Microcontroller Interface to the ADC12038 Families

National Semiconductor
Application Note 929
Emmy Denton
February 1994

Table of Contents

COPS™, HPC™ and MICROWIRE™ are trademarks of National Semiconductor Corporation.

© 2002 National Semiconductor Corporation

1.0 General Overview

The ADC12038 families are 12-bit plus sign sampling ADC converters with serial I/O. These devices have configurable analog multiplexers with 2, 4, or 8 input channels. On request, these A/Ds perform a self calibration routine that minimizes linearity, zero, and full-scale errors. To minimize power consumption these devices have a power down mode that can be accessed by hardware (PD pin) or by a software instruction.

The serial I/O is configured to comply with the NSC MICROWIRE™ serial data exchange standard for easy interface to the COPS™ and HPC families of controllers, and can easily interface with standard shift registers and microprocessors. The conversion resolution can be selected by a software instruction to be 8-bits, 8-bits+sign, 12-bits or 12-bits+sign. 8-bit and 8-bit+sign conversions take less time than 12-bit and 12-bit+sign conversions (21 clock periods versus 44). In addition, selection of the output data format can be software programmable to be:

1. 8-bits, 8-bits+sign, 12-bits, 12-bits+sign, 16-bits or 16-bits+sign in length
2. MSB or LSB first
3. Left or Right justified

There are three ADC12038 families: Low voltage, High speed and standard. Each family includes four different combinations of analog inputs and features as summarized in *Table 1*.

TABLE 1. Summary of the Differences of the Devices in the Three ADC12038 Families

Device Number	Operating Supply Voltage and Power Dissipation	Maximum Clock Frequency (MHz)	Maximum Sampling Rate (kHz)	Number of MUX Inputs	MUX OUT and A/D IN Pins	Hardware Power Down Control (PD Pin)	Package Size and Type
ADC12030	5V ±10% 33 mW (max) @5V	5 MHz	73 kHz	2	NO	NO	16-pin DIP & SO
ADC12032				2	YES	NO	20-pin DIP & SO
ADC12034				4	YES	YES	24-pin DIP & SO
ADC12038				8	YES	YES	28-pin DIP & SO
ADC12L030	3.3V ±10% 15 mW (max) @3.3V			2	NO	NO	16-pin DIP & SO
ADC12L032				2	YES	NO	20-pin DIP & SO
ADC12L034				4	YES	YES	24-pin DIP & SO
ADC12L038				8	YES	YES	28-pin DIP & SO
ADC12H030	5V ±10% 36 mW (max) @5V	8 MHz	116 kHz	2	NO	NO	16-pin DIP & SO
ADC12H032				2	YES	NO	20-pin DIP & SO
ADC12H034				4	YES	YES	24-pin DIP & SO
ADC12H038				8	YES	YES	28-pin DIP & SO

Throughout this application note we will refer to the ADC12038. Any of this information will also apply to all the devices in the ADC12038 families. The device data sheets should be used in conjunction with this application note to help you understand the operation of these devices. The scope of this application note will focus on the digital interface. A brief overview of the digital functionality of these devices is included.

1.1 THE SERIAL INTERFACE

The ADC12038 families of analog-to-digital converters can be programmed for many modes of operation through their serial digital interface. The serial interface for the ADC12038 is comprised of the digital control lines SCLK, $\overline{CS}$, DO, DI, EOC, $\overline{DOR}$, PD and $\overline{CONV}$. *Table 2* gives a brief pin description for each of these control lines.

1.0 General Overview (Continued)

TABLE 2. Digital Control Pin Descriptions

Pin Name	Description
SCLK	This is the serial data clock input. The clock applied to this input controls the rate at which the serial data exchange occurs. With $\overline{CS}$ low the rising edge of SCLK loads the information on the DI pin into the multiplexer address and mode select shift register. This address controls which channel of the analog input multiplexer (MUX) is selected and the mode of operation for the ADC. With $\overline{CS}$ low the falling edge shifts of SCLK the data resulting from the previous ADC conversion out on DO, with the exception of the first bit of data. When $\overline{CS}$ is low continuously, the first bit of the data is clocked out on the rising edge of EOC (end of conversion). When $\overline{CS}$ is toggled the falling edge of $\overline{CS}$ always clocks out the first bit of data. $\overline{CS}$ should be brought low when SCLK is low.
$\overline{CS}$	This is the chip select pin. When a logic low is applied to this pin the device is selected, activating the DO, DI, and SCLK serial interface lines. The falling edge of $\overline{CS}$ resets a conversion in progress and starts the sequence for a new conversion. When $\overline{CS}$ is brought low during a conversion in progress, the conversion is prematurely ended and the data in the output latches may be corrupted, requiring the data output at this time to be ignored. $\overline{CS}$ should be brought low when SCLK is low.
DI	The data input pin. The data applied to this pin is shifted by the rising edge of SCLK into the multiplexer address and mode select register. *Tables 4, 5, 6, 7* show the assignments of the multiplexer address and the mode select data.
DO	The data output pin. This pin is an active push/pull output when $\overline{CS}$ is Low. When $\overline{CS}$ is High this output is in TRI-STATE. The ADC conversion result and converter status data are clocked out by the falling edge of SCLK on this pin.
EOC	This pin is an active push/pull output and indicates the status of the device. When Low, it signals that the ADC is busy with a conversion, auto-calibration, auto-zero or power down cycle. The rising edge of EOC signals the end of one of these cycles.
$\overline{DOR}$	This is the data output ready pin. This pin is an active push/pull output. It is useful only when $\overline{CS}$ is toggled.
$\overline{CONV}$	A logic Low is required on this pin to program any mode or change the ADC's configuration (12-bit conversion, 8-bit conversion, Auto Cal, Auto Zero etc.) as listed in the Mode Programming Table (*Table 4*). When this pin is high the ADC is placed in the read data only mode. While in the read data only mode, bringing $\overline{CS}$ low and Pulsing SCLK will only clock out on DO any data stored in the ADCs output shift register. The data on DI will be neglected. A new conversion will not be started and the ADC will remain in the mode and/or configuration previously programmed. Read data only cannot be performed while a conversion, Auto-Cal or Auto-Zero are in progress.
PD	This is the power down pin. When PD is high, the ADC is powered down; when PD is low, the ADC is powered up.

