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	Not For New Designs
Core Processor	CPU12
Core Size	16-Bit
Speed	8MHz
Connectivity	CANbus, MI Bus, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	68
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 16x8/10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc912d60ccpve

Section 17. MSCAN Controller

Section 18. Analog-to-Digital Converter

Table 2-1. M68HC12 Addressing Mode Summary

Addressing Mode	Source Format	Abbreviation	Description
Inherent	**INST** (no externally supplied operands)	INH	Operands (if any) are in CPU registers
Immediate	**INST** *#opr8i* or **INST** *#opr16i*	IMM	Operand is included in instruction stream 8- or 16-bit size implied by context
Direct	**INST** *opr8a*	DIR	Operand is the lower 8-bits of an address in the range $0000 – $00FF
Extended	**INST** *opr16a*	EXT	Operand is a 16-bit address
Relative	**INST** *rel8* or **INST** *rel16*	REL	An 8-bit or 16-bit relative offset from the current pc is supplied in the instruction
Indexed (5-bit offset)	**INST** *oprx5*,*xysp*	IDX	5-bit signed constant offset from x, y, sp, or pc
Indexed (auto pre-decrement)	**INST** *oprx3*,–*xys*	IDX	Auto pre-decrement x, y, or sp by 1 ~ 8
Indexed (auto pre-increment)	**INST** *oprx3*,+*xys*	IDX	Auto pre-increment x, y, or sp by 1 ~ 8
Indexed (auto post-decrement)	**INST** *oprx3*,*xys*–	IDX	Auto post-decrement x, y, or sp by 1 ~ 8
Indexed (auto post-increment)	**INST** *oprx3*,*xys*+	IDX	Auto post-increment x, y, or sp by 1 ~ 8
Indexed (accumulator offset)	**INST** *abd*,*xysp*	IDX	Indexed with 8-bit (A or B) or 16-bit (D) accumulator offset from x, y, sp, or pc
Indexed (9-bit offset)	**INST** *oprx9*,*xysp*	IDX1	9-bit signed constant offset from x, y, sp, or pc (lower 8-bits of offset in one extension byte)
Indexed (16-bit offset)	**INST** *oprx16*,*xysp*	IDX2	16-bit constant offset from x, y, sp, or pc (16-bit offset in two extension bytes)
Indexed-Indirect (16-bit offset)	**INST [***oprx16*,*xysp***]**	[IDX2]	Pointer to operand is found at... 16-bit constant offset from x, y, sp, or pc (16-bit offset in two extension bytes)
Indexed-Indirect (D accumulator offset)	**INST [D,***xysp***]**	[D,IDX]	Pointer to operand is found at... x, y, sp, or pc plus the value in D

output to indicate that an internal failure has been detected in either the clock monitor or COP watchdog circuit. The MCU goes into reset asynchronously and comes out of reset synchronously. This allows the part to reach a proper reset state even if the clocks have failed, while allowing synchronized operation when starting out of reset.

It is important to use an external low-voltage reset circuit (such as MC34064 or MC34164) to prevent corruption of RAM or EEPROM due to power transitions.

The reset sequence is initiated by any of the following events:

- Power-on-reset (POR)
- COP watchdog enabled and watchdog timer times out
- Clock monitor enabled and Clock monitor detects slow or stopped clock
- User applies a low level to the reset pin

External circuitry connected to the reset pin should not include a large capacitance that would interfere with the ability of this signal to rise to a valid logic one within nine bus cycles after the low drive is released. Upon detection of any reset, an internal circuit drives the reset pin low and a clocked reset sequence controls when the MCU can begin normal processing. In the case of POR or a clock monitor error, a 4096 cycle oscillator startup delay is imposed before the reset recovery sequence starts (reset is driven low throughout this 4096 cycle delay). The internal reset recovery sequence then drives reset low for 16 to 17 cycles and releases the drive to allow reset to rise. Nine cycles later this circuit samples the reset pin to see if it has risen to a logic one level. If reset is low at this point, the reset is assumed to be coming from an external request and the internally latched states of the COP timeout and clock monitor failure are cleared so the normal reset vector ($FFFE:FFFF) is taken when reset is finally released. If reset is high after this nine cycle delay, the reset source is tentatively assumed to be either a COP failure or a clock monitor fail. If the internally latched state of the clock monitor fail circuit is true, processing begins by fetching the clock monitor vector ($FFFC:FFFD). If no clock monitor failure is indicated, and the latched state of the COP timeout is true, processing begins by fetching the COP

3.6.10 Port S

Port S is the 8-bit interface to the standard serial interface consisting of the two serial communications interfaces (SCI1 and SCI0) and the serial peripheral interface (SPI) subsystems. Port S pins are available for general-purpose parallel I/O when standard serial functions are not enabled.

