Using the Am79C961/Am79C961A (PCnet™-ISA+/PCnet-ISA II) Survival Guide

Application Note

Introduction

This document will describe how to use the hardware and software interfaces of the Am79C961 and Am79C961A (PCnet-ISA+ and PCnet-ISA II) single-chip Ethernet controllers when they are used in a non-Plug-and-Play (PnP) environment . Both cases, with external EEPROM and without are covered. In addition, this document will point the reader to existing support software and application notes. Information on using the part in a standard PnP mode will also be included to clarify the text in the data sheets.

The information in this note has come from customers that have actually used the Am79C961 and Am79C961A in a non-PnP environment, from factory applications engineers, and from research into the available literature and software.

The following discussions assume the reader has read either the Am79C961 or the Am79C961A Data Sheet. It does not assume the reader is an expert in PnP or that the reader has read the PnP specification.

Document References

All references (page numbers, figures, etc.) to the Am79C961 will be to the Am79C961 Data Sheet, PID#18183, Rev.B, Issue Date April,1994 while references to the Am79C961A will be to the Am79C961A Data Sheet, PID#19364, Rev.A, Issue Date October,1994. References to the 79C961A Amendment/2 will be to the Am79C961A Data Sheet Amendment, PID19364, Rev.A, Amendment/2, Issues Date June 1995. This amendment contains missing paragraphs and corrections to the 79C961A Data Sheet.

All page number references in this application note are to the Am79C961 Data Sheet (PID#18183B) unless otherwise stated. Page numbers enclosed in square brackets [pg.xx] reference the Am79C961A Data Sheet (PID#19364A). Document names and numbers enclosed in square brackets [xxxx] reference the Am79C961A. Finally, when the syntax Am79C961/A is used, it refers to both the Am79C961 and the Am79C961A.

Late Breaking News

A new bit was added to the PnP Register 0xF0 of the Am79C961A PCnet-ISA II that allows a user to turn off the standard PnP mechanism. This new feature makes the chip ignore the PnP software's special initiation key sequence (6A,...). This will effectively turn the chip into "Legacy" mode operation, where it will be visible in the I/O space, and only special set-up programs will be able to reconfigure it, but standard PnP software will have no effect on the chip. The chip will still recognize AMD's special initiation key sequence (6B,...). The function of this bit does not eliminate the need to follow the procedures for resource assignments as defined in this application note. The details of this new bit are documented in Appendix A at the end of this application note.

Publication# **21625** Rev. **A** Amendment/0
Issue Date: **November 1996**

And now we begin.

Background

First, a little background. The PCnet Family is really a number of devices (chips) whose core is a complete MAC and PHY Ethernet engine. The differences in the family are centered around the host bus interface (ISA, VESA VL, PCI, and Slave are supported by different family members) and whether the Ethernet engine is Half Duplex or Full Duplex. While all of these devices are basically software compatible, there are some enhancements in certain chips that will create minor differences in the configuration register bits (these differences are transparent to the AMD Drivers). It is strongly recommended that the designer read the appropriate data sheet for the specific family member that meets the needs of the application. In the case of this application note, the Am79C961 and the Am79C961A will be the center of discussion.

Bus Master Mode or Shared Memory Mode

Before you even apply power, you must first decide whether you will use the part in the Bus Master-the most common mode used by customers- or Shared Memory mode (a special note-the Am79C961A has a mode called Bus Slave Mode. This mode has two sub modes; Shared Memory Mode and Programmable I/O Mode). This is discussed on page 29[pg.30]: Bus Master Mode, System Interface. The paragraphs on page 29[pg.30] state that these modes are determined through hard wiring on the circuit board, not software. But what hard wiring? It turns out that Pin 2 determines the mode. This pin is named /SMA in Shared Memory (or Bus Slave) Mode -see pg.27[pg.28] and /MASTER in Bus Master Mode-see pg.17[pg.17]. The /SMA pin description on page 27[pg.28] states that this pin must be pulled permanently low to set the mode to Shared Memory. The PCnet internal logic will sample this pin after reset. In Master Mode Pin 2 is named /MASTER-see page 17[pg.17]-and is an I/O that is driven by the PCnet device to indicate that it has become Current bus master (reference the IEEE Terminology: page 16[pg.16]). So to set Master Mode, Pin 2 must be pulled high with a pull up resistor or driven high during power up reset.

EEPROM or no EEPROM

The Am79C961/A provides an interface to a Serial EEPROM. Immediately after Reset, the Am79C961/A will attempt to auto-load information from the Serial EEPROM into its internal registers. The data stored in this EEPROM is shown in the Serial EEPROM Byte Map on pages 41[pg.43] and 42[pg.44]. The information in the EEPROM map on page 41[pg.43] is the generic EEPROM layout for the controller. If an AMD driver is being ported to the target platform, then the EEPROM map on page 42[pg.44] must be used.

