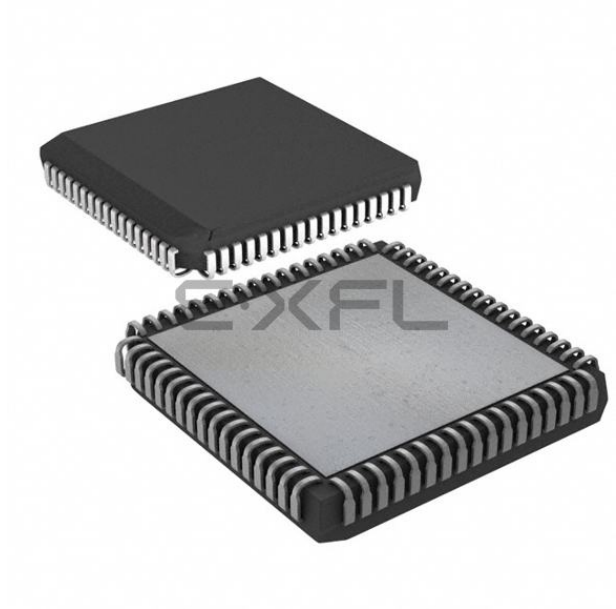
Analog Devices Inc./Maxim Integrated - DS80C390-QCR+ Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	8051
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, EBI/EMI, SIO, UART/USART
Peripherals	Power-Fail Reset, WDT
Number of I/O	32
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3.85V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.23x24.23)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/ds80c390-qcr

AC ELECTRICAL CHARACTERISTICS—(MULTIPLEXED ADDRESS/DATA BUS) (Note 10, Note 11)

PARAMETER	SYMBOL	CONDITIONS	40MHz		VARIABLE CLOCK		UNITS
			MIN	MAX	MIN	MAX	
Oscillator Frequency	$1 / t_{CLCL}$	External oscillator	0	40	0	40	MHz
		External crystal	1	40	1	40	
ALE Pulse Width	t_{LHLL}				$0.375\ t_{MCS} - 5$		ns
Port 0 Instruction Address or $\overline{CE0–4}$ Valid to ALE Low	t_{AVLL}				$0.125\ t_{MCS} - 5$		ns
Address Hold After ALE Low	t_{LLAX1}				$0.125\ t_{MCS} - 5$		ns
ALE Low to Valid Instruction In	t_{LLIV}					$0.625\ t_{MCS} - 20$	ns
ALE Low to $\overline{PSEN}$ Low	t_{LLPL}				$0.125\ t_{MCS} - 5$		ns
$\overline{PSEN}$ Pulse Width	t_{PLPH}				$0.5\ t_{MCS} - 8$		ns
$\overline{PSEN}$ Low to Valid Instruction In	t_{PLIV}					$0.5\ t_{MCS} - 20$	ns
Input Instruction Hold After $\overline{PSEN}$	t_{PXIX}		0		0		ns
Input Instruction Float After $\overline{PSEN}$	t_{PXIZ}					$0.25\ t_{MCS} - 5$	ns
Port 0 Address to Valid Instruction In	t_{AVIV1}					$0.75\ t_{MCS} - 22$	ns
Port 2, 4 Address to Valid Instruction In	t_{AVIV2}					$0.875\ t_{MCS} - 30$	ns
$\overline{PSEN}$ Low to Address Float	t_{PLAZ}			0		0	ns

Note 11: All parameters apply to both commercial and industrial temperature operation unless otherwise noted. The value t_{MCS} is a function of the machine cycle clock in terms of the processor's input clock frequency. These relationships are described in the *Stretch Value Timing* table. All signals characterized with load capacitance of 80pF except Port 0, ALE, $\overline{PSEN}$, $\overline{RD}$, and $\overline{WR}$ with 100pF. Interfacing to memory devices with float times (turn off times) over 25ns can cause bus contention. This does not damage the parts, but causes an increase in operating current. Specifications assume a 50% duty cycle for the oscillator. Port 2 and ALE timing changes in relation to duty cycle variation. Some AC timing characteristic drawings contain references to the CLK signal. This waveform is provided to assist in determining the relative occurrence of events and cannot be used to determine the timing of signals relative to the external clock. AC timing is characterized and guaranteed by design but is not production tested.

AC SYMBOLS

The DS80C390 uses timing parameters and symbols similar to the original 8051 family. The following list of timing symbols is provided as an aid to understanding the timing diagrams.

SYMBOL	FUNCTION
t	Time
A	Address
C	Clock
CE	Chip Enable
D	Input Data
H	Logic Level High
L	Logic Level Low
I	Instruction
P	$\overline{\text{PSEN}}$
Q	Output Data
R	$\overline{\text{RD}}$ Signal
V	Valid
W	$\overline{\text{WR}}$ Signal
X	No longer a valid logic level.
Z	Tri-State

Figure 1. Multiplexed External Program Memory Read Cycle

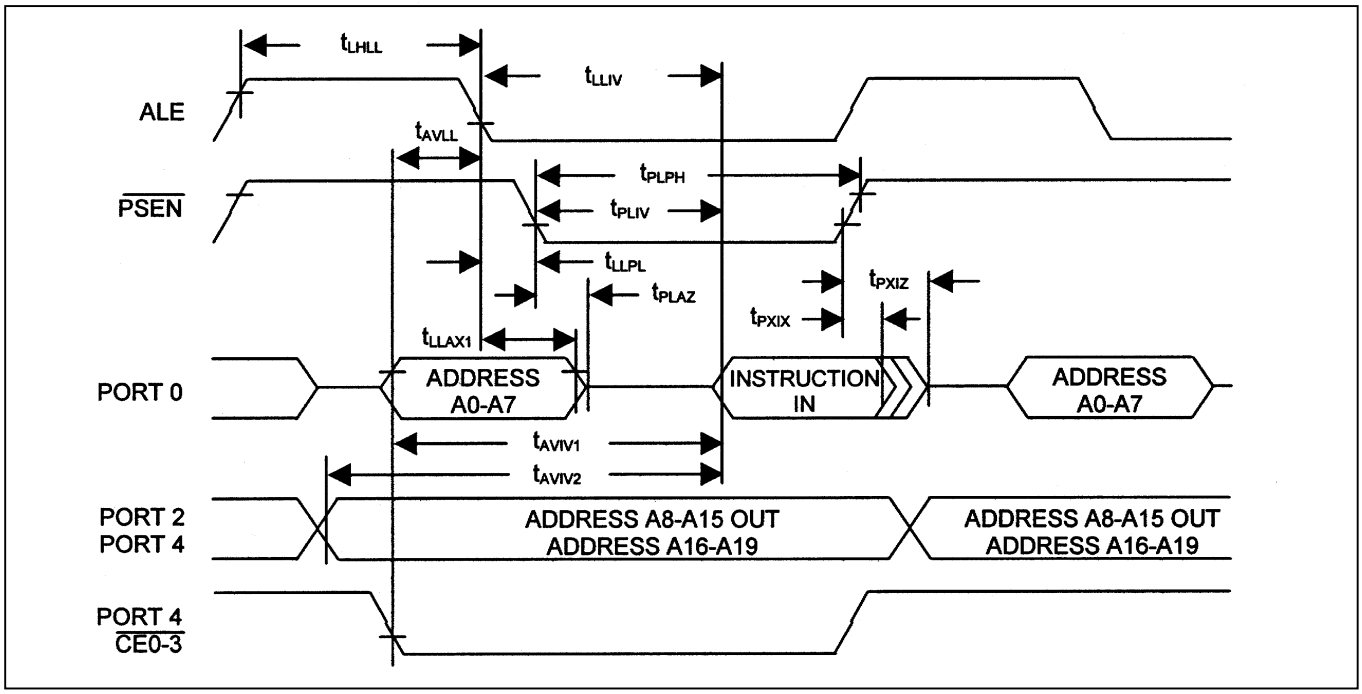


Figure 3. Multiplexed 9-Cycle Address/Data $\overline{\text{PCE0-3}}$ MOVX Read/Write Operation

