Protecting Data in Serial EEPROMs

Fairchild
Application Note 860

FAIRCHILD SEMICONDUCTOR™

TABLE OF CONTENTS

Serial Electrically Erasable PROMs (EEPROMs) are non-volatile memories whose contents can be changed by unique bit patterns called instructions, which are input to the memory array using serial data and clock pins. Serial EEPROMs operate from a single V_{CC} voltage supply (typically 2.7V–5.5V); an on-board charge pump provides the higher voltages required during the programming operation. These features offer the designer an easy to use and very flexible device. The ability to have a non-volatile memory whose contents can be changed "in-system" gives design engineers much greater flexibility. Serial EEPROMs provide a highly reliable and cost effective solution for a wide range of applications which need to store information such as calibration data, setting levels, and other user programmable data.

However, one factor a system designer needs to be aware of is the possibility of data corruption caused by "erroneous" or "false" data writes. Full featured EEPROMs (single voltage operation, self timed write cycle) can have data corruption problems due to noise spikes, glitches, bus contention, etc., which may initiate a false write or erase cycle. This data corruption is a concern for the designer since the non-volatile nature of the EEPROM means that after data corruption has occurred, it cannot be cleared simply by removing the power (for example as with a volatile memory such as SRAM).

This application note looks at the different types of serial EEPROMs and the techniques used by (a) the IC manufacturer, and (b) the system designer to overcome data corruption problems. The use of the three industry standard EEPROMs (MICROWIRE, I^2C and SPI) are discussed before an in-depth application example is presented for Fairchild Semiconductor's new NM25C04 SPI EEPROM.

2.0 SERIAL EEPROM INTERFACE STANDARDS

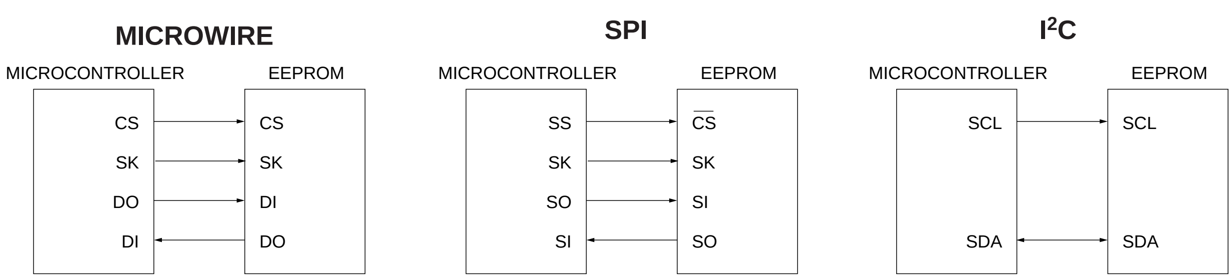
There are three main serial EEPROM interface standards; MICROWIRE™, I^2C™ and SPI™.

MICROWIRE is a three or four wire standard using a serial clock (SK), a Chip Select (CS), Data In (DI) and Data Out (DO) lines. These devices are available in standard form (NM93C06/46/56/66), security form (NM93CS06/46/56/66) or original form (NM93C46/56/66/86A).

I^2C (Inter-Integrated Circuit) is a two wire synchronous bus which uses SCL (clock) and SDA (data) to clock data between a master (for example a microcontroller) and a slave (the EEPROM). These devices are available in either standard form (NM24C02/04/08/16) or write protected form (NM24C03/05/09/17).

SPI (Serial Peripheral Interface) is a three or four wire synchronous bus which uses a chip select (CS), a serial clock (SCK), Data In (SI) and Data Out (SO) lines.

The three standardized serial interfaces are shown in Figure 1. The key specifications of these three interfaces are compared n Figure 2.



Note: If a standardized bus is not available, general purpose I/O port lines can be used to communicate with a serial device.
Fairchild Semiconductor has serial EEPROMs for all 3 standards

FIGURE 1. Serial EEPROM Interfaces

$I^2C^®$ is a registered trademark of Philips.
SPI™ is a trademark of Motorola Corporation.
GAL® is a registered trademark of Lattice Semiconductor Corporation.

