Analog Devices Inc./Maxim Integrated - DS80C411-FNY+ 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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	75MHz
Connectivity	1-Wire®, CANbus, EBI/EMI, Ethernet, SIO, UART/USART
Peripherals	Power-Fail Reset, WDT
Number of I/O	64
Program Memory Size	64KB (64K x 8)
Program Memory Type	ROM
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/ds80c411-fny

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Input Pin Relative to Ground……..-0.5V to +5.5V
Voltage Range on Any Output Pin Relative to Ground……..-0.5V to (V_{CC3} + 0.5V)
Voltage Range on V_{CC3} Relative to Ground……..-0.5V to +3.6V
Voltage Range on V_{CC1} Relative to Ground……..-0.3V to +2.0V
Operating Temperature Range……..-40°C to +85°C
Junction Temperature……..+150°C max
Storage Temperature Range……...-55°C to +160°C
Soldering Temperature……See IPC/JEDEC J-STD-020 Standard

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods can affect device reliability.

DC ELECTRICAL CHARACTERISTICS

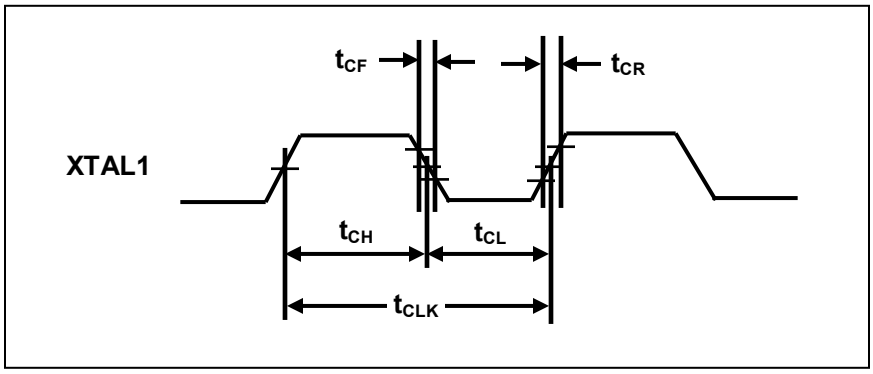
(V_{CC3} = 3.0V to 3.6V, V_{CC1} = 1.8V ±10%, T_A = -40°C to +85°C.) (Note 1)

	PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS
VCC3	Supply Voltage (V_{CC3}) (Note 2)	V_{CC3}	3.0	3.3	3.6	V
	Power-Fail Warning (V_{CC3}) (Note 3)	V_{PFW3}	2.85	3.00	3.15	V
	Power-Fail Reset Voltage (V_{CC3}) (Note 3)	V_{RST3}	2.76	2.90	3.05	V
	Active Mode Current (V_{CC3}) (Note 4)	I_{CC3}		16	35	mA
	Idle Mode Current (V_{CC3}) (Note 4)	I_{IDLE3}		7	15	mA
	Stop Mode Current (V_{CC3}) (Not 4)	I_{STOP3}		1	10	µA
	Stop Mode Current, Bandgap Enabled (V_{CC3}) (Note 4)	I_{SPBG3}		100	150	µA
VCC1	Supply Voltage (V_{CC1}) (Note 2)	V_{CC1}	1.62	1.8	1.98	V
	Power-Fail Warning (V_{CC1}) (Note 5)	V_{PFW1}	1.52	1.60	1.68	V
	Power-Fail Reset Voltage (V_{CC1}) (Note 5)	V_{RST1}	1.47	1.55	1.63	V
	Active Mode Current (V_{CC1}) (Note 4)	I_{CC1}		30	60	mA
	Idle Mode Current (V_{CC1}) (Note 4)	I_{IDLE1}		20	50	mA
	Stop Mode Current (V_{CC1}) (Note 4)	I_{STOP1}		3	20	mA
	Stop Mode Current, Bandgap Enabled (V_{CC1}) (Note 4)	I_{SPBG1}		3	20	mA
Input Low Level		V_{IL1}			0.8	V
Input Low Level for XTAL1, RST, OW		V_{IL2}			1.0	V
Input High Level		V_{IH1}	2.0			V
Input High Level for XTAL1, RST, OW		V_{IH2}	2.4			V
Output Low Current for Port 1, 3–7 at V_{OL} = 0.4V		I_{OL1}	6	10		mA
Output Low Current for Port 0, 2, TX_EN, TXD[3:0], MDC, MDIO, $\overline{\text{RSTOL}}$, ALE, $\overline{\text{PSEN}}$, and Ports 3–7 (when used as any of the following: A21–A0, $\overline{\text{WR}}$, $\overline{\text{RD}}$, $\overline{\text{CE0-7}}$, $\overline{\text{PCE0-3}}$) at V_{OL} = 0.4V (Note 6)		I_{OL2}	12	20		mA
Output Low Current for OW, $\overline{\text{OWSTP}}$ at V_{OL}= 0.4V		I_{OL3}	10	16		mA
Output High Current for Port 1, 3–7 at V_{OH} = V_{CC3} - 0.4V (Note 7)		I_{OH1}		-75	-50	µA
Output High Current for Port 1, 3–7 at V_{OH}= V_{CC3} - 0.4V (Note 8)		I_{OH2}		-8	-4	mA
Output High Current for Port 0, 2, TX_EN, TXD[3:0], MDC, MDIO, $\overline{\text{RSTOL}}$, ALE, $\overline{\text{PSEN}}$, and Ports 3–7 (when used as any of the following: A21–A0, $\overline{\text{WR}}$, $\overline{\text{RD}}$, $\overline{\text{CE0-7}}$, $\overline{\text{PCE0-3}}$) at V_{OH} = V_{CC3} - 0.4V (Notes 6, 9)		I_{OH3}		-16	-8	mA
Input Low Current for Port 1–7 at 0.4V (Note 10)		I_{IL}	-50	-20	-10	µA
Logic 1-to-0 Transition Current for Port 1, 3–7 (Note 11)		I_{TL}	-650	-400		µA
Input Leakage Current, Port 0 Bus Mode, V_{IL} = 0.8V (Note 12)		I_{TH0}	20	50	200	µA
Input Leakage Current, Port 0 Bus Mode, V_{IH} = 2.0V (Note 12)		I_{TL0}	-200	-50	-20	µA
Input Leakage Current, Input Mode (Note 13)		I_L	-10	0	10	µA
RST Pulldown Resistance		R_{RST}	50	100	200	kΩ

EXTERNAL CLOCK OSCILLATOR (XTAL1) CHARACTERISTICS

PARAMETER	SYMBOL	MIN	MAX	UNITS
Clock Oscillator Period	t_{CLK}	See *External Clock Oscillator Frequency*		
Clock Symmetry at 0.5 x V_{CC3}	t_{CH}	0.45 t_{CLK}	0.55 t_{CLK}	ns
Clock Rise Time	t_{CR}		3	ns
Clock Fall Time	t_{CF}		3	ns

EXTERNAL CLOCK DRIVE



SYSTEM CLOCK TIME PERIODS (t_{CLCL}, t_{CHCL}, t_{CLCH})

SYSTEM CLOCK SELECTION			SYSTEM CLOCK PERIOD t_{CLCL}	SYSTEM CLOCK HIGH (t_{CHCL}) AND SYSTEM CLOCK LOW (t_{CLCH})	
4X/$\overline{2X}$	CD1	CD0		MIN	MAX
1	0	0	t_{CLK} / 4	0.45 (t_{CLK} / 4)	0.55 (t_{CLK} / 4)
0	0	0	t_{CLK} / 2	0.45 (t_{CLK} / 2)	0.55 (t_{CLK} / 2)
X	1	0	t_{CLK}	0.45 t_{CLK}	0.55 t_{CLK}
X	1	1	256 t_{CLK}	0.45 (256 t_{CLK})	0.55 (256 t_{CLK})

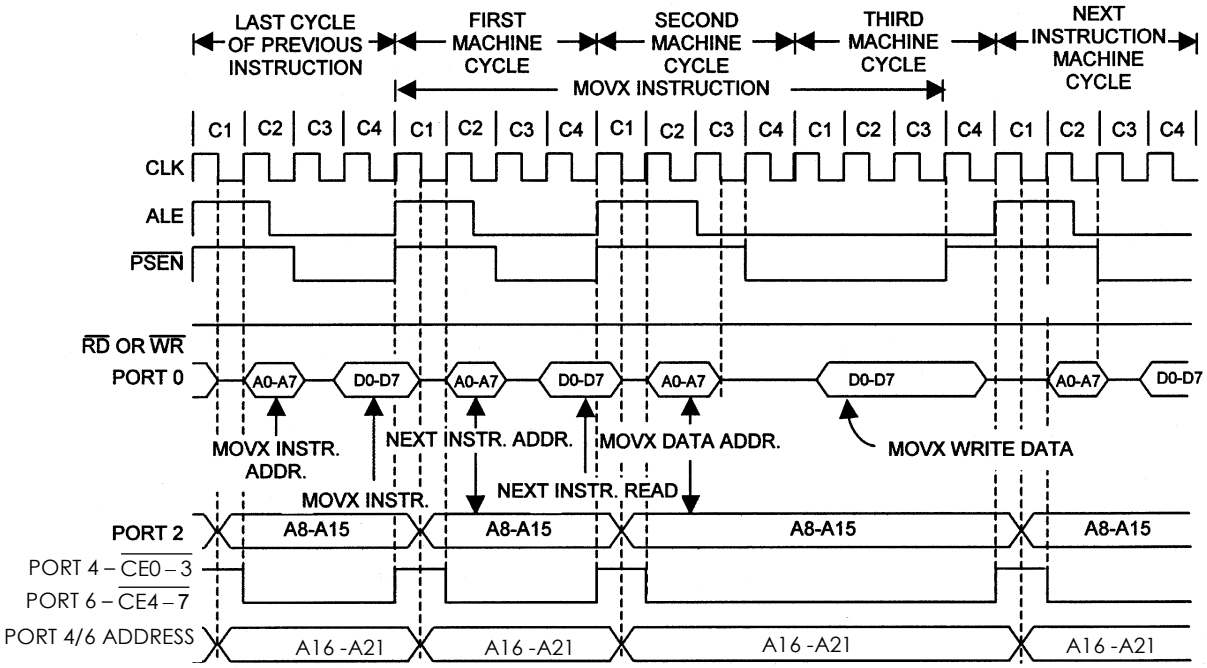
Note 1: Figure 21 shows a detailed description and illustration of the system clock selection.
Note 2: When an external clock oscillator is used in conjunction with the default system clock selection (CD1:CD0 = 10b), the minimum/maximum system clock high (t_{CHCL}) and system clock low (t_{CLCH}) periods are directly related to clock oscillator duty cycle.