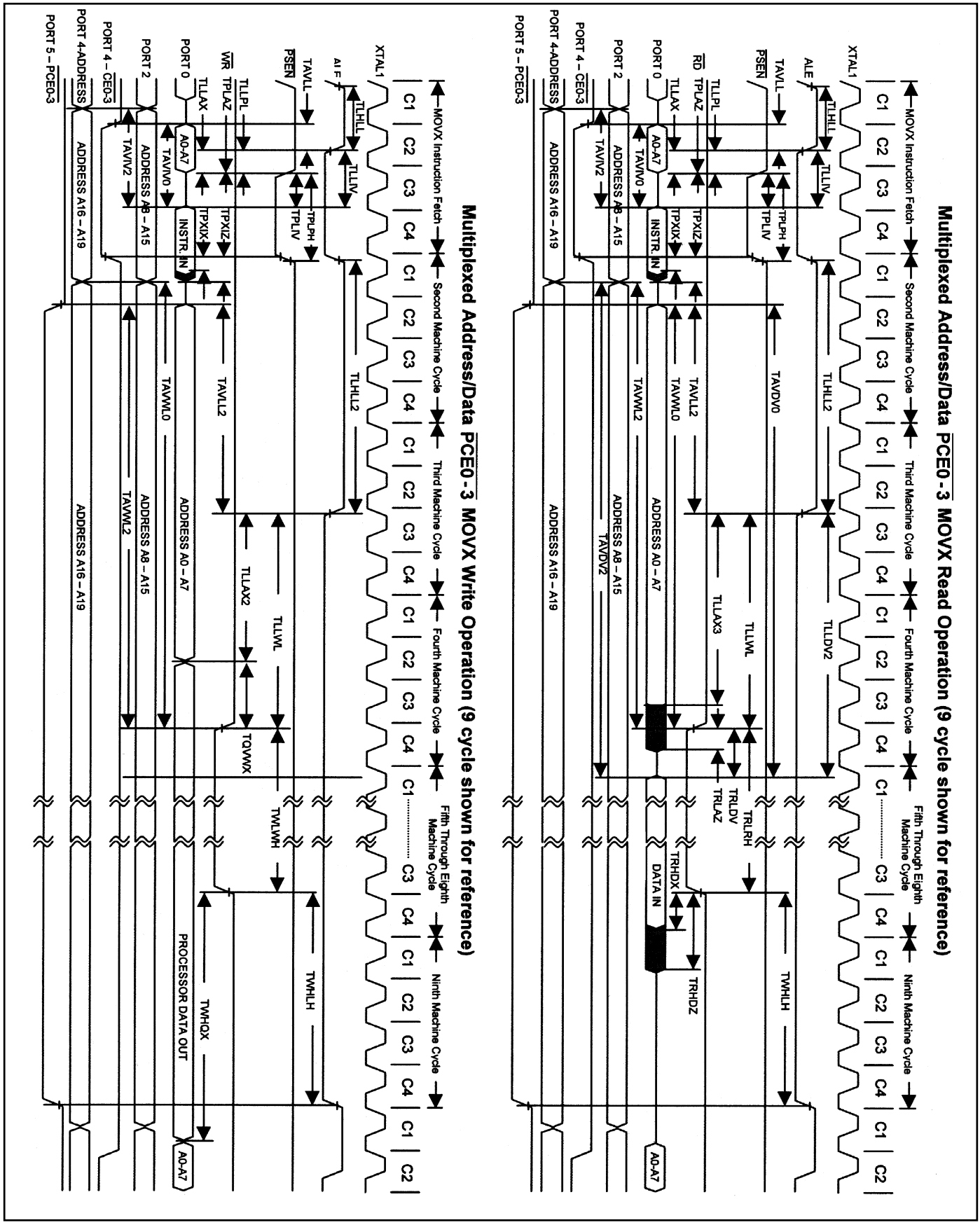


Figure 6. Multiplexed 2-Cycle Data Memory $\overline{CE0\text{-}3}$ Write

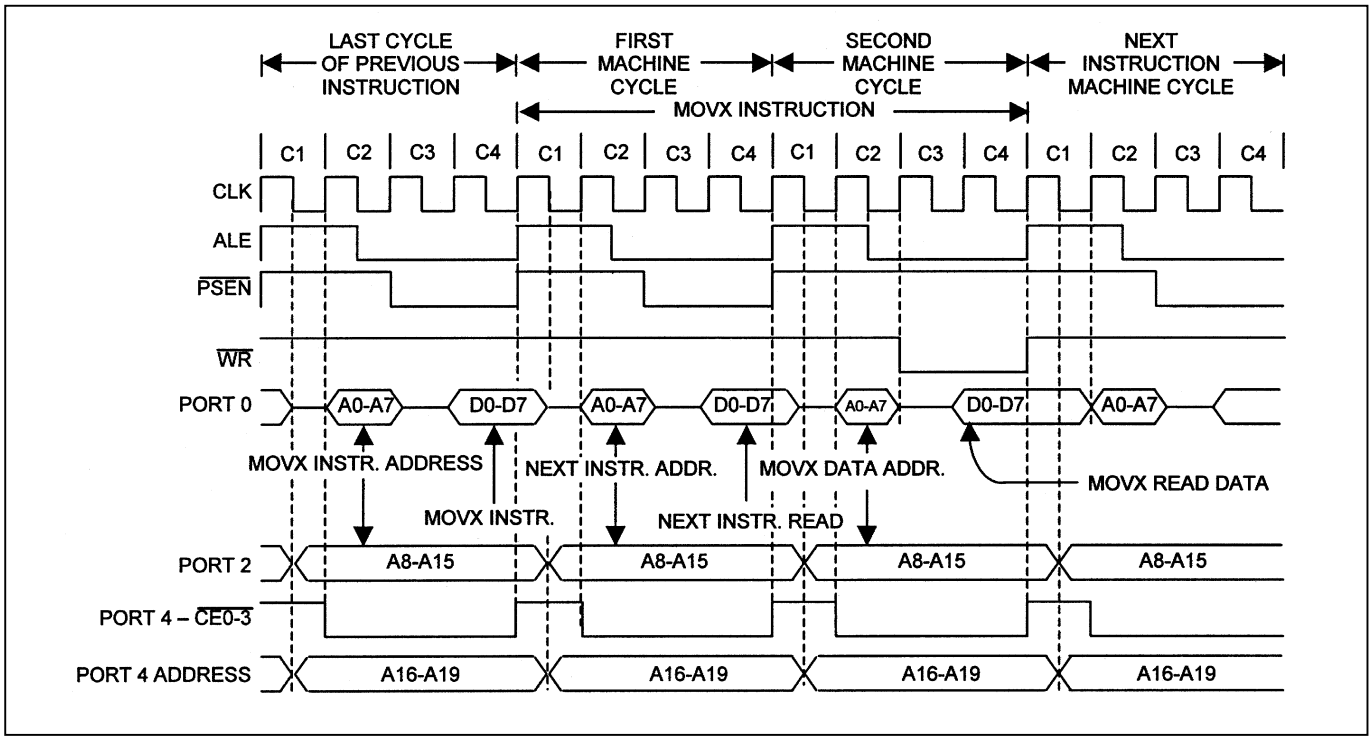


Figure 7. Multiplexed 3-Cycle Data Memory $\overline{PCE0\text{-}3}$ Read or Write

