORY CYCLE

The DS80C320 allows the application software to adjust the speed of data memory access. The micro is capable of performing the MOVX in as little as two instruction cycles. However, this value can be stretched as needed so that both fast memory and slow memory or peripherals can be accessed with no glue logic. Even in high–speed systems, it may not be necessary or desirable to perform data memory access at full speed. In addition, there are a variety of memory mapped peripherals such as LCD displays or UARTs that are not fast.

The Stretch MOVX is controlled by the Clock Control Register at SFR location 8Eh as described below. This allows the user to select a stretch value between zero and seven. A Stretch of zero will result in a two machine cycle MOVX. A Stretch of seven will result in a MOVX of nine machine cycles. The value can be changed dynamically under software control depending on the type of memory or peripheral to be accessed.

On reset, the Stretch value will default to a one resulting in a three cycle MOVX. Therefore, RAM access will not be performed at full speed. This is a convenience to existing designs that may not have fast RAM in place. When maximum speed is desired, the software should select a Stretch value of zero. When using very slow RAM or peripherals, a larger stretch value can be selected. Note that this affects data memory only and the only way to slow program memory (ROM) access is to use a slower crystal.

Using a Stretch value between one and seven causes the microcontroller to stretch the read/write strobe and all related timing. This results in a wider read/write strobe allowing more time for memory/peripherals to respond. The timing of the variable speed MOVX is shown in the Electrical Specifications. Note that full speed access is not the reset default case. Table 3 below shows the resulting strobe widths for each Stretch value. The memory stretch is implemented using the Clock Control Special Function Register at SFR location 8Eh. The stretch value is selected using bits CKCON.2–0. In the table, these bits are referred to as M2 through M0. The first stretch (default) allows the use of common 120 ns or 150 ns RAMs without dramatically lengthening the memory access.

DATA MEMORY CYCLE STRETCH VALUES Table 3

CKCON.2–0 MD2	MD1	MD0	MEMORY CYCLES	$\overline{RD}$ or $\overline{WR}$ STROBE WIDTH IN CLOCKS	STROBE WIDTH TIME @ 25 MHz
0	0	0	2	2	80 ns
0	0	1	3 (default)	4	160 ns
0	1	0	4	8	320 ns
0	1	1	5	12	480 ns
1	0	0	6	16	640 ns
1	0	1	7	20	800 ns
1	1	0	8	24	960 ns
1	1	1	9	28	1120 ns

DUAL DATA POINTER

Data memory block moves can be accelerated using the DS80C320 Dual Data Pointer (DPTR). The standard 8032 DPTR is a 16–bit value that is used to address off–chip data RAM or peripherals. In the DS80C320, the standard data pointer is called DPTR 0 and is located at SFR addresses 82h and 83h. These are the standard locations. No modification of standard code is needed to use DPTR0. The new DPTR is located at SFR 84h and 85h and is called DPTR1. The DPTR Select bit (DPS) chooses the active pointer and is located at the LSB of the SFR location 86h. No other bits in register 86h have any effect and are set to 0. The user switches between data pointers by toggling the LSB of register 86h. The increment (INC) instruction is the fastest way to accomplish this. All DPTR–related instructions use the currently selected DPTR for any activity. Therefore only one instruction is required to switch from a source to a destination address. Using the Dual–Data Pointer saves code from needing to save source and destination addresses when doing a block move. Once loaded, the software simply switches between DPTR0 and 1. The relevant register locations are as follows.

DPL0	82h	Low byte original DPTR
DPH0	83h	High byte original DPTR
DPL1	84h	Low byte new DPTR
DPH1	85h	High byte new DPTR
DPS	86h	DPTR Select (LSB)

Sample code listed below illustrates the saving from using the dual DPTR. The example program was original code written for an 8051 and requires a total of 1869 machine cycles on the DS80C320. This takes 299 µs to execute at 25 MHz. The new code using the Dual DPTR requires only 1097 machine cycles taking 175.5 µs. The Dual DPTR saves 772 machine cycles or 123.5 µs for a 64 byte block move. Since each pass through the loop saves 12 machine cycles when compared to the single DPTR approach, larger blocks gain more efficiency using this feature.

64 BYTE BLOCK MOVE WITHOUT DUAL DATA POINTER

```
; SH and SL are high and low byte source address.
; DH and DL are high and low byte of destination address.
                                                              # CYCLES

MOV     R5, #64d        ; NUMBER OF BYTES TO MOVE                 2
MOV     DPTR, #SHSL     ; LOAD SOURCE ADDRESS                     3
MOV     R1, #SL         ; SAVE LOW BYTE OF SOURCE                 2
MOV     R2, #SH         ; SAVE HIGH BYTE OF SOURCE                2
MOV     R3, #DL         ; SAVE LOW BYTE OF DESTINATION            2
MOV     R4, #DH         ; SAVE HIGH BYTE OF DESTINATION           2

MOVE:
; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64

MOVX    A, @DPTR        ; READ SOURCE DATA BYTE                   2
MOV     R1, DPL         ; SAVE NEW SOURCE POINTER                 2
MOV     R2, DPH         ;                                         2
```

```
MOV     DPL, R3         ; LOAD NEW DESTINATION                          2
MOV     DPH, R4         ;                                               2
MOVX    @DPTR, A        ; WRITE DATA TO DESTINATION                     2
INC     DPTR            ; NEXT DESTINATION ADDRESS                      3
MOV     R3, DPL         ; SAVE NEW DESTINATION POINTER                  2
MOV     R4, DPH         ;                                               2
MOV     DPL, R1         ; GET NEW SOURCE POINTER                        2
MOV     DPH, R2         ;                                               2
INC     DPTR            ; NEXT SOURCE ADDRESS                           3
DJNZ    R5, MOVE        ; FINISHED WITH TABLE?                          3
```

64 BYTE BLOCK MOVE WITH DUAL DATA POINTER

```
; SH and SL are high and low byte source address.
; DH and DL are high and low byte of destination address.
; DPS is the data pointer select. Reset condition is DPS=0, DPTR0 is selected.
                                                                  # CYCLES

EQU     DPS, #86h       ; TELL ASSEMBLER ABOUT DPS

MOV     R5, #64         ; NUMBER OF BYTES TO MOVE                       2
MOV     DPTR, #DHDL     ; LOAD DESTINATION ADDRESS                      3
INC     DPS             ; CHANGE ACTIVE DPTR                            2
MOV     DPTR, #SHSL     ; LOAD SOURCE ADDRESS                           2

MOVE:
; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64

MOVX    A, @DPTR        ; READ SOURCE DATA BYTE                         2
INC     DPS             ; CHANGE DPTR TO DESTINATION                    2
MOVX    @DPTR, A        ; WRITE DATA TO DESTINATION                     2
INC     DPTR            ; NEXT DESTINATION ADDRESS                      3
INC     DPS             ; CHANGE DATA POINTER TO SOURCE                 2
INC     DPTR            ; NEXT SOURCE ADDRESS                           3
DJNZ    R5, MOVE        ; FINISHED WITH TABLE?                          3
```

PERIPHERAL OVERVIEW

Peripherals in the DS80C320 are accessed using Special Function Registers (SFRs). The DS80C320 provides several of the most commonly needed peripheral functions in microcomputer–based systems. These functions are new to the 80C32 family and include a second serial port, Power–fail Reset, Power–fail Interrupt, and a programmable Watchdog Timer. These are described below, and more details are available in the High Speed Micro User's Guide.

SERIAL PORTS

The DS80C320 provides a serial port (UART) that is identical to the 80C32. Many applications require serial communication with multiple devices. Therefore the DS80C320 provides a second hardware serial port that is a full duplicate of the standard one. It optionally uses pins P1.2 (RXD1) and P1.3 (TXD1). This port has duplicate control functions included in new SFR locations. The second serial port operates in a comparable manner with the first. Both can operate simultaneously but can be at different baud rates.

