6 SERIAL PERIPHERAL INTERFACE

This section contains a description on the serial peripheral interface (SPI).

6.1 Overview and Features

The serial peripheral interface (SPI) is a synchronous interface which allows several SPI microcontrollers or SPI-type peripherals to be interconnected. In a serial peripheral interface, separate wires (signals) are required for data and clock. In the SPI format, the clock is not included in the data stream and must be furnished as a separate signal. The MC68HC11A8 SPI system may be configured either as a master or as a slave. The SPI contains the following features:

- Full Duplex, Three-Wire Synchronous Transfers
- Master or Slave Operation
- 1.5 MHz (Maximum) Master Bit Frequency
- 3 MHz (Maximum) Slave Bit Frequency
- Four Programmable Master Bit Rates
- Programmable Clock Polarity and Phase
- End-of-Transmission Interrupt Flag
- Write Collision Flag Protection
- Master-Master Mode Fault Protection
- Easily Interfaces to Simple Expansion Parts (PLLs, D/As, Latches, Display Drivers, etc.)

6.2 SPI Signal Descriptions

The four basic SPI signals (MISO, MOSI, SCK, and $\overline{SS}$) are discussed in the following paragraphs. Each signal is described for both the master and slave modes.

Any SPI output line has to have its corresponding data direction register bit set. If this bit is clear, the line is disconnected from the SPI logic and becomes a general-purpose input line. Any SPI input line is forced to act as an input regardless of what is in the corresponding data direction register bit.

6.2.1 Master In Slave Out (MISO)

The MISO line is configured as an input in a master device and as an output in a slave device. It is one of the two lines that transfer serial data in one direction, with the most significant bit sent first. The MISO line of a slave device is placed in the high-impedance state if the slave is not selected.

6.2.2 Master Out Slave In (MOSI)

The MOSI line is configured as an output in a master device and as an input in a slave device. It is one of the two lines that transfer serial data in one direction with the most significant bit sent first.

6.2.3 Serial Clock (SCK)

The serial clock is used to synchronize data movement both in and out of the device through its MOSI and MISO lines. The master and slave devices are capable of exchanging a byte of information during a sequence of eight clock cycles. Since SCK is generated by the master device, this line becomes an input on a slave device.

As shown in **Figure 6-1**, four possible timing relationships may be chosen by using control bits CPOL and CPHA in the serial peripheral control register (SPCR). Both master and slave devices must operate with the same timing. The master device always places data on the MOSI line a half-cycle before the clock edge (SCK), in order for the slave device to latch the data.

Two bits (SPR0 and SPR1) in the SPCR of the master device select the clock rate. In a slave device, SPR0 and SPR1 have no effect on the operation of the SPI.

6.2.4 Slave Select ($\overline{SS}$)

The slave select ($\overline{SS}$) input line is used to select a slave device. It has to be low prior to data transactions and must stay low for the duration of the transaction.

The $\overline{SS}$ line on the master must be tied high. If it goes low, a mode fault error flag (MODF) is set in the serial peripheral status register (SPSR). The $\overline{SS}$ pin can be selected to be a general-purpose output by writing a one in bit 5 of the port D data direction register, thus disabling the mode fault circuit. The other three SPI lines are dedicated to the SPI whenever the SPI is on.

