Software for Interfacing the COP8™ Family Microcontrollers to National's MICROWIRE™ EEPROMs

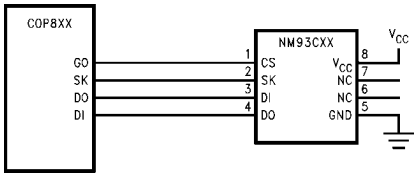
National Semiconductor
Application Note 841
Paul Bryant
July 1996

ABSTRACT

National's NM93Cxx and NM93CSxx family of serial EEPROMs have MICROWIRE ''slave'' interfaces that directly connect to the MICROWIRE ''master'' interfaces on the COP8™ family of 8-bit microcontrollers. Peak data transfer rates are as high as 1 MB on the 4 wire MICROWIRE bus. This application note includes the essential assembly language software to address MICROWIRE EEPROMs.

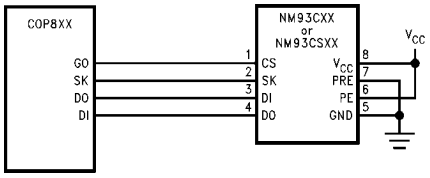
HARDWARE INTERFACE

A schematic for connecting a COP8 family microcontroller to one of Nationals NM93Cxx family MICROWIRE EEPROMs via the microcontrollers dedicated MICROWIRE port is shown in *Figure 1*. National makes two basic families of MICROWIRE EEPROMs, the classic NM93**C**xx series and the newer full featured NM93**CS**xx series. The NM93CSxx series EEPROMs have a sequential read capability and the ''S'' in ''CS'' stands for sequential (the ''C'' implies CMOS). As can be seen the ''CS'' series devices have two additional pins which control write protect features (consult a data sheet for information on their function). By connection these pins to V_{CC} and GND as shown in *Figure 2*, it becomes possible to use either ''C'' or ''CS'' family parts in the same physical socket. This would be desirable if a change from a ''C'' series part to a ''CS'' series part is possible for reasons of product upgrades or simply manufacturing inventory control. Pins 6 and 7 on the ''C'' series parts are true no connects and thus can be tied to V_{CC} or ground harmlessly.



TL/D/11491–1

FIGURE 1. Basic National MICROWIRE Interconnection to a COP8 Family Part



TL/D/11491–2

FIGURE 2. This schematic allows either a NM93Cxx or NM93CSxx part to be used in the same socket. Software can then be adjusted to take advantage of the ''CS'' series advantages without making changes to the board.

If it's desirable to use both types in the same socket without being forced to make software changes, one must be careful not to use the sequential read capability of the ''CS'' series. Both types of parts should be tested in the socket before the software is frozen.

NM93C06 to COP8 Family Software Details

Always consult the latest data sheets for information about timing variables mentioned in the text that follows. These numbers were correct at the time that this application note was written but are subject to change.

1. The SK clock frequency must not exceed 1 MHz. Consult the processor data sheet for details.
2. The CS low time following a write must exceed 250 ns. This starts the internally timed write operation. The DO line will leave the high impedance state if CS goes high again and will drive low until the internal write cycle is complete. After DO returns high, indicating ''ready'' the first rising edge of SK with CS high and DI high will return the DO pin to the high impedance condition. This condition is normally the start bit of the next instruction.

   The DO pin will be low for up to 10 ms and then go high to indicate that the write is complete. If a new instruction is attempted before the DO pin returns high it will be ignored and the DO pin will not go tristate. The DO pin will always go to the tristate condition when CS is low.
3. Opcodes are either 2-bits or 4-bits long depending on the instruction type and are always preceded by a ''start-bit'' of a logic one. Any number of leading zeros can be clocked in before the start-bit (the sample assembly code inserts seven). Addresses are either 6 or 8-bits long depending on the density of the device. The combined opcode and address field is 8-bits for the smaller devices (93C06 and 93C46) and 10-bits for the larger devices (93C56 and 93C66). On the opcode types that do not use addresses, all of the ''dummy'' address bits must be clocked anyway (the combined opcode/address field is constant number of clock cycles).
4. On read operations the data out stream starts with a dummy zero. On NM93Cxx family EEPROMs, it is acceptable but not required to have extra clocks after the 16th actual data bit. On NM93CSxx family EEPROMs, extra clocks after the 16th actual data bit will begin to read the next data word.