MOVX CHARACTERISTICS (MULTIPLEXED ADDRESS/DATA BUS) (Note 1)

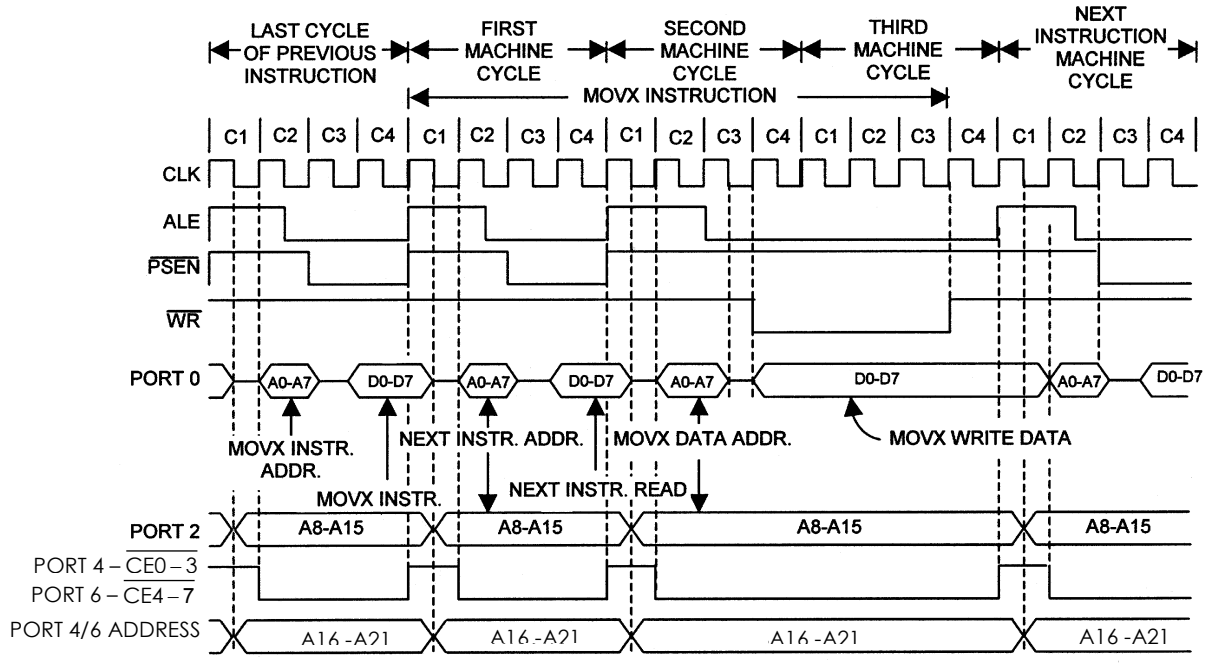
(V_{CC3} = 3.0V to 3.6V, V_{CC1} = 1.8V ±10%, T_A = -40 °C to +85°C.)

PARAMETER	SYMBOL	MIN	MAX	UNITS	STRETCH VALUES C_{ST} (MD2:0)
MOVX ALE Pulse Width	t_{LHLL2}	t_{CLCL} + t_{CHCL} - 5		ns	C_{ST} = 0
		2t_{CLCL} - 5			1≤ C_{ST} ≤ 3
		6t_{CLCL} - 5			4 ≤ C_{ST} ≤ 7
Port 0 MOVX Address Valid to ALE Low	t_{AVLL2}	t_{CHCL} - 5		ns	C_{ST} = 0
		t_{CLCL} - 6			1≤ C_{ST} ≤ 3
		5t_{CLCL} - 6			4 ≤ C_{ST} ≤ 7
Port 0 MOVX Address Hold after ALE Low	t_{LLAX2} and t_{LLAX3}	t_{CLCH} - 2		ns	C_{ST} = 0
		t_{CLCL} - 2			1≤ C_{ST} ≤3
		5t_{CLCL} - 2			4 ≤ C_{ST} ≤ 7
$\overline{RD}$ Pulse Width (P3.7 or $\overline{PSEN}$)	t_{RLRH}	2t_{CLCL} - 5		ns	C_{ST} = 0
		(4 x C_{ST}) t_{CLCL} - 3			1 ≤ C_{ST} ≤ 7
$\overline{WR}$ Pulse Width (P3.6)	t_{WLWH}	2t_{CLCL} - 5		ns	C_{ST} = 0
		(4 x C_{ST}) t_{CLCL} - 3			1 ≤ C_{ST} ≤ 7
$\overline{RD}$ (P3.7 or $\overline{PSEN}$) Low to Valid Data In	t_{RLDV}		2t_{CLCL} - 18	ns	C_{ST} = 0
			(4 x C_{ST}) t_{CLCL} - 18		1 ≤ C_{ST} ≤ 7
Data Hold After $\overline{RD}$ (P3.7 or $\overline{PSEN}$) High	t_{RHDX}	-2		ns	

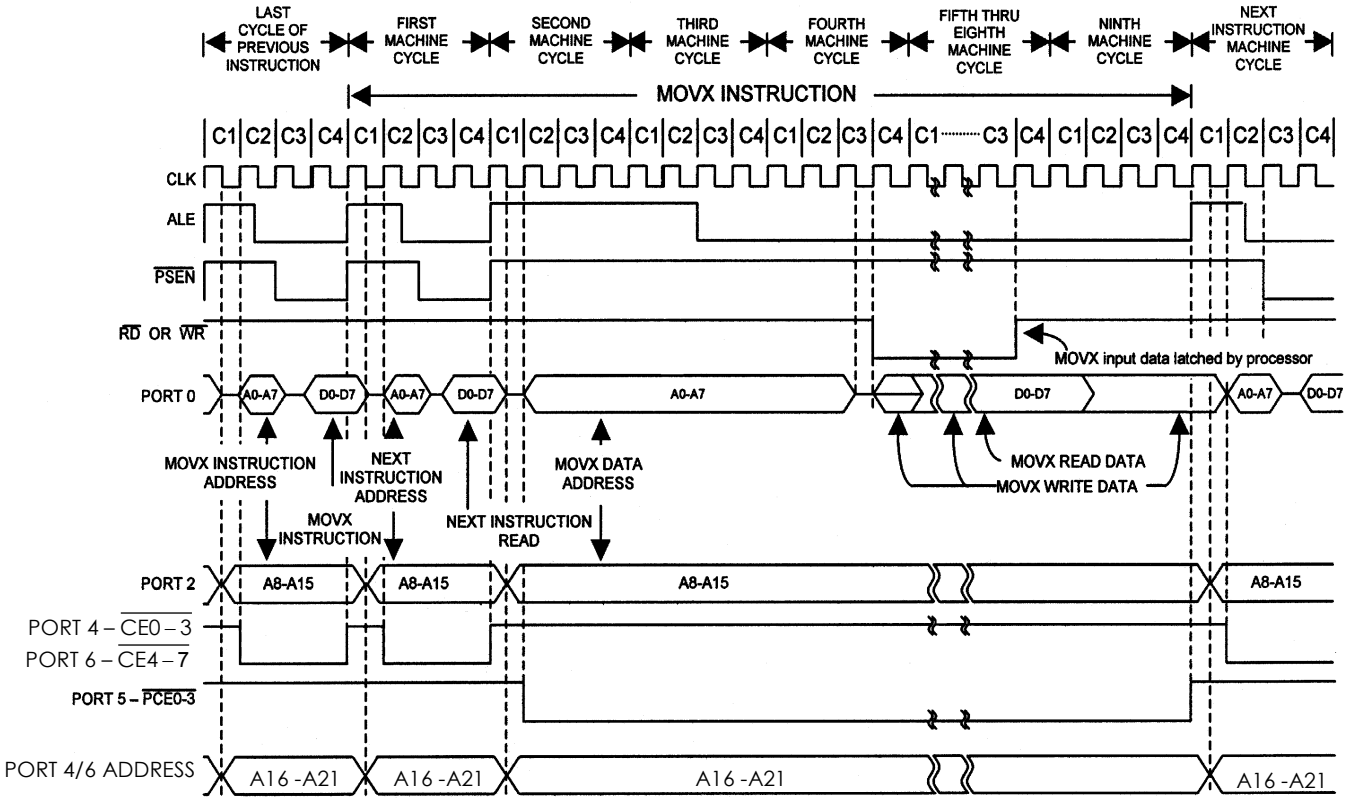
MULTIPLEXED, 3-CYCLE DATA MEMORY $\overline{\text{CE0-7}}$ READ



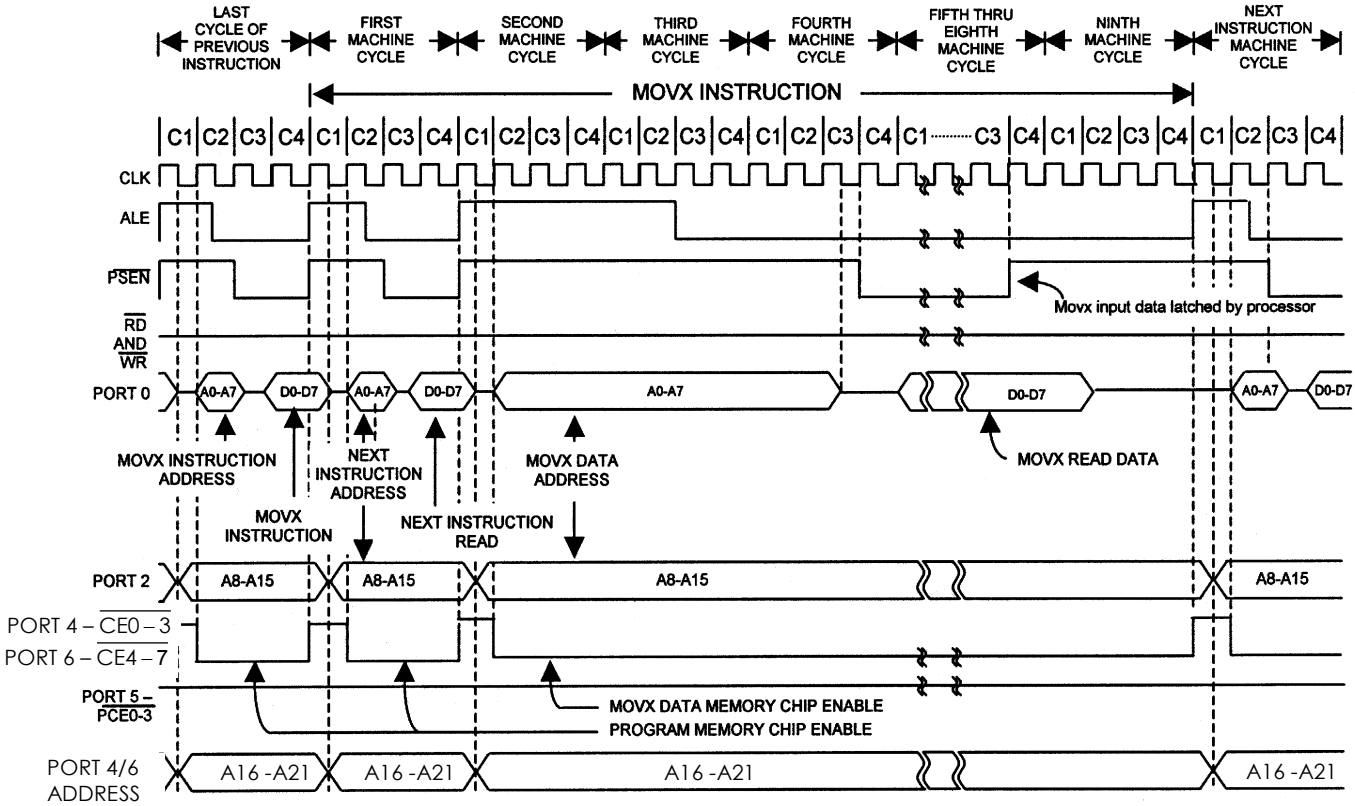
MULTIPLEXED, 3-CYCLE DATA MEMORY $\overline{\text{CE0-7}}$ WRITE