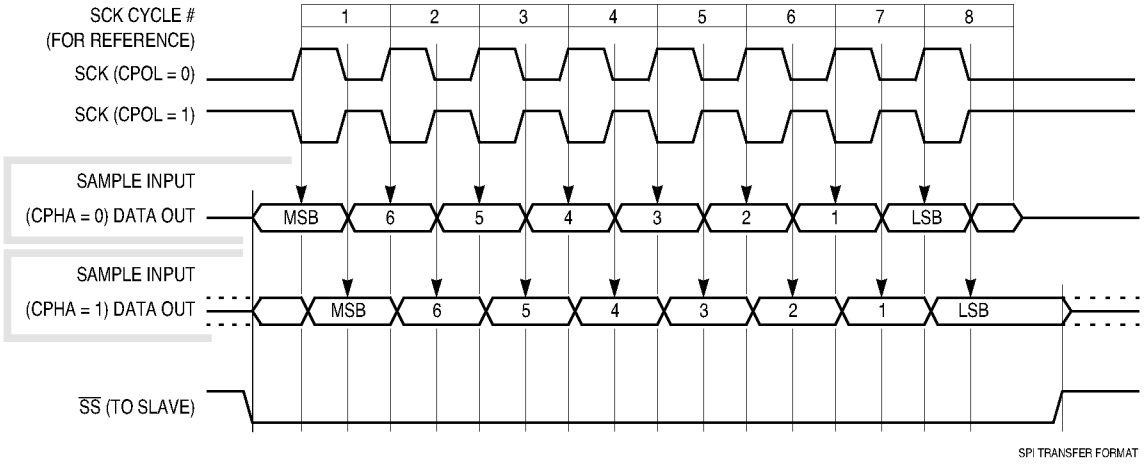

Figure 6-1 Data Clock Timing Diagram

When CPHA = 0, the shift clock is the OR of $\overline{SS}$ with SCK. In this clock phase mode, $\overline{SS}$ must go high between successive characters in an SPI message. When CPHA = 1, $\overline{SS}$ may be left low for several SPI characters. In cases where there is only one SPI slave MCU, its $\overline{SS}$ line could be tied to V_{SS} as long as CPHA = 1 clock modes are used.

6.3 Functional Description

Figure 6-2 shows a block diagram of the serial peripheral interface circuitry. When a master device transmits data to a slave device via the MOSI line, the slave device responds by sending data to the master device via the master's MISO line. This implies full duplex transmission with both data out and data in synchronized with the same clock signal. Thus, the byte transmitted is replaced by the byte received and eliminates the need for separate transmit-empty and receiver-full status bits. A single status bit (SPIF) is used to signify that the I/O operation has been completed.

The SPI is double buffered on read, but not on write. If a write is performed during data transfer, the transfer occurs uninterrupted, and the write will be unsuccessful. This condition will cause the write collision (WCOL) status bit in the SPSR to be set. After a data byte is shifted, the SPIF flag of the SPSR is set.