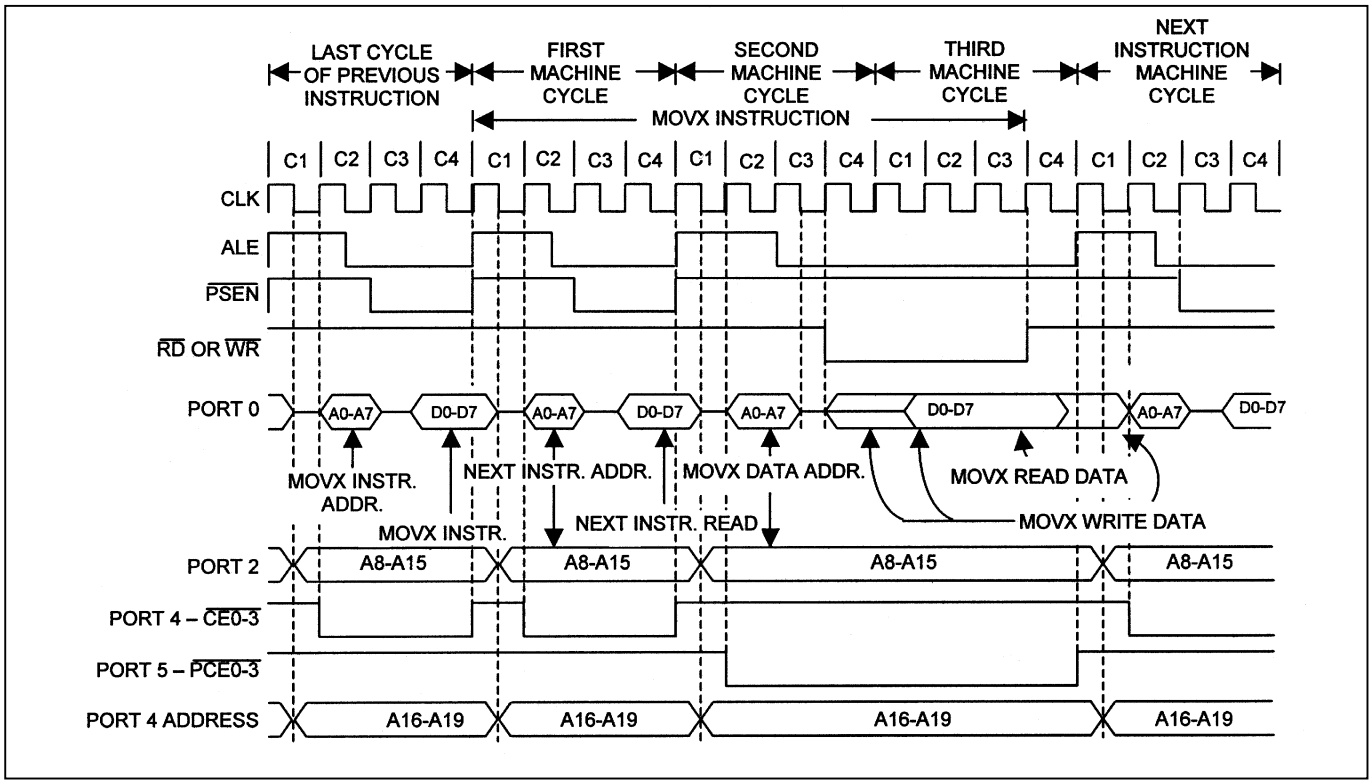


Figure 8. Multiplexed 3-Cycle Data Memory $\overline{CE0\text{-}3}$ Read

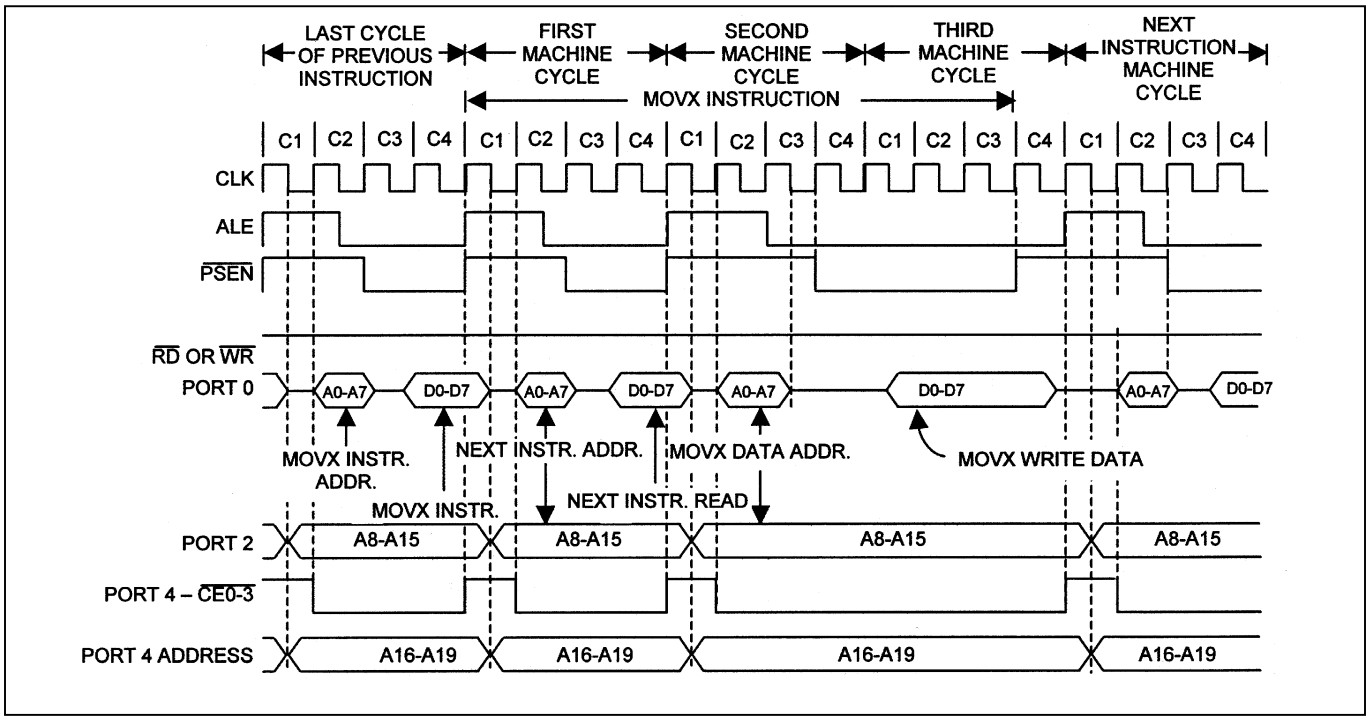


Figure 9. Multiplexed 3-Cycle Data Memory $\overline{CE0\text{-}3}$ Write

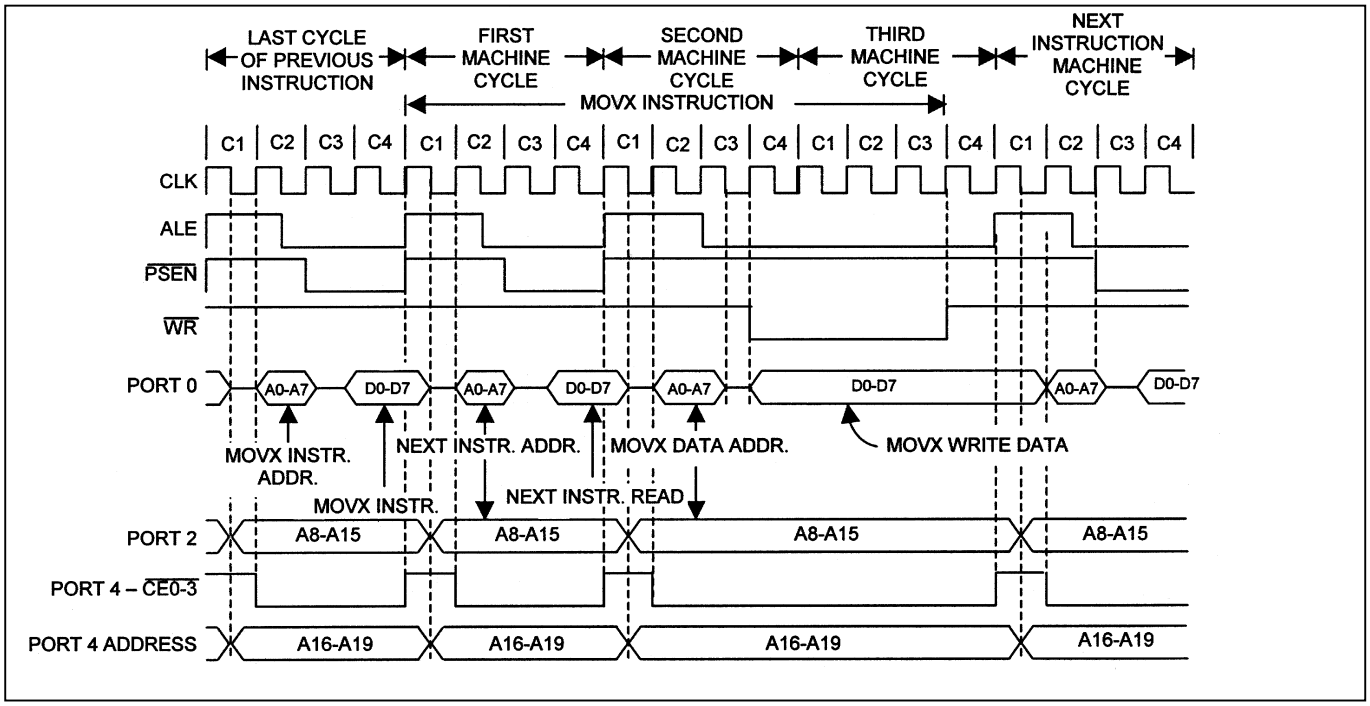


Figure 12. Multiplexed 9-Cycle Data Memory $\overline{CE0\text{-}3}$ Write