	MICROWIRE NM93Cxx	SPI NM25Cxx	I^2C NM24Cxx
Max Bus Speed	1 MHz	2.1 MHz	400 KHz
No Active Pins	3 or 4	3 or 4	2
Max Memory Size	N/A	N/A	16 kbit
Largest Device	4 kbit¨16 kbit	16 kbit	16 kbit
Acknowledge	NO	NO	YES
Data Size	8 bits or 16 bits	8 bits	8
Block Write	NO	YES	YES
Sequential Read	YES (CS version)	YES	YES
No Device on Bus	Limited by Port Pins	Limited by Port Pins	Up to 16 kbits
Security Feature	YES (CS version)	YES	YES (03/05/09/17)

FIGURE 2. Serial EEPROM Bus Comparison

Serial EEPROM devices are available from Fairchild Semiconductor in all three industry standards, in a variety of sizes as shown in Figure 3.

	MICROWIRE			I^2C		SPI
	Standard	Security	Original	Standard	Write Protected	
256-bit	NM93C06	NM93CS06				
1 kbit	NM93C46	NM93CS46	NM93C46A			
2 kbit	NM93C56	NM93CS56	NM93C56A	NM24C02	NM24C08	NM25C020
4 kbit	NM93C66	NM93CS66	NM93C66A	NM24C04	NM24C05	NM25C040/04
8 kbit				NM24C08	NM24C09	
16 kbit			NM93C86A	NM24C16	NM24C17	NM25C160

Note: Contact Customer Support Center at (888) 522-5372 for latest details of availability.

FIGURE 3. Serial EEPROM Availability

3.0 ACCESSING SERIAL EEPROMs

No matter which type of serial interface standard used by an EEPROM, they all have two basic instructions: READ and WRITE data.

READ: This is a non-destructive instruction which reads data from the memory array.

WRITE: This is a destructive instruction. The data in the memory array is either erased or over written by the new data.

A typical set of EEPROM instructions, using the NM25C04 SPI EEPROM as an example is shown in Figure 4.

Instruction Name	Operation
WREN	Set Write Enable Latch
WRDI	Reset Write Enable Latch
RDSR	Read Status Register
WRSR	Write Status Register
READ	Read Data from Memory Array
WRITE	Write Data to Memory Array

Note: EEPROM powers-up in Write Disable Mode

Note: Must execute WREN before a Write instruction

Note: Status Register is used to:
— Poll READY/BUSY
— Set zone write protection ranges
— Indicate Write enable status

FIGURE 4. NM25C04 SPI EEPROM Instructions

In order to ensure data integrity, the system designer needs to understand the possible causes of inadvertent data writes which may cause data corruption, and the ways of overcoming this problem.

4.0 SOFTWARE WRITE PROTECT METHODS

All Fairchild Semiconductor EEPROMs incorporate common features to protect against inadvertent data writing to offer high reliability operation. A program disable mode is included which will ensure that devices power-up in the “Write Disable mode”. This means that unless the “Write Enable” instruction is executed, the EEPROM will abort any requested write or erase cycles. This is especially useful for protecting against data corruption during power transitions.

4.1 SPI EEPROMs (NM25C04)

Fairchild Semiconductor’s SPI EEPROMs such as the 4 kbit NM25C04 includes the following design features to guard against inadvertent data writes:

— Write Protect (WP) pin to disable memory writes

— Write Disable Instructions

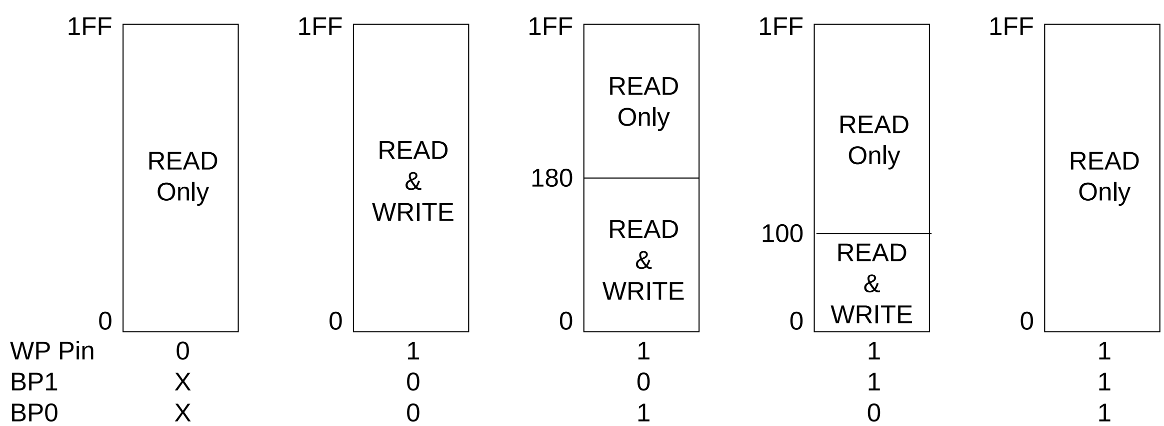
— Software write protection: the user can define a portion of the memory to be READ Only.