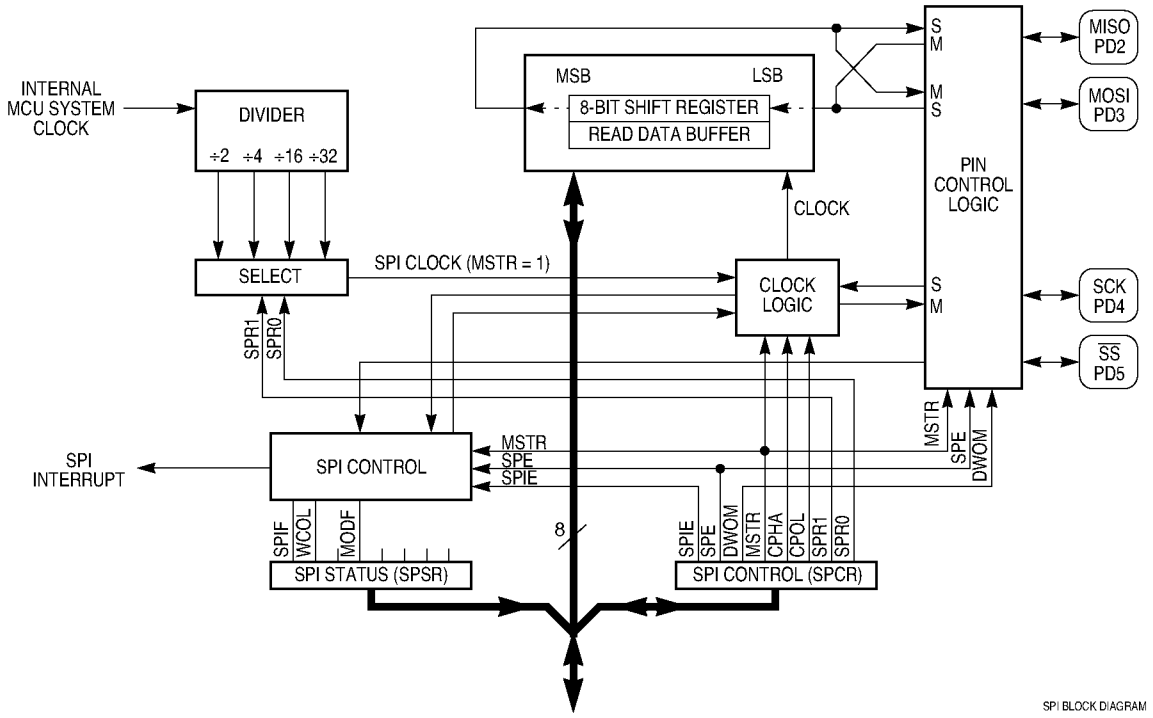

Figure 6-2 Serial Peripheral Interface Block Diagram

In the master mode, the SCK pin is an output. It idles high or low, depending on the CPOL bit in the SPCR, until data is written to the shift register, at which point eight clocks are generated to shift the eight bits of data and then SCK goes idle again.

In a slave mode, the slave start logic receives a logic low at the $\overline{SS}$ pin and a clock input at the SCK pin. Thus, the slave is synchronized with the master. Data from the master is received serially at the slave MOSI line and loads the 8-bit shift register. After the 8-bit shift register is loaded, its data is parallel transferred to the read buffer. During a write cycle, data is written into the shift register, then the slave waits for a clock train from the master to shift the data out on the slave's MISO line.

Figure 6-3 illustrates the MOSI, MISO, SCK, and $\overline{SS}$ master-slave interconnections.

Due to data direction register control of SPI outputs and the port D wire-OR mode (DWOM) option, the SPI system can be configured in a variety of ways. Systems with a single bidirectional data path rather than separate MISO and MOSI paths can be accommodated. Since MC68HC11A8 SPI slaves can selectively disable their MISO output, a broadcast message protocol is also possible.

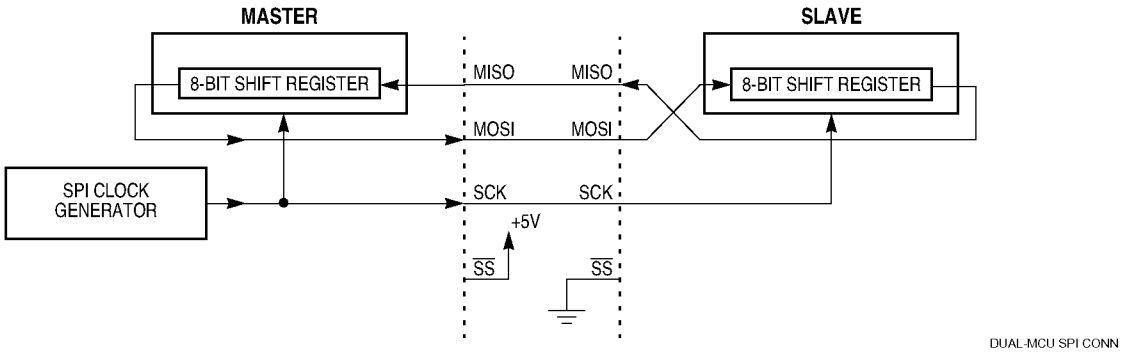

Figure 6-3 Serial Peripheral Interface Master-Slave Interconnection

6.4 SPI Registers

There are three registers in the serial peripheral interface which provide control, status, and data storage functions. These registers are called the serial peripheral control register (SPCR), serial peripheral status register (SPSR), and serial peripheral data I/O register (SPDR) and are described in the following paragraphs.

6.4.1 Serial Peripheral Control Register (SPCR)

	7	6	5	4	3	2	1	0	
$1028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
RESET	0	0	0	0	0	1	U	U	

SPIE — Serial Peripheral Interrupt Enable
0 = SPIF interrupts disabled
1 = SPI interrupt if SPIF = 1

SPE — Serial Peripheral System Enable
0 = SPI system off
1 = SPI system on

DWOM — Port D Wire-OR Mode Option
DWOM affects all six port D pins together.
0 = Port D outputs are normal CMOS outputs
1 = Port D outputs act as open-drain outputs

MSTR — Master Mode Select

0 = Slave mode

1 = Master mode

CPOL — Clock Polarity

When the clock polarity bit is cleared and data is not being transferred, a steady state low value is produced at the SCK pin of the master device. Conversely, if this bit is set, the SCK pin will idle high. This bit is also used in conjunction with the clock phase control bit to produce the desired clock-data relationship between master and slave. See **Figure 6-1**.