The second serial port has similar control registers (SCON1 at C0h, SBUF1 at C1h) to the original. One difference is that for timer based baud rates, the original serial port can use Timer 1 or Timer 2 to generate baud rates. This is selected via SFR bits. The new serial port can only use Timer 1.

TIMER RATE CONTROL

One important difference exists between the DS80C320 and 80C32 regarding timers. The original 80C32 used a 12 clock per cycle scheme for timers and consequently for some serial baud rates (depending on the mode). The DS80C320 architecture normally runs using 4 clocks per cycle. However, in the area of timers, the DS80C320 will default to a 12 clock per cycle

scheme on a reset. This allows existing code with real–time dependencies such as baud rates to operate properly. If an application needs higher speed timers or serial baud rates, the timers can be set to run at the 4 clock rate.

The Clock Control register (CKCON – 8Eh) determines these timer speeds. When the relevant CKCON bit is a logic 1, the DS80C320 uses 4 clocks per cycle to generate timer speeds. When the control bit is set to a 0, the DS80C320 uses 12 clocks for Timer speeds. The reset condition is a 0. CKCON.5 selects the speed of Timer 2. CKCON.4 selects Timer 1 and CKCON.3 selects Timer 0. Note that unless a user desires very fast timing, it is unnecessary to alter these bits. Note that the timer controls are independent.

POWER FAIL RESET

The DS80C320 incorporates a precision band–gap voltage reference to determine when V_{CC} is out–of–tolerance. While powering up, internal circuits will hold the DS80C320 in a reset state until V_{CC} rises above the V_{RST} reset threshold. Once V_{CC} is above this level, the oscillator will begin running. An internal reset circuit will then count 65536 clocks to allow time for power and the oscillator to stabilize. The uC will then exit the reset condition. No external components are needed to generate a power on reset. During power down or during a severe power glitch, as V_{CC} falls below V_{RST}, the uC will also generate its own reset. It will hold the reset condition as long as power remains below the threshold. This reset will occur automatically, needing no action from the user or from the software. Refer to the Electrical Specifications for the exact value of V_{RST}.

POWER FAIL INTERRUPT

The same reference that generates a precision reset threshold can also generate an optional early warning Power–fail Interrupt (PFI). When enabled by the application software, this interrupt always has the highest priority. On detecting that the V_{CC} has dropped below V_{PFW} and that the PFI is enabled, the processor will vector to ROM address 0033h. The PFI enable is located in the Watchdog Control SFR (WDCON – D8h). Setting WDCON.5 to a logic one will enable the PFI. The application software can also read a flag at WDCON.4. This bit is set when a PFI condition has occurred. The flag is independent of the interrupt enable and software must manually clear it.

WATCHDOG TIMER

For applications that can not afford to run out–of–control, the DS80C320 incorporates a programmable Watchdog Timer circuit. It resets the uC if software fails to reset the Watchdog before the selected time interval has elapsed. The user selects one of four time–out values. After enabling the Watchdog, software must reset the timer prior to expiration of the interval, or the CPU will be reset. Both the Watchdog Enable and the Watchdog Reset bits are protected by a "Timed Access" circuit. This prevents accidentally clearing the Watchdog. Time–out values are precise since they are related to the crystal frequency as shown below in Table 4. For reference, the time periods at 25 MHz are also shown.

The DS80C320 Watchdog also provides a useful option for systems that may not require a reset. If enabled, then 512 clocks before giving a reset, the Watchdog will give an interrupt. The interrupt can also serve as a convenient time–base generator, or be used to wake–up the processor from IDLE mode. The Watchdog function is controlled in the Clock Control (CKCON – 8Eh), Watchdog Control (WDCON – D8h), and Extended Interrupt Enable (EIE – E8h) SFRs. CKCON.7 and CKCON.6 are called WD1 and WD0 respectively and are used to select the Watchdog time–out period as shown in Table 4.

WATCHDOG TIME–OUT VALUES Table 4

WD1	WD0	INTERRUPT TIME–OUT	TIME (@25 MHz)	RESET TIME–OUT	TIME (@25 MHz)
0	0	2^{17} clocks	5.243 ms	2^{17} + 512 clocks	5.263 ms
0	1	2^{20} clocks	41.94 ms	2^{20} + 512 clocks	41.96 ms
1	0	2^{23} clocks	335.54 ms	2^{23} + 512 clocks	335.56 ms
1	1	2^{26} clocks	2684.35 ms	2^{26} + 512 clocks	2684.38 ms

As shown above, the Watchdog Timer uses the crystal frequency as a time base. A user selects one of four counter values to determine the time–out. These clock counter lengths are 2^{17}= 131,072 clocks; 2^{20} = 1,048,576; 2^{23} = 8,388,608 clocks; or 2^{26} = 67,108,864 clocks. The times shown in Table 4 above are with a 25 MHz crystal frequency. Note that once the counter chain has reached a conclusion, the optional interrupt is generated. Regardless of whether the user enables this interrupt, there are then 512 clocks left until a reset occurs. There are five control bits in special function registers that affect the Watchdog Timer and two status flags that report to the user.

WDIF (WDCON.3) is the interrupt flag that is set when there are 512 clocks remaining until a reset occurs. WTRF (WDCON.2) is the flag that is set when a Watchdog reset has occurred. This allows the application software to determine the source of a reset.

EWT (WDCON.1) is the enable for the Watchdog Timer. Software sets this bit to enable the timer. The bit is protected by Timed Access discussed below. RWT (WDCON.0) is the bit that software uses to restart the Watchdog Timer. Setting this bit restarts the timer for another full interval. Application software must set this bit prior to the time–out. As mentioned previously, WD1 and 0 (CKCON .7 and 6) select the time–out. Finally, the Watchdog Interrupt is enabled using EWDI (EIE.4). The Special Function Register map is shown below.

INTERRUPTS

The DS80C320 provides 13 sources of interrupt with three priority levels. The Power–fail Interrupt (PFI), if enabled, always has the highest priority. There are two remaining user selectable priorities : high and low. If two interrupts that have the same priority occur simultaneously, the natural precedence given below determines which is a acted upon. Except for the PFI, all interrupts that are new to the 8051 family have a lower natural priority than the originals.

INTERRUPT PRIORITY Table 5

NAME	DESCRIPTION	VECTOR	NATURAL PRIORITY	OLD/NEW
PFI	Power Fail Interrupt	33h	1	NEW
$\overline{\text{INT0}}$	External Interrupt 0	03h	2	OLD
TF0	Timer 0	0Bh	3	OLD
$\overline{\text{INT1}}$	External Interrupt 1	13h	4	OLD
TF1	Timer 1	1Bh	5	OLD
SCON0	TI0 or RI0 from serial port 0	23h	6	OLD
TF2	Timer 2	2Bh	7	OLD
SCON1	TI1 or RI1 from serial port 1	3Bh	8	NEW
INT2	External Interrupt 2	43h	9	NEW
$\overline{\text{INT3}}$	External Interrupt 3	4Bh	10	NEW
INT4	External Interrupt 4	53h	11	NEW
$\overline{\text{INT5}}$	External Interrupt 5	5Bh	12	NEW
WDTI	Watchdog Time–out Interrupt	63h	13	NEW

POWER MANAGEMENT

The DS80C320 provides the standard IDLE and power down (STOP) that are available on the standard 80C32. However the DS80C320 has enhancements that make these modes more useful, and allow more power saving.