The various write protection configurations are shown in Figure 5.

4.2 MICROWIRE EEPROMs

All NM93CSxx devices have the security feature which allows the user to define a portion of the memory to be write protected, either permanently or temporarily. This is useful for storing secure information in a system, such as calibration data. To control the secure memory involves a combination of setting a hardware pin and various software instructions as shown in Figure 6.

Data Protect Features

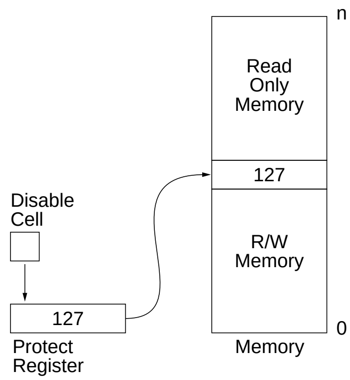


#1. Software Write Enable/Disable
— Powers-up in write disable state
— Must execute WREN before a write Instruction

#2. Zone Write protect
— Controlled by BP1 and BP0 in WRSR (WRite Status Register)

#3. Write Protect Pin (WP)
— Hardware method of preventing erroneous write cycles
— WP must be high to allow writes to EEPROM or the status register

FIGURE 5. NM25C04 Write Protect Features



— Protect Register:
    Input PRE must be high and PREN instruction executed before a write to protect register

— Disable Cell:
    Set via PRDS instruction, input PRE must be high and PREN instruction executed
    PRDS is a one time only instruction

— Address in register defines first location to be protected

— Protect register may be altered unless PRDS is executed

FIGURE 6. NM93CSxx Memory Protect Register

5.0 HARDWARE WRITE PROTECT METHODS

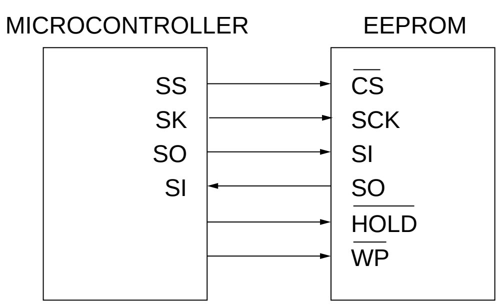
5.1 Write Protect Pin

The Write Protect pin (WP) allows the system designer to include a hardware method for protecting against false data writes. The basic principle is the same for each family of serial EEPROMs; for this example the NM25C04 SPI device is considered.

The connection between a microcontroller and a NM25C04 is shown in Figure 7. If the WP pin is held at 0V then the device is READ only, all WRITE cycles are inhibited.

This interface can be modified by the addition of some external logic to give software control for the WP pin to give increased immunity from data corruption. The basic principle is shown in Figure 8.

NM25C04 SPI EEPROM



Pin Names

CS	Chip Select Input
SO	Serial Data Output
SI	Serial Data Input
SCK	Serial Clock Input
WP	Write Protect
HOLD	Suspends Serial Input

FIGURE 7. NM25C04 EEPROM Interface to Microcontroller

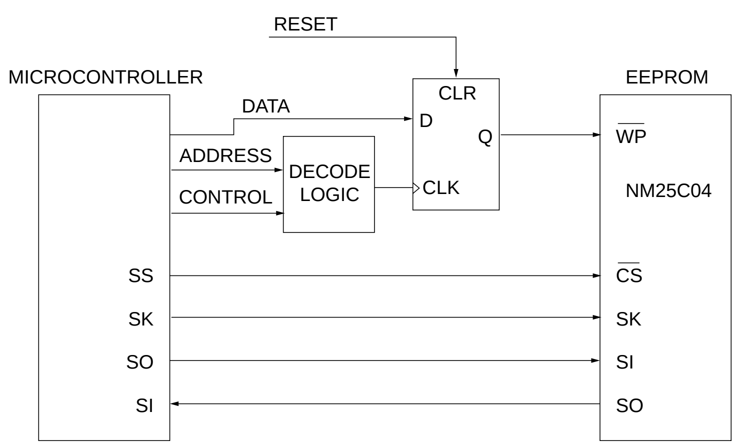


FIGURE 8. Hardware Controlled Write Protect

The theory of operation is as follows:

System RESET signal clears latch output Q, setting WP at logic low level making EEPROM READ only,

For a WRITE instruction to the EEPROM, the microcontroller must first write a logic “1” to the latch to enable the WP pin before executing the normal EEPROM WRITE instruction,

After the WRITE instruction the microcontroller writes a logic “0” to the latch to disable further EEPROM WRITEs.