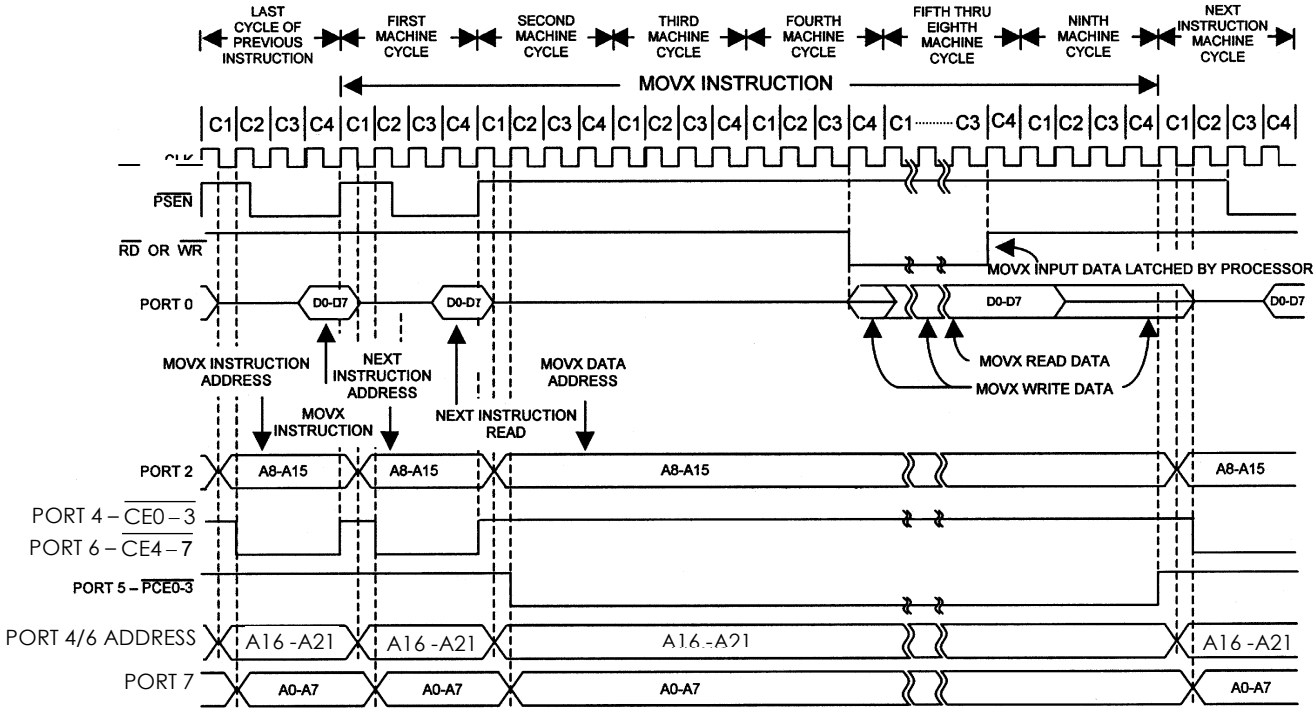
MULTIPLEXED, 9-CYCLE DATA MEMORY $\overline{\text{PCE0-3}}$ READ OR WRITE



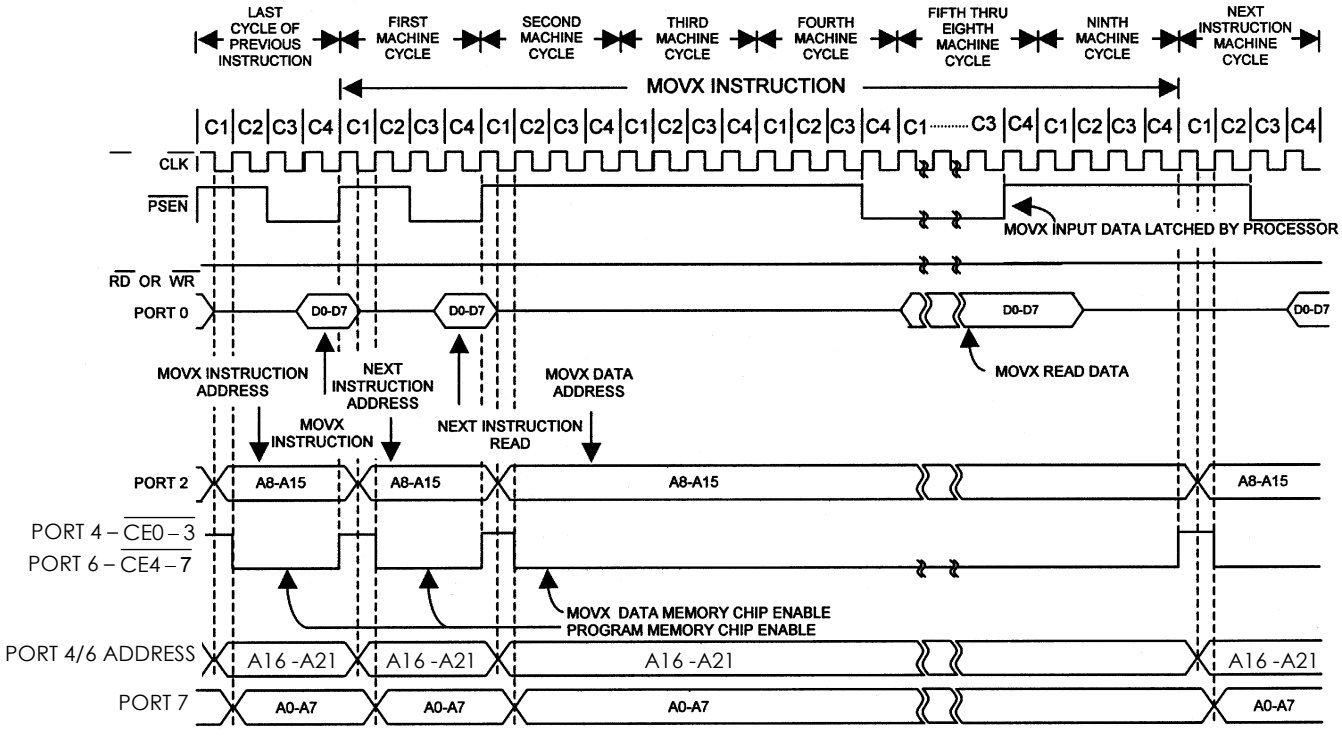
MULTIPLEXED, 9-CYCLE DATA MEMORY $\overline{\text{CE0-7}}$ READ



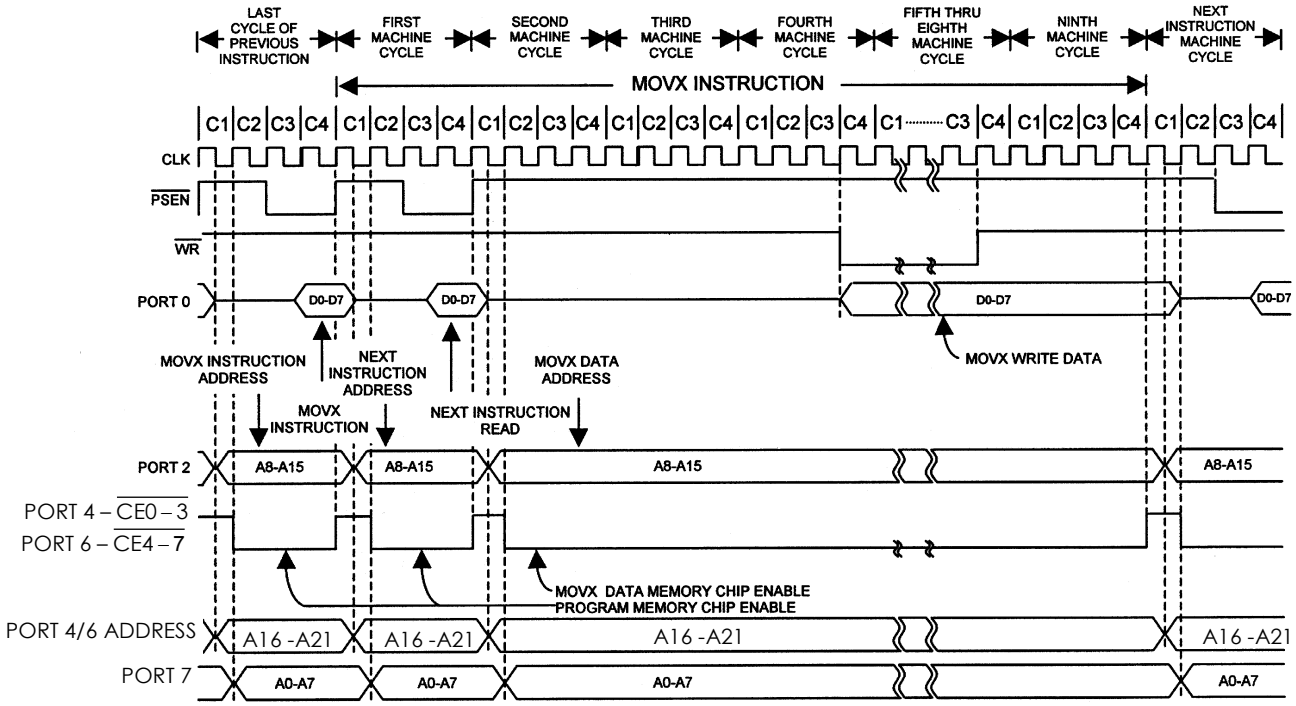
NONMULTIPLEXED, 9-CYCLE DATA MEMORY $\overline{\text{PCE0-3}}$ READ OR WRITE



NONMULTIPLEXED, 9-CYCLE DATA MEMORY $\overline{\text{CE0-7}}$ READ



NONMULTIPLEXED, 9-CYCLE DATA MEMORY $\overline{CE0\text{-}7}$ WRITE



OW PIN TIMING CHARACTERISTICS (Note 1)

(V_{CC3} = 3.0V to 3.6V, V_{CC1} = 1.8V ±10%, T_A = -40°C to +85°C.)