Notes on the Assembly Code:

The subroutines that follow are adequate to quickly pilot the programmers task of addressing a serial EEPROM of the NM93Cxx family. Additional subroutines can very easily be adapted from these to handle the additional opcode types of the NM93CSxx series parts. Enough code has been included to allow the code to operate in a stand-alone fashion. However, when integrating the routines in to another program, initialization statements affecting global variables such as initializing the stack point or the X or B registers will need to be moved, deleted or replaced by statements in the main program.

The assembly code uses a software timer loop to time out the write time of the EEPROM. The programmer should be aware that it is possible to use the EEPROMs own internal timer to accomplish this task. This is done by monitoring the EEPROMs DO line after taking the EEPROMs CS line low to start a write and then setting CS high again to re-enable the DO output. The write is complete when the DO (of the EEPROM) drives high. Using the EEPROMs internal timer will allows the microcontroller time to accomplish some other task in the 10 ms that the write or erase operation requires. If the DO line is to be used to indicate that the write is complete, other MICROWIRE components on the bus must wait for the EEPROM writes to time out before being accessed (the DO line is in use).

The code was tested on a COP820 device via a Metalink In Circuit Emulator. The code should translate to other COP8 devices with little or no modification.

NM93C06 and NM93C46 Opcodes and Address Fields*

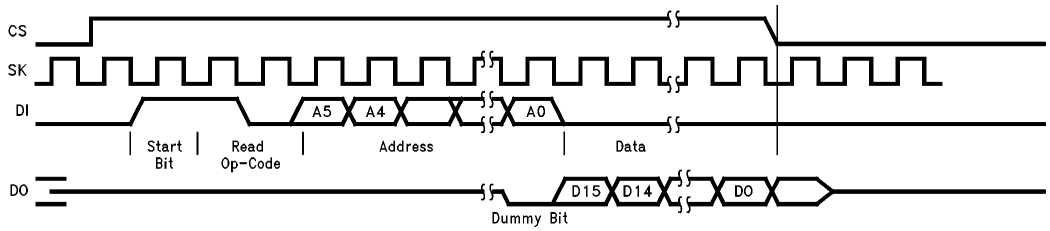
WREN	Write Enable	0	0	1	1	X	X	X	X
WRDI	Write Disable	0	0	0	0	X	X	X	X
ERAL	Erase All	0	0	1	0	X	X	X	X
WRAL	Write All	0	0	0	1	X	X	X	X
READ	Read	1	0	A5	A4	A3	A2	A1	A0
WRITE	Write	0	1	A5	A4	A3	A2	A1	A0

NM93C56 and NM93C66 Opcodes and Address Fields*

WREN	Write Enable	0	0	1	1	X	X	X	X	X	X
WRDI	Write Disable	0	0	0	0	X	X	X	X	X	X
ERAL	Erase All	0	0	1	0	X	X	X	X	X	X
WRAL	Write All	0	0	0	1	X	X	X	X	X	X
READ	Read	1	0	A7	A6	A5	A4	A3	A2	A1	A0
WRITE	Write	0	1	A7	A6	A5	A4	A3	A2	A1	A0

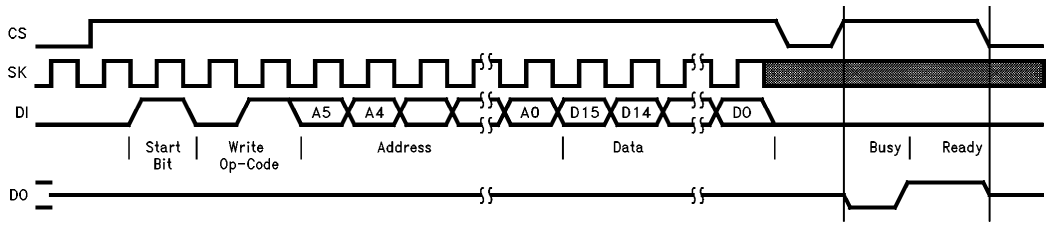
***Note:** All Opcode/Address Fields must be preceded with a leading ''1'' as a start-bit.