The IDLE mode is invoked by setting the LSB of the Power Control register (PCON – 87h). IDLE will leave internal clocks, serial port and timer running. No memory access will be performed so power is dramatically reduced. Since clocks are running, the IDLE power consumption is related to crystal frequency. It should be approximately 1/2 of the operational power. The CPU can exit the IDLE state with any interrupt or a reset.

The power–down or STOP mode is invoked by setting the PCON.1 bit. STOP mode is a lower power state than IDLE since it turns off all internal clocking. The I_{CC} of a standard STOP mode is approximately 1 µA but is specified in the Electrical Specifications. The CPU will exit stop mode from an external interrupt or a reset condi-

tion. Note that internally generated interrupts (timer, serial port, watchdog) are not useful since they require clocking activity.

IDLE MODE ENHANCEMENTS

A simple enhancement to IDLE mode makes it substantially more useful. The innovation involves not the IDLE mode itself, but the watchdog timer. As mentioned above, the Watchdog Timer provides an optional interrupt capability. This interrupt can provide a periodic interval timer to bring the DS80C320 out of IDLE mode. This can be useful even if the Watchdog is not normally used. By enabling the Watchdog Timer and its interrupt prior to invoking IDLE, a user can periodically come out of IDLE perform an operation, then return to IDLE until the next operation. This will lower the overall power consumption. When using the Watchdog Interrupt to cancel the IDLE state, make sure to restart the Watchdog Timer or it will cause a reset.

STOP MODE ENHANCEMENTS

The DS80C320 provides two enhancements to the STOP mode. As documented above, the DS80C320 provides a band-gap reference to determine Power-fail Interrupt and Reset thresholds. The default state is that the band-gap reference is off when STOP mode is invoked. This allows the extremely low power state mentioned above. A user can optionally choose to have the band-gap enabled during STOP mode. This means that PFI and power-fail reset will be activated and are valid means for leaving STOP mode.

In STOP mode with the band-gap on, I_{CC} will be approximately 50 µA compared with 1 µA with the band-gap off. If a user does not require a Power-fail Reset or Interrupt while in STOP mode, the band-gap can remain turned off. Note that only the most power sensitive applications should turn off the band-gap, as this results in an uncontrolled power down condition.

The control of the band-gap reference is located in the Extended Interrupt Flag register (EXIF – 91h). Setting BGS (EXIF.0) to a one will leave the band-gap reference enabled during STOP mode. The default or reset condition is with the bit at a logic 0. This results in the band-gap being turned off during STOP mode. Note that this bit has no control of the reference during full power or IDLE modes.

The second feature allows an additional power saving option. This is the ability to start instantly when exiting STOP mode. It is accomplished using an internal ring oscillator that can be used when exiting STOP mode in response to an interrupt. The benefit of the ring oscillator is as follows.

Using STOP mode turns off the crystal oscillator and all internal clocks to save power. This requires that the oscillator be restarted when exiting STOP mode. Actual start-up time is crystal dependent, but is normally at least 4 ms. A common recommendation is 10 ms. In an application that will wake-up, perform a short operation, then return to sleep, the crystal start-up can be longer than the real transaction. However, the ring oscillator will start instantly. The user can perform a simple operation and return to sleep before the crystal has even stabilized. If the ring is used to start and the processor remains running, hardware will automatically switch to the crystal once a power-on reset interval (65536 clocks) has expired. This value is used to guarantee stability even though power is not being cycled.

If the user returns to STOP mode prior to switching of crystal, then all clocks will be turned off again. The ring oscillator runs at approximately 4 MHz but will not be a precision value. No real-time precision operations (including serial communication) should be conducted during this ring period. Figure 7 shows how the operation would compare when using the ring, and when starting up normally. The default state is to come out of STOP mode without using the ring oscillator.

This function is controlled using the RGSL – Ring Select bit at EXIF.1 (EXIF – 91h). When EXIF.1 is set, the ring oscillator will be used to come out of STOP mode quickly. As mentioned above, the processor will automatically switch from the ring (if enabled) to the crystal after a delay of 65536 crystal clocks. For a 3.57 MHz crystal, this is approximately 18 ms. The processor sets a flag called RGMD – Ring Mode to tell software that the ring is being used. This bit at EXIF.2 will be a logic 1 when the ring is in use. No serial communication or precision timing should be attempted while this bit is set, since the operating frequency is not precise.

RING OSCILLATOR START–UP Figure 4

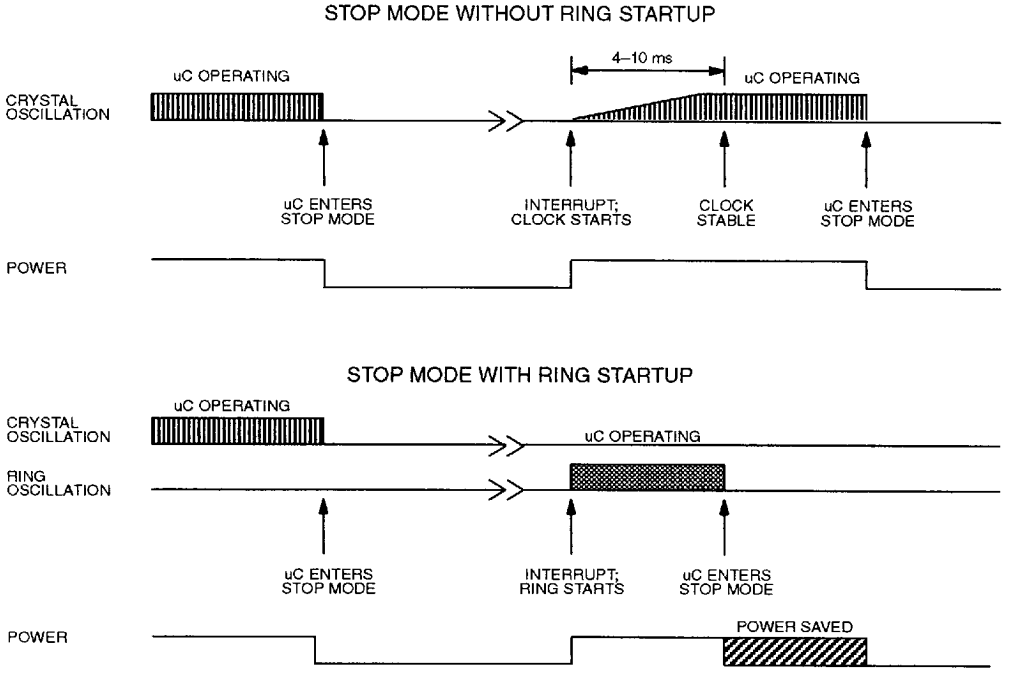


Diagram assumes that the operation following STOP requires less than 18 ms complete.

TIMED ACCESS PROTECTION

Selected SFR bits are critical to operation, making it desirable to protect against an accidental write operation. The Timed Access procedure prevents an errant cpu from accidentally altering a bit that would cause difficulty. The Timed Access procedure requires that the write of a protected bit be preceded by the following instructions :

```
MOV     0C7h, #0AAh
MOV     0C7h, #55h
```

By writing an AAh followed by a 55h to the Timed Access register (location C7h), the hardware opens a two cycle window that allows software to modify one of the protected bits. If the instruction that seeks to modify the protected bit is not immediately proceeded by these instructions, the write will not take effect. The protected bits are:

EXIF.0	BGS Band–gap Select
WDCON.6	POR Power–on Reset flag
WDCON.1	EWT Enable Watchdog
WDCON.0	RWT Reset Watchdog
WDCON.3	WDIF Watchdog Interrupt Flag

SPECIAL FUNCTION REGISTERS

Most special features of the DS80C320 or 80C32 are controlled by bits in special function registers (SFRs). This allows the DS80C320 to add many features but use the same instruction set. When writing software to use a new feature, the SFR must be defined to an assembler or compiler using an equate statement. This is the only change needed to access the new function. The DS80C320 duplicates the SFRs that are contained in the standard 80C32. Table 6 shows the register addresses and bit locations. Many are standard 80C32 registers. The High Speed Micro User's Guide describes all SFRs.