The following is an informational comment. In the EEPROM map on page 42[pg.44], the pair of ASCII characters at word location 7, WW(0x57, 0x57), are an ID code used by the AMD Drivers in support of Novell networks.

Note that this auto-loading will only load data from address 00h through address 1AH on the Am79C961 and from address 00h through address 1BH on the Am79C961A.

If no EEPROM is used, the designer must notify the Am79C961/A by floating, or tying down (through a pull down resistor), the SHFBUSY pin (page 19 of the Am79C961A data sheet and page 47 of the Am79C961 data sheet).

EEPROM Mode

In an application with an EEPROM, the Am79C961/A will load the PnP resource registers with values from the EEPROM. It is the responsibility of the system designer to assign valid values to the I/O port, interrupt and DMA channels, as well as to the ROM and RAM addresses. Valid in this context means that there is no conflict with any other I/O or memory device in the system.

Bit 9 of ISACSR2 is the P&P_ACT bit. P&P_ACT must be set to ONE via the EEPROM, so that the Am79C961/A comes up active and is ready to accept accesses to its I/O ports after having read the EEPROM.

Note, that this mechanism fails, if the EEPROM read operation results in a checksum error. Instead, the Am79C961/A will go into software relocatable mode. All register values read from the EEPROM will be discarded and the registers, including the PnP registers, will contain default values. At this time, the activation of the Am79C961/A must follow the same path as described for the non-EEPROM mode.

It is also important to understand that even if the PnP registers are programmed by the EEPROM read operation and the chip is activated, the Am79C961/A continues to accept the PnP key. Any generic PnP software (e.g. PnP BIOS or ISA Configuration Utility), that knows how to deal with ISA PnP devices, can come in and reprogram the PnP registers in the Am79C961/A. This can be avoided by setting the new LGCY_EN bit. By setting LGCY_EN, the Am79C961/A will only accept the special AMD key to gain access to its PnP registers.

No-EEPROM Mode

If there is no EEPROM present or the EEPROM read operation failed due to a checksum error, the part will enter the Software Relocatable Mode (refer to pg 46 of the Am79C961 Data Sheet and pg.1 of the Am79C961A amendment/2) . The Am79C961/A will set the contents of Plug and Play registers from Address 0x00 to 0xF0 to their default values. The following default values are in binary.

PnP Register	Am79C961 Default Values	Am79C961A Default Values
0x00	00000000	00000000
0x02	00000000	00000000
0x03	00000000	00000000
0x06	00000000	00000000
0x07	00000000	00000000
0x30	00000000	00000000
0x31	00000000	00000000
0x60	0000001x	0000001x
0x61	xxx00000	xxx00000
0x70	00000011	00000011
0x71	00000010	00000010
0x74	00000011	00000011
0x40	00001100	00000000
0x41	00000000	00000000
0x42	00000010	00000010
0x43	11111110	00000000
0x44	xxx00000	xxx00000
0x48	00001100	00000000
0x49	00000000	00000000
0x4A	00000010	00000010
0x4B	11111110	00000000
0x4C	xxx00000	xxx00000
0xF0	00000000	00000000

At this point, it is necessary to “unlock” the Am79C961/A with the AMD initiation key.

Now turn to the Plug and Play ISA card State Transition Diagram on page 37[pg.39] of the Am79C961 data sheet. No matter whether you want to operate in PnP mode or in non-PnP mode, the PnP or AMD Initiation Key sequence respectively must be entered (to unlock the Am79C961), followed by the operations needed to move through all the states shown in this diagram until the CONFIG state is reached. Once in this CONFIG state, the PnP registers can be programmed and the controller can be "activated" for normal I/O operations.

Before continuing, let’s review our current status. At this point, reset is over, the Am79C961/A has been notified (by the state of the SHFBUSY pin) that Software Relocatable mode is required, the PnP registers contain their default values, and the Am79C961/A is in the WAIT FOR KEY State-waiting to be “unlocked”. The unlocking operation requires that a predefined sequence of bytes be written to the ADDRESS PORT at address 0x0279 (page 34[pg.36] of the Am79C961 data sheet).

Important Note: During the I/O write operations to address 0x0279, the data must by placed on the low byte of the data bus. The upper byte of the data bus is disabled while in this state.

But, read on to learn the right way to accomplish this loading.

You will see that there are two Initiation Key sequences described for the Am79C961/A. The one on page 34[pg.37] of the Am79C961 data sheet (6A B5 DA---) is the official PnP Initiation Key sequence used when astandard PnP configuration is performed.

Since we are in Software Relocatable Mode due to no EEPROM loading, it is necessary to use the AMD Initiation Key sequence shown on page 46 [pg.1 of the Am79C961A amendment] of the Am79C961 data sheet that starts with the byte pattern 6B 35 9A---.