Read Cycle



TL/D/11491–3

Write Cycle



TL/D/11491–4

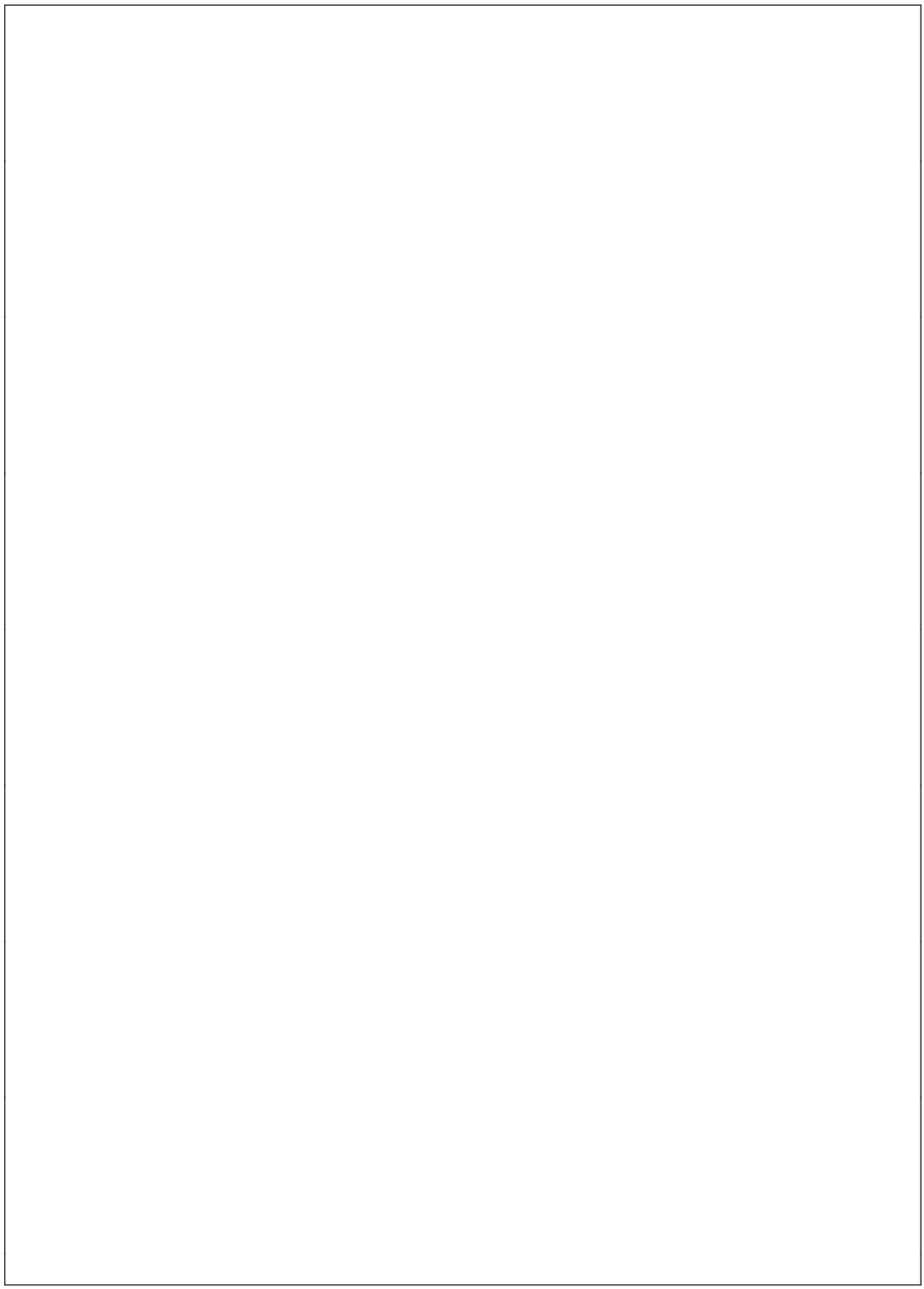
FIGURE 3. Read and Write cycle waveforms. Notice that one leading zero is shown before the start-bit. The actual code inserts seven.

```
;*******************************************************
; THIS PROGRAM PROVIDES SUBROUTINES TO HANDLE COP820 OPERATIONS ON
; THE NM93C06 EEPROM I.E., WRITES, READS, ERASES, ENABLES AND DISABLES
;*******************************************************
;
 .INCLD COP820.INC
;Reserving RAM locations for key variables
 RDATL = 1  ;LOWER BYTE OF THE NM93C06 MEMORY DATA READ
 RDATH = 2  ;UPPER BYTE OF THE NM93C06 MEMORY DATA READ
 WDATL = 3  ;LOWER BYTE OF THE DATA TO BE WRITTEN TO NM93C06
 WDATH = 4  ;UPPER BYTE OF THE DATA TO BE WRITTEN TO NM93C06
 ADRESS = 5 ;THE LOWER 4-BITS OF THIS LOCATION CONTAINS THE ADDRESS
            ;OF THE NM93C06 MEMORY LOCATIONS TO BE READ/WRITTEN
            ;THE UPPER NIBBLE MUST BE ZEROS
 SNDBUF = 0 ;USED FOR THE COMMAND BYTE TO BE WRITTEN (Local Scratch Pad)
 DLYH = 0F0 ;LOCATIONS RESERVED FOR WRITE TIMEOUT VALUES
 DLYL = 0F1
 FLAGS = 6  ;USED FOR PROGRAM FLAGS (Local Scratch Pad)
            ;
            ;FLAG VALUE DEFINITIONS
            ;00        ERASE, ENABLE, DISABLE, ERASE ALL
            ;01        READ CONTENTS OF NM93C06 REGISTER
            ;03        WRITE TO NM93C06 REGISTER
            ;OTHERS    ILLEGAL COMBINATION
```

```
;THE INTERFACE BETWEEN THE COP820C/840C AND THE NM93C06 (256-BIT EEPROM)
;CONSISTS OF FOUR LINES. THE G0(CHIP SELECT LINE), G4(SERIAL OUT SO),
;G5(SERIAL CLOCK SK) AND G6(SERIAL IN SI).
;**************************************
;
;INITIALIZATION, MODIFY MOVE OR DELETE WHEN INTEGRATING INTO MAIN PROGRAM
                      ;USE ONLY IF SP WAS NOT PREVIOUSLY INITIALIZED
       LD  SP,#02F    ;INITIALIZE STACK POINTER
       LD  PORTGC,#031;SETUP G0, G4, G5 AS OUTPUT
       LD  PORTGD,#000;INITIALIZE G DATA REG TO ZERO
       LD  CNTRL,#008 ;ENABLE MSEL, SELECT MW RATE OF 2TC