SPECIAL FUNCTION REGISTER LOCATIONS Table 6

REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	ADDRESS
STACK POINTER									81h
DPL									82h
DPH									83h
DPL1									84h
DPH1									85h
DPS	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**SEL**	86h
PCON	SMOD	**SMOD0**	–	–	GF1	GF0	STOP	IDLE	87h
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	88h
TMOD	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	89h
TL0									8Ah
TL1									8Bh
TH0									8Ch
TH1									8Dh
CKCON	**WD1**	**WD0**	**T2M**	**T1M**	**T0M**	**MD2**	**MD1**	**MD0**	8Eh
PORT1	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	90h
EXIF	**IE5**	**IE4**	**IE3**	**IE2**	**–**	**RGMD**	**RGSL**	**BGS**	91h
SCON0	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI	98h
SBUF0									99h
PORT2	P2.0	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	A0h
IE	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0	A8h
SADDR0									A9h
SADDR1									AAh
PORT3	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	B0h
IP	–	PS1	PT2	PS0	PT1	PX1	PT0	PX0	B8h
SADEN0									B9h
SADEN1									BAh
SCON1	**SM0/FE**	**SM1**	**SM2**	**REN**	**TB8**	**RB8**	**TI**	**RI**	C0h
SBUF1									C1h
TA									C7h
T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/RL2	C8h
T2MOD	–	–	–	–	–	–	T2OE	DCEN	C9h
RCAP2L									CAh
RCAP2H									CBh
TL2									CCh
TH2									CDh
PSW	CY	AC	F0	RS1	RS0	OV	FL	P	D0h
WDCON	**SMOD**	**POR**	**EPFI**	**PFI**	**WDIF**	**WTRF**	**EWT**	**RWT**	D8h
ACC									E0h
EIE	**–**	**–**	**–**	**EWDI**	**EX5**	**EX4**	**EX3**	**EX2**	E8h
B									F0h
EIP	**–**	**–**	**–**	**PWDI**	**PX5**	**PX4**	**PX3**	**PX2**	F8h

ELECTRICAL SPECIFICATIONS

V_{CC}=+5V ± 10%; t_A=0°C to 70°C

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	–1.0V to 7.0V
Operating Temperature	–40°C to +85°C
Storage Temperature	–55°C to 125°C
Soldering Temperature	260°C for 10 seconds

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

DC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.0V to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Operating Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
Power Fail Warning	V_{PFW}	4.25	4.38	4.5	V	1
Minimum Operating Voltage	V_{RST}	4.0	4.1	4.25	V	1
Supply Current Active Mode @ 25 MHz	I_{CC}		30		mA	2
Supply Current Idle Mode @ 25 MHz	I_{IDLE}		20		mA	3
Supply Current Stop Mode	I_{STOP}		.01	1	µA	4
Supply Current Stop Mode, Band–gap On	I_{SPBG}		51		µA	4, 5
Input Low Level	V_{IL}	–0.3		+0.8	V	1
Input High Level (Except XTAL1 and RST)	V_{IH1}	2.0		V_{CC}+0.3	V	1
Input High Level XTAL1 and RST	V_{IH2}	3.5		V_{CC}+0.3	V	1
Output Low Voltage @I_{OL}=1.6 mA Ports 1, 3	V_{OL1}			0.45	V	1
Output Low Voltage Ports 0, 2, ALE, $\overline{PSEN}$ @I_{OL}=3.2 mA	V_{OL2}			0.45	V	1, 6
Output High Voltage Ports 0, 1, 2, 3, ALE, $\overline{PSEN}$, @I_{OH}=–50 µA	V_{OH1}	2.4			V	1, 7
Output High Voltage Ports 1, 3 @I_{OH}=–50 µA	V_{OH2}	2.4			V	1, 8
Output High Voltage Ports 1, 3 @I_{OH}=–1.5 mA	V_{OH3}	2.4			V	1, 9
Output High Voltage Ports 0, 2, ALE, $\overline{PSEN}$ I_{OH}=–8 mA	V_{OH4}	2.4			V	1, 6
Input Low Current Ports 1, 3 @0.45V	I_{IL}			–55	µA	
Transition Current from 1 to 0 Ports 1, 3, @2V	I_{TL}			–650	µA	10
Input Leakage Port 0	I_L		±300		µA	11
RST Pull–down Resistance	R_{RST}	50		170	kΩ	

NOTES:

1. All voltages referenced to ground.
2. Active current is measured with a 25 MHz clock source driving XTAL1, +5V between V_{CC} and GND, RST at +5V, all other pins disconnected.
3. Idle mode current is measured with a 25 MHz clock source driving XTAL1, +5V between V_{CC} and GND, RST at ground, all other pins disconnected.
4. Stop mode current measured with XTAL1 and RST grounded, +5V between V_{CC} and GND, all other pins disconnected.
5. Band–gap reference is enabled.
6. When addressing.
7. During a reset condition.
8. At a steady state, not being pulled below V_{IH1} externally.
9. During a 0 to 1 transition, a one–shot drives the ports hard for two clock cycles.
10. Ports 1, 2, 3 source transition current when being pulled down externally. It reaches its maximum at approximately 2V.
11. V_{IN} between ground and V_{CC}–0.3V. Not a high impedance input. This port is a weak address holding latch. Peak current occurs near the input transition point of the latch.

AC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.0V to 5.5V)

PARAMETER	SYMBOL	25 MHz MIN	25 MHz MAX	VARIABLE CLOCK MIN	VARIABLE CLOCK MAX	UNITS
Oscillator Frequency	$1/t_{CLCL}$	0	25	0	25	MHz
ALE Pulse Width	t_{LHLL}	50		$1.5t_{CLCL}-10$		ns
Address Valid to ALE Low	t_{AVLL}	9		$.5t_{CLCL}-11$		ns
Address Hold After ALE Low	t_{LLAX1}	5	note 5	$.25t_{CLCL}-5$	note 5	ns
Address Hold After ALE Low for MOVX $\overline{WR}$	t_{LLAX2}	13		$.5t_{CLCL}-7$		ns
ALE Low to Valid Instruction In	t_{LLIV}		73		$2.5t_{CLCL}-27$	ns
ALE Low to $\overline{PSEN}$ Low	t_{LLPL}	3		$.25t_{CLCL}-7$		ns
$\overline{PSEN}$ Pulse Width	t_{PLPH}	83		$2.25t_{CLCL}-7$		ns
$\overline{PSEN}$ Low to Valid Instr. In	t_{PLIV}		69		$2.25t_{CLCL}-21$	ns
Input Instruction Hold After $\overline{PSEN}$	t_{PXIX}	0		0		ns
Input Instruction Float After $\overline{PSEN}$	t_{PXIZ}		35		$t_{CLCL}-5$	ns
Port 0 Address to Valid Instr. In	t_{AVIV1}		93		$3t_{CLCL}-27$	ns
Port 2 Address to Valid Instr. In	t_{AVIV2}		107		$3.5t_{CLCL}-33$	ns
$\overline{PSEN}$ Low to Address Float	t_{PLAZ}		note 5		note 5	ns
Data Hold after Read	t_{RHDX}	0		0		ns
Data Float after Read	t_{RHDZ}		35		$t_{CLCL}-5$	ns
$\overline{RD}$ Low to Address Float	t_{RLAZ}		note 5		note 5	ns
$\overline{RD}$ or $\overline{WR}$ High to ALE High	t_{WHLH}	0	see table	0	see table	ns
$\overline{RD}$ Pulse Width	t_{RLRH}	see table		see table		ns
$\overline{WR}$ Pulse Width	t_{WLWH}	see table		see table		ns
$\overline{RD}$ Low to Valid Data In	t_{RLDV}		see table		see table	ns
ALE Low to Valid Data In	t_{LLDV}		see table		see table	ns
Address to Valid Data In	t_{AVDV}		see table		see table	ns
ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	t_{LLWL}	see table	see table	see table	see table	ns
Port 0 Address Valid to $\overline{RD}$ or $\overline{WR}$ Low	t_{AVWL1}	see table		see table		ns
Port 2 Valid to $\overline{RD}$ or $\overline{WR}$ Low	t_{AVWL2}	see table		see table		ns
Data Valid to $\overline{WR}$ Transition	t_{QVWX}	see table		see table		ns
Data Hold After Write	t_{WHQX}	see table		see table		ns