Port S pins serve several functions depending on the various internal control registers. If WOMS bit in the SC0CR1register is set, the P-channel drivers of the output buffers are disabled for bits 0 through 1 for the SCSI1 (2 through 3 for the SCI0). If SWOM bit in the SP0CR1 register is set, the P-channel drivers of the output buffers are disabled for bits 4 through 7 (wire-ORed mode). The open drain control effects to both the serial and the general-purpose outputs. If the RDPSx bits in the PURDS register are set, the appropriate Port S pin drive capabilities are reduced. If PUPSx bits in the PURDS register are set, the appropriate pull-up device is connected to each port S pin which is programmed as a general-purpose input. If the pin is programmed as a general-purpose output, the pull-up is disconnected from the pin regardless of the state of the individual PUPSx bits. See Multiple Serial Interface.

3.6.11 Port T

This port provides eight general-purpose I/O pins when not enabled for input capture and output compare in the timer and pulse accumulator subsystem. The TEN bit in the TSCR register enables the timer function. The pulse accumulator subsystem is enabled with the PAEN bit in the PACTL register.

Register DDRT determines pin direction of port T when used for general-purpose I/O. When DDRT bits are set, the corresponding pin is configured for output. On reset the DDRT bits are cleared and the corresponding pin is configured for input.

When the PUPT bit in the TMSK2 register is set, all input pins are pulled up internally by an active pull-up device. Pull-ups are disabled after reset.

Special Expanded Narrow Mode — This mode can be used for emulation of normal expanded narrow mode. Ports A and B are used for the16-bit address bus. Port A is used as the data bus, multiplexed with addresses. In this mode, 16-bit data is presented one byte at a time, the high byte followed by the low byte. The address is automatically incremented on the second cycle.

Special Peripheral Mode — The CPU is not active in this mode. An external master can control on-chip peripherals for testing purposes. It is not possible to change to or from this mode without going through reset. Background debugging should not be used while the MCU is in special peripheral mode as internal bus conflicts between BDM and the external master can cause improper operation of both modes.

5.4 Background Debug Mode

Background debug mode (BDM) is an auxiliary operating mode that is used for system development. BDM is implemented in on-chip hardware and provides a full set of debug operations. Some BDM commands can be executed while the CPU is operating normally. Other BDM commands are firmware based, and require the BDM firmware to be enabled and active for execution.

In special single-chip mode, BDM is enabled and active immediately out of reset. BDM is available in all other operating modes, but must be enabled before it can be activated. BDM should not be used in special peripheral mode because of potential bus conflicts.

Once enabled, background mode can be made active by a serial command sent via the BKGD pin or execution of a CPU12 BGND instruction. While background mode is active, the CPU can interpret special debugging commands, and read and write CPU registers, peripheral registers, and locations in memory.

While BDM is active, the CPU executes code located in a small on-chip ROM mapped to addresses $FF20 to $FFFF, and BDM control registers are accessible at addresses $FF00 to $FF06. The BDM ROM replaces

7.7.2 Normal Operation

The Flash EEPROM allows a byte or aligned word read in one bus cycle. A misaligned word read requires an additional bus cycle. The Flash EEPROM array responds to read operations only. Write operations are ignored.

7.7.3 Program/Erase Operation

An unprogrammed Flash EEPROM bit has a logic state of one. A bit must be programmed to change its state from one to zero. Erasing a bit returns it to a logic one. The Flash EEPROM has a minimum program/erase life of 100 cycles. Programming or erasing the Flash EEPROM is accomplished by a series of control register writes.

The Flash EEPROM must be completely erased prior to programming final data values.

Programming and erasing of Flash locations cannot be performed by code being executed from the Flash memory. While these operations must be performed in the order shown, other unrelated operations may occur between the steps. Do not exceed t_{FPGM} maximum (40µs).