1.0 General Overview (Continued)

The interplay of these lines can be graphically seen in the timing diagram of *Figure 1*.

The chip select pin ($\overline{CS}$) enables the logic inputs and DO output. Eight bits of data that control the ADC are clocked in on the digital input pin (DI) by the rising edge of the serial clock (SCLK) when $\overline{CS}$ is low. Taking $\overline{CS}$ will output the first bit of data (DBO) on DO. While $\overline{CS}$ is low, the falling edge of SCLK clocks the data out on the digital output pin (DO). $\overline{CS}$ should only be brought low when SCLK is low. The functions of the convert input ($\overline{CONV}$), data output ready ($\overline{DOR}$) and end of conversion output (EOC) pins are covered in more detail in the data sheet. The simplest interface to the ADC12038 requires only 4 control lines: DO, DI, SCLK and $\overline{CS}$. For this case $\overline{CONV}$ and PD are grounded and EOC and $\overline{DOR}$ outputs are not used.

1.2 THE SERIAL OUTPUT WORD FORMAT

The diagram in *Figure 2* shows a 16-bit serial output word. The ADC12038 family can be programmed to provide unsigned output data in 8-bit, 12-bit, or 16-bit word lengths or signed data in 9-bit, 13-bit, or 17-bit word lengths. The data format can be right- or left-justified, MSB or LSB first. *Table 3* summarizes the available serial output data formats. *Table 4* describes the serial input word required to select the available serial output data formats.

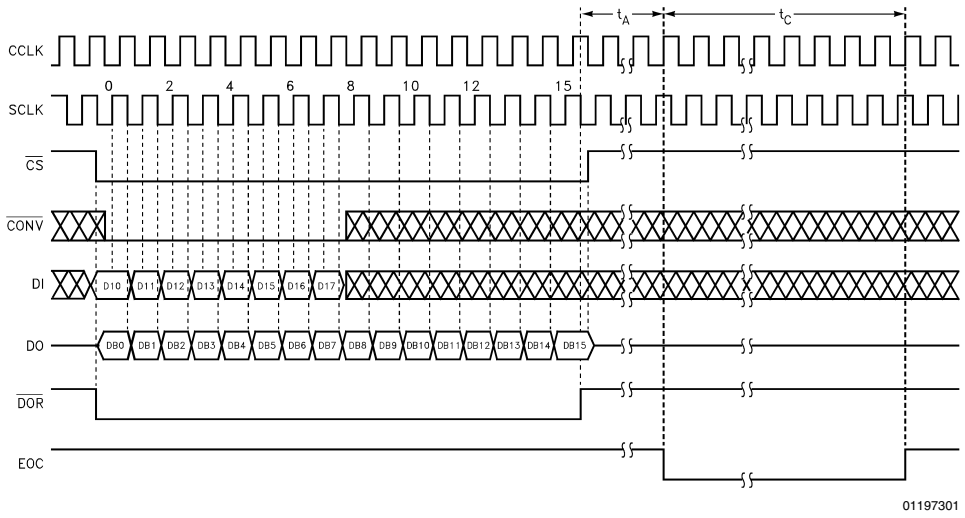

FIGURE 1. Timing Diagram for a 12-Bit Plus Sign Conversion with a 16-Bit Serial Output Word Format on DO

1.0 General Overview (Continued)

TABLE 3. Data Out Formats

DO Formats			DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	DB8	DB9	DB10	DB11	DB12	DB13	DB14	DB15	DB16
with Sign	MSB First	17 Bits	Sign	Sign	Sign	Sign	Sign	MSB	10	9	8	7	6	5	4	3	2	1	LSB
		13 Bits	Sign	MSB	10	9	8	7	6	5	4	3	2	1	LSB				
		9 Bits	Sign	MSB	6	5	4	3	2	1	LSB								
	LSB First	17 Bits	LSB	1	2	3	4	5	6	7	8	9	10	MSB	Sign	Sign	Sign	Sign	Sign
		13 Bits	LSB	1	2	3	4	5	6	7	8	9	10	MSB	Sign				
		9 Bits	LSB	1	2	3	4	5	6	MSB	Sign								
without Sign	MSB First	16 Bits	0	0	0	0	MSB	10	9	8	7	6	5	4	3	2	1	LSB	
		12 Bits	MSB	10	9	8	7	6	5	4	3	2	1	LSB					
		8 Bits	MSB	6	5	4	3	2	1	LSB									
	LSB First	16 Bits	LSB	1	2	3	4	5	6	7	8	9	10	MSB	0	0	0	0	
		12 Bits	LSB	1	2	3	4	5	6	7	8	9	10	MSB					
		8 Bits	LSB	1	2	3	4	5	6	MSB									

The falling edge of SCLK strobes out the digital word on DO when $\overline{CS}$ is low. The digital word length will vary in accord with the digital word format. Thus for 8-bits + sign resolution 9 clock cycles are required.

As shown in the timing diagram (*Figure 1*), the acquisition time (the period of time during which the analog input is being sampled) starts on the falling edge of the last data clock cycle. For 16 bits of data that would be the 16th clock; for 8 bits of data that would be the 8th clock. The length of the acquisition time may be programmed by the user with an instruction, (see *Table 4*). The acquisition time can be set to 6, 10, 18, or 34 CCLK cycles.

1.3 SELECTING OUTPUT WORD FORMAT AND MODE

While $\overline{CS}$ is low, the rising edge of SCLK strobes in the data bits DI0–DI7 on the DI control line. For the ADC12038, the values of DI0–DI7 determine the digital output word format, mode select, and multiplexer configuration. For the ADC12034, 7 bits of data (DI0–DI6) are required. The ADC12032, and ADC12030 require only 6 bits of data (DI0–DI5). Mode Select determines the number of clock periods for the acquisition time (t_A), software power up/down, Auto Cal, Auto Zero and other functions as shown in *Table 4*.