NOTES:

1. All signals rated over operating temperature.
2. All signals characterized with load capacitance of 80 pF except Port 0, ALE, $\overline{PSEN}$, $\overline{RD}$ and $\overline{WR}$ at 100 pF. Note that loading should be approximately equal for valid timing.
3. Interfacing to memory devices with float times (turn off times) over 35 ns may cause contention. This will not damage the parts, but will cause an increase in operating current.
4. Specifications assume a 50% duty cycle for the oscillator. Port 2 timing will change with the duty cycle variations.
5. Address is held in a weak latch until over driven by external memory.

MOVX CHARACTERISTICS USING STRETCH MEMORY CYCLES

(0°C to 70°C; V_{CC}=4.0 to 5.5V)

PARAMETER	SYMBOL	VARIABLE CLOCK MIN	VARIABLE CLOCK MAX	UNITS	STRETCH
$\overline{RD}$ Pulse Width	t_{RLRH}	$2t_{CLCL}-11+t_{MCS}$		ns	
$\overline{WR}$ Pulse Width	t_{WLWH}	$2t_{CLCL}-11+t_{MCS}$		ns	
$\overline{RD}$ Low to Valid Data In	t_{RLDV}		$2t_{CLCL}-25+t_{MCS}$	ns	
ALE Low to Valid Data In	t_{LLDV}		$2.5t_{CLCL}-26$ $3.5\ t_{CLCL}-28+t_{MCS}$	ns	t_{MCS}=0 t_{MCS}>0
Port 0 Address to Valid Data In	t_{AVDV1}		$3t_{CLCL}-24$ $4t_{CLCL}-31+t_{MCS}$	ns	t_{MCS}=0 t_{MCS}>0
Port 2 Address to Valid Data In	t_{AVDV2}		$3.5t_{CLCL}-32$ $4.5t_{CLCL}-34+t_{MCS}$	ns	t_{MCS}=0 t_{MCS}>0
$\overline{RD}$ or $\overline{WR}$ High to ALE High	t_{WHLH}	0 $1.0t_{CLCL}-5$	10 $1.0t_{CLCL}+11$	ns ns	t_{MCS}=0 t_{MCS}>0
ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	t_{LLWL}	$0.5t_{CLCL}-5$ $1.5t_{CLCL}-5$	$0.5t_{CLCL}+6$ $1.5t_{CLCL}+8$	ns ns	t_{MCS}=0 t_{MCS}>0
Port 0 Address Valid to $\overline{RD}$ or $\overline{WR}$ Low	t_{AVWL1}	$1.0t_{CLCL}-9$ $2.0t_{CLCL}-10$		ns ns	t_{MCS}=0 t_{MCS}>0
Port 2 Address Valid to $\overline{RD}$ or $\overline{WR}$ Low	t_{AVWL2}	$1.5t_{CLCL}-9$ $2.5t_{CLCL}-13$		ns ns	t_{MCS}=0 t_{MCS}>0
Data Valid to $\overline{WR}$ Transition	t_{QVWX}	-9 $1.0t_{CLCL}-10$		ns ns	t_{MCS}=0 t_{MCS}>0
Data Hold After Write	t_{WHQX}	$1.0t_{CLCL}-7$ $2.0t_{CLCL}-5$		ns ns	t_{MCS}=0 t_{MCS}>0

NOTES:

t_{MCS} is a time period related to the Stretch memory cycle selection. The following table shows the value of t_{MCS} for each Stretch selection.

MD2	MD1	MD0	MOVX CYCLES	t_{MCS}
0	0	0	2 machine cycles	0
0	0	1	3 machine cycles	$2t_{CLCL}$
0	1	0	4 machine cycles	$6t_{CLCL}$
0	1	1	5 machine cycles	$10t_{CLCL}$
1	0	0	6 machine cycles	$14t_{CLCL}$
1	0	1	7 machine cycles	$18t_{CLCL}$
1	1	0	8 machine cycles	$22t_{CLCL}$
1	1	1	9 machine cycles	$26t_{CLCL}$

EXTERNAL CLOCK CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.0 to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Clock High Time	t_{CHCX}	20			ns	
Clock Low Time	t_{CLCX}	20			ns	
Clock Rise Time	t_{CLCH}			15	ns	
Clock Fall Time	t_{CHCL}			15	ns	

SERIAL PORT MODE 0 TIMING CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.0 to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Serial Port Clock Cycle Time SM2=0 12 clocks per cycle SM2=1 4 clocks per cycle	t_{XLXL}		 $12t_{CLCL}$ $4t_{CLCL}$		ns	
Output Data Setup to Clock Rising Edge SM2=0 12 clocks per cycle SM2=1 4 clocks per cycle	t_{QVXH}		 $10t_{CLCL}$ $3t_{CLCL}$		ns	
Output Data Hold from Clock Rising SM2=0 12 clocks per cycle SM2=1 4 clocks per cycle	t_{XHQX}		 $2t_{CLCL}$ t_{CLCL}		ns	
Input Data Hold after Clock Rising SM2=0 12 clocks per cycle SM2=1 4 clocks per cycle	t_{XHDX}		 t_{CLCL} t_{CLCL}		ns	
Clock Rising Edge to Input Data Valid SM2=0 12 clocks per cycle SM2=1 4 clocks per cycle	t_{XHDV}		 $11t_{CLCL}$ $3t_{CLCL}$		ns	

EXPLANATION OF AC SYMBOLS

In an effort to remain compatible with the original 8051 family, this device specifies the same parameter as such devices, using the same symbols. For completeness, the following is an explanation of the symbols.

t	Time
A	Address
C	Clock
D	Input data
H	Logic level high
L	Logic level low
I	Instruction
P	$\overline{PSEN}$
Q	Output data
R	$\overline{RD}$ signal
V	Valid
W	$\overline{WR}$ signal
X	No longer a valid logic level
Z	Tristate