7.8 Programming the Flash EEPROM

Programming the Flash EEPROM is done on a row basis. A row consists of 32 consecutive words (64 bytes) with rows starting from addresses $XX00, $XX40, $XX80 and $XXC0. When writing a row care should be taken not to write data to addresses outside of the row. Programming is restricted to aligned word i.e. data writes to select rows/blocks for programming/erase should be to even adresses and writes to any row for programming should be to aligned words.

9.5 Interrupt Control and Priority Registers

	Bit 7	6	5	4	3	2	1	Bit 0
	IRQE	IRQEN	DLY	0	0	0	0	0
RESET:	0	1	1	0	0	0	0	0

INTCR — Interrupt Control Register **$001E**

IRQE — $\overline{\text{IRQ}}$ Select Edge Sensitive Only

0 = $\overline{\text{IRQ}}$ configured for low-level recognition.

1 = $\overline{\text{IRQ}}$ configured to respond only to falling edges (on pin PE1/$\overline{\text{IRQ}}$).

IRQE can be read anytime and written once in normal modes. In special modes, IRQE can be read anytime and written anytime, except the first write is ignored.

IRQEN — External $\overline{\text{IRQ}}$ Enable

The $\overline{\text{IRQ}}$ pin has an active pull-up. See Table 3-4.

0 = External $\overline{\text{IRQ}}$ pin is disconnected from interrupt logic.

1 = External $\overline{\text{IRQ}}$ pin is connected to interrupt logic.

IRQEN can be read and written anytime in all modes.

DLY — Enable Oscillator Start-up Delay on Exit from STOP

The delay time of about 4096 cycles is based on the X clock rate chosen.

0 = No stabilization delay imposed on exit from STOP mode. A stable external oscillator must be supplied.

1 = Stabilization delay is imposed before processing resumes after STOP.

DLY can be read anytime and written once in normal modes. In special modes, DLY can be read and written anytime.

11.6.15 PLL Register Descriptions

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	SYN5	SYN4	SYN3	SYN2	SYN1	SYN0
RESET:	0	0	0	0	0	0	0	0

SYNR — Synthesizer Register **$0038**

Read anytime, write anytime, except when BCSP = 1 (PLL selected as bus clock).

If the PLL is on, the count in the loop divider (SYNR) register effectively multiplies up the bus frequency from the PLL reference frequency by SYNR + 1. Internally, SYSCLK runs at twice the bus frequency. Caution should be used not to exceed the maximum rated operating frequency for the CPU.

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	REFDV2	REFDV1	REFDV0
RESET:	0	0	0	0	0	0	0	0

REFDV — Reference Divider Register **$0039**

Read anytime, write anytime, except when BCSP = 1.

The reference divider bits provides a finer granularity for the PLL multiplier steps. The reference frequency is divided by REFDV + 1.

	Bit 7	6	5	4	3	2	1	Bit 0
	TSTOUT7	TSTOUT6	TSTOUT5	TSTOUT4	TSTOUT3	TSTOUT2	TSTOUT1	TSTOUT0
RESET:	0	0	0	0	0	0	0	0

CGTFLG — Clock Generator Test Register **$003A**

Always reads zero, except in test modes.

Maximum ESR vs. EXTAL–VSS or XTAL–VSS capacitor value, 8MHz resonators

Shunt Capacitance (pF) (VDDPLL=VDD)	**3**	60	80	100	130	165	200	270	325
	5	55	70	85	110	135	165	210	250
	7	50	60	75	95	115	135	170	195
	10	40	50	60	75	90	105	125	145
Shunt Capacitance (pF) (VDDPLL=0)	**3**	60	75	95	125	155	190	255	305
	5	50	65	80	105	130	155	200	235
	7	45	55	70	90	105	125	160	185
	10	40	45	55	70	85	95	120	130
$C_{EXTAL-VSS}$ (pF)		47pF	39pF	33pF	27pF	22pF	18pF	13pF	10pF

Maximum ESR vs. EXTAL–VSS or XTAL–VSS capacitor value,10MHz resonators

Shunt Capacitance (pF) (VDDPLL=0)	**3**	35	45	60	80	100	120	165	200
	5	30	40	50	65	80	100	125	150
	7	25	35	45	55	65	80	100	120
	10	20	30	35	40	50	60	70	80
$C_{EXTAL-VSS}$ (pF)		47pF	39pF	33pF	27pF	22pF	18pF	13pF	10pF

Maximum ESR vs. EXTAL–VSS or XTAL–VSS capacitor value,16MHz resonators

Shunt Capacitance (pF) (1) (VDDPLL=0)	**3**		10	15	20	30	35	50	60
	5			10	15	20	25	30	45
	7				10	15	20	30	35
	10								20
$C_{EXTAL-VSS}$ (pF)		47pF	39pF	33pF	27pF	22pF	18pF	13pF	10pF

1. Please refer to point 1 in 12.5.3.1 How to Use This Information for important information regarding shunt capacitance.