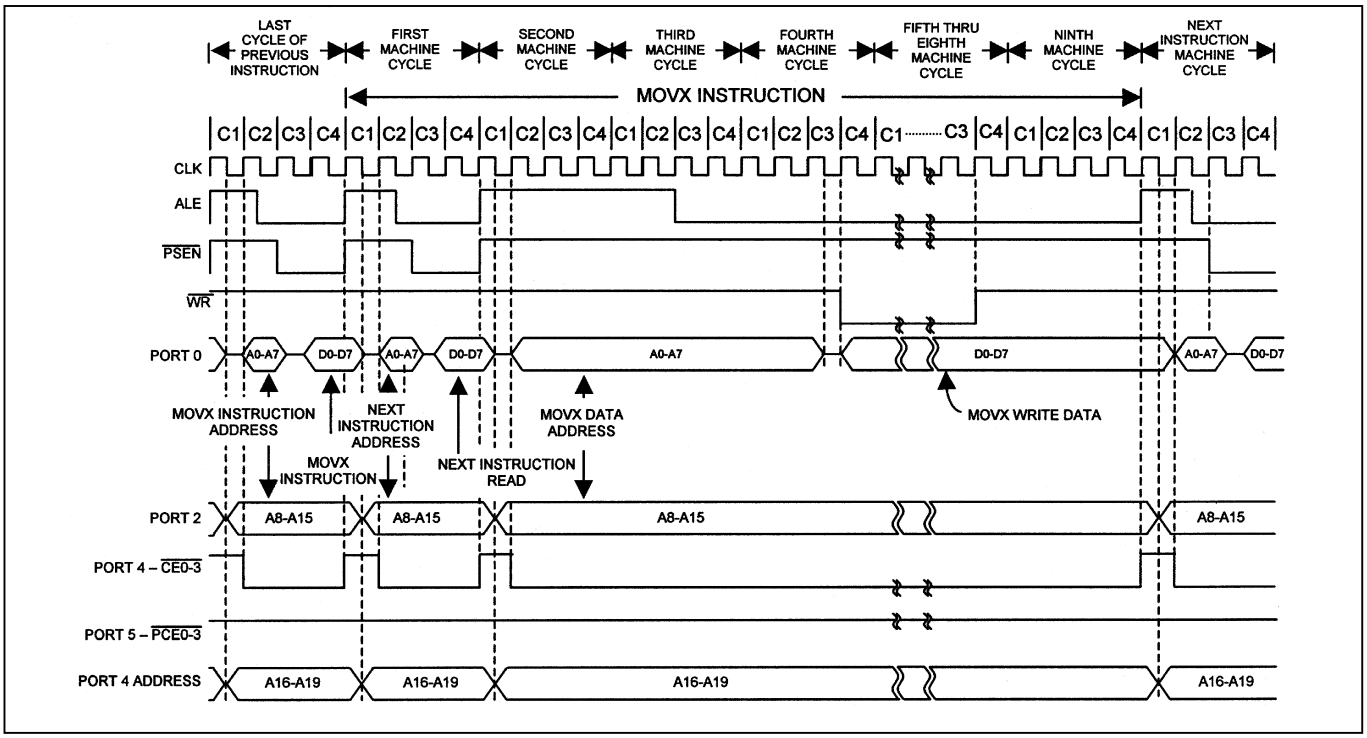


Figure 18. Nonmultiplexed 2-Cycle Data Memory $\overline{CE0\text{-}3}$ Write

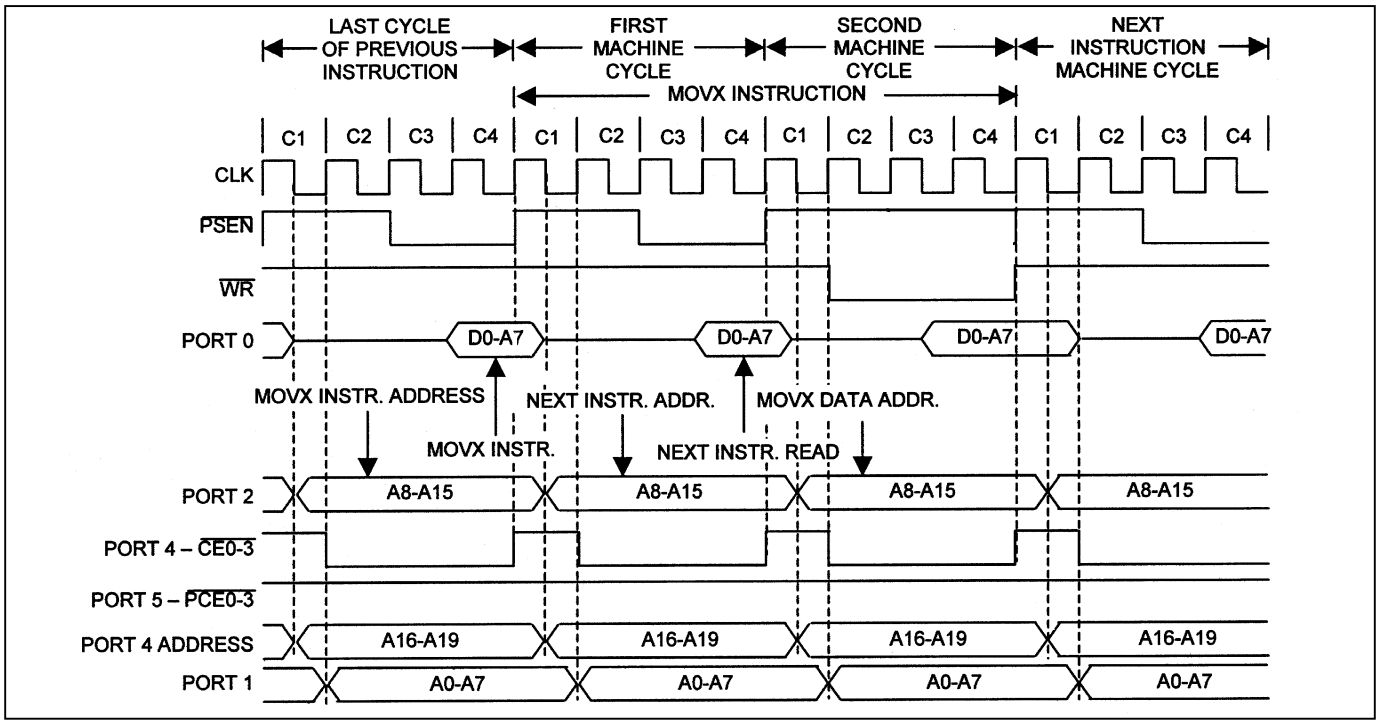
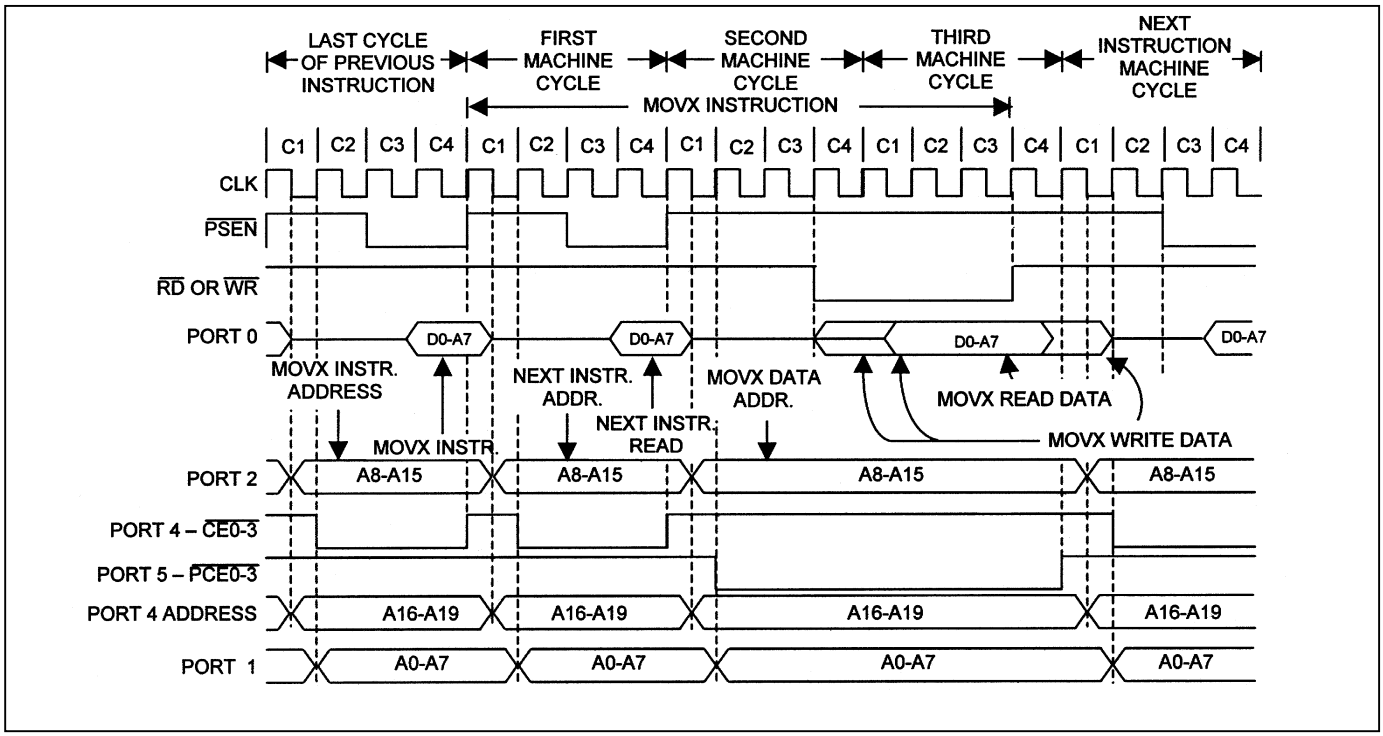


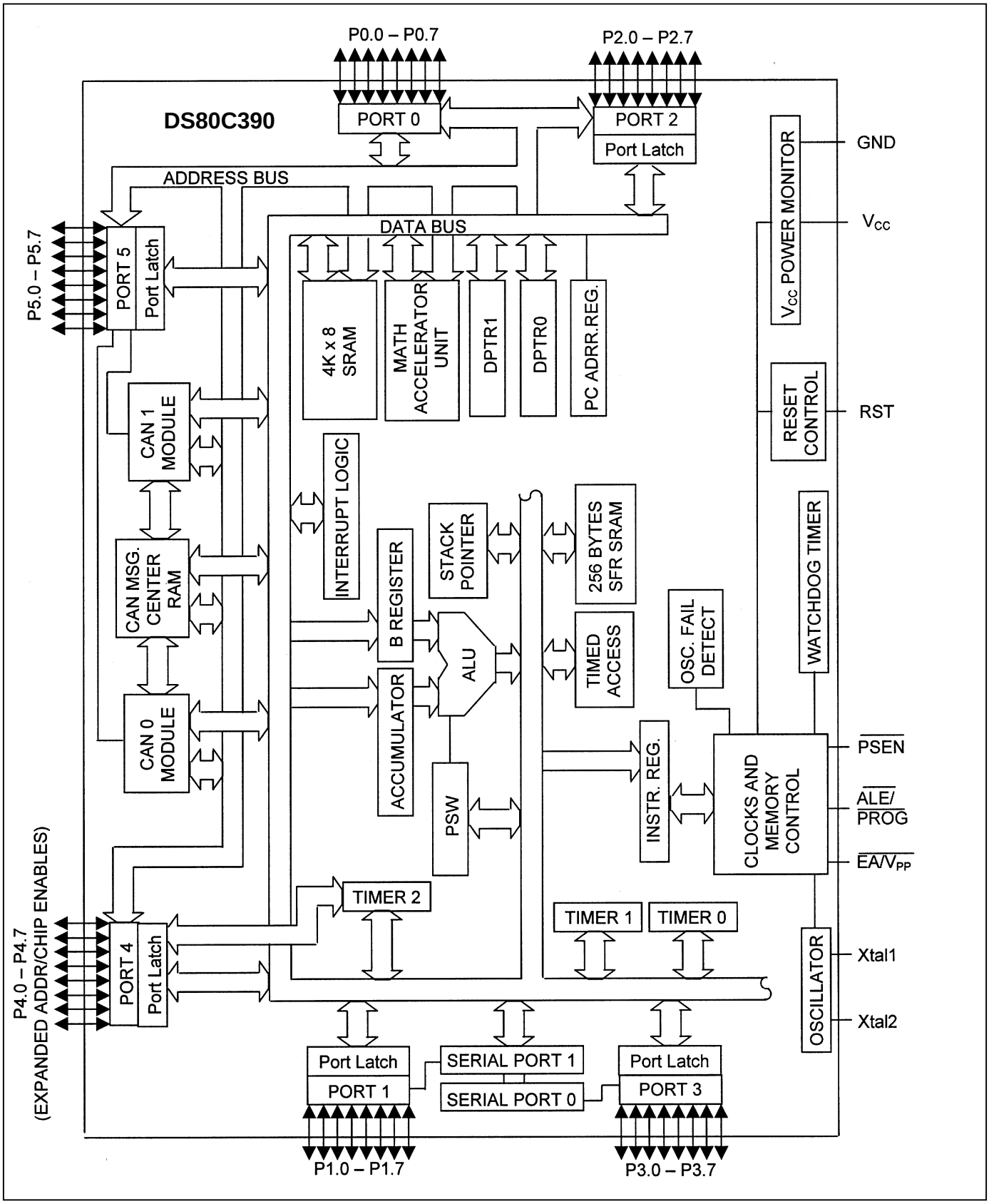
Figure 19. Nonmultiplexed 3-Cycle Data Memory $\overline{PEC0\text{-}3}$ Read or Write



PIN DESCRIPTION (continued)