PARAMETER	SYMBOL	STANDARD		OVERDRIVE		LONGLINE		UNITS
		MIN	MAX	MIN	MAX	MIN	MAX	
Transmit Reset Pulse Low Time (Note 2)	t_{RSTL}	500.8	626	50.4	63	500.8	626	µs
Transmit Reset Pulse High Time (Note 2)	t_{RSTH}	508.8	636	59.2	74	508.8	636	µs
Wait Time for Transmit of Presence Pulse (Notes 2, 3)	t_{PDH}	15	60	2	6	15	60	µs
Wait Time for Absence of Presence Pulse (Notes 2, 4)	t_{PDHCNT}	60	75	6.4	8	60	75	µs
Presence Pulse Width (Note 2)	t_{PDL}	60	240	8	24	60	240	µs
Presence Pulse Sampling Time (Note 2)	t_{PDS}	24	31	2.4	4	30.4	38	µs
Read/Write Data Time Slot	t_{SLOT}	68.8	86	12	15	68.8	86	µs
Low Time for Write 1	t_{LOW1}	4.8	6	0.8	1	7.2	9	µs
Low Time for Write 0	t_{LOW0}	62.4	78	8	10	62.4	78	µs
Write Data Sampling Time	t_{WDV}	15	60	2	6	25	60	µs
Read Data Sampling Time	t_{RDV}	12	15	1.6	2	20	25	µs

Note 1: AC electrical characteristics assume 50% duty cycle for the oscillator, oscillator frequency ≤ 75MHz, and are not 100% production tested, but are guaranteed by design.

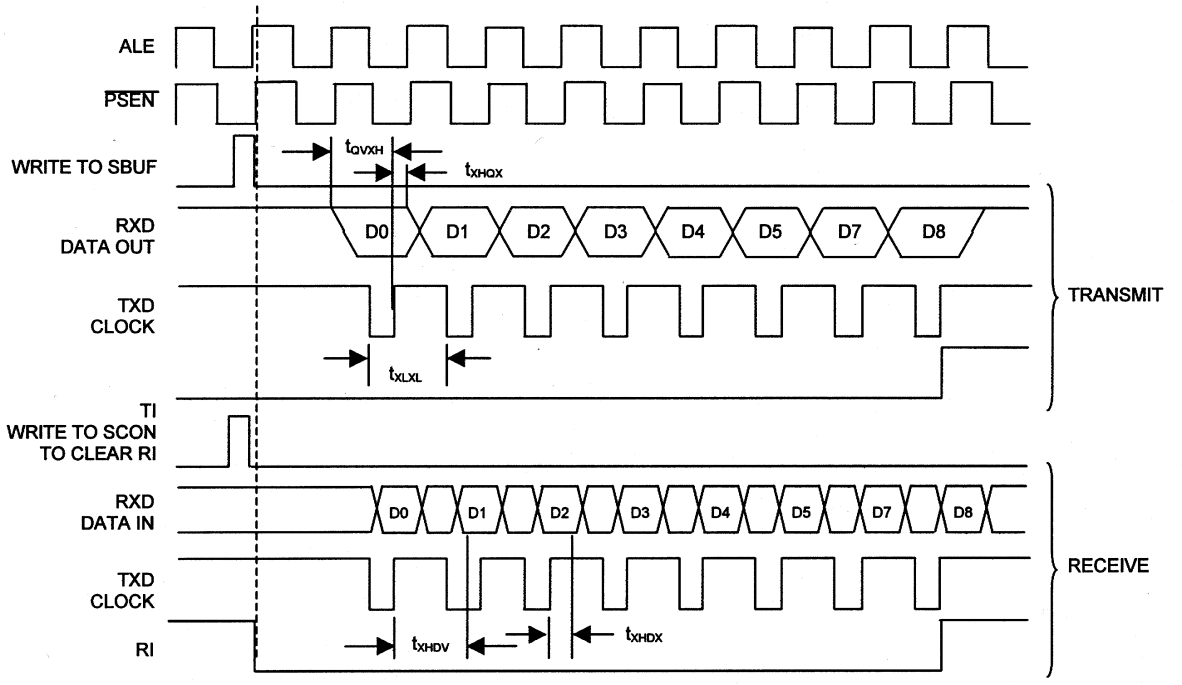
Note 2: In PMM mode, the master pulls the line low after the first 15µs for the remainder of the standard speed 1-Wire routine.

Note 3: This parameter quantifies the wait time for the slave devices to respond to the reset pulse and is dependent on the slave device timing.

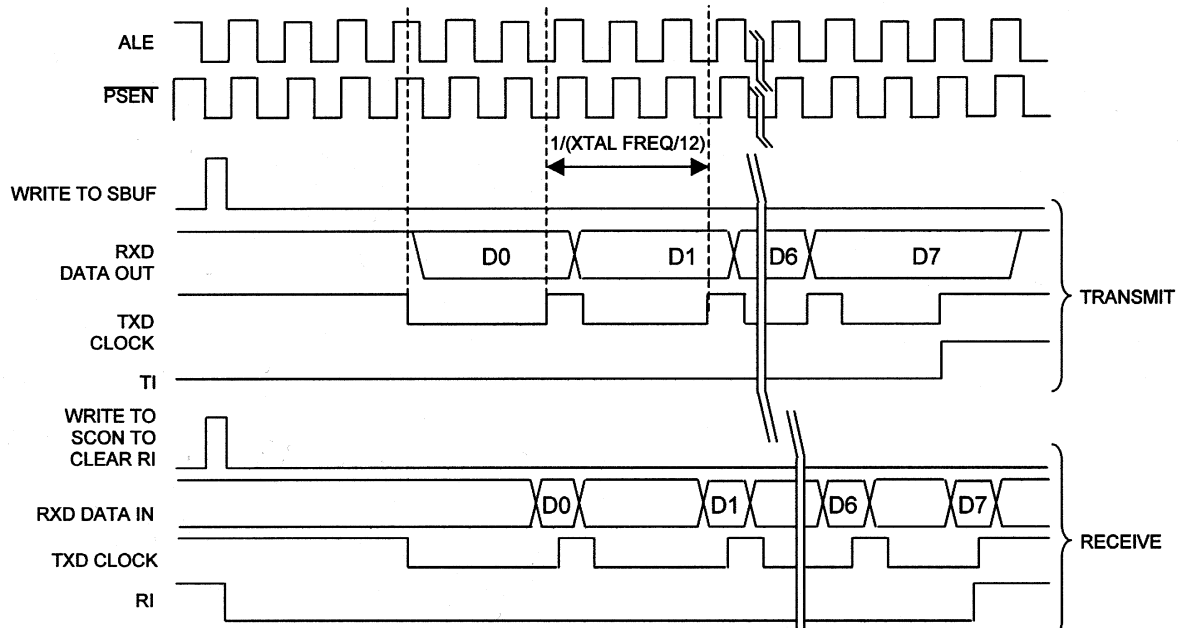
Note 4: This parameter quantifies the wait time for the case when no presence pulse detected.

Note 5: The maximum timing figures shown apply only when an exact 1-Wire clock frequency can be achieved from the microcontroller input clock.

SERIAL PORT 0 (SYNCHRONOUS MODE)



HIGH-SPEED OPERATION, TXD CLK = SYSCLK/4 (SM2 = 1)



TRADITIONAL 8051 OPERATION, TXD CLOCK = XTAL/12 (SM2 = 0)

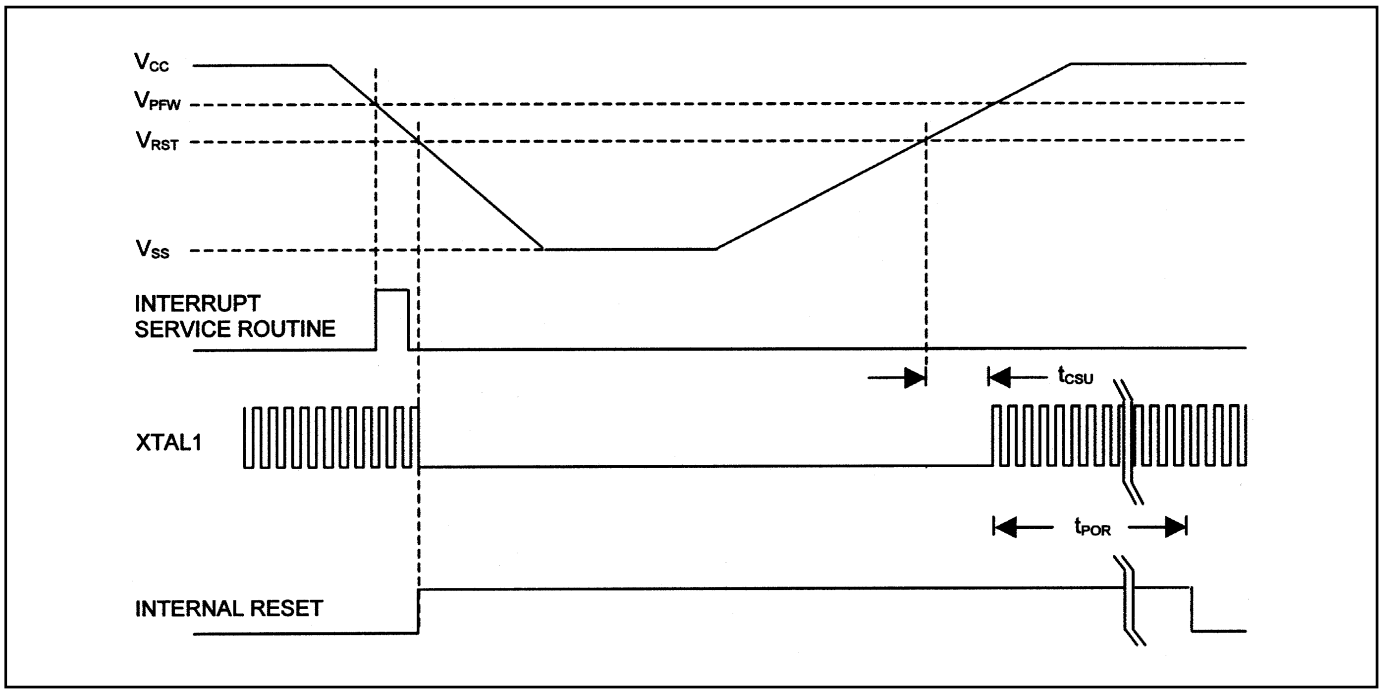
POWER-CYCLE TIMING CHARACTERISTICS

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS
Crystal Startup Time (Note 1)	t_{CSU}		1.8		ms
Power-On Reset Delay (Note 2)	t_{POR}			65,536	t_{CLCK}