       LD  X,#SIOR    ;SET THE X REGISTER TO POINT TO SIOR
       LD  B,#PSW     ;SET THE B REGISTER TO POINT TO PSW

;EXAMPLE SUBROUTINE CALLS ONLY, DO NOT INCLUDE IN FINAL CODE LOAD
;ADDRESS IN LOCATION "ADRESS" HIGH AND LOW BYTE TO BE WRITTEN INTO
;WDATH AND WDATL AND CALL THE SUBROUTINE,
       JSR EWEN
       JSR WRITE
       JSR EWDS
       JSR READ
DONE:  JP  DONE

;THIS ROUTINE ERASES THE MEMORY LOCATION POINTED TO BY THE ADDRESS
;CONTAINED IN THE LOCATION "ADRESS". THE LOWER NIBBLE OF THE VALUE
;IN THE LOCATION "ADRESS" IS THE NM93C06 REGISTER ADDRESS. THE UPPER
;NIBBLE SHOULD BE SET TO ZERO.
;

ERASE: LD  A,ADRESS
       OR  A,#0C0
       X   A,SNDBUF
       LD  FLAGS,#00
       JSR INIT
       RET
;
;THIS ROUTINE ENABLES PROGRAMMING THE NM93C06 (EWEN).
;
EWEN:  LD  SNDBUF,#030
       LD  FLAGS,#00
       JSR INIT
       RET
;
;THIS ROUTINE DISABLES PROGRAMMING OF NM93C06.
;
EWDS:  LD  SNDBUF,#00
       LD  FLAGS,#00
       JSR INIT
       RET
;
;THIS ROUTINE ERASES ALL REGISTERS OF NM93C06.
;
ERAL:  LD  SNDBUF,#020
       LD  FLAGS,#00
       JSR INIT
       RET
```

```
;
;THIS ROUTINE READS THE CONTENTS OF THE NM93C06 REGISTER. THE ADDRESS
;IS SPECIFIED IN THE LOWER NIBBLE OF LOCATION "ADRESS". THE UPPER
;NIBBLE SHOULD BE SET TO ZERO. THE 16-BIT CONTENTS OF NM93C06 REGISTER ARE
;STORED IN RDATL AND RDATH.
;
READ:  LD    A,ADRESS
       OR    A,#080
       X     A,SNDBUF
       LD    FLAGS,#01
       JSR   INIT
       RET
;
;THIS WRITES A 16-BIT VALUE STORED IN WDATL AND WDADTH TO THE EEPROM
;REGISTER WHOSE ADDRESS IS CONTAINED IN THE LOWER NIBBLE OF THE
;LOCATION "ADRESS". THE UPPER NIBBLE OF THE ADDRESS SHOULD BE SET TO ZERO.
;
WRITE: LD    A,ADRESS
       OR    A,#040
       X     A,SNDBUF
       LD    FLAGS,#03
       JSR   INIT
       RET
;
;THIS ROUTINE SENDS OUT THE START BIT AND COMMAND BYTE. IT ALSO
;DECIPHERS THE CONTENTS OF THE FLAG LOCATION AND MAKES A DECISION
;REGARDING WRITE, READ OR RETURN TO THE CALLING ROUTINE.
;
INIT:  SBIT  0,PORTGD    ;SET CHIP SELECT HIGH
       LD    SIOR,#001   ;LOAD SIOR WITH START BIT