PIN		NAME	FUNCTION
LQFP	PLCC		
58–64, 1	2–8, 10	P1.0–P1.7	**Port 1, I/O.** Port 1 can function as an 8-bit bidirectional I/O port, the nonmultiplexed A0–A7 signals (when the $\overline{MUX}$ pin = 1), and as an alternate interface for internal resources. Setting the SP1EC bit relocates RXD1 and TXD1 to Port 5. The reset condition of Port 1 is all bits at logic 1 through a weak pullup. The logic 1 state also serves as an input mode, since external circuits writing to the port can overdrive the weak pullup. When software clears any port pin to 0, a strong pulldown is activated that remains on until either a 1 is written to the port pin or a reset occurs. Writing a 1 after the port has been at 0 activates a strong transition driver, followed by a weaker sustaining pullup. Once the momentary strong driver turns off, the port once again becomes the output (and input) high state. **Port — Alternate Function** P1.0 — T2 External I/O for Timer/Counter 2 P1.1 — T2EX Timer/Counter 2 Capture/Reload Trigger P1.2 — RXD1 Serial Port 1 Input P1.3 — TXD1 Serial Port 1 Output P1.4 — INT2 External Interrupt 2 (Positive Edge Detect) P1.5 — $\overline{INT3}$ External Interrupt 3 (Negative Edge Detect) P1.6 — INT4 External Interrupt 4 (Positive Edge Detect) P1.7 — $\overline{INT5}$ External Interrupt 5 (Negative Edge Detect)
58	2	A0	
59	3	A1	
60	4	A2	
61	5	A3	
62	6	A4	
63	7	A5	
64	8	A6	
1	10	A7	
35	46	A8 (P2.0)	**A15–A8 (Port 2), Output.** Port 2 serves as the MSB for external addressing. The port automatically asserts the address MSB during external ROM and RAM access. Although the Port 2 SFR exists, the SFR value never appears on the pins (due to memory access). Therefore, accessing the Port 2 SFR is only useful for MOVX A, @Ri or MOVX @Ri, A instructions, which use the Port 2 SFR as the external address MSB.
36	47	A9 (P2.1)	
37	48	A10 (P2.2)	
38	49	A11 (P2.3)	
39	50	A12 (P2.4)	
42	53	A13 (P2.5)	
43	54	A14 (P2.6)	
44	55	A15 (P2.7)	
4–7, 10–13	13–16, 19–22	P3.0–P3.7	**Port 3, I/O.** Port 3 functions as an 8-bit bidirectional I/O port and as an alternate interface for several resources found on the traditional 8051. The reset condition of Port 1 is all bits at logic 1 through a weak pullup. The logic 1 state also serves as an input mode, since external circuits writing to the port can overdrive the weak pullup. When software clears any port pin to 0, the device activates a strong pulldown that remains on until either a 1 is written to the port pin or a reset occurs. Writing a 1 after the port has been at 0 activates a strong transition driver, followed by a weaker sustaining pullup. Once the momentary strong driver turns off, the port once again becomes the output (and input) high state. **Port — Alternate Function** P3.0 — RXD0 Serial Port 0 Input P3.1 — 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/XCLK Timer 1 External Input/External Clock Output P3.6 — $\overline{WR}$ External Data Memory Write Strobe P3.7 — $\overline{RD}$ External Data Memory Read Strobe
4	13		
5	14		
6	15		
7	16		
10	19		
11	20		
12	21		
13	22		

Figure 28. Block Diagram



40-BIT ACCUMULATOR

The accelerator also incorporates an automatic accumulator function, permitting the implementation of multiply-and-accumulate and divide-and-accumulate functions without any additional delay. Each time the accelerator is used for a multiply or divide operation, the result is transparently added to a 40-bit accumulator. This can greatly increase speed of DSP and other high-level math operations.

The accumulator can be accessed anytime the multiply/accumulate status flag (MCNT1;D2h) is cleared. The accumulator is initialized by performing five writes to the multiplier C register (MC;D5h), LSB first. The 40-bit accumulator can be read by performing five reads of the multiplier C register, MSB first.

MEMORY ADDRESSING

The DS80C390 incorporates three internal memory areas:

- 256 bytes of scratchpad (or direct) RAM
- 4kB of SRAM configurable as various combinations of MOVX data memory, stack memory, and MOVC program memory
- 512 bytes of RAM reserved for the CAN message centers.

Up to 4MB of external memory is addressed via a multiplexed or demultiplexed 20-bit address bus/8-bit data bus and four chip-enable (active during program memory access) or four peripheral-enable (active during data memory access) signals. Three different addressing modes are supported, as selected by the AM1, AM0 bits in the ACON SFR.

16-Bit Address Mode

Memory is accessed by 16-bit address mode similarly to the traditional 8051. It is op-code compatible with the 8051 microprocessor and identical to the byte and cycle count of the Dallas Semiconductor High-Speed Microcontroller family. A device operating in this mode can access up to 64kB of program and data memory. The device defaults to this mode following any reset.

22-Bit Paged-Address Mode

The 22-bit paged-address mode retains binary-code compatibility with the 8051 instruction set, but adds one machine cycle to the ACALL, LCALL, RET, and RETI instructions with respect to Dallas Semiconductor's High-Speed Microcontroller family timing. This is transparent to standard 8051 compilers. Interrupt latency is also increased by one machine cycle. In this mode, interrupt vectors are fetched from 0000xxh.

22-Bit Contiguous Address Mode

The 22-bit contiguous addressing mode uses a full 22-bit program counter, and all modified branching instructions automatically save and restore the entire program counter. The 22-bit branching instructions such as ACALL, AJMP, LCALL, LJMP, MOV DPTR, RET, and RETI instructions require an assembler, compiler, and linker that specifically supports these features. The INC DPTR is lengthened by one cycle but remains byte-count-compatible with the standard 8051 instruction set.

Internally, the device uses a 22-bit program counter. The lowest order 22 bits are used for memory addressing, with a special 23rd bit used to map the 4kB SRAM above the 4MB memory space in bootstrap loader applications. Address bits 16–23 for the 22-bit addressing modes are generated through additional SFRs dependent on the type of instruction as shown in Table 4.

Table 4. Extended Address Generation

INSTRUCTION	ADDRESS BITS 23–16	ADDRESS BITS 15–8	ADDRESS BITS 7–0
MOVX instructions using DPTR	DPX;93h	DPH;83h	DPL;82h
MOVX instructions using DPTR1	DPX1;95h	DPH1;85h	DPL1;84h
MOVX instructions using @Ri	MXAX;EAh	P2;A0h	Ri
Addressing program memory in 22-bit paged mode	AP;9Ch	—	—
10-bit stack pointer mode	—	ESP;9Bh	SP;81h

INTERNAL MOVX SRAM

The DS80C390 contains 4kB of SRAM that can be configured as user accessible MOVX memory, program memory, or optional stack memory. The specific configuration and locations are governed by the internal data memory configuration bits (IDM1, IDM0) in the memory control register (MCON;C6h). Note that when the SA bit (ACON.2) is set, the first 1kB of the MOVX data memory is reserved for use by the 10-bit expanded stack. Internal memory accesses will not generate $\overline{WR}$, $\overline{RD}$, or $\overline{PSEN}$ strobes.

The DS80C390 can configure its 4kB of internal SRAM as combined program and data memory. This allows the application software to execute self-modifiable code. The technique loads the 4kB SRAM with bootstrap loader software, and then modifies the IDM1 and IDM0 bits to map the 4kB starting at memory location 40000h. This allows the system to run the bootstrap loader without disturbing the 4MB external memory bus, making the device in-system reprogrammable for flash or NV RAM.

Table 5. Internal MOVX SRAM Configuration

IDM1	IDM0	CMA	MEMORY		
			MOVX DATA	CAN MESSAGE	SHARED PROGRAM/DATA
0	0	0	00F000h–00FFFFh	00EE00h–00EFFFh	—
0	0	1	00F000h–00FFFFh	401000h–4011FFh	—
0	1	0	000000h–000FFFh	00EE00h–00EFFFh	—
0	1	1	000000h–000FFFh	401000h–4011FFh	—
1	0	0	400000h–400FFFh	00EE00h–00EFFFh	—
1	0	1	400000h–400FFFh	401000h–4011FFh	—
1	1	0	—	00EE00h–00EFFFh	400000h–400FFFh*
1	1	1	—	401000h–4011FFh	400000h–400FFFh*

**10-bit expanded stack is not available in shared program/data memory mode.*

EXTERNAL MEMORY ADDRESSING

The enabling and mapping of the chip-enable signals is done through the Port 4 control register (P4CNT;92h) and memory control register (MCON; 96h). Table 7 shows which chip-enable and address line signals are active on Port 4. Following reset, the device will be configured with P4.7–P4.4 as address lines and P4.3–P4.0 configured as $\overline{CE3\text{-}0}$, with the first program fetch being performed from 00000h with $\overline{CE0}$ active. The following tables illustrate which memory ranges are controlled by each chip enable as a function of which address lines are enabled.