POWER CYCLE TIMING CHARACTERISTICS

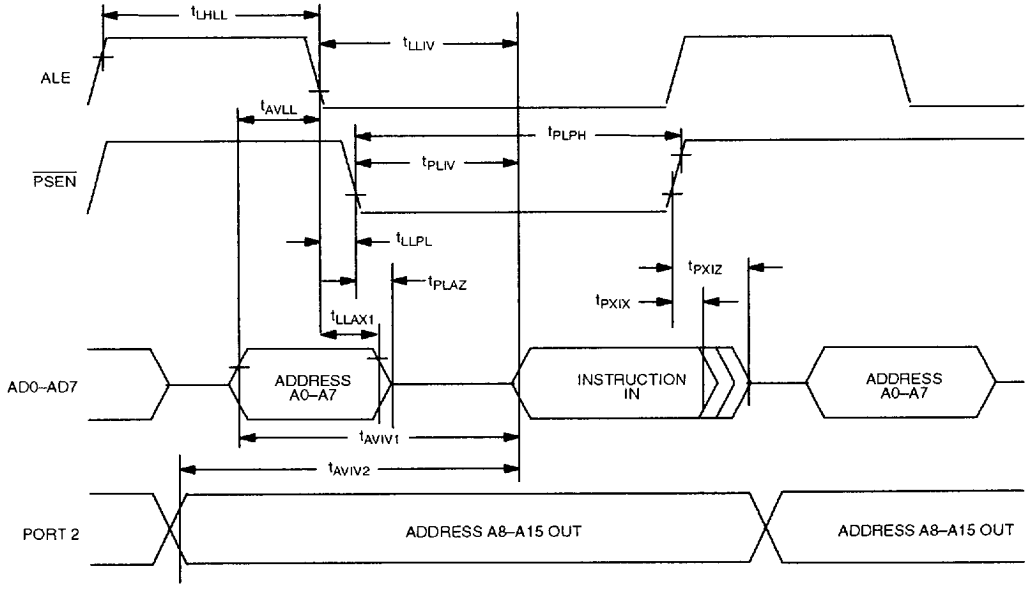
(0°C to 70°C; V_{CC}=4.0 to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Crystal Start–up Time	t_{CSU}		1.8		ms	1
Power–on Reset Delay	t_{POR}			65536	t_{CLCL}	2

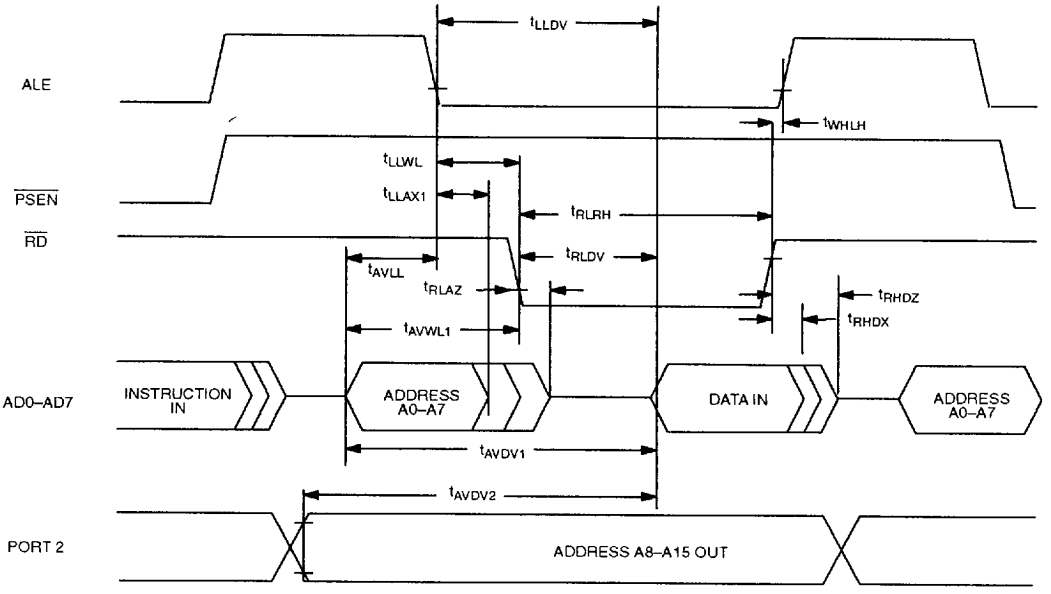
NOTES:

1. Start–up time for crystals varies with load capacitance and manufacturer. Time shown is for an 11.0592 MHz crystal manufactured by Fox crystal.
2. Reset delay is a synchronous counter of crystal oscillations after crystal start–up. At 25 MHz, this time is 2.62 ms.

