NM95C12 Applications in a PC-AT® Ethernet® Adapter

National Semiconductor
Application Note 792
Sean Long
November 1991

INTRODUCTION

This application describes a typical Ethernet adapter card designed to be plugged into a PC-AT expansion slot. The board is designed around the National Semiconductor DP83932 SONIC™ Network Controller device. This application note will detail the system design and focus on the functions performed by the NM95C12 EEPROM.

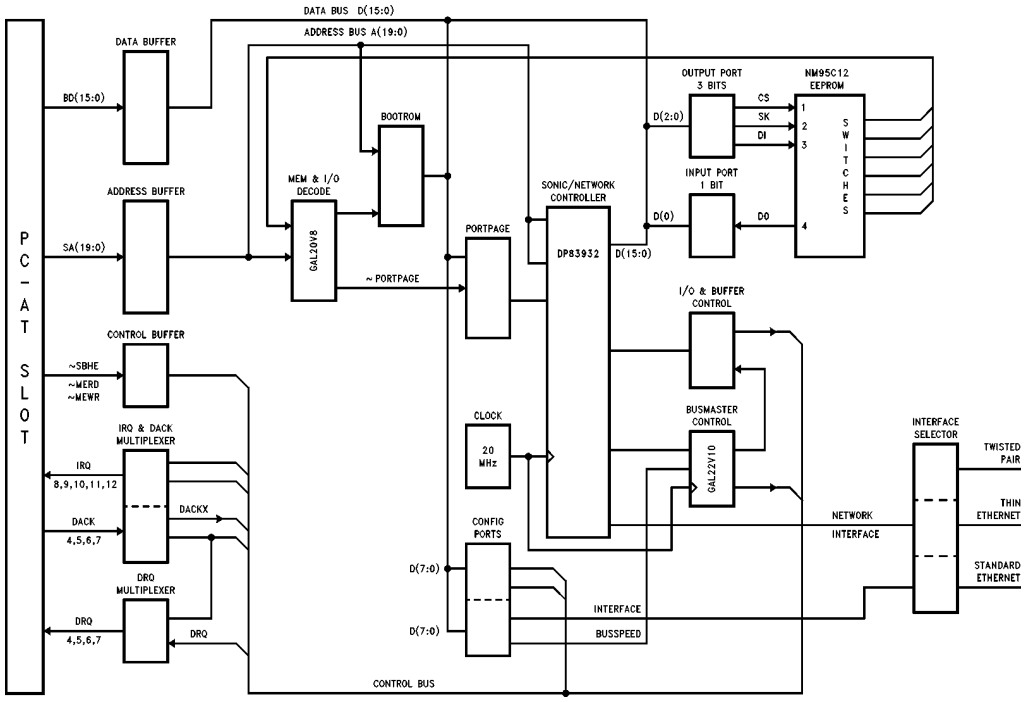
This application note assumes that the reader is familiar with the PC-AT architecture, the DP83932 device, the NM95C12 EEPROM and designing with GAL® Programmable Logic Devices (PLDs).

SYSTEM DESCRIPTION

The network controller card has been designed to meet the following specifications:

- Designed around high performance 32-bit DP83932 Ethernet Controller
- 16-Bit bus master operation to give higher performance
- Fully software configurable (no jumpers or mechanical DIP switches)
- Extensive test and configuration capabilities
- Supports different media interfaces
- Bootrom option

The system block diagram is shown in *Figure 1*.



TL/D/11265–1

*Denotes an active low signal.

FIGURE 1. System Block Diagram

OPAL™ and SONIC™ are trademarks of National Semiconductor Corporation.
PC-AT® is a registered trademark of International Business Machines Corp.
Ethernet® is a registered trademark of Xerox Corp.
GAL® is a registered trademark of Lattice Semiconductor Corp.
ABEL™ is a trademark of Data I/O Corp.

  RRD-B30M75/Printed in U. S. A.

FUNCTIONAL DESCRIPTION OF THE BOARD

The system contains the following logical functions:

1. Network controller (DP83932)
2. Cable interfaces
3. Busmaster interface logic, including data and address buffers
4. EPROM option for remote boot loader

This system uses both the EEPROM locations and the switch logic terminals of the NM95C12 to perform various functions within the system as detailed below.