1.0 General Overview (Continued)

TABLE 4. Mode Programing

ADC12038	DI0	DI1	DI2	DI3	DI4	DI5	DI6	DI7	Mode Select (Current)	DO Format (next Conversion Cycle)
ADC12034	DI0	DI1	DI2		DI3	DI4	DI5	DI6		
ADC12030 and ADC12032	DI0	DI1			DI2	DI3	DI4	DI5		
	MUX Address see *Tables 5, 6* or *Table 7*				L	L	L	L	12-Bit Conversion	12- or 13-Bit MSB First
	MUX Address see *Tables 5, 6* or *Table 7*				L	L	L	H	12-Bit Conversion	16-Bit MSB First
	MUX Address see *Tables 5, 6* or *Table 7*				L	L	H	L	8-Bit Conversion	8- or 9-Bit MSB First
	L	L	L	L	L	L	H	H	12-Bit Conversion of Full-Scale	12- or 13-Bit MSB First
	MUX Address see *Tables 5, 6* or *Table 7*				L	H	L	L	12-Bit Conversion	12- or 13-Bit LSB First
	MUX Address see *Tables 5, 6* or *Table 7*				L	H	L	H	12-Bit Conversion	16-Bit LSB First
	MUX Address see *Tables 5, 6* or *Table 7*				L	H	H	L	8-Bit Conversion	8- or 9-Bit LSB First
	L	L	L	L	L	H	H	H	12-Bit Conversion of Offset	12- or 13-Bit LSB First
	L	L	L	L	H	L	L	L	Auto Cal	No Change
	L	L	L	L	H	L	L	H	Auto Zero	No Change
	L	L	L	L	H	L	H	L	Power Up	No Change
	L	L	L	L	H	L	H	H	Power Down	No Change
	L	L	L	L	H	H	L	L	Read Status Register (LSB First)	No Change
	L	L	L	L	H	H	L	H	Data Out without Sign	No Change
	H	L	L	L	H	H	L	H	Data Out with Sign	No Change
	L	L	L	L	H	H	H	L	Acquisition Time—4 CCLK Cycles	No Change
	L	H	L	L	H	H	H	L	Acquisition Time—8 CCLK Cycles	No Change
	H	L	L	L	H	H	H	L	Acquisition Time—16 CCLK Cycles	No Change
	H	H	L	L	H	H	H	L	Acquisition Time—32 CCLK Cycles	No Change
	L	L	L	L	H	H	H	H	User Mode	No Change
	H	L	L	L	H	H	H	H	Test Mode (CH1–CH7 become Active Outputs)	No Change

Note: The A/D powers up with No CAL, No Auto-Zero, 10 CCLK Cycles Acquisition time, sign bit on, 13-bit MSB First format, power up, and user mode.

1.0 General Overview (Continued)

1.4 MULTIPLEXER ADDRESSING

The analog input channel configuration is selected during mode programming using the "MUX address" bits in *Table 4*. These bits and their effects are defined in *Tables 5, 6, 7*.

TABLE 5. ADC12038 Multiplexer Addressing

MUX Address				Analog Channel Addressed and Assignment with A/D IN1 tied to MUXOUT1 and A/D IN2 tied to MUXOUT2									A/D Input Polarity Assignment		Multiplexer Output Channel Assignment		Mode
DI0	DI1	DI2	DI3	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7	COM	A/D IN1	A/D IN2	MUXOUT1	MUXOUT2	
L	L	L	L	+	−								+	−	CH0	CH1	
L	L	L	H			+	−						+	−	CH2	CH3	
L	L	H	L					+	−				+	−	CH4	CH5	
L	L	H	H							+	−		+	−	CH6	CH7	Differential
L	H	L	L	−	+								−	+	CH0	CH1	
L	H	L	H			−	+						−	+	CH2	CH3	
L	H	H	L					−	+				−	+	CH4	CH5	
L	H	H	H							−	+		−	+	CH6	CH7	
H	L	L	L	+								−	+	−	CH0	COM	
H	L	L	H			+						−	+	−	CH2	COM	
H	L	H	L					+				−	+	−	CH4	COM	
H	L	H	H							+		−	+	−	CH6	COM	Single-Ended
H	H	L	L		+							−	+	−	CH1	COM	
H	H	L	H				+					−	+	−	CH3	COM	
H	H	H	L						+			−	+	−	CH5	COM	
H	H	H	H								+	−	+	−	CH7	COM	

TABLE 6. ADC12034 Multiplexer Addressing

MUX Address			Analog Channel Addressed and Assignment with A/D IN1 tied to MUXOUT1 and A/D IN2 tied to MUXOUT2					A/D Input Polarity Assignment		Multiplexer Output Channel Assignment		Mode
DI0	DI1	DI2	CH0	CH1	CH2	CH3	COM	A/D IN1	A/D IN2	MUXOUT1	MUXOUT2	
L	L	L	+	−				+	−	CH0	CH1	
L	L	H			+	−		+	−	CH2	CH3	Differential
L	H	L	−	+				−	+	CH0	CH1	
L	H	H			−	+		−	+	CH2	CH3	
H	L	L	+				−	+	−	CH0	COM	
H	L	H			+		−	+	−	CH2	COM	Single-
H	H	L		+			−	+	−	CH1	COM	Ended
H	H	H				+	−	+	−	CH3	COM	

1.0 General Overview (Continued)

TABLE 7. ADC12032 and ADC12030 Multiplexer Addressing

MUX Address		Analog Channel Addressed and Assignment with A/D IN1 tied to MUXOUT1 and A/D IN2 tied to MUXOUT2			A/D Input Polarity Assignment		Multiplexer Output Channel Assignment		Mode
DI0	DI1	CH0	CH1	COM	A/D IN1	A/D IN2	MUXOUT1	MUXOUT2	
L	L	+	−		+	−	CH0	CH1	Differential
L	H	−	+		−	+	CH0	CH1	
H	L	+		−	+	−	CH0	COM	Single-
H	H		+	−	+	−	CH1	COM	Ended

Note: MUXOUT1, MUXOUT2, A/D IN1 and A/D IN2 pins are not available on the ADC12030. A/D IN1 is tied internally to MUXOUT1: A/D IN2 is tied internally to MUXOUT2.