PCLK0 — PWM Channel 0 Clock Select

0 = Clock A is the clock source for channel 0.
1 = Clock S0 is the clock source for channel 0.

If a clock select is changed while a PWM signal is being generated, a truncated or stretched pulse may occur during the transition.

The following four bits apply in left-aligned mode only:

PPOL3 — PWM Channel 3 Polarity

0 = Channel 3 output is low at the beginning of the period; high when the duty count is reached.
1 = Channel 3 output is high at the beginning of the period; low when the duty count is reached.

PPOL2 — PWM Channel 2 Polarity

0 = Channel 2 output is low at the beginning of the period; high when the duty count is reached.
1 = Channel 2 output is high at the beginning of the period; low when the duty count is reached.

PPOL1 — PWM Channel 1 Polarity

0 = Channel 1 output is low at the beginning of the period; high when the duty count is reached.
1 = Channel 1 output is high at the beginning of the period; low when the duty count is reached.

PPOL0 — PWM Channel 0 Polarity

0 = Channel 0 output is low at the beginning of the period; high when the duty count is reached.
1 = Channel 0 output is high at the beginning of the period; low when the duty count is reached.

Depending on the polarity bit, the duty registers may contain the count of either the high time or the low time. If the polarity bit is zero and left alignment is selected, the duty registers contain a count of the low time. If the polarity bit is one, the duty registers contain a count of the high time.

PWEN0 — PWM Channel 0 Enable

The pulse modulated signal will be available at port P, bit 0 when its clock source begins its next cycle.

0 = Channel 0 is disabled.

1 = Channel 0 is enabled.

	Bit 7	6	5	4	3	2	1	Bit 0
	0	Bit 6	5	4	3	2	1	Bit 0
RESET:	0	0	0	0	0	0	0	0

PWPRES — PWM Prescale Counter **$0043**

PWPRES is a free-running 7-bit counter. Read anytime. Write only in special mode (SMOD = 1).

	Bit 7	6	5	4	3	2	1	Bit 0
	Bit 7	6	5	4	3	2	1	Bit 0
RESET:	0	0	0	0	0	0	0	0

PWSCAL0 — PWM Scale Register 0 **$0044**

Read and write anytime. A write will cause the scaler counter PWSCNT0 to load the PWSCAL0 value unless in special mode with DISCAL = 1 in the PWTST register.

PWM channels 0 and 1 can select clock S0 (scaled) as its input clock by setting the control bit PCLK0 and PCLK1 respectively. Clock S0 is generated by dividing clock A by the value in the PWSCAL0 register + 1 and dividing again by two. When PWSCAL0 = $FF, clock A is divided by 256 then divided by two to generate clock S0.

MCEN — Modulus Down-Counter Enable

0 = Modulus counter disabled.

1 = Modulus counter is enabled.

When MCEN=0, the counter is preset to $FFFF. This will prevent an early interrupt flag when the modulus down-counter is enabled.

MCPR1, MCPR0 — Modulus Counter Prescaler select

These two bits specify the division rate of the modulus counter prescaler.

The newly selected prescaler division rate will not be effective until a load of the load register into the modulus counter count register occurs.

MCPR1	MCPR0	Prescaler division rate
0	0	1
0	1	4
1	0	8
1	1	16

	BIT 7	6	5	4	3	2	1	BIT 0
	MCZF	0	0	0	POLF3	POLF2	POLF1	POLF0
RESET:	0	0	0	0	0	0	0	0

MCFLG — 16-Bit Modulus Down-Counter FLAG Register **$00A7**

Read: any time

Write: Only for clearing bit 7

MCZF — Modulus Counter Underflow Interrupt Flag

The flag is set when the modulus down-counter reaches $0000.

Writing a1 to this bit clears the flag (if TFFCA=0). Writing zero has no effect.