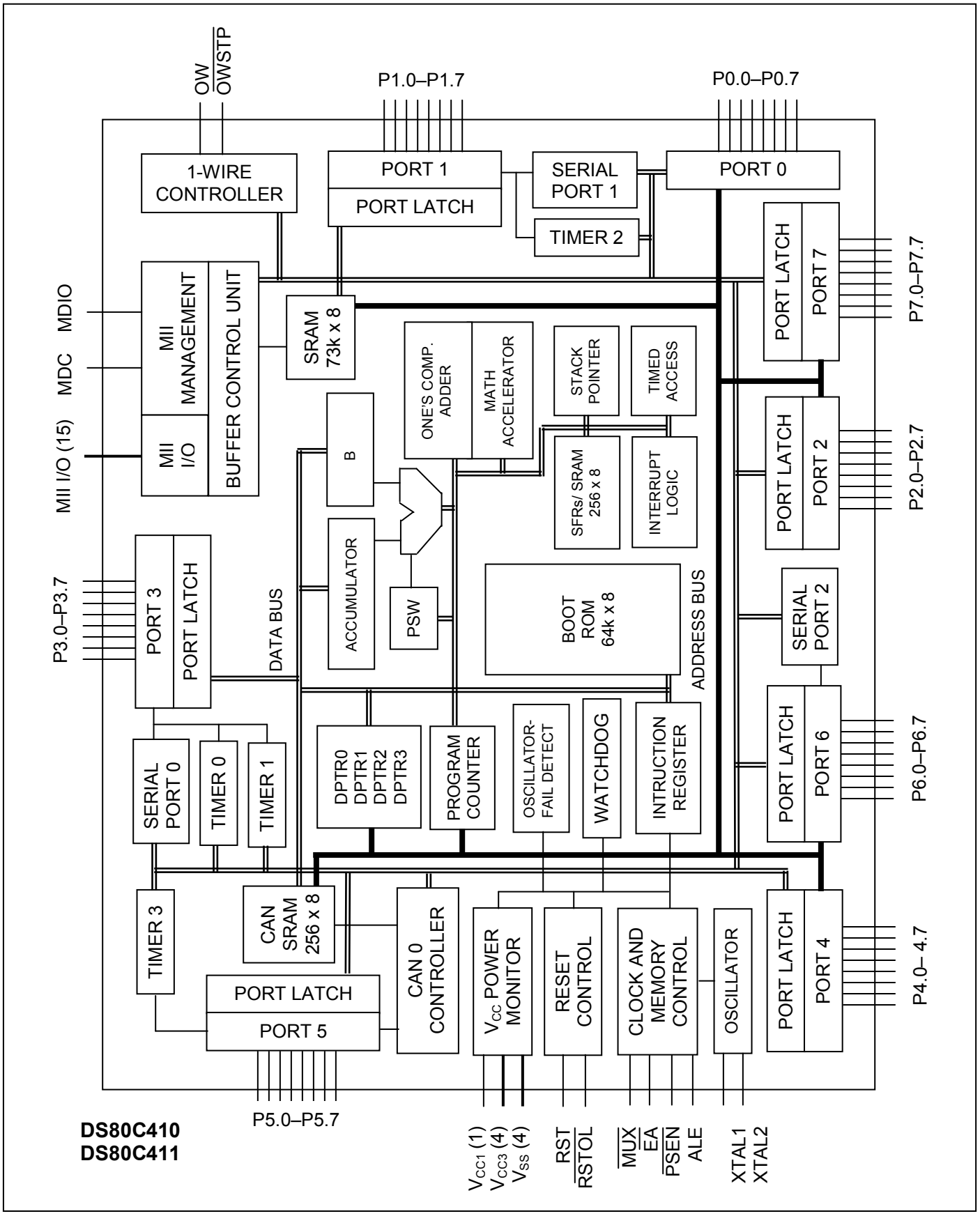
Note 1: Startup time for crystals varies with load capacitance and manufacturer. Time shown is for an 11.0592MHz crystal manufactured by Fox Electronics.

Note 2: Reset delay is a synchronous counter of crystal oscillations during crystal startup. Counting begins when the level on the XTAL1 input meets the V_{IH2} criteria. At 40MHz, this time is approximately 1.64ms.

POWER-CYCLE TIMING



BLOCK DIAGRAM



PIN	NAME	FUNCTION
19	MDIO	**MII Management Input/Output.** The MII management I/O is the data pin for serial communication with the external Ethernet PHY controller. In a read cycle, data is driven by the PHY to the MAC synchronously with respect to the MDC clock. In a write cycle, data from the MAC is output to the external PHY synchronously with respect to the MDC clock.
99	OW	**1-Wire Data, I/O.** The 1-Wire data pin is an open-drain, bidirectional data bus for the 1-Wire Bus Master. External 1-Wire slave devices are connected to this pin. This pin must be pulled high by an external resistor, normally 2.2kΩ.
100	$\overline{\text{OWSTP}}$	**Strong Pullup Enable, Output.** This 1-Wire pin is an open-drain active-low output used to enable an external strong pullup for the 1-Wire bus. This pin must be pulled high by an external resistor, normally 10kΩ. This functionality helps recovery times when the 1-Wire bus is operated in overdrive and long-line standard communication modes. It can optionally be enabled while the bus master is in the idle state for slave devices requiring sustained high-current operation.

FEATURES (*continued*)

- **Full-Function CAN 2.0B Controller**
  - 15 Message Centers
  - Supports Standard (11-Bit) and Extended (29-Bit) Identifiers and Global Masks
  - Media Byte Filtering to Support DeviceNet™, SDS, and Higher Layer CAN Protocols
  - Auto-Baud Mode and SIESTA Low-Power Mode
- **Integrated Primary System Logic**
  - 16 Total Interrupt Sources with Six External
  - Four 16-Bit Timer/Counters
  - 2x/4x Clock Multiplier Reduces Electromagnetic Interference (EMI)
  - Programmable Watchdog Timer
  - Oscillator-Fail Detection
  - Programmable IrDA Clock
- **Advanced Power Management**
  - Energy Saving 1.8V Core
  - 3.3V I/O Operation, 5V Tolerant
  - Power-Management, Idle, and Stop Mode Operations with Switchback Feature
  - Ethernet and CAN Shutdown Control for Power Conservation
  - Early Warning Power-Fail Interrupt
  - Power-Fail Reset
- **Enhanced Memory Architecture**
  - Selectable 8/10-Bit Stack Pointer for High-Level Language Support
  - 64kBytes Additional On-Chip SRAM Usable as Program/Data Memory
  - 16-Bit/24-Bit Paged/24-Bit Contiguous Modes
  - Selectable Multiplexed/Nonmultiplexed External Memory Interface
  - Merged Program/Data Memory Space Allows In-System Programming
  - Defaults to True 8051-Memory Compatibility

DeviceNet is a trademark of Open DeviceNet Vendor Association, Inc.

Table 2. SFR Reset Values

REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	ADDRESS
P4	1	1	1	1	1	1	1	1	80h
SP	0	0	0	0	0	0	0	0	81h
DPL	0	0	0	0	0	0	0	0	82h
DPH	0	0	0	0	0	0	0	0	83h
DPL1	0	0	0	0	0	0	0	0	84h
DPH1	0	0	0	0	0	0	0	0	85h
DPS	0	0	0	0	0	0	0	0	86h
PCON	0	0	Special	0	0	0	0	0	87h
TCON	0	0	0	0	0	0	0	0	88h
TMOD	0	0	0	0	0	0	0	0	89h
TL0	0	0	0	0	0	0	0	0	8Ah
TL1	0	0	0	0	0	0	0	0	8Bh
TH0	0	0	0	0	0	0	0	0	8Ch
TH1	0	0	0	0	0	0	0	0	8Dh
CKCON	0	0	0	0	0	0	0	1	8Eh
P1	1	1	1	1	1	1	1	1	90h
EXIF	0	0	0	0	Special	Special	Special	0	91h
P4CNT	1	1	1	1	1	1	1	1	92h
DPX	0	0	0	0	0	0	0	0	93h
DPX1	0	0	0	0	0	0	0	0	95h
C0RMS0	0	0	0	0	0	0	0	0	96h
C0RMS1	0	0	0	0	0	0	0	0	97h
SCON0	0	0	0	0	0	0	0	0	98h
SBUF0	0	0	0	0	0	0	0	0	99h
ESP	1	1	1	1	1	1	0	0	9Bh
AP	0	0	0	0	0	0	0	0	9Ch
ACON	1	1	0	0	Special	0	0	0	9Dh
C0TMA0	0	0	0	0	0	0	0	0	9Eh
C0TMA1	0	0	0	0	0	0	0	0	9Fh
P2	1	1	1	1	1	1	1	1	A0h
P5	1	1	1	1	1	1	1	1	A1h
P5CNT	1	0	0	0	0	0	0	0	A2h
C0C	0	0	0	0	1	0	0	1	A3h
C0S	0	0	0	0	0	0	0	0	A4h
C0IR	0	0	0	0	0	0	0	0	A5h
C0TE	0	0	0	0	0	0	0	0	A6h
C0RE	0	0	0	0	0	0	0	0	A7h
IE	0	0	0	0	0	0	0	0	A8h
SADDR0	0	0	0	0	0	0	0	0	A9h
SADDR1	0	0	0	0	0	0	0	0	AAh
C0M1C	0	0	0	0	0	0	0	0	ABh
C0M2C	0	0	0	0	0	0	0	0	ACh
C0M3C	0	0	0	0	0	0	0	0	ADh
C0M4C	0	0	0	0	0	0	0	0	AEh
C0M5C	0	0	0	0	0	0	0	0	AFh
P3	1	1	1	1	1	1	1	1	B0h
P6	1	1	1	1	1	1	1	1	B1h
P6CNT	0	0	0	0	0	0	0	0	B2h
C0M6C	0	0	0	0	0	0	0	0	B3h
C0M7C	0	0	0	0	0	0	0	0	B4h
C0M8C	0	0	0	0	0	0	0	0	B5h
C0M9C	0	0	0	0	0	0	0	0	B6h
C0M10C	0	0	0	0	0	0	0	0	B7h
IP	1	0	0	0	0	0	0	0	B8h
SADEN0	0	0	0	0	0	0	0	0	B9h
SADEN1	0	0	0	0	0	0	0	0	BAh
C0M11C	0	0	0	0	0	0	0	0	BBh
C0M12C	0	0	0	0	0	0	0	0	BCh
C0M13C	0	0	0	0	0	0	0	0	BDh
C0M14C	0	0	0	0	0	0	0	0	BEh
C0M15C	0	0	0	0	0	0	0	0	BFh
SCON1	0	0	0	0	0	0	0	0	C0h
SBUF1	0	0	0	0	0	0	0	0	C1h
PMR	1	0	0	0	0	0	1	1	C4h