FUNCTIONAL DESCRIPTION OF NM95C12 EEPROM

Use of the Switches:

The switch terminals of the NM95C12 EEPROM are used as part of the memory map address decoding and the I/O map decoding circuitry, feeding as inputs to a GAL20V8 which performs the address decoding logic from the system address inputs.

The NM95C12 switches control:

1. The base I/O address of the network controller board.
2. The base memory address of the bootrom EPROM option on the board.

ADDRESS DECODING

The address decoding is controlled by a GAL20V8 PLD (refer to the 1990 National Semiconductor PLD Databook and Design Guide for further information) as shown in *Figure 2*.

The inputs to the GAL20V8 are the system address lines, the memory and I/O control signals, and the switch terminals from the NM95C12. The outputs from the GAL20V8 are the various chip select signals for the memory and I/O ports. The system address bus transmits the current address value and the M/∼IO signal determines if a memory or I/O cycle is in progress.

Address lines A0–A19 allow up to 1 Meg (0–FFFFF) of memory to be addressed, while address lines A0–A15 allow up to 64K (0-FFFF) of I/O ports to be addressed. If the control signal M/∼IO is logical ''1'' (high) then the processor is performing a memory cycle and if the M/∼IO signal is logical ''0'' then an I/O cycle is in operation.

For a PC-AT various memory and I/O locations are reserved for standard functions such as system memory and I/O (refer to PC-AT documentation to determine which memory and I/O locations are free for add-in boards).

The switch outputs from the NM95C12 are connected as inputs to the GAL address decode logic and are used to determine the base memory and I/O locations for the add-in card. *Figure 2* shows the typical use of a GAL for address decoding.

The advantage of using a PLD for the address decoding is that it is an easy way to implement different address decode functions by logic equations. The logic equations can be implemented with a standard PLD design compiler such as OPAL™ from National Semiconductor or a third party software package such as ABEL™ from Data I/O. The PLD compiler will take the logic equations and convert them into the GAL fuse map which can be used for programming on a wide range of device programmers. A typical set of logic equations using National Semiconductors OPAL software package is shown in *Figure 3*.

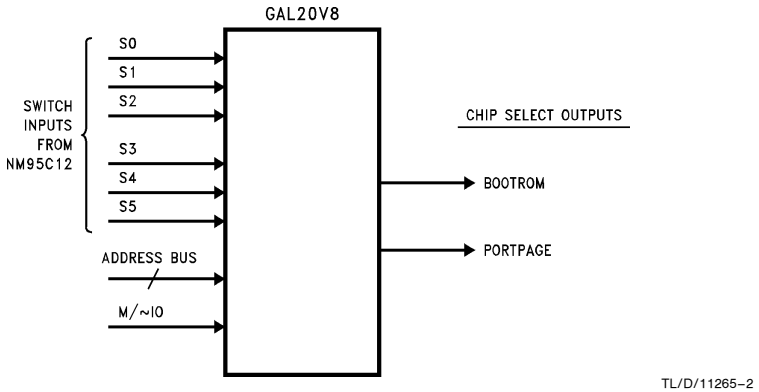


FIGURE 2. Address Decoding

```
BEGIN HEADER
TITLE    Address decoding for PC AT Ethernet adapter card
PATTERN  Addr_Dec
REVISION Rev 0
AUTHOR   Dave Engineer
COMPANY  National Semiconductor
DATE     June 1991
Everything in the header command is copied directly into the JEDEC map as a comment field for
easy documentation
END HEADER

BEGIN DEFINITIONS
device G20V8;                                  { specify the device used }
inputs s0, s1, s2, s3, s4, s5;                 { define the inputs }
inputs m_~io, a0, a1, a2, a3, a4, a5;
outputs(com) bootroom, portpage;               {define the outputs}
{ OPAL will perform automatic pin assignment }
set ioselect=[s2,s1,s0], memselect=[s5,s4,s3]; { define the switch sets }
set address=[a5,a4,a3,a2,a1,a0];
END DEFINITIONS

BEGIN EQUATIONS
{ " / " = logical NOT function (i.e. logical 0)
  " & " = logical AND function
  " + " = logical OR function }
{ if m_~io is logical 0, then decode switch set s2, s1, s0 and address lines for the various
base I/O locations.
  Refer to PC-AT system I/O address map before selecting free I/O ports, the decodes shown are
  for example only - change for specific applications as required. }
bootrom = /m_~io & ( (ioselect == 0) & (address == ↑h00)
                   + (ioselect == 1) & (address == ↑h01)
                   + (ioselect == 2) & (address == ↑h02)
                   + (ioselect == 3) & (address == ↑h03)
                   + (ioselect == 4) & (address == ↑h04)
                   + (ioselect == 5) & (address == ↑h05)
                   + (ioselect == 6) & (address == ↑h06));
{ if m_~io is logical 1, then decode switch set s5, s4, s3 and address line for the various
base memory locations.
  Refer to PC-AT system I/O address map before selecting free Memory locations, the decodes
  shown are for example purposes only - change for specific applications as required. }
portpage = m_~io & ( (memselect == 0) & (address == ↑h18)
                   + (memselect == 1) & (address == ↑h20)
                   + (memselect == 2) & (address == ↑h28)
                   + (memselect == 3) & (address == ↑h30)
                   + (memselect == 4) & (address == ↑h38);
END EQUATIONS
```