As can be seen in the tables, 4, 3 or 2 bits of the serial digital input word control the channel selection. These bits are part of an 8-, 7-, or 6-bit serial word that controls the function of the devices.

1.5 STATUS REGISTER DEFINITION

On request, the ADC12038 provides status information indicating power up or power down status, output data format, Auto-Cal status, and User/Test Mode status. *Table 8* defines the digital output data obtained after requesting a "Status Read".

When $\overline{CS}$ is used it is not necessary to clock all the status bits out. $\overline{CS}$ may be brought high at any time to restart a new serial data communication.

1.6 PROGRAMMING PROCEDURE

The example in *Figure 2* shows a typical sequence of events after power is applied to the ADC12038:

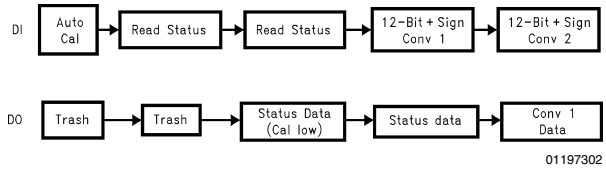

FIGURE 2. Typical Instruction Sequence after Power Up

The first instruction to the ADC via DI initiates Auto Cal. The data output on DO at that time is meaningless and is completely random. To obtain the specified accuracy of the device it is necessary to issue an Auto Cal instruction after the power supply and reference voltage to the device have been given enough time to stabilize. The Auto Cal instruction initiates an internal calibration sequence without which the specified accuracy of the device would be unattainable. To determine whether the Auto Cal has been completed, a Read Status instruction is issued to the device. Again, the data obtained while issuing the Read Status instruction has no significance since the Auto Cal instruction modifies the data in the output shift register. To retrieve the status information an additional read status instruction is issued to the ADC. At this time the status data is available on DO. If the Cal signal in the status word is low, Auto Cal has been completed. Therefore, the next instruction issued can start a conversion. The data output, while clocking in the "start conversion request", is again status information. Status can not be read during a conversion. To preserve the integrity of the A/D conversion, there is no end of conversion bit in the status word. If $\overline{CS}$ is brought low during a conversion, that conversion is stopped and never completed. EOC can be used to determine the end of a conversion or the A/D controller can keep track in software of when it would be appropriate to communicate to the A/D again. Once it has been determined that the A/D has completed a conversion another instruction can be transmitted to the A/D. The data from this conversion can be accessed when the next instruction is issued to the A/D.

1.0 General Overview (Continued)

TABLE 8. Status Register

Status Bit Location	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	DB8
Status Bit	PU	PD	Cal	8 or 9	12 or 13	16 or 17	Sign	Justification	Test Mode
	Device Status			DO Output Format Status					
Function	"High" indicates a Power Up State	"High" indicates a Power Down State	"High" indicates an Auto-Cal Sequence is in progress	"High" indicates an 8- or 9-bit format	"High" indicates a 12- or 13-bit format	"High" indicates a 16- or 17-bit format	"High" indicates that the sign bit is included. When "Low", the sign bit is not included.	When "High", the conversion result will be output MSB first. When "Low", the result will be output LSB first.	When "High", the device is in test mode. When "Low", the device is in user mode.

2.0 General Flow Chart for a Microcontroller Interface

Below is a flow chart that can be used for a microcontroller interface to the ADC12038. The data required by the ADC12038 is given in parentheses.

The timing diagrams shown to the right are suggested for each instruction issued to the ADC.