ADDRESSING MODES

Three different addressing modes are supported, as selected by the AM1, AM0 bits in the address control (ACON; 9Dh) SFR.

AM1:0	ADDRESS MODE
00b	16-bit (default when internal ROM disabled)
01b	24-bit paged
1xb	24-bit contiguous (default if internal ROM enabled)

16-Bit Address Mode

The 16-bit address mode accesses memory in a similar manner as a traditional 8051. It is op-code compatible with the 8051 microprocessor and identical to the byte and cycle count of the Maxim high-speed microcontroller family. A device operating in this mode can access up to 64kB of program and data memory. The DS80C410 defaults to this mode following any reset.

24-Bit Paged Address Mode

The 24-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 the Maxim 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.

24-Bit Contiguous Address Mode

The 24-bit contiguous addressing mode uses a full 24-bit program counter, and all modified branching instructions automatically save and restore the entire program counter. The 24-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.

Visit www.maxim-ic.com/microcontrollers for a list of tools that support the DS80C410.

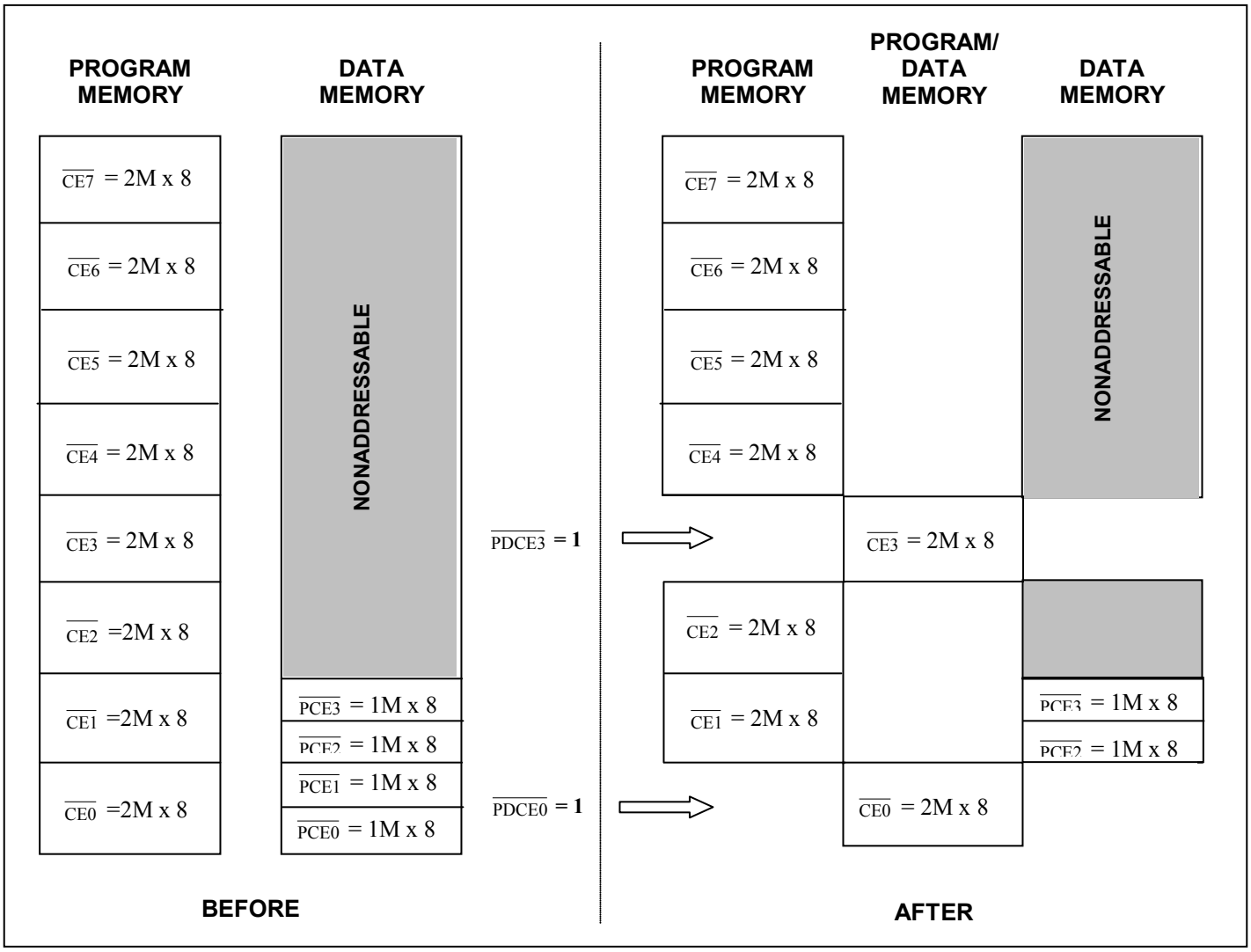
Extended Address Generation

FUNCTION	ADDRESS BITS 23–16	ADDRESS BITS 15–8	ADDRESS BITS 7–0
MOVX Instructions Using DPTRn	DPXn	DPHn	DPLn
MOVX Instructions Using @Ri	MXAX;EAh	P2;A0h	Ri
Addressing Program Memory In 24-Bit Paged Mode	AP;9Ch	—	—
10-Bit Stack Pointer Mode	—	ESP;9Bh	SP;81h

External Program Memory Addressing

Since the DS80C410 is not bound to the 8051's traditional 16-bit address mode, on-chip hardware enhancements were made to accommodate the larger memory interfaces associated with 24-bit addressing. The DS80C410 provides SFR bits to configure certain port pins as upper address lines and chip enables. The Port 4 control register (P4CNT; 92h) and Port 6 control register (P6CNT; B2h) control the number of chip enables that are used and the maximum amount of program memory that can be accessed per chip enable. Tables 3 and 4 illustrate which port pins are converted to address lines or chip enables as a result of the P4CNT and P6CNT bit settings.

Figure 1. Example External Memory Map—Merged Program/Data



Enhanced Quad Data Pointers

The DS80C410 offers enhanced features for accelerating the access and movement of data. It contains four data pointers (DPTR0, DPTR1, DPTR2, and DPTR3), in comparison to the single data pointer offered on the original 8051, and allows the user to define, for each data pointer, whether the INC DPTR instruction increments or decrements the selected pointer. Also, realizing that many data accesses occur in large contiguous blocks, the DS80C410 can be configured to automatically increment or decrement a data pointer on execution of certain instructions. This improvement greatly speeds access to consecutive pieces of data since hardware can now accomplish a task (advancing the data pointer) that previously required software execution time. Finally, each pair of data pointers (DPTR0, DPTR1 or DPTR2, DPTR3) can be configured for an auto-toggle mode. When placed into this mode, certain data pointer-related instructions toggle the active data-pointer selection to the other pointer in the pair. Enabling the auto-toggle feature, with one pointer to source data and a second pointer to destination data, greatly speeds the copying of large data blocks.

DPTR0 is located at the same address as the original 8051 data pointer, allowing the DS80C410 to execute standard 8051 code with no modifications. The registers making up the second, third, and fourth data pointers are located at SFR address locations not used in the original 8051. To access the extended 24-bit address range supported by the DS80C410, a third, high-order byte (DPXn) has been added to each pointer so that each data pointer is now composed of the SFR combination DPXn+DPHn+DPLn. Table 8 summarizes the SFRs that make up each data pointer.

```
Auto-Increment/Decrement (if AID = 1)
MOVC A, @A+DPTR
MOVX A, @DPTR
MOVX @DPTR, A
```

When used in conjunction, the auto-toggle and auto-increment/decrement features can produce very fast and efficient routines for copying or moving data. For example, suppose you want to copy three bytes of data from a source location (pointed to by DPTR2) to a destination location (pointed to by DPTR3). Assuming that DPTR2 is the active pointer (SEL1 = 1, SEL = 0), with TSL = 1 and AID = 1, the instruction sequence below copies the three bytes:

```
MOVX A, @DPTR
MOVX @DPTR, A
MOVX A, @DPTR
MOVX @DPTR, A
MOVX A, @DPTR
MOVX @DPTR, A
```

Stretch Memory Cycles

The DS80C410 allows user-application software to select the number of machine cycles it takes to execute a MOVX instruction, allowing access to both fast and slow off-chip data memory and/or peripherals without glue logic. High-speed systems often include memory-mapped peripherals such as LCDs or UARTs with slow access times, so it may not be necessary or desirable to access external devices at full speed. The microprocessor can perform a MOVX instruction in as little as two machine cycles or as many as 12 machine cycles. Accesses to internal MOVX SRAM always use two cycles. Note that stretch cycle settings affect external MOVX memory operations only and there is no way to slow the accesses to program memory other than to use a slower crystal (or external clock).

External MOVX timing is governed by the selection of 0-to-7 stretch cycles, controlled by the MD2–MD0 SFR bits in the clock control register (CKCON.2–0). A stretch of 0 results in a 2-machine cycle MOVX instruction. A stretch of 7 results in a MOVX of 12 machine cycles. Software can dynamically change the stretch value depending on the particular memory or peripheral being accessed. The default of one stretch cycle allows the use of commonly available SRAMs without dramatically lengthening the memory access times.