       SBIT  BUSY,[B]    ;SEND OUT THE START BIT
PUNT1: IFBIT BUSY,[B]
       JP    PUNT1
       LD    A,SNDBUF
       X     A,[X]       ;LOAD SIOR WITH COMMAND BYTE
       SBIT  BUSY,[B]    ;SEND OUT COMMAND BYTE
PUNT2: IFBIT BUSY,[B]
       JP    PUNT2
       IFBIT 0,FLAGS     ;ANY FURTHER PROCESSING?
       JP    NOTDON      ;YES
       RBIT  0,PORTGD    ;NO, RESET CS AND RETURN
       RET
;
NOTDON:IFBIT 1,FLAGS     ;READ OR WRITE?
       JP    WR93C       ;JMP TO WRITE ROUTINE
       LD    SIOR,#000   ;NO READ NM93C06
       SBIT  BUSY,PSW    ;DUMMY CLOCK TO READ ZERO
       RBIT  BUSY,[B]
       SBIT  BUSY,[B]
PUNT3: IFBIT BUSY,[B]
       JP    PUNT3
       X     A,[X]
       SBIT  BUSY,[B]
       X     A,RDATH
```

```
PUNT4: IFBIT BUSY,[B]
       JP    PUNT4
       LD    A,[X]
       X     A,RDATL
       RBIT  0,PORTGD
       RET
WR93C: LD    A,WDATH
       X     A,[X]
       SBIT  BUSY,[B]
PUNT5: IFBIT BUSY,[B]
       JP    PUNT5
       LD    A,WDATL
       X     A,[X]
       SBIT  BUSY,[B]
PUNT6: IFBIT BUSY,[B]
       JP    PUNT6
       RBIT  0,PORTGD
       JSR   TOUT
       RET
;
;ROUTINE TO GENERATE DELAY FOR WRITE
;***************************************
;
TOUT:  LD    DLYH,#007   ;CHECK YOUR OSCILLATOR--PROCESSOR COMBINATION
                         ;TUNE FOR 10 MS DELAY
WAIT:  LD    DLYL,#0FF
WAIT1: DRSZ  DLYL
       JP    WAIT1
       DRSZ  DLYH
       JP    WAIT
       RET
       .END
```



Software for Interfacing the COP8 Family Microcontrollers to National's MICROWIRE EEPROMs

AN-841

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 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
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

http://www.national.com

National Semiconductor Europe
Fax: +49 (0) 180-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 180-530 85 85
English Tel: +49 (0) 180-532 78 32
Français Tel: +49 (0) 180-532 93 58
Italiano Tel: +49 (0) 180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2308
Fax: 81-043-299-2408