Table 6. External Memory Addressing Pin Assignments

ADDRESS/DATA BUS	$\overline{CE3}$–$\overline{CE0}$	$\overline{PCE3}$–$\overline{PCE0}$	ADDR 19–16	ADDR 15–8	ADDR 7–0	DATA BUS
Multiplexed	P4.3–P4.0	P5.7–P5.4	P4.7–P4.4	P2	P0	P0
Demultiplexed	P4.3–P4.0	P5.7–P5.4	P4.7–P4.4	P2	P1	P0

Table 7. Extended Address and Chip-Enable Generation

P4CNT.5–3	PORT 4 PIN FUNCTION				P4CNT.2–0	PORT 4 PIN FUNCTION			
	P4.7	P4.6	P4.5	P4.4		P4.3	P4.2	P4.1	P4.0
000	I/O	I/O	I/O	I/O	000	I/O	I/O	I/O	I/O
100	I/O	I/O	I/O	A16	100	I/O	I/O	I/O	$\overline{CE0}$
101	I/O	I/O	A17	A16	101	I/O	I/O	$\overline{CE1}$	$\overline{CE0}$
110	I/O	A18	A17	A16	110	I/O	$\overline{CE2}$	$\overline{CE1}$	$\overline{CE0}$
111(default)	A19	A18	A17	A16	111(default)	$\overline{CE3}$	$\overline{CE2}$	$\overline{CE1}$	$\overline{CE0}$

Table 9. Data Memory Cycle Stretch Values

MD2	MD1	MD0	STRETCH CYCLE COUNT	MOVX MACHINE CYCLES	$\overline{RD}$, $\overline{WR}$ PULSE WIDTH (IN OSCILLATOR CLOCKS)			
					t_{MCS} (4X/$\overline{2X}$ = 1 CD1:0 = 00)	t_{MCS} (4X/$\overline{2X}$ = 0 CD1:0 = 00)	t_{MCS} (4X/$\overline{2X}$ = X CD1:0 = 10)	t_{MCS} (4X/$\overline{2X}$ = X CD1:0 = 11)
0	0	0	0*	2	0.5 t_{CLCL}	1 t_{CLCL}	2 t_{CLCL}	2048 t_{CLCL}
0	0	1	1**	3	t_{CLCL}	2 t_{CLCL}	4 t_{CLCL}	4096 t_{CLCL}
0	1	0	2	4	2 t_{CLCL}	4 t_{CLCL}	8 t_{CLCL}	8192 t_{CLCL}
0	1	1	3	5	3 t_{CLCL}	6 t_{CLCL}	12 t_{CLCL}	12,288 t_{CLCL}
1	0	0	4	9	4 t_{CLCL}	8 t_{CLCL}	16 t_{CLCL}	16,384 t_{CLCL}
1	0	1	5	10	5 t_{CLCL}	10 t_{CLCL}	20 t_{CLCL}	20,480 t_{CLCL}
1	1	0	6	11	6 t_{CLCL}	12 t_{CLCL}	24 t_{CLCL}	24,576 t_{CLCL}
1	1	1	7	12	7 t_{CLCL}	14 t_{CLCL}	28 t_{CLCL}	28,672 t_{CLCL}

**All internal MOVX operations execute at the 0 Stretch setting.*
***Default stretch setting for external MOVX operations following reset.*

EXTENDED STACK POINTER

The DS80C390 supports both the traditional 8-bit and an extended 10-bit stack pointer that improves the performance of large programs written in high-level languages such as C. Enable the 10-bit stack pointer feature by setting the stack address mode bit, SA (ACON.2). The bit is cleared following a reset, forcing the device to use an 8-bit stack located in the scratchpad RAM area. When the SA bit is set, the device will address up to 1kB of stack memory in the first 1kB of the internal MOVX memory. The 10-bit stack pointer address is generated by concatenating the lower two bits of the extended stack pointer (ESP;9Bh) and the traditional 8051 stack pointer (SP;81h). The 10-bit stack pointer cannot be enabled when the 4kB of SRAM is mapped as both program and data memory.

ENHANCED DUAL DATA POINTERS

The DS80C390 contains two data pointers, DPTR0 and DPTR1, designed to improve performance in applications that require high data throughput. Incorporating a second data pointer allows the software to greatly speed up block data (MOVX) moves by using one data pointer as a source register and the other as the destination register.

DPTR0 is located at the same address as the original 8051 data pointer, allowing the DS80C390 to execute standard 8051 code with no modifications. The second data pointer, DPTR1, is split between the DPH1 and DPL1 SFRs, similar to the DPTR0 configuration. The active data pointer is selected with the data pointer select bit SEL (DPS.0). Any instructions that reference the DPTR (i.e., MOVX A, @DPTR), will select DPTR0 if SEL = 0, and DPTR1 if SEL = 1. Because the bits adjacent to SEL are not implemented, the state of SEL (and thus the active data pointer) can be quickly toggled by the `INC DPS` instruction without disturbing other bits in the DPS register.

Unlike the standard 8051, the DS80C390 has the ability to decrement as well as increment the data pointers without additional instructions. When the `INC DPTR` instruction is executed, the active DPTR increments or decrements according to the ID1, ID0 (DPS.7-6), and SEL (DPS.0) bits as shown. The inactive DPTR is not affected.

Table 10. Data Pointer Auto Increment/ Decrement Configuration

ID1	ID0	SEL	INC DPTR RESULT
X	0	0	Increment DPTR0
X	1	0	Decrement DPTR0
0	X	1	Increment DPTR1
1	X	1	Decrement DPTR1

Another useful feature of the device is its ability to automatically switch the active data pointer after a DPTR-based instruction is executed. This feature can greatly reduce the software overhead associated with data memory block moves, which toggle between the source and destination registers. When the toggle-select bit (TSL;DPS.5) is set to 1, the SEL bit (DPS.0) is automatically toggled every time one of the following DPTR-related instructions is executed.

POWER MANAGEMENT MODE (PMM) AND SWITCHBACK

Power consumption in PMM is less than in idle mode, and approximately one quarter of that consumed in divide-by-four mode. While PMM and Idle modes leave the power-hungry internal timers running, PMM runs all clocked functions such as timers at the rate of crystal divided by 1024, rather than crystal divided by 4. Even though instruction execution continues in PMM (albeit at a reduced speed), it still consumes less power than idle mode. As a result there is little reason to use idle mode in new designs.

When enabled, the switchback feature allows serial ports and interrupts to automatically switch back from divide by 1024 (PMM) to divide-by-4 (standard speed) operation. This feature makes it very convenient to use the PMM in real-time applications. Software can simply set the CD1 and CD0 clock control bits to the 4 clocks-per-cycle mode to exit PMM. However, the microcontroller provides hardware alternatives for automatic Switchback to standard speed (divide-by-4) operation.

Setting the SFR bit SWB (PMR.5) to 1 enables the switchback feature. Once it is enabled, and when PMM is selected, two possible events can cause an automatic switchback to divide-by-4 mode. First, if an interrupt occurs and is acknowledged, the system clock reverts from PMM to divide-by-4 mode. For example, if $\overline{\text{INT0}}$ is enabled and the CPU is not servicing a higher priority interrupt, then switchback occurs on $\overline{\text{INT0}}$. However, if $\overline{\text{INT0}}$ is not enabled or the CPU is servicing a higher priority interrupt, then activity on $\overline{\text{INT0}}$ does not cause switchback to occur.

A switchback can also occur when an enabled UART detects the start bit indicating the beginning of an incoming serial character or when the SBUF register is loaded initiating a serial transmission. Note that a serial character's start bit does not generate an interrupt. The interrupt occurs only on reception of a complete serial word. The automatic switchback on detection of a start bit allows timer hardware to return to divide-by-4 operation (and the correct baud rate) in time for a proper serial reception or transmission. So with switchback enabled and a serial port enabled, the automatic switch to divide-by-4 operation occurs in time to receive or transmit a complete serial character as if nothing special had happened.

STATUS

The status register (STATUS;C5h) provides information about interrupt and serial port activity to assist in determining if it is possible to enter PMM. The microprocessor supports three levels of interrupt priority: power-fail, high, and low. The PIP (power-fail priority interrupt status; STATUS.7), HIP (high-priority interrupt status; STATUS.6), and LIP (low-priority interrupt status; STATUS.5) status bits, when set to logic 1, indicate the corresponding level is in service.

Software should not rely on a lower-priority level interrupt source to remove PMM (switchback) when a higher level is in service. Check the current priority service level before entering PMM. If the current service level locks out a desired switchback source, then it would be advisable to wait until this condition clears before entering PMM. Alternately, software can prevent an undesired exit from PMM by intentionally entering a low priority interrupt service level before entering PMM. This will prevent other low priority interrupts from causing a switchback.

Entering PMM during an ongoing serial port transmission or reception can corrupt the serial port activity. To prevent this, a hardware lockout feature ignores changes to the clock divisor bits while the serial ports are active. Serial port activity can be monitored via the serial port activity bits located in the status register.

IDLE MODE

Setting the IDLE bit (PCON.0) invokes the idle mode. Idle leaves internal clocks, serial ports, and timers running. Power consumption drops because memory is not being accessed and instructions are not being executed. Since clocks are running, the idle power consumption is a function of crystal frequency. It should be approximately one-half of the operational power at a given frequency. The CPU can exit idle mode with any interrupt or a reset. Because PMM consumes less power than idle mode, as well as leaving timers and CPU operating, idle mode is no longer recommended for new designs, and is included for backward software compatibility only.

TIMED-ACCESS PROTECTION

Selected SFR bits are critical to operation, making it desirable to protect them against an accidental write operation. The timed-access procedure prevents an errant processor from accidentally altering bits that would seriously affect processor operation. The timed-access procedure requires that the write of a protected bit be immediately preceded by the following two instructions:

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

Writing an AAh followed by a 55h to the timed-access register (location C7h) opens a three-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 preceded by these instructions, the write is ignored. The protected bits are:

WDCON.6	POR	Power-On Reset Flag
WDCON.3	WDIF	Watchdog Interrupt Flag
WDCON.1	EWT	Watchdog Reset Enable
WDCON.0	RWT	Reset Watchdog Timer
RCON.0	BGS	Bandgap Select
ACON.2	SA	Stack Address Mode
ACON.1–0	AM1–AM0	Address Mode Select bits
MCON.7–6	IDM1–IDM0	Internal Memory Configuration and Location bits
MCON.5	CMA	CAN Data Memory Assignment
MCON.3–0	PDCE3–PDCE.0	Program/Data Chip Enables
C0C.3	CRST	CAN 0 Reset
C1C.3	CRST	CAN 1 Reset
P4CNT.6	SBCAN	Single Bus CAN
P4CNT.5–0		Port 4 Pin Configuration Control Bits
P5CNT.2–0	P5.7–P5.5	Configuration Control Bits
COR.7	IRDACK	IRDA Clock Output Enable
COR.6–5	C1BPR7–C1BPR6	CAN 1 Baud Rate Prescale Bits
COR.4–3	C0BPR7–C0BPR6	CAN 0 Baud Rate Prescale Bits
COR.2–1	COD1–COD0	CAN Clock Output Divide Bit 1 and Bit 0
COR.0	CLKOE	CAN Clock Output Enable

EMI REDUCTION

One of the major contributors to radiated noise in an 8051-based system is the toggling of ALE. The microcontroller allows software to disable ALE when not used by setting the ALEOFF (PMR.2) bit to 1. When ALEOFF = 1, ALE automatically toggles during an off-chip MOVX. However, ALE remains static when performing on-chip memory access. The default state of ALEOFF is 0 so ALE normally toggles at a frequency of XTAL/4.