Stretch cycle settings affect external MOVX timing in three gradations. Changing the stretch value from 0 to 1 adds an additional clock cycle each to the data setup and hold times. Stretch values of 2 and 3 each stretch the $\overline{WR}$ or $\overline{RD}$ signal by an additional machine cycle. When a stretch value of 4 or above is selected, the interface timing changes dramatically to allow for very slow peripherals. First, the ALE signal is lengthened by one machine cycle. This increases the address setup time into the peripheral by this amount. Next, the address is held on the bus for one additional machine cycle, increasing the address hold time by this amount. The $\overline{WR}$ and $\overline{RD}$ signals are then lengthened by a machine cycle. Finally, during a MOVX write the data is held on the bus for one additional machine cycle, thereby increasing the data hold time by this amount. For every stretch value greater than 4, the setup and hold times remain constant, and only the width of the read or write signal is increased. These three gradations are reflected in the *AC Electrical Characteristics* section, where the eight MOVX timing specifications are represented by only three timing diagrams.

The reset default of one stretch cycle results in a three-cycle MOVX for any external access. Therefore, the default off-chip RAM access is not at full speed. This is a convenience to existing designs that use slower RAM. When maximum speed is desired, software should select a stretch value of 0. When using very slow RAM or peripherals, the application software can select a larger stretch value.

The specific timing of MOVX instructions as a function of stretch settings is provided in the *Electrical Specifications* section of this data sheet. As an example, Table 9 shows the read and write strobe widths corresponding to each stretch value.

CAN Interrupts

The DS80C410 provides one interrupt source for the CAN controller. The CAN interrupt source can be triggered by a receive/transmit acknowledgment from one of the 15 message centers or an error condition.

Each message center has individual ETI (transmit) and ERI (receive) interrupt enable bits and INTRQ flag bits that are found in the corresponding message control (C0MxC) SFR. If the ETI or ERI bits have been set for a message center, the successful transmission or receipt of a message, respectively, sets the INTRQ bit for that message center. The INTRQ bit can only be cleared through software. All message center interrupt flags (INTRQ) of the CAN module are ORed together to produce a single interrupt source for the CAN controller. For the microcontroller to acknowledge any individual message center interrupt request, the global interrupt enable bit (IE.7) and the CAN 0 interrupt enable bit, EIE.6, must both be set.

Interrupt assertion of error and status conditions associated with the CAN module is controlled by the ERIE and STIE bits located in the CAN 0 control (C0C) SFR. These interrupt sources also require that the global interrupt enable (EA = IE.7) and the CAN 0 interrupt enable (C0IE = EIE.6) bits be set in order to be acknowledged by the microcontroller.

Arbitration and Masking

After the CAN module has ascertained that an incoming message is bit error-free, the identification field of that message is then compared against one or more arbitration values to determine if they are loaded into a message center. Each enabled message center (see the MSRDY bit in the CAN message control register) is tested in order from 1 to 15. The first message center to successfully pass the test receives the incoming message and ends the testing. The use of masking registers allows the use of more complex identification schemes, as tests can be made based on bit patterns rather than an exact match between all bits in the identification field and arbitration values. The CAN controller also incorporates a set of five masks to allow messages with different IDs to be grouped and successfully loaded into a message center; note that some of these masks are optional as per the bits shown in Table 19.

There are several possible arbitration tests, varying according to which message center is involved. If all of the enabled tests succeed, the message is loaded into the respective message center. The most basic test, performed on all messages, compares either 11 (CAN 2.0A) or 29 (CAN 2.0B) bits of the identification field to the appropriate arbitration register, based on the EX/$\overline{ST}$ bit in the CAN 0 format register. The MEME bit (C0MxF.1) controls whether the arbitration and ID registers are compared directly or through a mask register. A special set of arbitration registers dedicated to message center 15 allows added flexibility in filtering this location.

If desired, further arbitration can be performed by comparing the first two bytes of the data field in each message against two 8-bit media arbitration register bytes. The MDME bit in the CAN message center format registers (C0MxF.0) either disables (MDME = 0) arbitration, or enables (MDME = 1) arbitration using the media ID mask registers 0–1.

If the 11-bit or 29-bit arbitration and the optional media-byte arbitration are successful, the message is loaded into the respective message center. The format register also allows the microcontroller to program each message center to function in a receive or transmit mode through the T/$\overline{R}$ bit and to use from 0 to 8 data bytes within the data field of a message. Note that message center 15 can only be used in a receive mode. To avoid a priority inversion, the DS80C410 CAN controller is configured to reload the transmit buffer with the message of the highest priority (lowest message center number) whenever an arbitration is lost or an error condition occurs.

EN_FOW (Bit 2): Enable Force OW. Setting the EN_FOW bit to a logic 1 allows the bus master to force the OW line low using FOW (bit 2 of the command register). Clearing the EN_FOW bit to a logic 0 disables the use of the FOW bit.

STPEN (Bit 3): Strong Pullup Enable. Setting the STPEN bit to a logic 1 enables functionality for the $\overline{\text{OWSTP}}$ output pin. The $\overline{\text{OWSTP}}$ pin serves as the enable signal to an external strong pullup device. This functionality is used for meeting the recovery time requirement in overdrive mode and long-line standard communication. When enabled (STPEN = 1), $\overline{\text{OWSTP}}$ goes active-low any time the master is not pulling the OW line low or waiting to read data from a slave during a communication sequence. Once the communication sequence is complete, the $\overline{\text{OWSTP}}$ output is released. Note that when the master is in the idle state, the STP_SPLY bit must also be set to logic 1 (in addition to STPEN = 1) in order for the $\overline{\text{OWSTP}}$ pin to remain in the active-low state. Clearing the STPEN bit to logic 0 disables all $\overline{\text{OWSTP}}$ pin functionality.

STP_SPLY (Bit 4): Strong Pullup Supply Mode. When the $\overline{\text{OWSTP}}$ pin is enabled (STPEN = 1), setting the STP_SPLY bit to logic 1 results in an active-low output for the $\overline{\text{OWSTP}}$ pin being sustained when the master is in an idle state. Thus, when the $\overline{\text{OWSTP}}$ signal gates an external P-channel pullup, STP_SPLY = 1 can be used to enable a stiff supply voltage to slave devices requiring high current during operation. Clearing the bit to logic 0 disables the strong pullup on the $\overline{\text{OWSTP}}$ pin when the master is idle. This bit has no affect when the $\overline{\text{OWSTP}}$ pin is disabled (STPEN = 0).

BIT_CTL (Bit 5): Bit-Banging Mode. Setting this bit to logic 1 place the master into the bit-banging mode of operation. In the bit-banging mode, only the least significant bit of the transmit/receive register is sent or received before the associated interrupt flag occurs (signaling the end of the transaction). Clearing the bit leaves the bus master operating in full-byte boundaries.

OD (Bit 6): Overdrive Mode. Setting this bit to a logic 1 places the master into overdrive mode, effectively changing the bus master timing to match the 1-Wire timing for overdrive mode as outlined in *The Book of iButton Standards*. Clearing the OD bit to a logic 0 leaves the master operating with standard mode timing.

EOWMI (Bit 7): Enable 1-Wire Master Interrupt. Setting this bit to a logic 1 enables the 1-Wire master interrupt request to the CPU for any of the 1-Wire interrupt sources that have been individually enabled in the interrupt enable register (xxxxx011b). Since the 1-Wire master interrupt and external interrupt 5 share the same interrupt flag (IE5; EXIF.7), both cannot be used simultaneously. Thus, enabling the 1-Wire interrupt source effectively disables the external interrupt 5 source.

1-Wire Interrupts

The 1-Wire bus master can be configured to generate an interrupt request to the CPU on the occurrence of a number of 1-Wire-related events or conditions. These include the following: presence-detect, transmit buffer empty, transmit shift-register empty, receive buffer full, receive shift-register full, 1-Wire short, and 1-Wire low. Each of these potential 1-Wire interrupt sources has a corresponding enable bit and flag bit. Each flag bit in the interrupt flag register (xxxxx010b) is set, independent of the interrupt enable bit, when the associated event or condition occurs. In order for the interrupt flag to generate an interrupt request to the CPU, however, the individual enable bit for the source along with the 1-Wire bus master interrupt enable bit (EOWMI; control register bit 7), and global interrupt enable bit (EA; IE.7) must both be set to a logic 1. To clear the 1-Wire bus master interrupt, a read of the interrupt flag register must always be performed by software. Table 22 summarizes the 1-Wire bus master interrupt sources.

Timers

The microcontroller provides four general-purpose timer/counters. Timers 0, 1, and 3 have three common modes of operation. Each of the three can be used as a 13-bit timer/counter, 16-bit timer/counter, or 8-bit timer/counter with auto-reload. Timer 0 can also operate as two 8-bit timer/counters. When operated as a counter, timers 0, 1, and 3 count pulses on the corresponding T0, T1, and T3 external pins. Timer 2 is a true 16-bit timer/counter with several additional operating modes. With a 16-bit reload register, timer 2 supports other features such as 16-bit auto-reload, capture, up/down count, and output clock generation. All four timer/counters default to the standard oscillator frequency divided by 12 input clock but can be configured to run from the system clock divided by 4. Timers 1 and 2 can also be configured to operate with an input clock equal to the system clock divided by 13. Table 24 shows the SFRs and bits associated with the four timer/counters.

Table 24. Timer/Counter SFRs

TIMER/COUNTER FUNCTION	TIMER/ COUNTER 0	TIMER/ COUNTER 1	TIMER/ COUNTER 2	TIMER/ COUNTER 3
Timer/Counter Mode Selection and Control	TMOD, TCON	TMOD, TCON	T2MOD, T2CON	T3CM
Count Registers	TH0, TL0	TH1, TL1	TH2, TL2	TH3, TL3
8-Bit Reload Register	TH0	TH1	—	TH3
16-Bit Reload/Capture Registers	—	—	RCAP2H, RCAP2L	—
Timer Input Clock-Select Bit	CKCON.3	CKCON.4	CKCON.5	T3CM.5
Divide-by-13 Clock-Option Bit	—	T2MOD.4	T2MOD.3	—

Watchdog Timer

The watchdog is a free-running, programmable timer that can set a flag, cause an interrupt, and/or reset the microcontroller if allowed to reach a preselected timeout. It can be restarted by software.