Any access to the MCCNT register will clear the MCZF flag in this register when TFFCA bit in register TSCR($86) is set.

15.5 Serial Peripheral Interface (SPI)

The serial peripheral interface allows the MC68HC912D60A to communicate synchronously with peripheral devices and other microprocessors. The SPI system in the MC68HC912D60A can operate as a master or as a slave. The SPI is also capable of interprocessor communications in a multiple master system.

When the SPI is enabled, all pins that are defined by the configuration as inputs will be inputs regardless of the state of the DDRS bits for those pins. All pins that are defined as SPI outputs will be outputs only if the DDRS bits for those pins are set. Any SPI output whose corresponding DDRS bit is cleared can be used as a general-purpose input.

A bidirectional serial pin is possible using the DDRS as the direction control.

15.5.1 SPI Baud Rate Generation

The E Clock is input to a divider series and the resulting SPI clock rate may be selected to be E divided by 2, 4, 8, 16, 32, 64, 128 or 256. Three bits in the SP0BR register control the SPI clock rate. This baud rate generator is activated only when SPI is in the master mode and serial transfer is taking place. Otherwise this divider is disabled to save power.

15.5.2 SPI Operation

In the SPI system the 8-bit data register in the master and the 8-bit data register in the slave are linked to form a distributed 16-bit register. When a data transfer operation is performed, this 16-bit register is serially shifted eight bit positions by the SCK clock from the master so the data is effectively exchanged between the master and the slave. Data written to the SP0DR register of the master becomes the output data for the slave and data read from the SP0DR register of the master after a transfer operation is the input data from the slave.