5.2 System Design Example

This method can be implemented in a practical way, as shown in the design in Figure 9.

The common system block diagram built around an 8- or 16-bit microcontroller will have EPROM, SRAM, Various I/O Ports and a serial EEPROM. The address decode and control logic implementation typically uses a programmable logic device such as a GAL®. The flexibility and user programmability of the GAL allows a designer to integrate the write protect logic for the serial EEPROM in a reliable and cost effective manner.

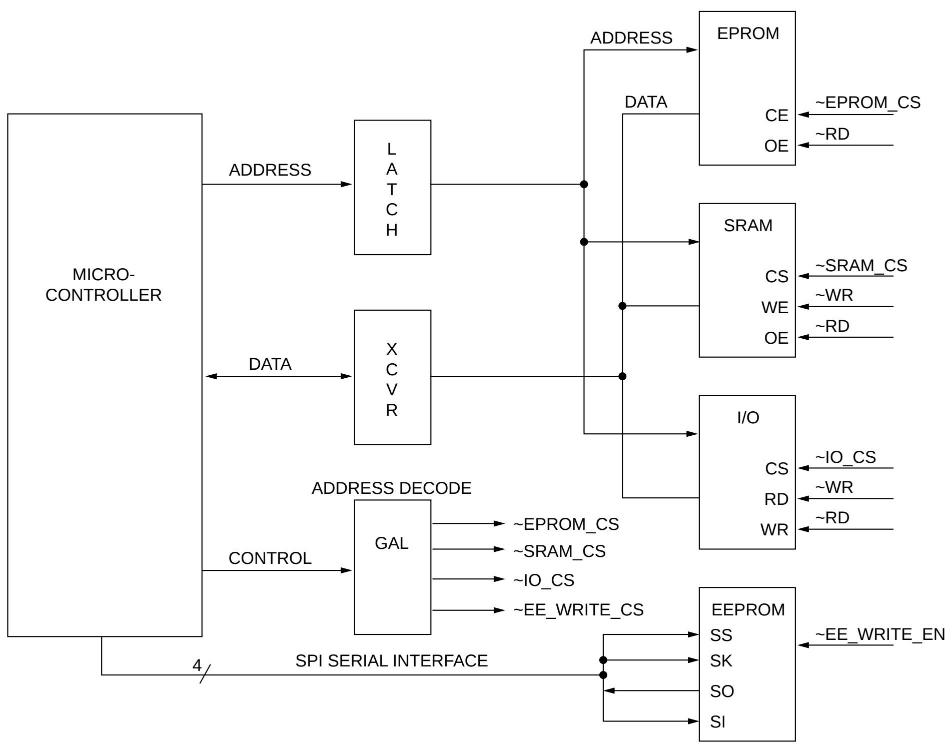


FIGURE 9. Integrated Address Decode/Write Protect Logic

The system memory map is shown in Figure 10.

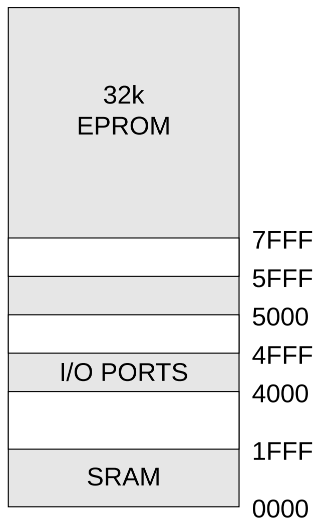


FIGURE 10. System Memory Map

The EEPROM typically connects directly to the microcontroller, either via a standard serial interface (e.g., SPI port for Motorola 68HC05/11 micro's, MICROWIRE port for Fairchild Semiconductor COPS™/HPC™ micro's) or to parallel port pins which can be toggled by software as required. For the purpose of the write protect logic the EEPROM is "mapped" to address space. Note in this example fully exhaustive address decoding was not used.

The logic equations for the GAL can be created using a wide range of PLD design software tools; this example uses Fairchild Semiconductor's OPAL™ design software and is shown in the appendix at the end of this application note.