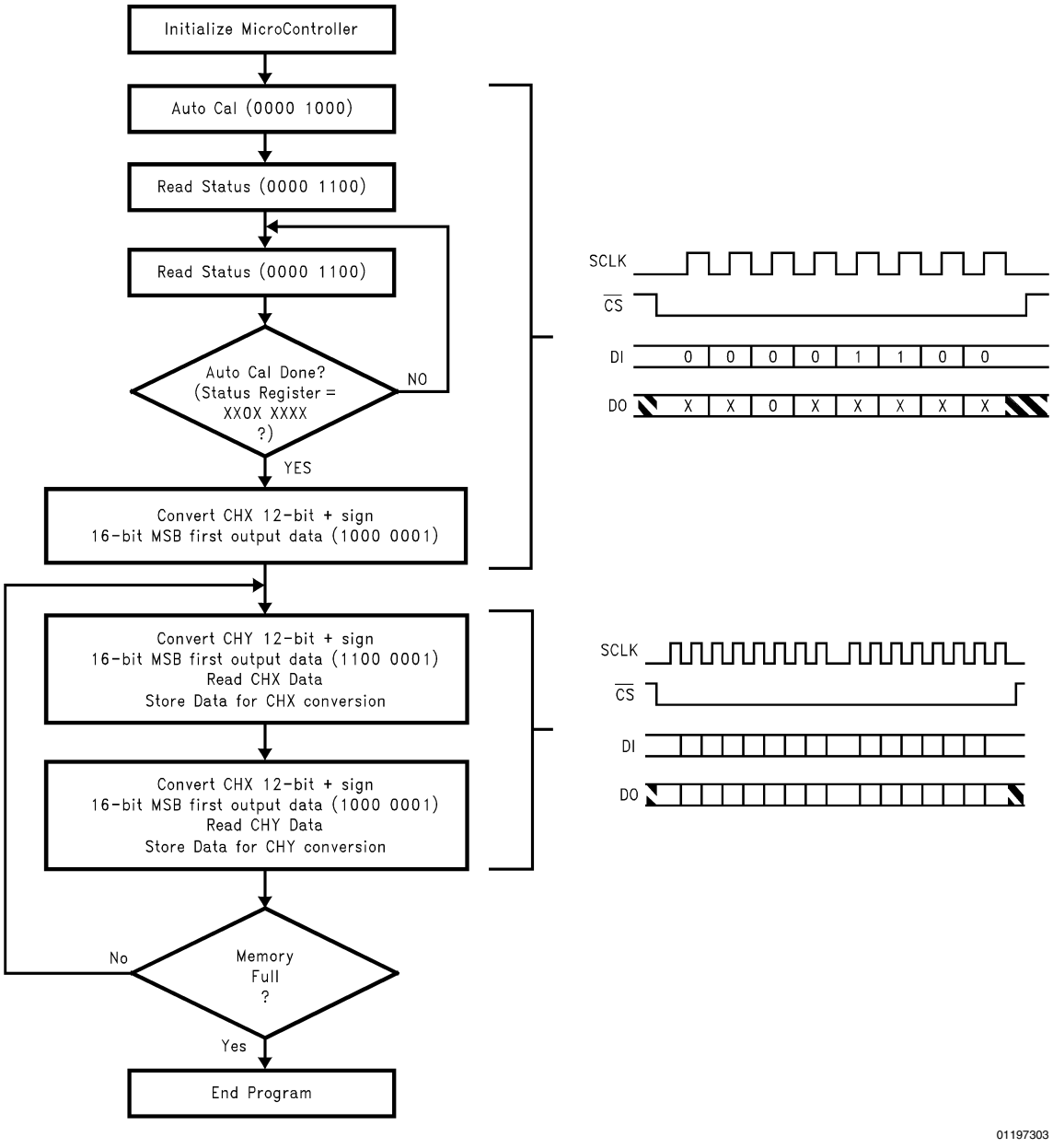

FIGURE 3. ADC12038 Program Flow Chart and Timing

3.0 Examples of Microcontroller Hardware Implementations

3.1 THE 68HC11

Figure 4 shows the hardware interface to a Motorola M68HC11 microcontroller. Motorola's SPI (Serial Peripheral Interface) SCK, MISO, and MOSI lines are directly tied to the SCLK, DO and DI of the ADC12038. Port B bit 0 is used to generate the $\overline{CS}$ to the ADC.

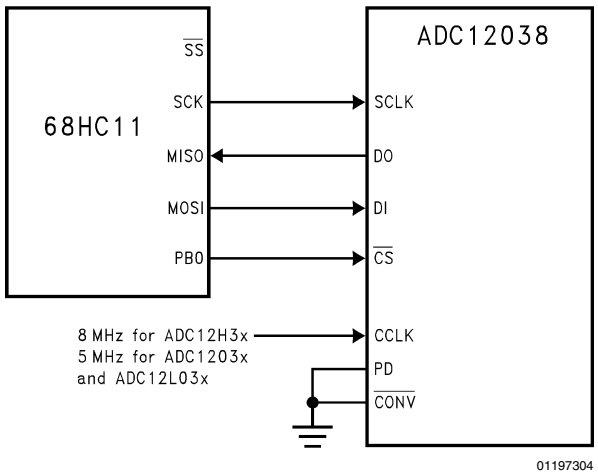

FIGURE 4. 68HC11 ADC12038 Hardware Interface

3.2 NATIONAL'S HPC AND COP

The serial I/O for these devices is configured to comply with the NSC MICROWIRE™ serial data exchange standard for easy interface to the COPS™ and HPC™ families of controllers. The output data format is software-programmable, making the serial interface extremely flexible and an ideal choice for many applications. Shown in *Figure 5* is an implementation of an National Semiconductor HPC microcontroller interface. The SK (Serial clocK), SI (Serial Input data) and SO (Serial Output data) lines of the HPC, used in Nationals MICROWIRE interface, are tied directly to the ADC12038. Port B, bit 6 is used to generate a $\overline{CS}$ for the ADC.

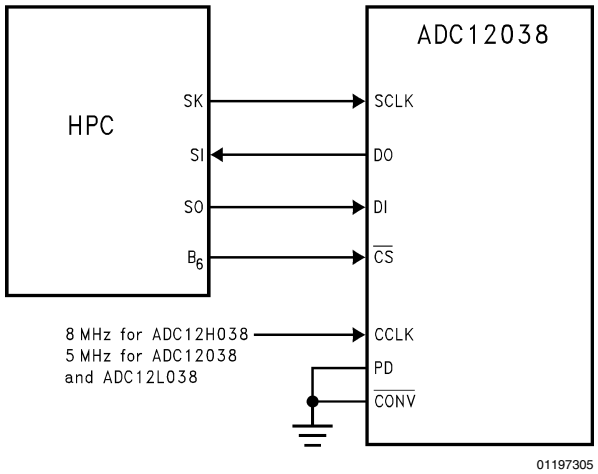

FIGURE 5. HPC to ADC12038 Hardware Interface

3.3 THE 8051

Figure 6 shows the ADC12038 connected to an Intel 8051. The 8051 serial interface does not support the protocol of the serial interface for this device. Therefore three port lines from the 8051 (P1.0, P1.1 and P1.2) can be used to talk to the ADC. The software toggles these lines directly to form the signals necessary to control the ADC.

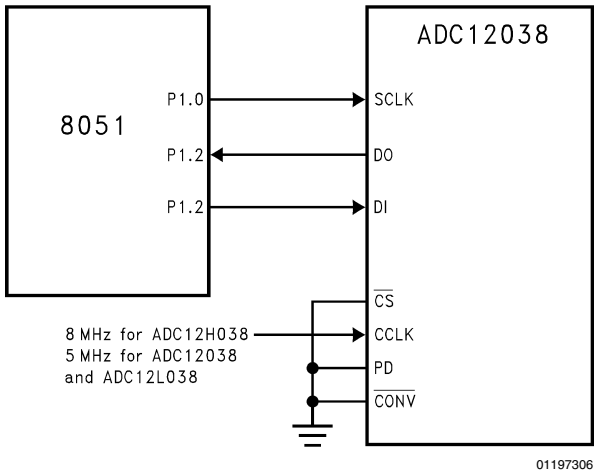

FIGURE 6. 8051 ADC12038 Hardware Interface

4.0 68HC11 SPI Interface

This section will describe in detail an SPI interface to the ADC12038. *Figure 7*, shown below, is a detailed schematic of the interface. The Motorola M68HC11EVB evaluation board was used to verify the program included at the end of this section. Therefore, the schematic shown here shows the connections required to the 68HC11 evaluation board.