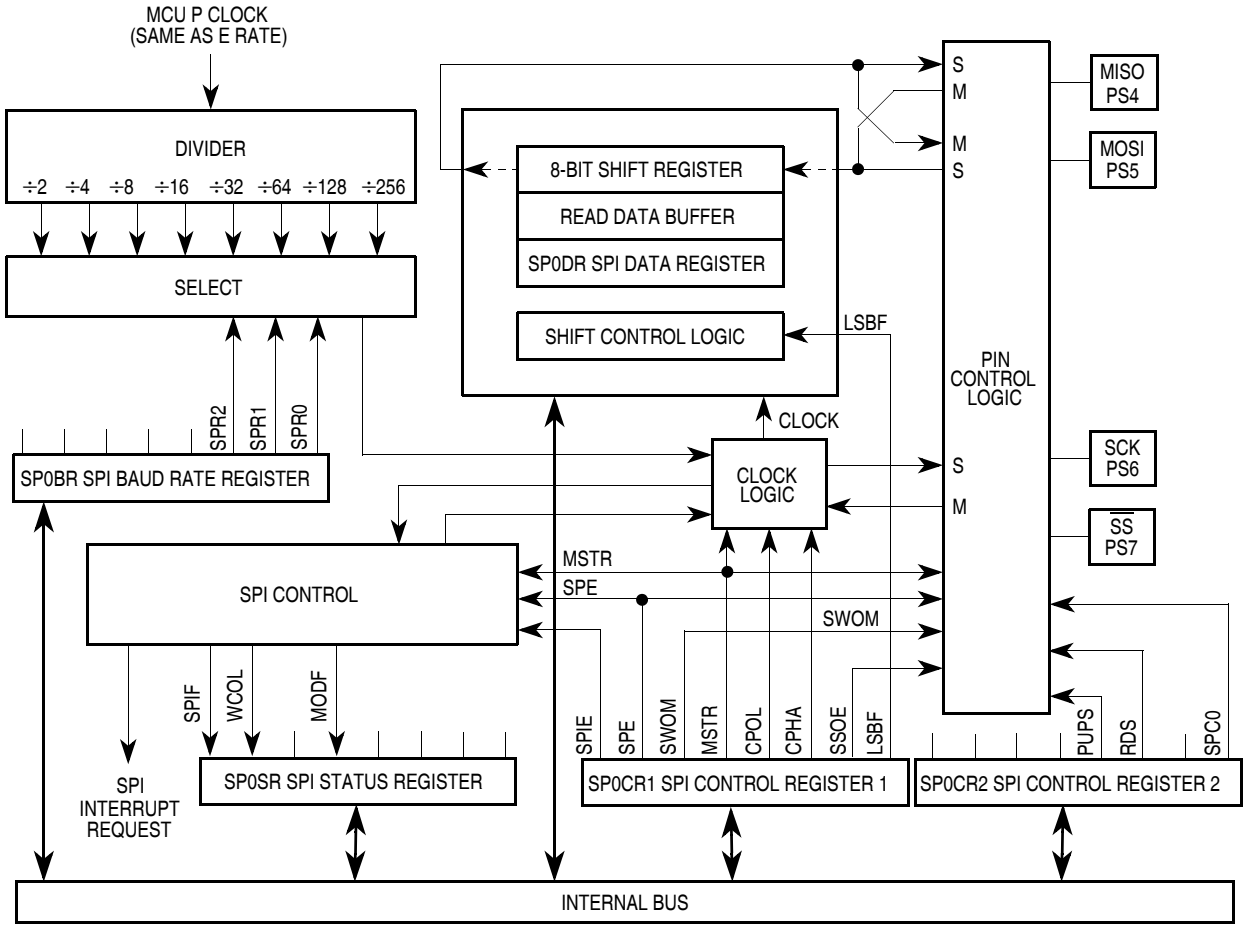

Figure 15-3. Serial Peripheral Interface Block Diagram

A clock phase control bit (CPHA) and a clock polarity control bit (CPOL) in the SP0CR1 register select one of four possible clock formats to be used by the SPI system. The CPOL bit simply selects non-inverted or inverted clock. The CPHA bit is used to accommodate two fundamentally different protocols by shifting the clock by one half cycle or no phase shift.

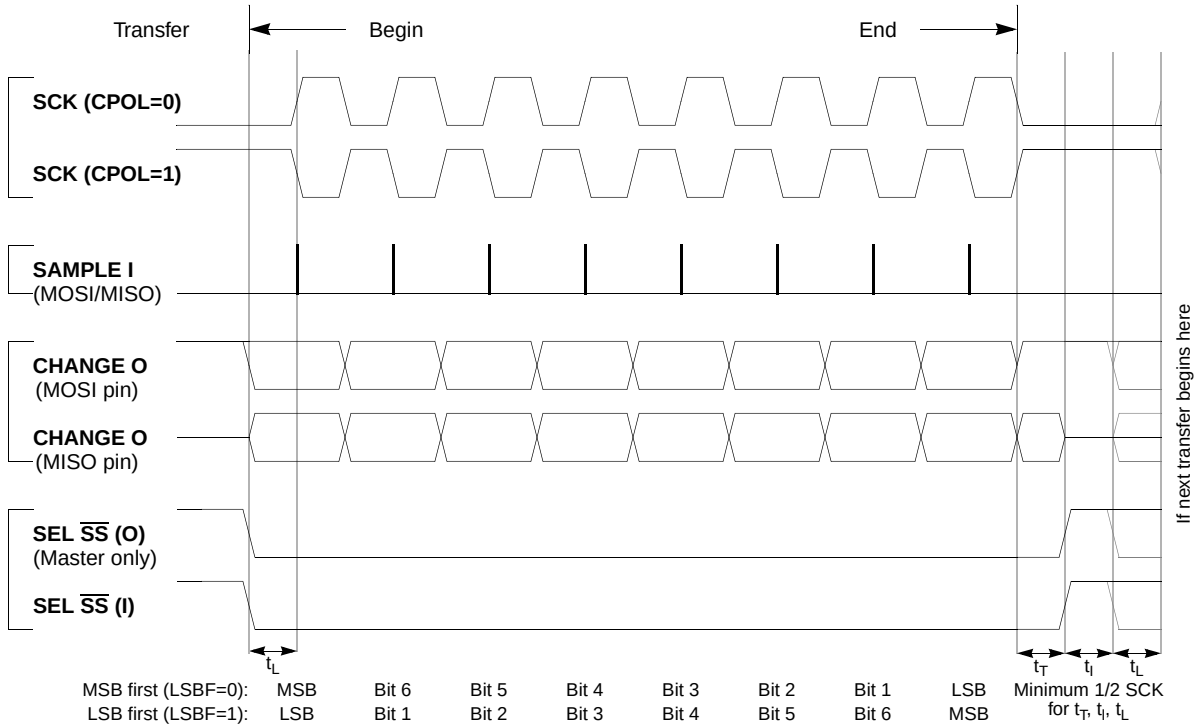

Figure 15-4. SPI Clock Format 0 (CPHA = 0)

SPIE — SPI Interrupt Enable

0 = SPI interrupts are inhibited
1 = Hardware interrupt sequence is requested each time the SPIF or MODF status flag is set

SPE — SPI System Enable

0 = SPI internal hardware is initialized and SPI system is in a low-power disabled state.
1 = PS[4:7] are dedicated to the SPI function

When MODF is set, SPE always reads zero. SP0CR1 must be written as part of a mode fault recovery sequence.