PERIPHERAL OVERVIEW

The DS80C390 provides several of the most commonly needed peripheral functions in microcomputer-based systems. New functions include a second serial port, power-fail reset, power-fail interrupt flag, and a programmable watchdog timer. In addition, the microcontroller contains two CAN modules for industrial communication applications. Each of these peripherals is described in the following paragraphs. More details are available in the *High-Speed Microcontroller User's Guide* and the *DS80C390 Supplement*.

SERIAL PORTS

The microcontroller provides a serial port (UART) that is identical to the 80C52. In addition it includes a second hardware serial port that is a full duplicate of the standard one. This second port optionally uses pins P1.2 (RXD1) and P1.3 (TXD1). It has duplicate control functions included in new SFR locations. The second serial port can alternately be mapped to P5.2 and P5.3 to allow use of both serial ports in nonmultiplexed mode.

Both ports can operate simultaneously but can be at different baud rates or even in different modes. The second serial port has similar control registers (SCON1, SBUF1) to the original. The new serial port can only use Timer 1 for baud-rate generation.

POWER-FAIL RESET

The microcontroller incorporates an internal precision bandgap voltage reference and comparator circuit that provide a power-on and power-fail reset function. This circuit monitors the processor's incoming power supply voltage (V_{CC}), and holds the processor in reset while V_{CC} is below the minimum voltage level. When power exceeds the reset threshold, a full power-on reset is performed. In this way, this internal voltage monitoring circuitry handles both power-up and power-down conditions without the need for additional external components.

Once V_{CC} has risen above V_{RST}, the device automatically restarts the oscillator for the external crystal and counts 65,536 clock cycles before program execution begins at location 0000h. This helps the system maintain reliable operation by only permitting processor operation when the supply voltage is in a known good state. Software can determine that a power-on reset has occurred by checking the power-on reset flag (POR;WDCON.6). Software should clear the POR bit after reading it.

POWER-FAIL INTERRUPT

The bandgap voltage reference that sets a precise reset threshold also generates an optional early warning power-fail interrupt (PFI). When enabled by software, the processor vectors to ROM address 0033h if V_{CC} drops below V_{PFW}. PFI has the highest priority. The PFI enable is in the watchdog control SFR (EPFI;WDCON.5). Setting this bit to logic 1 enables the PFI. Application software can also read the PFI flag at WDCON.4. A PFI condition sets this bit to 1. The flag is independent of the interrupt enable and must be cleared by software.

EXTERNAL RESET PINS

The DS80C390 has reset input (RST) and reset output ($\overline{\text{RSTOL}}$) pins. The $\overline{\text{RSTOL}}$ pin supplies an active-low reset when the microprocessor is issued a reset from either a high on the RST pin, a timeout of the watchdog timer, a crystal oscillator fail, or an internally detected power fail. The timing of the $\overline{\text{RSTOL}}$ pin is dependent on the source of the reset.

RESET TYPE/SOURCE	$\overline{\text{RSTOL}}$ DURATION
Power-On Reset	65,536 t_{CLCL} (as described in *Power Cycle Timing Characteristics*)
External Reset	<1.25 machine cycles
Power Fail	65,536 t_{CLCL} (as described in *Power Cycle Timing Characteristics*)
Watchdog Timer Reset	2 machine cycles
Oscillator-Fail Detect	65,536 t_{CLCL} (as described in *Power Cycle Timing Characteristics*)

INTERRUPTS

The microcontroller provides 16 interrupt sources with three priority levels. All interrupts, with the exception of the power-fail interrupt, are controlled by a series combination of individual enable bits and a global interrupt-enable, EA (IE.7). Setting EA to 1 allows individual interrupts to be enabled. Clearing EA disables all interrupts regardless of their individual enable settings.

The three available priority levels are low, high, and highest. The highest priority level is reserved for the power-fail interrupt only. All other interrupt priority levels have individual priority bits that, when set to 1, establish the particular interrupt as high priority. In addition to the user-selectable priorities, each interrupt also has an inherent natural priority, used to determine the priority of simultaneously occurring interrupts. The available interrupt sources, their flags, their enables, their natural priority, and their available priority selection bits are identified in Table 13.

Table 13. Interrupt Summary

NAME	DESCRIPTION	VECTOR	NATURAL PRIORITY	FLAG BIT	ENABLE BIT	PRIORITY CONTROL BIT
PFI	Power-Fail Interrupt	33h	0	PFI (WDCON.4)	EPFI (WDCON.5)	N/A
INT0	External Interrupt 0	03h	1	IE0 (TCON.1)**	EX0 (IE.0)	PX0 (IP.0)
TF0	Timer 0	0Bh	2	TF0 (TCON.5)*	ET0 (IE.1)	PT0 (IP.1)
INT1	External Interrupt 1	13h	3	IE1 (TCON.3)**	EX1 (IE.2)	PX1 (IP.2)
TF1	Timer 1	1Bh	4	TF1 (TCON.7)*	ET1 (IE.3)	PT1 (IP.3)
SCON0	TI0 or RI0 from Serial Port 0	23h	5	RI_0 (SCON0.0); TI_0 (SCON0.1)	ES0 (IE.4)	PS0 (IP.4)
TF2	Timer 2	2Bh	6	TF2 (T2CON.7)	ET2 (IE.5)	PT2 (IP.7)
SCON1	TI1 or RI1 from Serial Port 1	3Bh	7	RI_1 (SCON1.0); TI_1 (SCON1.1)	ES1 (IE.6)	PS1 (IP.6)
INT2	External Interrupt 2	43h	8	IE2 (EXIF.4)	EX2 (EIE.0)	PX2 (EIP.0)
INT3	External Interrupt 3	4Bh	9	IE3 (EXIF.5)	EX3 (EIE.1)	PX3 (EIP.1)
INT4	External Interrupt 4	53h	10	IE4 (EXIF.6)	EX4 (EIE.2)	PX4 (EIP.2)
INT5	External Interrupt 5	5Bh	11	IE5 (EXIF.7)	EX5 (EIE.3)	PX5 (EIP.3)
C0I	CAN0 Interrupt	6Bh	12	various	C0IE (EIE.6)	C0IP (EIP.6)
C1I	CAN1 Interrupt	73h	13	various	C1IE (EIE.5)	C1IP (EIP.5)
WDTI	Watchdog Timer	63h	14	WDIF (WDCON.3)	EWDI (EIE.4)	PWDI (EIP.4)
CANBUS	CAN0/1 Bus Activity	7Bh	15	various	CANBIE (EIE.7)	CANBIP (EIP.7)

Unless marked, all flags must be cleared by the application software.
**Cleared automatically by hardware when the service routine is entered.*
***If edge-triggered, flag is cleared automatically by hardware when the service routine is entered. If level-triggered, flag follows the state of the interrupt pin.*

CONTROLLER AREA NETWORK (CAN) MODULE

The DS80C390 incorporates two CAN controllers that are fully compliant with the CAN 2.0B specification. CAN is a highly robust, high-performance communication protocol for serial communications. Popular in a wide range of applications including automotive, medical, heating, ventilation, and industrial control, the CAN architecture allows for the construction of sophisticated networks with a minimum of external hardware.

The CAN controllers support the use of 11-bit standard or 29-bit extended acceptance identifiers for up to 15 messages, with the standard 8-byte data field, in each message. Fourteen of the 15 message centers are programmable in either transmit or receive modes, with the 15th designated as a FIFO-buffered, receive-only message center to help prevent data overruns. All message centers support two separate 8-bit media masks and media arbitration fields for incoming message verification. This feature supports the use of higher-level protocols, which make use of the first and/or second byte of data as a part of the acceptance layer for storing incoming messages. Each message center can also be programmed independently to test incoming data with or without the use of the global masks.

Global controls and status registers in each CAN unit allow the microcontroller to evaluate error messages, generate interrupts, locate and validate new data, establish the CAN bus timing, establish identification mask bits, and verify the source of individual messages. Each message center is individually equipped with the necessary status and control bits to establish direction, identification mode (standard or extended), data field size, data status, automatic remote frame request and acknowledgment, and perform masked or non-masked identification acceptance testing.