4.1 68HC11 SPI PORT AND REGISTER INITIALIZATION FOR THE ADC12038

The 68HC11 SPI (Serial Peripheral Interchange) interface is ideal for driving the ADC12038. The SCK, MISO, and MOSI lines of the SPI tie directly to the SCLK, DO and DI lines of the ADC. $\overline{CS}$ for the ADC is generated using a line of the 68HC11's output port B. Here is a brief overview of the 68HC11 ports and registers used by the SPI.

The 68HC11 has four I/O ports. Port D can be set up as a general purpose I/O port or it can be used for the SPI interface and SCI (Serial Control Interface). The signal assignments for port D when used for SPI or SCI follow:

PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
X	X	SS	SCK	MOSI	MISO	TXD	RXD

SS (Slave Select), SCK (Serial ClocK), MOSI (Master Output Slave Input), MISO (Master Input Slave Output) are used for the SPI. SCI uses TXD and RXD.

There are two registers in the 68HC11 that need to be initialized: the DDRD (Data Direction Register for port D) and the SPCR (Serial Peripheral Control Register).

4.0 68HC11 SPI Interface (Continued)

4.1.1 DDRD

If ones are placed in the locations corresponding to the signal assignments for port D, those signals will be selected as outputs (except for the SS location). A one placed in the SS location disables that function. The SS input can be used for synchronizing master/slave communications between 68HC11s on the SPI bus. If SS is enabled and the 68HC11 is set as master then the SS input should be hard wired to a logical "high" for our case. Shown below is the data required to initialize DDRD for the ADC12038 interface. SS is disabled; SCK and MOSI are set as outputs; MISO is set as an input. TXD and RXD are not used but are set as input and output.

PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
X	X	1	1	1	0	1	0

DDRD resides at address i009. On the 68HC11 development board this address is 1009. All register addresses on the development board start at 1000. 0000 through 0FFF are used by the software that controls the development board. In an actual system the registers can be remapped to any 4k boundary by software.

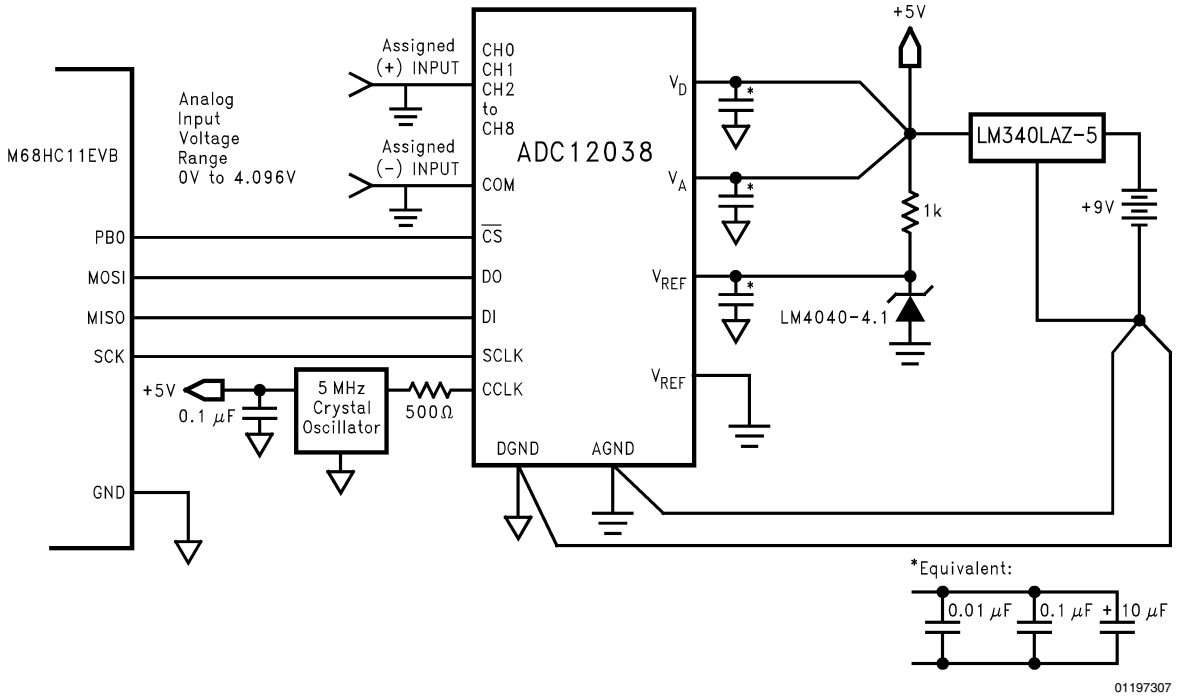

FIGURE 7. Detailed Schematic of ADC12038 to M68HC11EVB Interface

4.1.2 SPCR

On power up the SPI is disabled. The data stored in the SPCR (Serial Peripheral Control Register) controls how the SPI functions. SPCR resides at address 1028 for the development board. The table below summarizes the functions of the bits in this register. The SPIE (Serial Peripheral Interrupt Enable) bit when set to 1 allows the use of an interrupt to signal when an I/O exchange has completed. The SPE (Serial Peripheral Enable) bit when set to 1 enables the SPI. DWOM bit when set to 1 sets the outputs of port D to open drain. When this bit is low port D has totem pole outputs. MSTR bit controls whether this 68HC11 is a master or slave. When set to a 1 the 68HC11 is set as a master. In the slave mode SCK is an input. The CPOL and CPHA control the inactive level of the SCK output as well as which edge of the SCK output strobes the data out or in on the MISO or MOSI pins of port D. With both these bits set low the timing is as shown in *Figure 8*.

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0
0	1	0	1	0	0	0	0

The 68HC11 clocks in the data on MISO using the rising edge of SCK. Data on MOSI changes on the falling edge of SCK. This timing matches what the ADC12038 expects. SPR1 and SPR2 control the frequency of SCK as shown in *Table 9*.