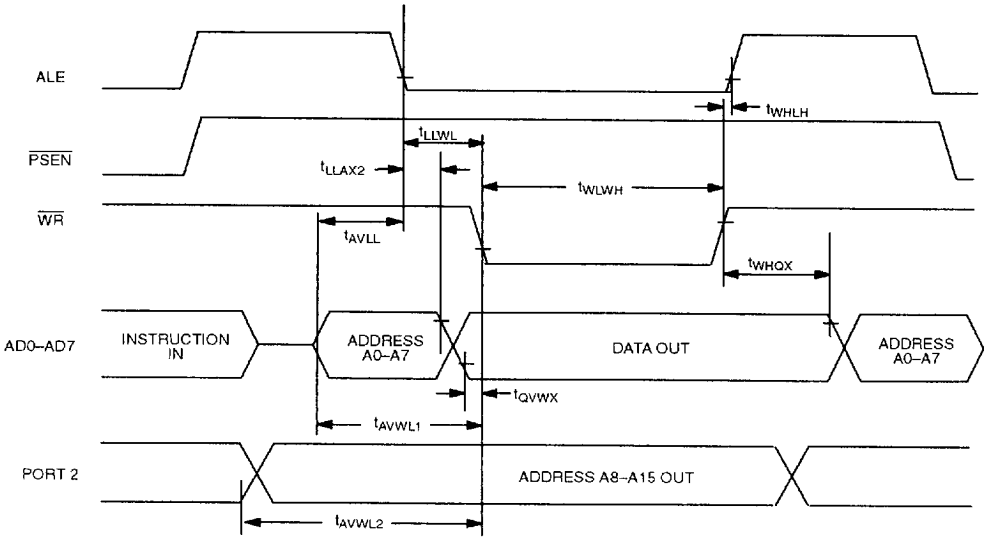
PROGRAM MEMORY READ CYCLE



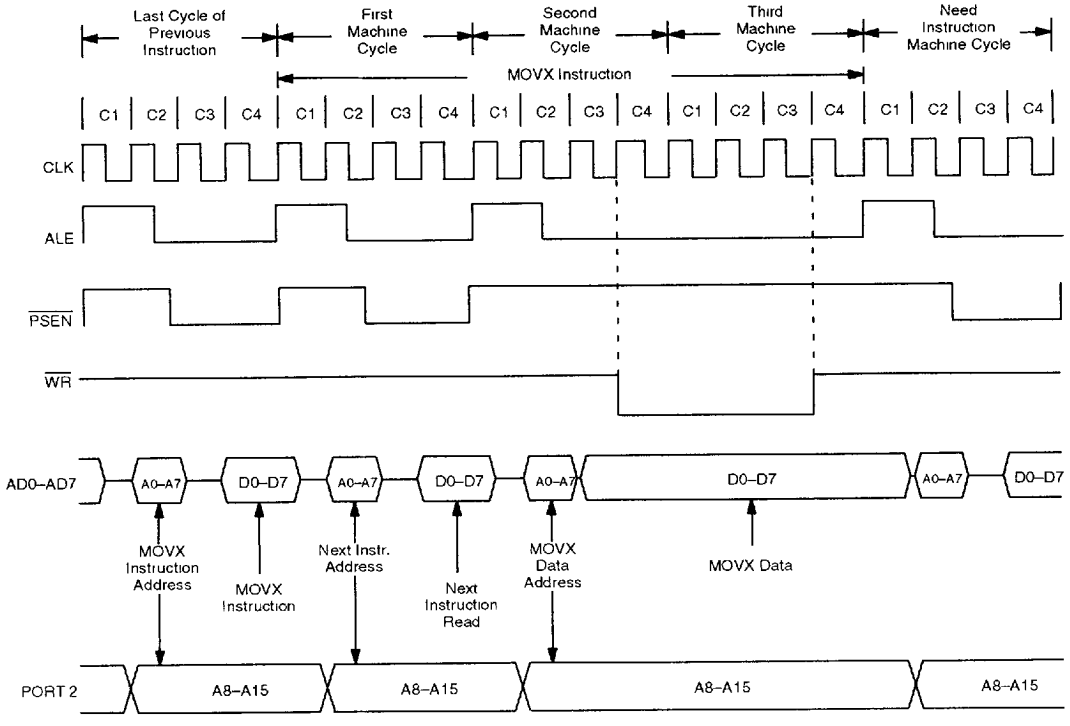
DATA MEMORY READ CYCLE



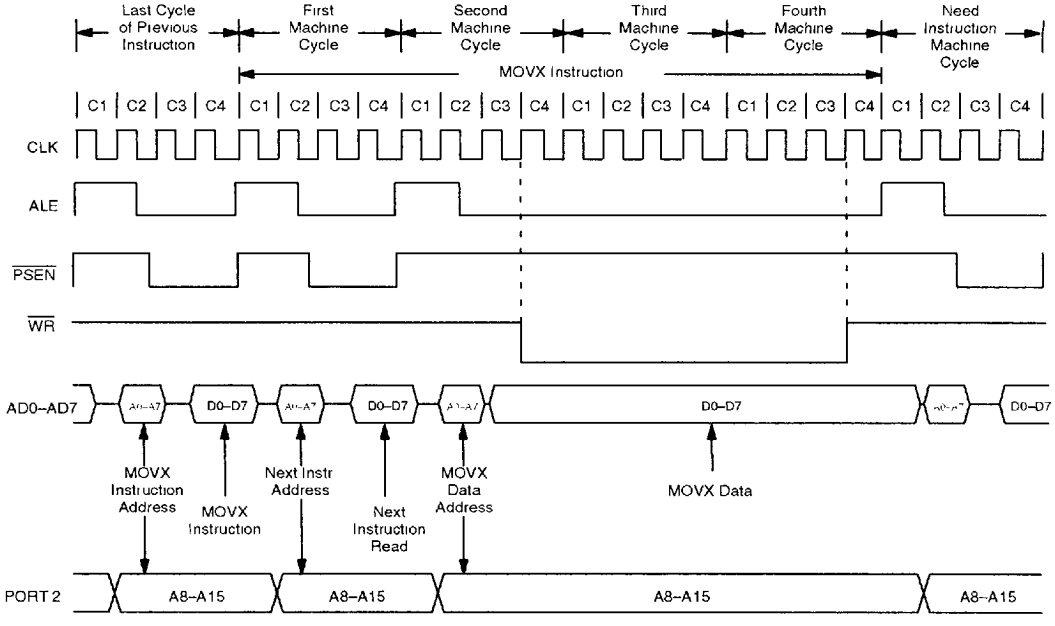
DATA MEMORY WRITE CYCLE



DATA MEMORY WRITE WITH STRETCH=1

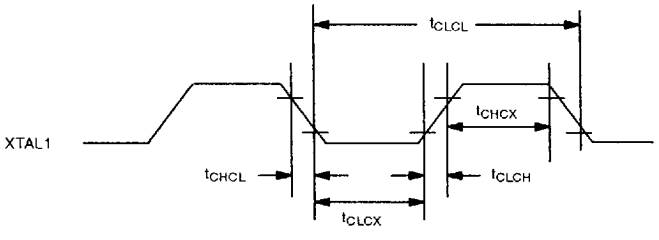


DATA MEMORY WRITE WITH STRETCH=2



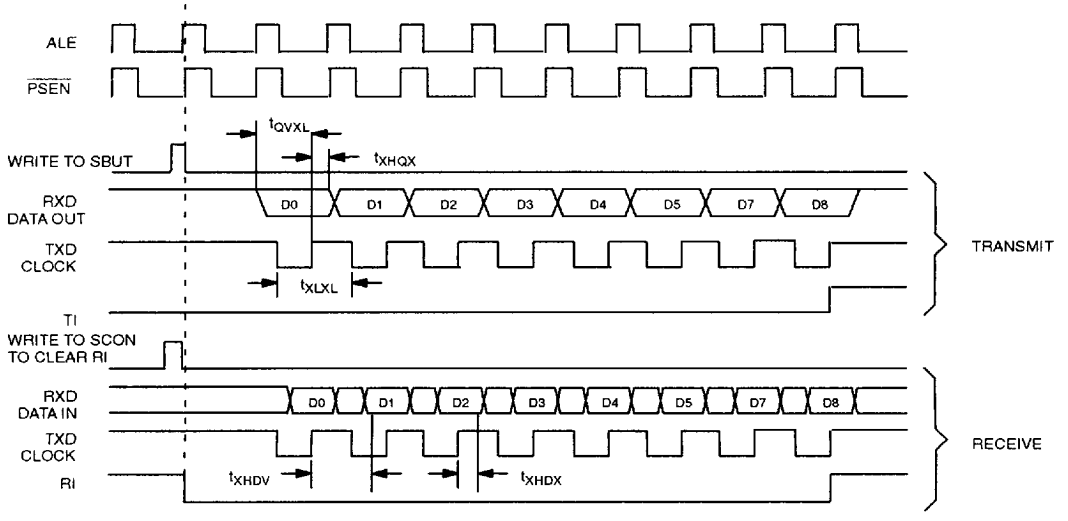
FOUR CYCLE DATA MEMORY WRITE
STRETCH VALUE=2

EXTERNAL CLOCK DRIVE

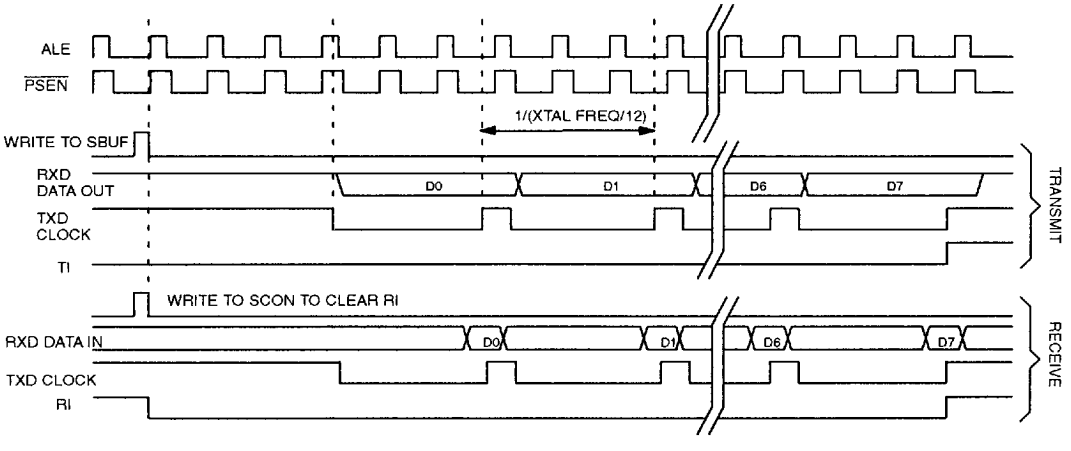


SERIAL PORT MODE 0 TIMING

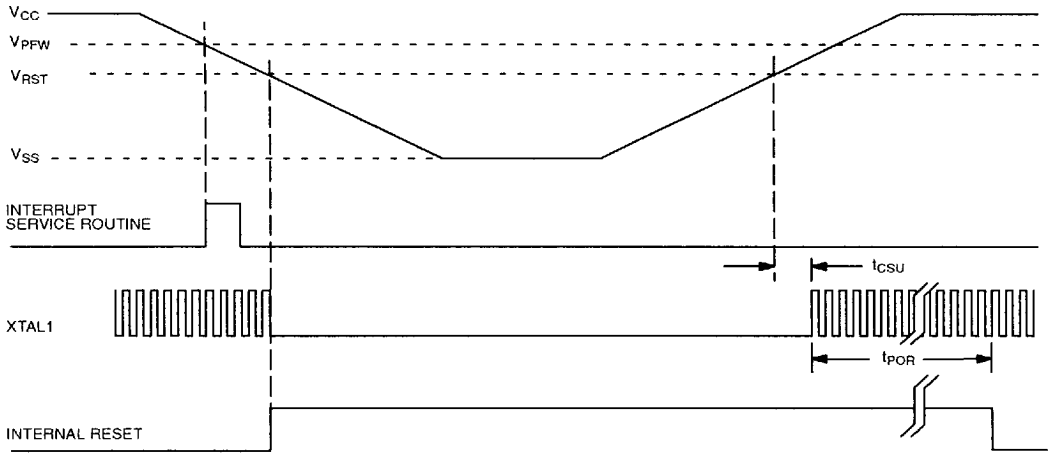
SERIAL PORT 0 (SYNCHRONOUS MODE)
HIGH SPEED OPERATION SM2=1=>TXD CLOCK=XTAL/4



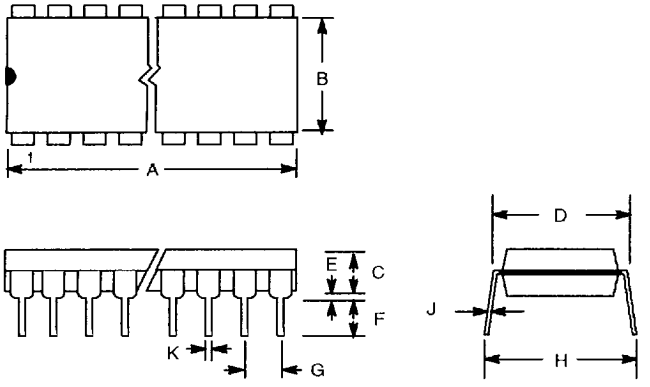
SERIAL PORT 0 (SYNCHRONOUS MODE)
SM2=0=>TXD CLOCK=XTAL/12



POWER CYCLE TIMING

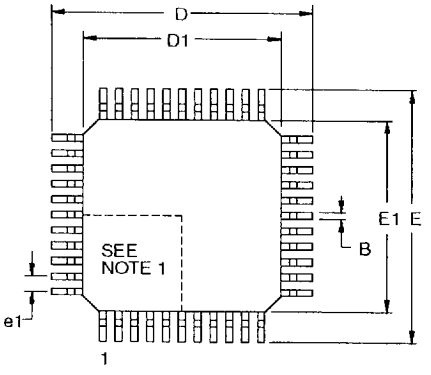


40–PIN PLASTIC DIP (600 MIL)



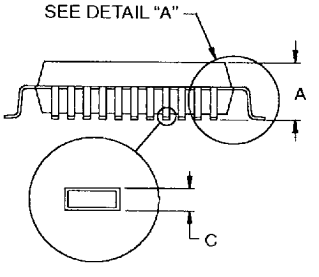
PKG	40-PIN	
DIM	MIN	MAX
A IN. MM	2 050 52.07	2.075 52.71
B IN. MM	0.530 13.46	0 550 13 97
C IN. MM	0.140 3.56	0.160 4.06