FIGURE 3. GAL® Logic Equations

USE OF THE NM95C12 EEPROM LOCATIONS

1. Three locations are used to store the ethernet address of the card.
2. One location is used to store the interrupt number and the DMA channel of the board.
3. One location is used to store the busmaster speed setting of the card.
4. Two locations are used to store information about the production flow of the board e.g.; the version number of the out-going inspection, and serialization program which stores a unique ethernet address in the EEPROM.
5. There are also some EEPROM locations used to enable some special features in the network driver such as protocol, DMA priority, etc.

Figure 4 below shows the memory usage of the NM95C12.

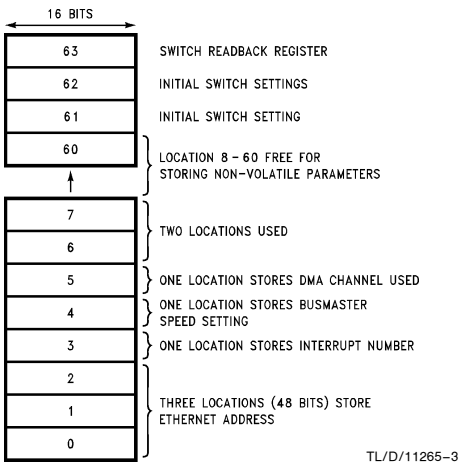


FIGURE 4. Memory Locations Used in NM95C12

FUNCTIONAL DESCRIPTION OF THE SOFTWARE

The driver for the card can be supplied in two ways:

1. As a driver which is loaded from the disk.

2. As a bootrom which is located at the card.

The driver determines the base I/O address of the card. This is done by scanning the possible I/O map where the card can be located (seven possible locations) and testing if the NM95C12 EEPROM can be found.

The EEPROM is found if, after an address is shifted in the EEPROM, the DO output from the NM95C12 has become logical ''zero''. Then the CS pin will be disabled and there will be a check if the DO output pin will become high (this pin is pulled up with a 47K resistor).

When the software finds the base address of the card, it reads the locations which contain the DMA and IRQ number to use and programs these values into the corresponding output latches. These latches will enable and/or multiplex the corresponding DMA (DACKx, DRQx) and INT (IRQx) to the busmaster logic and interrupt logic.

The same operation is done for the busmaster speed, one location in the EEPROM determines the active low and high time for busmaster cycles, the output of this latch will go to the busmaster state machine (implemented in a GAL22V10).

The ethernet address will be read by the driver and copied to a private location in the driver data area for use with the network software.

The bootrom can be located at five locations in memory (controlled by the NM95C12 switch logic) and can be disabled if required.

CONCLUSION

This application has shown the many advantages of the NM95C12 EEPROM with DIP Switches. In this example the NM95C12 replaces the functions typically performed by a Bipolar PROM (store ethernet address), mechanical DIP switches/jumpers (select options), and general read/write logic (software testing of the hardware configuration). The use of the switch terminals as part of the address decode logic makes the address decode function more flexible and allows for software control.

The easy interfacing to the NM95C12 (just four pins) and the simple, but powerful instruction set allows the NM95C12 to give the system designer:

- Greater flexibility
- Fully software controllable and testable
- Greater reliability (no mechanical switches or jumpers)
- Reduced component count
- Lower component cost

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

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 Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998