4.0 68HC11 SPI Interface (Continued)

TABLE 9. SCK Frequency Control

XTAL Frequency	Internal Processor Clock	SCK Frequency	Internal Processor Clock Divide by	SPR1	SPR0
8 MHz	2 MHz	1 MHz	2	0	0
8 MHz	2 MHz	500 kHz	4	0	1
8 MHz	2 MHz	250 kHz	8	1	0
8 MHz	2 MHz	125 kHz	16	1	1

The SPSR (SPI Status Register) logs the status of the SPI I/O interchange. The only bit that is of concern is the SPIF which when set signals that the SPI interchange is complete.

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SPIF							

SPDR (SPI Data Register) is an 8-bit register used to exchange input and output data on the SPI. A write to this register will initiate an SPI exchange. The data input to the 68HC11 after an SPI exchange will reside in this register. This register resides at address 1029 for the development system.

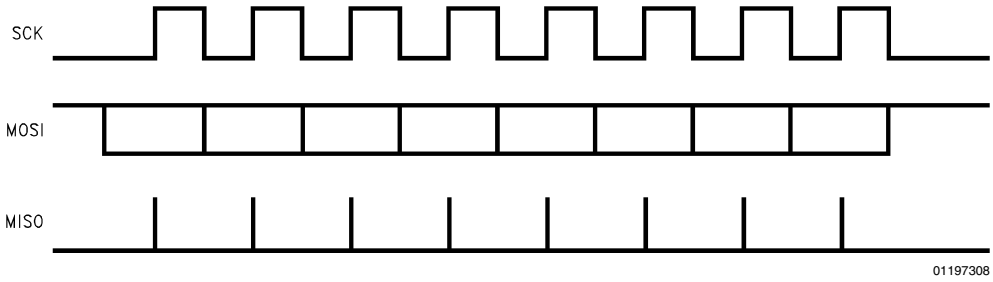

FIGURE 8. SPI Timing Diagram Required for the ADC12038

4.0 68HC11 SPI Interface (Continued)

4.2 68HC11 Program Listing

The following program listing follows the flow chart given in Section 2.0.

```
0001*****************************************************************************************
0002                        *       Emmy Denton                 3/4/93
0003                        *
0004                        *       ADC12'38 MC68HC11 SPI Interface
0005                        *
0006                        *
0007                        *       This program
0008                        *               1. Initializes the SPI interface
0009                        *               2. Starts a self calibration
0010                        *               3. Fills memory locations C200-C2FF with
0011                        *                conversions of CH0 and CH1 set up as single ended,
0012                        *                12-bit +sign MSB first
0013                        *
0014*****************************************************************************************
0015                        *
0016 0081                   CH0CONV EQU     %10000001       ADC DI FOR CH0 CONVERSION
0017 00c1                   CH1CONV EQU     %11000001       ADC DI FOR CH1 CONVERSION
0018 0008                   CAL     EQU     $08             ADC DI FOR CALIBRAITON
0019 000c                   STATUS  EQU     $0C             ADC DI FOR STATUS READ
0020 c1ff                   STARTDATA EQU   $C1FF           START ADDRESS - 1 FOR CONVERSION RESULTS
0021 c2ff                   ENDDATA EQU     $C2FF           END ADDRESS FOR CONVERSION RESULTS
0022 1009                   DDRD    EQU     $1009           DATA DIRECTION REGISTER ADDRESS
0023 1028                   SPCR    EQU     $1028           SPI CONTROL REGISTER ADDRESS
0024 1029                   SPSR    EQU     $1029           SPI STATUS REGISTER ADDRESS
0025 102a                   SPDR    EQU     $102A           SPI DATA REGISTER ADDRESS
0026 1004                   PORTB   EQU     $1004           PORT B ADDRESS
0027 1008                   PORTD   EQU     $1008           PORT D ADDRESS (SPI OUTPUT)
0028
0029
0030 c000                           ORG     $C000           STARTING ADDRESS OF PROGRAM*
0031
0032
0033
0034
0035
0036                        *******
0037                        *       INITIALIZE SPI INTERFACE PORT
0038                        *******
0039 c000 86 20                     LDAA    #$20
0040 c002 b7 10 08                  STAA    PORTD           SET SCK TO 0, MISO TO 0, SS TO 1
0041
0042 c005 86 3a                     LDAA    #$3A            SET DDRD: DISABLE SS; SCK, MOSI, TXD - OUTPUTS;
0043 c007 b7 10 09                  STAA    DDRD                      MISO,RXD - INPUTS.
0044                        *
0045 c00a 86 50                     LDAA    #$50            SET SPCR
0046 c00c b7 10 28                  STAA    SPCR
0047
0048
0049
0050
0051
0052                        *******
0053                        *       INITIALIZE PORT B AND X INDEX REGISTER
0054                        *******
0055 c00f f6 10 04                  LDAB    PORTB           PLACE PORT B DATA INTO ACC B
0056 c012 ca 01                     ORAB    #$01            (BIT 0 OF PORT B IS ADC CS) SET CS OF ADC HIGH
0057 c014 f7 10 04                  STAB    PORTB
0058 c017 ce c1 ff                  LDX     #STARTDATA      SET X INDEX REGISTER TO START OF ADC DATA
0059                        *
0060
0061
0062
0063                        *
```