SWOM — Port S Wired-OR Mode

Controls not only SPI output pins but also the general-purpose output pins (PS[4:7]) which are not used by SPI.

0 = SPI and/or PS[4:7] output buffers operate normally
1 = SPI and/or PS[4:7] output buffers behave as open-drain outputs

MSTR — SPI Master/Slave Mode Select

0 = Slave mode
1 = Master mode

CPOL, CPHA — SPI Clock Polarity, Clock Phase

These two bits are used to specify the clock format to be used in SPI operations. When the clock polarity bit is cleared and data is not being transferred, the SCK pin of the master device is low. When CPOL is set, SCK idles high. See Figure 15-4 and Figure 15-5.

SSOE — Slave Select Output Enable

The $\overline{SS}$ output feature is enabled only in the master mode by asserting the SSOE and DDS7.

LSBF — SPI LSB First enable

0 = Data is transferred most significant bit first
1 = Data is transferred least significant bit first

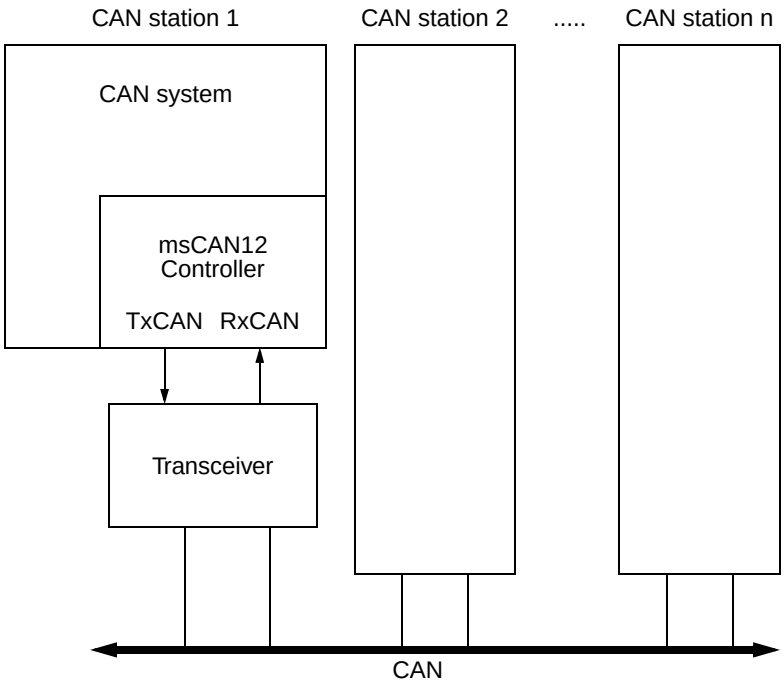

Figure 17-1. The CAN System

17.4 Message Storage

msCAN12 facilitates a sophisticated message storage system which addresses the requirements of a broad range of network applications.

17.4.1 Background

Modern application layer software is built upon two fundamental assumptions:

1. Any CAN node is able to send out a stream of scheduled messages without releasing the bus between two messages. Such nodes will arbitrate for the bus right after sending the previous message and will only release the bus when arbitration is lost.
2. The internal message queue within any CAN node is organized such that if more than one message is ready to be sent, the highest priority message will be sent out first.