MOVX MESSAGE CENTERS FOR CAN 0

CAN 0 CONTROL/STATUS/MASK REGISTERS

REGISTER	7	6	5	4	3	2	1	0	MOVX DATA ADDRESS[1]
C0MID0	MID07	MID06	MID05	MID04	MID03	MID02	MID01	MID00	xxxx00h
C0MA0	M0AA7	M0AA6	M0AA5	M0AA4	M0AA3	M0AA2	M0AA1	M0AA0	xxxx01h
C0MID1	MID17	MID16	MID15	MID14	MID13	MID12	MID11	MID10	xxxx02h
C0MA1	M1AA7	M1AA6	M1AA5	M1AA4	M1AA3	M1AA2	M1AA1	M1AA0	xxxx03h
C0BT0	SJW1	SJW0	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0	xxxx04h
C0BT1	SMP	TSEG26	TSEG25	TSEG24	TSEG13	TSEG12	TSEG11	TSEG10	xxxx05h
C0SGM0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	xxxx06h
C0SGM1	ID20	ID19	ID18	0	0	0	0	0	xxxx07h
C0EGM0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	xxxx08h
C0EGM1	ID20	ID19	ID18	ID17	ID16	ID15	ID14	ID13	xxxx09h
C0EGM2	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	xxxx0Ah
C0EGM3	ID4	ID3	ID2	ID1	ID0	0	0	0	xxxx0Bh
C0M15M0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	xxxx0Ch
C0M15M1	ID20	ID19	ID18	ID17	ID16	ID15	ID14	ID13	xxxx0Dh
C0M15M2	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	xxxx0Eh
C0M15M3	ID4	ID3	ID2	ID1	ID0	0	0	0	xxxx0Fh

CAN 0 MESSAGE CENTER 1

	Reserved								xxxx10h–11h
C0M1AR0	CAN 0 MESSAGE 1 ARBITRATION REGISTER 0								xxxx12h
C0M1AR1	CAN 0 MESSAGE 1 ARBITRATION REGISTER 1								xxxx13h
C0M1AR2	CAN 0 MESSAGE 1 ARBITRATION REGISTER 2								xxxx14h
C0M1AR3	CAN 0 MESSAGE 1 ARBITRATION REGISTER 3							WTOE	xxxx15h
C0M1F	DTBYC3	DTBYC2	DTBYC1	DTBYC0	T/$\overline{R}$	EX/$\overline{ST}$	MEME	MDME	xxxx16h
C0M1D0–7	CAN 0 MESSAGE 1 DATA BYTES 0–7								xxxx17h–1Eh
	Reserved								xxxx1Fh

CAN 0 MESSAGE CENTERS 2–14

	MESSAGE CENTER 2 REGISTERS (similar to Message Center 1)	xxxx20h–2Fh
	MESSAGE CENTER 3 REGISTERS (similar to Message Center 1)	xxxx30h–3Fh
	MESSAGE CENTER 4 REGISTERS (similar to Message Center 1)	xxxx40h–4Fh
	MESSAGE CENTER 5 REGISTERS (similar to Message Center 1)	xxxx50h–5Fh
	MESSAGE CENTER 6 REGISTERS (similar to Message Center 1)	xxxx60h–6Fh
	MESSAGE CENTER 7 REGISTERS (similar to Message Center 1)	xxxx70h–7Fh
	MESSAGE CENTER 8 REGISTERS (similar to Message Center 1)	xxxx80h–8Fh
	MESSAGE CENTER 9 REGISTERS (similar to Message Center 1)	xxxx90h–9Fh
	MESSAGE CENTER 10 REGISTERS (similar to Message Center 1)	xxxxA0h–AFh
	MESSAGE CENTER 11 REGISTERS (similar to Message Center 1)	xxxxB0h–BFh
	MESSAGE CENTER 12 REGISTERS (similar to Message Center 1)	xxxxC0h–CFh
	MESSAGE CENTER 13 REGISTERS (similar to Message Center 1)	xxxxD0h–DFh
	MESSAGE CENTER 14 REGISTERS (similar to Message Center 1)	xxxxE0h–EFh

CAN 0 MESSAGE CENTER 15

—	Reserved								xxxxF0h–F1h
C0M15AR0	CAN 0 MESSAGE 15 ARBITRATION REGISTER 0								xxxxF2h
C0M15AR1	CAN 0 MESSAGE 15 ARBITRATION REGISTER 1								xxxxF3h
C0M15AR2	CAN 0 MESSAGE 15 ARBITRATION REGISTER 2								xxxxF4h
C0M15AR3	CAN 0 MESSAGE 15 ARBITRATION REGISTER 3							WTOE	xxxxF5h
C0M15F	DTBYC3	DTBYC2	DTBYC1	DTBYC0	0	EX/$\overline{ST}$	MEME	MDME	xxxxF6h
C0M15D0–C0M15D7	CAN 0 MESSAGE 15 DATA BYTE 0–7								xxxxF7h–FEh
	Reserved								xxxxFFh

[1]The first two bytes of the CAN 0 MOVX memory address are dependent on the setting of the CMA bit (MCON.5) CMA = 0, xxxx = 00EE; CMA = 1, xxxx = 4010.

MESSAGE BUFFERING/OVERWRITE

If a message center is configured for reception (T/$\overline{R}$ = 0) and the previous message has not been read (DTUP = 1), then the disposition of an incoming message to that message center is controlled by the WTOE bit (located in CAN Arbitration Register 3 of each message center). When WTOE = 0, the incoming message is discarded and the current message is untouched.

If the WTOE bit is set, the incoming message is received and written over the existing data bytes in that message center. The receiver overwrite bit (ROW) is also set in the corresponding message center control register, located in SFR memory.

Message center 15 is unique in that it incorporates a buffer that can receive up to two messages without loss. If a message is received by message center 15 while it contains an unread message, the new incoming message is held in an internal buffer. When the CAN processor reads the message-center-15 memory location and then clears DTUP = INTRQ = EXTRQ = 0, the contents of the internal buffer is automatically loaded into the message-center-15 MOVX-memory location.

The message-center-15 WTOE bit controls what happens if a third message is received when both the message-center-15 MOVX-memory location and the buffer contain unread messages. If WTOE = 0, the new message is discarded, leaving the message-center-15 MOVX-memory location and the buffer untouched. If WTOE = 1, then the third message writes over the buffered message but leaves the message-center-15 MOVX-memory location untouched.

ERROR COUNTER INTERRUPT GENERATION

Each CAN module can be independently configured to alert the microprocessor when either 96 or 128 errors have been detected by the transmit or receive error counters. The error count select bit, ERCS (C0C.1 or C1C.1) selects whether the limit is 96 (ERCS = 0) or 128 (ERCS = 1) errors. When the error limit is exceeded, the CAN error count exceeded bit, CECE (C0S.6 or C1S.6), bit is set. If the ERIE, C0IE (or C1IE), and EA SFR bits are configured, an interrupt is generated. If the ERCS bit is set, the device generates an interrupt when the CECE bit is set or cleared, if the interrupt is enabled.

BIT TIMING

Bit timing of the CAN transmission can be adjusted per the CAN 2.0B specification. The CAN 0/1 bus timing register zero (C0BT0 and C1BT0)—located in the control/status/mask register block in MOVX memory—controls the PHASE_SEG1 and PHASE_SEG2 time segments as well as the baud-rate prescaler (BPR5–BPR0). The CAN 0/1 bus timing register one (C0BT1 and C1BT1) contains the controls for the sampling rate and the number of clock cycles assigned to the Phase Segment 1 and 2 portions of the nominal bit time. The values of both bus timing registers are automatically loaded into the CAN processor following each software change of the SWINT bit from a 1 to a 0 by the microcontroller. The bit timing parameters must be set before starting operation of the CAN processor. These registers can only be modified during a software initialization, (SWINT = 1), when the CAN processor is **NOT** in a bus-off mode, and after the removal of a system reset or a CAN reset. To avoid unpredictable behavior of the CAN processor, the software cannot clear the SWINT bit when TSEG1 and TSEG2 are both cleared to 0.

PACKAGE INFORMATION

For the latest package outline information and land patterns, go to **www.maxim-ic.com/packages**.

PACKAGE TYPE	PACKAGE CODE	DOCUMENT NO.
68 PLCC	Q68-1	**21-0049**
64 LQFP	C64L-2	**21-0083**

REVISION HISTORY

REVISION	DESCRIPTION
062299	Initial preliminary release.
090799	Clarifies that unused/unimplemented bits in the CAN MOVX SRAM read 0. Corrected the t_{MCS} time period table. Corrected multiplexed 2-cycle date memory $\overline{CEO-3}$ read figure to show $\overline{RD}$ and $\overline{WR}$ inactive.
110199	Corrected P5.2 and P5.3 pin descriptions. Corrected description of sequence to activate the crystal frequency multiplier. Corrected references to PQFP to read LQFP. Added $\overline{RSTOL}$ timing information.
032904	Official release (removed "preliminary" status). Abs max soldering temp now references JEDEC standard. AC and DC specifications updated to reflect final characterization data. Clarified DC characteristics Note 6 concerning port 4 and 5. Removed Figure 1. *Typical I_{CC} vs. Frequency.* Added t_{LLAX3} specification (identical to t_{LLAX2}). Clarified that t_{RLAZ} is held weak latch until overdriven by external memory. Removed t_{PXIZ}, t_{PHAV}, t_{PHWL}, and t_{PHRL} from nonmultiplexed address/data bus table. Corrected PSEN trace in Figure 10 to not show assertion during MOVX write. Corrected Table 3 to show unnecessary steps during 16/16 divide. Supplied approximate oscillator-fail detection frequency. Removed text references to Stop mode current. Corrected location of PT2 in Table 14.
022305	In *Absolute Maximum Ratings* section (page 2): Removed "A" from IPC/JEDEC J-STD-020A specification to support lead-free devices. In *DC Electrical Characteristics* table (page 2): Changed V_{PFW} MIN to 4.10V from 4.20V Changed V_{PFW} MAX to 4.60V from 4.55V Changed V_{RST} MIN to 3.85V from 3.95V Changed V_{RST} MAX to 4.35V from 4.3V Changed V_{IH2} MIN reference to 0.7 x V_{CC} from 0.7 x V_{DD} Added Note 10 In *AC Electrical Characteristics* table (page 3): Added note to (now) Note 11 that AC timing is characterized and guaranteed by design but is not production tested.
060805	Added lead-free part numbers to *Ordering Information* table.
110905	Added new paragraph to page 33 stating "Software must ensure that the input value for the normalize operation is not zero or the function will not complete. Compilers such as the one from Keil Software have updated their libraries and compensate for this condition." Table 3: clarified text under "Normalize" function. Changed "Configure MCNTO register as required." to "Load MCNT0 with 00h."

Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408-737-7600