A typical EEPROM WRITE operation would follow the following routine:

Write a "1" to address 5000H (sets WP high)

Perform EEPROM WRITE cycle

Write a "0" to address 5000H (sets WP low)

By having to explicitly follow this set of operations, it protects the serial EEPROM from inadvertent data writes.

6.0 CONCLUSION

Serial EEPROMs are becoming a standard component in virtually every system; they offer the system designer an easy to use, very flexible solution for a wide range of non-volatile parameter storage applications. The addition of an EEPROM allows for increased system functionality and flexibility providing a superior solution to battery back-up RAM. Serial EEPROMs are highly reliable, offering endurances of 1 million data changes, and data retention of greater than 40 years. The combinations of good IC design practice and system design techniques help solve the issue of data corruption giving high integrity non-volatile memory solutions.

REFERENCES

Fairchild Semiconductor Memory Databook

Fairchild Semiconductor PLD Databook and Design Guide

Fairchild Semiconductor Microcontroller Databook

APPENDIX

EEAPPS.OPL

```
Begin Header

GAL Design for Address Decode Logic & EEPROM Write Protect Logic
Fairchild Semiconductor, 1992

End Header

Begin Definition

DEVICE GAL22V10;

INPUTS

SYS__CLK = 1,
       A15 = 2, A14 = 3, A13 = 4, A12 = 5,
       A11 = 6, A10 = 7, A9 = 8, A8 = 9,
       RD = 10, WR = 11, D0 = 13,
       RESET = 14;                          (RESET active high)

OUTPUTS (COM)

       !~EPROM__CS = 20, !~SHAM__CS = 19, !~IO__CS = 18; (active low outputs)

OUTPUT (REG)

EE__WRITE__EN = 17;

SET    ADDRESS = [Al5,A14,A13,A12,A11,A10,A9,A8];

End Definition

Begin Equations

~EPROM__CS =     !RESET & !RD & WR & ( (ADDRESS >= ^h7F) & (ADDRESS <= ^hFF) );

~SRAM__CS =      !RESET & (!RD + !WR)
                 &( (ADDRESS >= ^h00) & (ADDRESS <= ^h1F) );

~IO__CS =        !RESET & (!RD + !WR)
                 &( (ADDRESS >= ^h40) & (ADDRESS <= ^h4F) );

EE__WRITE__EN := !RESET & (RD & !WR)
                 &D0 & ( (ADDRESS >= ^h50) & (ADDRESS <= ^h5F) );

End Equations
```

EEAPPS.EQN

```
GAL Design for Address Decode Logic & EEPROM Write Protect Logic
```
Fairchild Semiconductor, 1992

```
;Translated from Fairchild formatted PLA file.

CHIP EEAPPS GAL22V10
  SYS__CLK=1 A15=2 A14=3 A13=4 A12=5 A11=6
  A10=7 A9=8 A8=9 RD=10 WR=11 D0=13 RESET=14
  EE__WRITE__EN=17 /~IO__CS=18 /~SRAM__CS=19 /~EPROM__CS=20

EQUATIONS

~EPROM__CS = A14 * A13 * A12 * All * A10 * A9 * A8 * /RD * WR
           * /RESET
       + A15 * /RD * WR * /reset
~SRAM__CS = /A15 * /A14 * /A13 * /WR * /RESET
       + /A15 * /A14 * /A13 * /RD * /RESET
~IO__CS = /A15 * A14 * /A13 * /A12 * /WR * /RESET
       + /A15 * A14 * /A13 * /A12 * /RD * /RESET

EE__WRITE__EN := /A15 * A14 * /A13 * A12 * RD * /WR * D0 * /RSET
```


Fairchild Semiconductor
Americas
Customer Response Center
Tel. 1-888-522-5372

Fairchild Semiconductor
Europe
Fax: +44 (0) 1793-856858
Deutsch Tel: +49 (0) 8141-6102-0
English Tel: +44 (0) 1793-856856
Français Tel: +33 (0) 1-6930-3696
Italiano Tel: +39 (0) 2-249111-1

Fairchild Semiconductor
Hong Kong
8/F, Room 808, Empire Centre
68 Mody Road, Tsimshatsui East
Kowloon. Hong Kong
Tel; +852-2722-8338
Fax: +852-2722-8383

Fairchild Semiconductor
Japan Ltd.
4F, Natsume Bldg.
2-18-6, Yushima, Bunkyo-ku
Tokyo, 113-0034 Japan
Tel: 81-3-3818-8840
Fax: 81-3-3818-8841