CPHA — Clock Phase

The clock phase bit, in conjunction with the CPOL bit, controls the clock-data relationship between master and slave. The CPOL bit can be thought of as simply inserting an inverter in series with the SCK line. The CPHA bit selects one of two fundamentally different clocking protocols. When CPHA = 0, the shift clock is the OR of SCK with $\overline{SS}$. As soon as $\overline{SS}$ goes low the transaction begins and the first edge on SCK invokes the first data sample. When CPHA = 1, the $\overline{SS}$ pin may be thought of as a simple output enable control. Refer to **Figure 6-1**.

Table 6-1 Serial Peripheral Rate Selection

SPR1	SPR0	Internal Processor, Clock Divide By
0	0	2
0	1	4
1	0	16
1	1	32

SPR1 and SPR0—SPI Clock Rate Selects

These two serial peripheral rate bits select one of four baud rates (**Table 6-1**) to be used as SCK if the device is a master; however, they have no effect in the slave mode.

6.4.2 Serial Peripheral Status Register (SPSR)

	7	6	5	4	3	2	1	0	
$1029	SPIF	WCOL	0	MODF	0	0	0	0	SPSR
RESET	0	0	0	0	0	0	0	0	

SPIF — SPI Transfer Complete Flag

The serial peripheral data transfer flag bit is set upon completion of data transfer between the processor and external device. If SPIF goes high, and if SPIE is set, a serial peripheral interrupt is generated. Clearing the SPIF bit is accomplished by reading the SPSR (with SPIF set) followed by an access of the SPDR. Unless SPSR is read (with SPIF set) first, attempts to write to SPDR are inhibited.

WCOL — Write Collision

The write collision bit is set when an attempt is made to write to the serial peripheral data register while data transfer is taking place. If CPHA is zero a transfer is said to begin when $\overline{SS}$ goes low and the transfer ends when $\overline{SS}$ goes high after eight clock cycles on SCK. When CPHA is one a transfer is said to begin the first time SCK becomes active while $\overline{SS}$ is low and the transfer ends when the SPIF flag gets set. Clearing the WCOL bit is accomplished by reading the SPSR (with WCOL set) followed by an access to SPDR.

Bit 5 — Not Implemented

This bit always reads zero.

MODF — Mode Fault

The mode fault flag indicates that there may have been a multi-master conflict for system control and allows a proper exit from system operation to a reset or default system state. The MODF bit is normally clear, and is set only when the master device has its $\overline{SS}$ pin pulled low. Setting the MODF bit affects the internal serial peripheral interface system in the following ways:

1. An SPI interrupt is generated if SPIE = 1.
2. The SPE bit is cleared. This disables the SPI.
3. The MSTR bit is cleared, thus forcing the device into the slave mode.
4. DDRD bits for the four SPI pins are forced to zeros.

Clearing the MODF bit is accomplished by reading the SPSR (with MODF set), followed by a write to the SPCR. Control bits SPE and MSTR may be restored by user software to their original state after the MODF bit has been cleared. It is also necessary to restore DDRD after a mode fault.

Bits 3-0 — Not Implemented

These bits always read zero.

6.4.3 Serial Peripheral Data I/O Register (SPDR)

The serial peripheral data I/O register is used to transmit and receive data on the serial bus. Only a write to this register will initiate transmission/reception of another byte, and this will only occur in the master device. At the completion of transmitting a byte of data, the SPIF status bit is set in both the master and slave devices.

When the user reads the serial peripheral data I/O register, a buffer is actually being read. The first SPIF must be cleared by the time a second transfer of data from the shift register to the read buffer is initiated or an overrun condition will exist. In cases of overrun the byte which causes the overrun is lost.

A write to the serial peripheral data I/O register is not buffered and places data directly into the shift register for transmission.