A typical application uses the watchdog timer as a reset source to prevent software from losing control. The watchdog timer is initialized, selecting the timeout period and enabling the reset and/or interrupt functions. After enabling the reset function, software must then restart the timer before its expiration or hardware resets the CPU. In this way, if the code execution goes awry and software does not reset the watchdog as scheduled, the microcontroller is put in reset, a known good state.

Software can select one of four timeout values as controlled by the WD1 and WD0 bits. Timeout values are precise since they are a function of the crystal frequency. When the watchdog times out, the watchdog interrupt flag (WDIF = WDCON.3) is set. If the watchdog interrupt source has been enabled, program execution immediately vectors to the watchdog timer interrupt-service routine (code address = 63h). To enable the watchdog interrupt source, both the EWDI (EIE.4) and EA (IE.7) bits must be set. Furthermore, setting the EWT (WDCON.1) bit allows the watchdog timer to generate a reset exactly 512 system clocks following a timeout. To prevent the watchdog reset from occurring in such a situation, the watchdog timer count must be reset (RWT = 1) or the watchdog-reset function itself must be disabled (EWT = 0). Both the enable watchdog timer (EWT) reset and the reset watchdog timer (RWT) control bits are protected by timed-access circuitry. This prevents errant software from accidentally clearing or disabling the watchdog. When a watchdog timer reset condition occurs, the watchdog timer reset flag (WTRF = WDCON.2) is set by the hardware. This flag can then be interrogated following a reset to determine whether the reset was caused by the watchdog timer.

The watchdog interrupt is useful for systems that do not require a reset circuit. It sets the WDIF (watchdog interrupt) flag 512 system clocks before setting the reset flag. Software can optionally enable this interrupt source, which is independent of the watchdog-reset function. The interrupt is commonly used during the debug process to determine where watchdog reset commands must be located in the application software. The interrupt also can serve as a convenient time-base generator or can wake up the microcontroller from power-saving modes.

The watchdog timer is controlled by the clock control (CKCON) and the watchdog control (WDCON) SFRs. CKCON.7 and CKCON.6 are WD1 and WD0 respectively, and they select the watchdog timeout period. Of course, the 4X/$\overline{2X}$ (PMR.3) and CD1:0 (PMR.7:6) system clock control bits also affect the timeout period. Table 25 shows the selection of timeout.

Table 25 demonstrates that, for a 40MHz crystal frequency, the watchdog timer is capable of producing timeout periods from 3.28ms (2^{17} x 1/40MHz) to greater than one and a half seconds (1.68 = 2^{26} x 1/40MHz) with the default setting of CD1:0 (= 10). This wide variation in timeout periods allows very flexible system implementation.

In a typical initialization, the user selects one of the possible counter values to determine the timeout. Once the counter chain has completed a full count, hardware sets the interrupt flag (WDIF = WDCON.3). Regardless of whether the software makes use of this flag, there are then 512 system clocks left until the reset flag (WTRF = WDCON.2) is set. Software can enable (1) or disable (0) the reset using the enable watchdog timer reset (EWT = WDCON.1) bit.

Table 25. Watchdog Timeout Values

4X/$\overline{2X}$	CD1:0	WATCHDOG INTERRUPT TIMEOUT			
		WD1:0 = 00	WD1:0 = 01	WD1:0 = 10	WD1:0 = 11
1	00	2^{15}	2^{18}	2^{21}	2^{24}
0	00	2^{16}	2^{19}	2^{22}	2^{25}
x	01	2^{17}	2^{20}	2^{23}	2^{26}
x	10	2^{17}	2^{20}	2^{23}	2^{26}
x	11	2^{25}	2^{28}	2^{31}	2^{34}

IrDA Clock

The DS80C410 has the ability to generate an output clock (CLKO) as a secondary function on port pin P3.5. Setting both the IrDA clock-output enable bit (IRDACK:COR.7) and external clock-output enable bit (XCLKOE:COR.1) to a logic 1 produces an output clock of 16 times the programmed baud rate for serial port 0. This 16X output clock used in conjunction with serial port 0 I/O (TXD0, RXD0) conveniently allows for direct connection to common IrDA encoder/decoder devices. If the XCLKOE bit alone is set to logic 1, the CLKO pin outputs the system clock frequency divided by 2, 4, 6, or 8 as defined by clock-output divide bits (COD1:0). Setting the IRDACK bit alone to logic 1 has no effect.

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 a 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 interrupts have individual priority bits that when set to a 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, enables, natural priority, and available priority selection bits are identified in Table 26. Note that external interrupts 2–5 and the 1-Wire bus master share a common interrupt vector (43h). Also note that external interrupt 5 and the 1-Wire bus master interrupt are multiplexed to form a single interrupt request. When the 1-Wire bus master interrupt is enabled (EOWMI = 1), it takes priority over external interrupt 5. In order for external interrupt 5 request to be used, the 1-Wire bus master interrupt must be disabled (EOWMI = 0).

Table 26. Interrupt Summary

NAME	FUNCTION	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) (Note 2)	EX0 (IE.0)	PX0 (IP.0)
TF0	Timer 0	0Bh	2	TF0 (TCON.5) (Note 1)	ET0 (IE.1)	PT0 (IP.1)
INT1	External Interrupt 1	13h	3	IE1 (TCON.3) (Note 2)	EX1 (IE.2)	PX1 (IP.2)
TF1	Timer 1	1Bh	4	TF1 (TCON.7) (Note 1)	ET1 (IE.3)	PT1 (IP.3)
TI0 or RI0	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.5)
TI1 or RI1	Serial Port 1	3Bh	7	RI_1(SCON1.0) TI_1(SCON1.1)	ES1 (IE.6)	PS1 (IP.6)
INT2	External Interrupts 2–5, 1-Wire Bus Master, Interrupt	43h	8	IE2 (EXIF.4)	EX2-5 (EIE.0)	PX2-5 (EIP.0)
INT3				IE3 (EXIF.5)	—	
INT4				IE4 (EXIF.6)	—	
INT5/OWMI	—	—	—	IE5 (EXIF.7) (Note 3)	EOWMI (Note 3)	
TF3	Timer 3	4Bh	9	TF3 (T3CM.7))	ET3 (EIE.1)	PT3 (EIP.1)
TI2 or RI2	Serial Port 2	53h	10	IE4 (EXIF.6)	ES2 (EIE.2)	PS2 (EIP.2)
WPI	Write Protect Interrupt	5Bh	11	WPIF (MCON2.7)	EWPI (EIE.3)	PWPI (EIP.3)
C0I	CAN0 Interrupt	6Bh	12	Various	C0IE (EIE.6)	C0IP (EIP.6)
EAI	Ethernet Activity	73h	13	TIF (BCUC.5) RIF (BCUC.4)	EAIE (EIE.5)	EAIP (EIP.5)
WDTI	Watchdog Timer	63h	14	WDIF (WDCON.3)	EWDI (EIE.4)	PWDI (EIP.4)
EPMI	Ethernet Power Mode	7Bh	15	EPMF (BCUC.6)	EPMIE (EIE.7)	EPMIP (EIP.7)

Unless marked, all flags must be cleared by the application software.

Note 1: Cleared automatically by hardware when the service routine is entered.

Note 2: If edge-triggered, the flag is cleared automatically by hardware when the service routine is entered. If level-triggered, the flag follows the state of the interrupt pin.

Note 3: The global 1-Wire interrupt-enable bit (EOWMI) and individual 1-Wire interrupt source enables are located in the internal 1-Wire bus master interrupt enable register, and must be accessed through the OWMAD and OWMDR SFRs. Individual 1-Wire interrupt source flag bits that are located in the internal 1-Wire bus master Interrupt flag register are accessed in the same way.

One's Complement Adder

The DS80C410 implements a one's complement adder to support the Internet checksum algorithm. The adder contains a 16-bit accumulator and is accessed through the one's complement adder data (OCAD) SFR.

Writing two bytes to the OCAD register initiates a summation between the 16-bit accumulator and the 16-bit value entered. When entering a new 16-bit value for summation, the MSB should be loaded first and the LSB loaded second. The calculation begins on the first machine cycle following the second write to the OCAD register and executes in a single machine cycle. This allows back-to-back writes of 16-bit data to the OCAD register for summation. The carry out bit from the high-order bit of the calculation is added back into the low-order bit of the accumulator.

Reading two bytes from the OCAD register downloads the contents of the 16-bit accumulator. When reading the 16-bit accumulator through the OCAD register, the MSB is unloaded first and the LSB is unloaded second. The 16-bit accumulator is cleared to 0000h following the second read of the OCAD SFR.

The following is an example sequence for producing an Internet checksum for transmission.

- Read OCAD twice to make certain that the 16-bit accumulator = 0000h
- Write MSB of 16-bit value to OCAD
- Write LSB of 16-bit value to OCAD

Status

The STATUS (C5h) register and STATUS1 (F7h) register provide information about interrupt and serial port activity to assist in determining if it is possible to enter PMM. The microcontroller 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 a 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 prevents 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 transmit and receive activity can be monitored through the serial port activity bits located in the STATUS and STATUS1 registers.

Oscillator-Fail Detect

The microcontroller contains a safety mechanism called an on-chip oscillator-fail detect circuit. When enabled, this circuit causes the microcontroller to be held in reset if the oscillator frequency falls below ~100kHz. When activated, this circuit complements the watchdog timer. Normally, the watchdog timer is initialized so that it times out and causes a reset in the event that the microcontroller loses control. In the event of a crystal or external oscillator failure, however, the watchdog timer does not function, and there is the potential to fail in an uncontrolled state. Using the oscillator-fail detect circuit forces the microcontroller to a known state (i.e., reset) even if the oscillator stops.

The oscillator-fail detect circuitry is enabled when software sets the enable bit OFDE (PCON.4) to a 1. Please note that software must use a timed-access procedure (described earlier) to write this bit. The OFDF (PCON.5) bit also sets to a 1 when the circuitry detects an oscillator failure, and the microcontroller is forced into a reset state. This bit can only be cleared to a 0 by a power-fail reset or by software. The oscillator-fail detect circuitry is not triggered when the oscillator is stopped upon entering stop mode.

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 incoming power supply voltages (V_{CC1} and V_{CC3}) and holds the microcontroller in reset if either supply is below a 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_{CC1} and V_{CC3} have risen above minimum voltages, V_{RST1} and V_{RST3} respectively, 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 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 precise reset thresholds also generates an optional early warning power-fail interrupt (PFI). When enabled by software, the microcontroller vectors to code address 0033h if either V_{CC1} or V_{CC3} drop below V_{PFW1} or V_{PFW3}, respectively. 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 a 1. The flag is independent of the interrupt enable and must be cleared by software.