D IN. MM	0.600 15.24	0.625 15.88
E IN. MM	0.015 0.380	0.040 1 02
F IN. MM	0.120 3.05	0 145 3.68
G IN. MM	0.090 2.29	0.110 2 79
H IN MM	0.625 15.88	0.675 17 15
J IN MM	0.008 0 20	0.012 0 30
K IN. MM	0.015 0.38	0 022 0 56

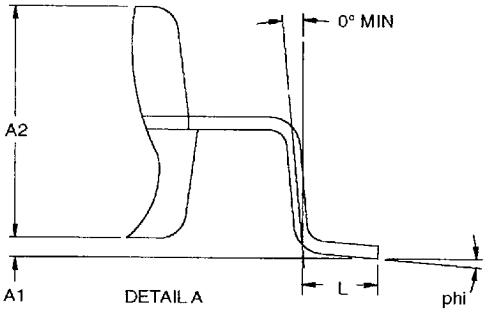
44–PIN METRIC QUAD FLAT PACK



NOTES:

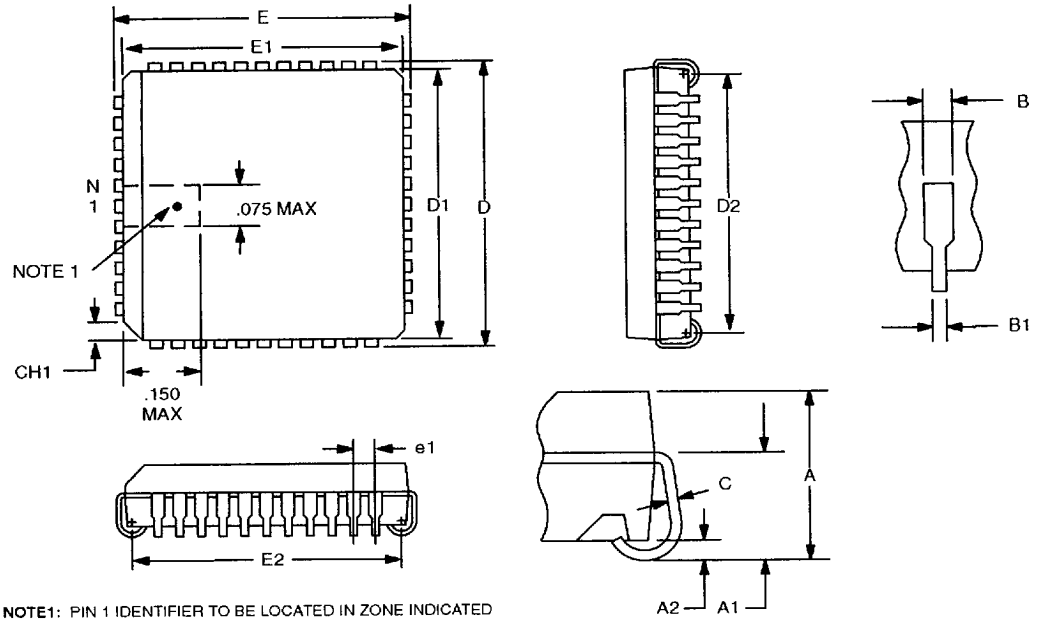
1. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL BUT MUST BE LOCATED WITHIN THE ZONE INDICATED.





PKG	10 X 10 X 2 BODY	
DIM	MIN	MAX
A MM	–	2 45
A1 MM	0.10	–
A2 MM	1.95	2 10
B MM	0 30	0.45
C MM	0 13	0 23
D MM	13.65	14 15
E MM	13 65	14 15
D1 MM	9.90	10 10
E1 MM	9 90	10.10
L MM	0 63	1 03
e1 MM	0 80 BSC	
phi	0°	8°

44–PIN PLCC



NOTE1: PIN 1 IDENTIFIER TO BE LOCATED IN ZONE INDICATED

DIM	INCHES	
	MIN	MAX
A	0.165	0 180
A1	0.090	0.120
A2	0.020	–
B	0.026	0.033
B1	0.013	0 021
C	0.009	0.012
CH1	0.042	0.048
D	0.685	0.695
D1	0.650	0.656
D2	0.590	0.630
E	0.685	0.695
E1	0.650	0.656
E2	0.590	0.630
e1	0.050 BSC	