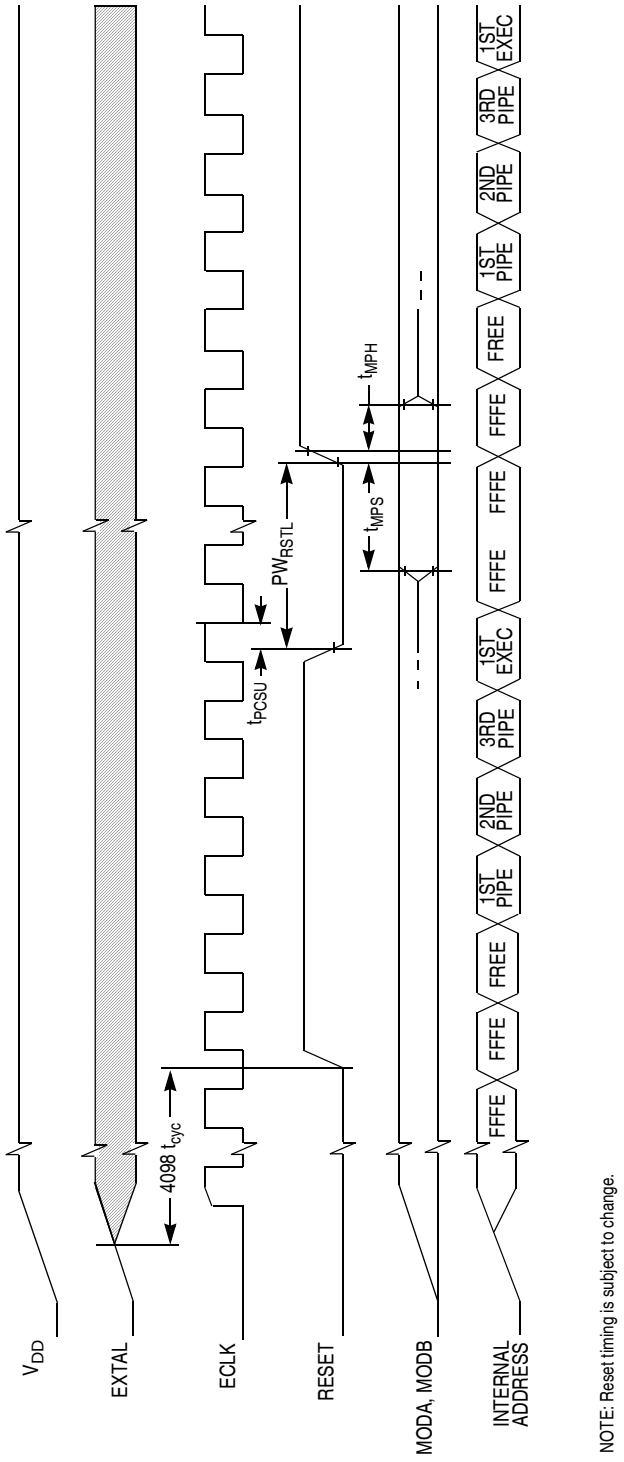

Figure 20-2. POR and External Reset Timing Diagram

operand — Data on which an operation is performed. Usually a statement consists of an operator and an operand. For example, the operator may be an add instruction, and the operand may be the quantity to be added.

oscillator — A circuit that produces a constant frequency square wave that is used by the computer as a timing and sequencing reference.

OTPROM — One-time programmable read-only memory. A nonvolatile type of memory that cannot be reprogrammed.

overflow — A quantity that is too large to be contained in one byte or one word.

page zero — The first 256 bytes of memory (addresses $0000–$00FF).

parity — An error-checking scheme that counts the number of logic 1s in each byte transmitted. In a system that uses odd parity, every byte is expected to have an odd number of logic 1s. In an even parity system, every byte should have an even number of logic 1s. In the transmitter, a parity generator appends an extra bit to each byte to make the number of logic 1s odd for odd parity or even for even parity. A parity checker in the receiver counts the number of logic 1s in each byte. The parity checker generates an error signal if it finds a byte with an incorrect number of logic 1s.

PC — See “program counter (PC).”

peripheral — A circuit not under direct CPU control.

phase-locked loop (PLL) — A clock generator circuit in which a voltage controlled oscillator produces an oscillation which is synchronized to a reference signal.

PLL — See "phase-locked loop (PLL)."

pointer — Pointer register. An index register is sometimes called a pointer register because its contents are used in the calculation of the address of an operand, and therefore points to the operand.

polarity — The two opposite logic levels, logic 1 and logic 0, which correspond to two different voltage levels, V_{DD} and V_{SS}.

polling — Periodically reading a status bit to monitor the condition of a peripheral device.

port — A set of wires for communicating with off-chip devices.

prescaler — A circuit that generates an output signal related to the input signal by a fractional scale factor such as 1/2, 1/8, 1/10 etc.

program — A set of computer instructions that cause a computer to perform a desired operation or operations.