01197309

4.0 68HC11 SPI Interface (Continued)

```
0064                        *******
0065                        *       ADC MAIN PROGRAM
0066                        *******
0067                        *
0068
0069 c01a 86 08             MAIN    LDAA    #CAL
0070 c01c bd c0 53                  JSR     EBWRADC         ACC A JUNK START CALIBRATION
0071 c01f 86 0c                     LDAA    #STATUS         READ STATUS
0072 c021 bd c0 53                  JSR     EBWRADC         JUNK IN ACCUMULATOR
0073                        *
0074 c024 86 0c             CALWAIT LDAA    #STATUS
0075 c026 bd c0 53                  JSR     EBWRADC         READ ADC STATUS
0076 c029 84 20                     ANDA    #$20            MASK STATUS BIT
0077 c02b 26 f7                     BNE     CALWAIT         IF Z=1 JUMP TO CALWAIT
0078                        *
0079 c02d 86 81                     LDAA    #CH0CONV
0080 c02f bd c0 53                  JSR     EBWRADC         START CH0 CONVERSION DO IS JUNK
0081                        *
0082                        *
0083 c032 86 c1             CONV    LDAA    #CH1CONV        START CH1 CONVERSION
0084 c034 bd c0 6e                  JSR     SBWRADC
0085
0086 c037 08                        INX
0087 c038 a7 00                     STAA    0,X             STORE CH0 DATA
0088 c03a 08                        INX
0089 c03b e7 00                     STAB    0,X
0090
0091 c03d 86 81                     LDAA    #CH0CONV        START CH0 CONVERSION
0092 c03f bd c0 6e                  JSR     SBWRADC
0093
0094 c042 08                        INX
0095 c043 a7 00                     STAA    0,X             STORE CH1 DATA
0096 c045 08                        INX
0097 c046 e7 00                 STAB    0,X
0098
0099 c048 8c c2 ff                  CPX     #ENDDATA        IS MEMORY FOR DATA FULL
0100 c04b 26 e5                     BNE     CONV            IF NOT DO ANOTHER 2 CONVERSIONS
0101 c04d ce c1 ff                  LDX     #STARTDATA
0102 c050 01                        NOP
0103 c051 01                        NOP
0104 c052 01                        NOP
0105                        *
0106                                END
0107
0108
0109
0110
0111                        *******
0112                        *       EBWRADC - Subroutine to Output/Input 8 bits to/from ADC (SPI port)
0113                        *       UPON ENTERING SUBROUTINe - ACCUMULATOR A HAS DATA TO OUTPUT TO ADC
0114                        *       UPON EXITING SUBROUTINE - ACCUMULATOR A HAS DATA FROM ADC
0115                        *******
0116 c053 f6 10 04          EBWRADC LDAB    PORTB           READ PREVIOUSE SETTING OF PORTB
0117 c056 c4 fe                     ANDB    #$FE            SET CS LOW (BIT 0 OF PORTB)
0118 c058 f7 10 04                  STAB    PORTB
0119 c05b b7 10 2a                  STAA    SPDR            WRITE ACCUMULATOR A TO SPI PORT AND READ SPI
0120
0121 c05e b6 10 29          SPIWTA  LDAA    SPSR            WAIT FOR SPI INTERFACE
0122 c061 84 80                     ANDA    #$80            AND RESET SPI FOR ANOTHER TIMING SEQUENCE
0123 c063 27 f9                     BEQ     SPIWTA
0124
0125 c065 b6 10 2a                  LDAA    SPDR            LOAD SPI DATA INTO ACCUMULATOR A
0126 c068 ca 01                     ORAB    #$01            SET CS HIGH
0127 c06a f7 10 04                  STAB    PORTB
0128 c06d 39                        RTS
0129
0130
0131
0132
```

01197310

4.0 68HC11 SPI Interface (Continued)

```
0133                        *
0134                        *******
0135                        *       SBWRADC - Subroutine to Output/Input 16 bits to/from ADC (SPI port)
0136                        *       UPON ENTERING SUBROUTINE - ACCUMULATOR A and B HAVE DATA TO OUTPUT TO ADC
0137                        *       UPON EXITING SUBROUTINE - ACCUMULATOR A AND B HAVE DATA FROM ADC
0138                        *******
0139
0140
0141 c06e f6 10 04          SBWRADC LDAB    PORTB           READ PREVIOUS SETTING OF PORTB
0142 c071 c4 fe             ANDB    #$FE            SET CS LOW (BIT 0 OF PORTB)
0143 c073 f7 10 04          STAB    PORTB
0144 c076 b7 10 2a          STAA    SPDR            WRITE ACCUMULATOR A TO SPI PORT AND READ BYTE 1
0145
0146 c079 b6 10 29          SPIWTB  LDAA    SPSR            WAIT FOR SPI INTERFACE
0147 c07c 84 80             ANDA    #$80            AND RESET SPI FOR ANOTHER TIMING SEQUENCE
0148 c07e 27 f9             BEQ     SPIWTB
0149
0150 c080 f6 10 2a          LDAB    SPDR            LOAD SPI DATA (BYTE 1) INTO ACCUMULATOR B
0151 c083 b7 10 2a          STAA    SPDR            WRITE ACCUMULATOR A TO SPI PORT AND READ BYTE 2
0152
0153 c086 b6 10 29          SPIWTC  LDAA    SPSR            WAIT FOR SPI INTERFACE
0154 c089 84 80             ANDA    #$80            AND RESET SPI FOR ANOTHER TIMING SEQUENCE
0155 c08b 27 f9             BEQ     SPIWTC
0156
0157 c08d b6 10 04          LDAA    PORTB           SET CS HIGH
0158 c090 8a 01             ORAA    #$01
0159 c092 b7 10 04          STAA    PORTB
0160 c095 b6 10 2a          LDAA    SPDR            LOAD SPI DATA (BYTE 2) INTO ACCUMULATOR A
0161
0162 c098 39                RTS
0163                        *
```

01197311

5.0 References

National Serniconductor Microcontroller Databook

Motorola MC68HC11 Reference Manual
#M68HC11RM/AD

Motorola MC68HC11A8/1/0 Data Sheet

"Design with Microcontrollers" John B. Peatman

Intel 8051 Databook

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
Americas
Email: support@nsc.com

www.national.com

National Semiconductor Europe
Fax: +49 (0) 180-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 69 9508 6208
English Tel: +44 (0) 870 24 0 2171
Français Tel: +33 (0) 1 41 91 8790

National Semiconductor Asia Pacific Customer Response Group
Tel: 65-2544466
Fax: 65-2504466
Email: ap.support@nsc.com

National Semiconductor Japan Ltd.
Tel: 81-3-5639-7560
Fax: 81-3-5639-7507

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.