Application Note AN012001 for the Am79C961[AN13001 for the Am79C961A], Rev. 1.01, Issue Date March, 1995 makes some important suggestions on page four[pg.4] with respect to this key loading operation. These suggestions are restated here for convenience and they are:

Suggestion 1: Software should reset the Am79C961/A internal Linear Feedback Shift Register hardware to its initial value by writing two 8-bit values of 0x00 to the ADDRESS port (0x0279) before the Initiation Key is sent.

Suggestion 2: The I/O writes of the Initiation Key must be contiguous with no other I/O cycles to address 0x0279. It is recommended that system interrupts be disabled while issuing the Initiation Key in order to avoid any extraneous I/O cycles.

Suggestion 3: In order to avoid the possibility of having the Am79C961/Am79C961A controller miss the Initiation Key because it happened to be in the wrong state, it is recommended that the Initiation Key be sent to the controller twice in succession.

Reference the PnP state transitions diagram on page 37[pg.39] of the Am79C961 data sheet. After the AMD Initiation Key is successfully received, the Am79C961/A will enter the SLEEP state. Note that this has nothing to do with the power down sleep mode described on page 64[pg.67] of the Am79C961 data sheet. From this state, only certain operations are possible. The recommended PnP register initialization sequence is described in application note AN012001[AN13001], page 8 and is included here for convenience. The data values used here are for reference only, the programmer can use whatever values are needed by the application.

The following code example assumes that the Am79C961/A is the only (E)ISA PnP device in the system. If there are two or more (E)ISA PnP devices present, the PnP Serial Isolation protocol (page 35[page 37]) must be used to isolate and configure one (E)ISA PnP device at a time. In the C-code below, the function calls contain two arguments. The first argument is the PnP register number and the second argument is the data value to be written to the register. Assume that the variable IOBASE has been initialized with a 16-bit I/O address value of 0x0340.

In the following description, the example code lines are referenced as line 1 through line 14. It is the reader’s responsibility to ensure that resource values such as the I/O address, interrupt request, and DMA channel are legal and are not in conflict with other used device resources in the system.

```
ip_cfg_w(0x02, 0x05);  /* RESET command. */
ip_cfg_w(0x03, 0x00);  /* WAKE[0] command. */
ip_cfg_w(0x06, 0x01);  /* SET_CSN[1] command. */
ip_cfg_w(0x60, (BYTE)((IOBASE & 0xFF00) >> 8));  /* I/O addr: 15:8 */
ip_cfg_w(0x61, (BYTE)(IOBASE & 0x00FF));  /* I/O addr: 7:0 */
ip_cfg_w(0x70, 0x00);  /* IRQ level: 0 = no IRQ selection. */
ip_cfg_w(0x71, 0);  /* IRQ type: Edge, active low. */
ip_cfg_w(0x74, 0x04);  /* DMA 4: Channel 4. */
ip_cfg_w(0x43, 0xFE);  /* Mem Desc 0: bit0 = 0 = disable. */
ip_cfg_w(0x4B, 0xFE);  /* Mem Desc 1: bit0 = 0 = disable. */
ip_cfg_w(0xF0, 0x00);  /* Vendor (AMD) Defined Byte. */
ip_cfg_w(0x31, 0x00);  /* Disable I/O range check. */
ip_cfg_w(0x30, 0x01);  /* Activate Reg: bit0 = 1 = active. */
ip_cfg_w(0x02, 0x02);  /* Cfg Ctl: bit1 = 1 = WAIT_FOR_KEY */
```

Line 1: Resets the PnP logical device forcing default values in the PnP registers. Resets the Card Select Number to 0.

Line 2: The WAKE command causes a transition to the ISOLATION state.

Line 3: The CSN for the PnP device is set to 1. This action automatically causes a transition to the CONFIG state.

Lines 4 through 12: These commands are assigning PnP resources in the various PnP registers. The most important resource assignment is in lines 4/5. This is where the base I/O address for the PCnet controller is programmed. The example code assigns the upper half of the I/O address (0x03) to register 0x60 and the lower half of the I/O address (0x40) to register 0x61.

The other resource assignments are less critical but necessary. No interrupt is being selected, and DMA channel 0 is being assigned.

Line13: This is where the controller is activated to respond to its I/O address. It is important that this step is only performed after a valid I/O address is assigned to the Am79C961/A.

Line 14: The final step is to force the PnP state machine back to the WAIT_FOR_KEY state so the controller will stop sharing the pins on the external private bus and, more importantly, to eliminate the possibility that the PnP registers will become corrupted by subsequent I/O accesses to the system "printer" port at 0x0279 and 0x0A79.

When an EEPROM is present in the system and the read operation successfully passed the checksum test, a number of PnP and ISACSR registers are programmed (review pg.42[pg.44]). Some of these PnP, and none of the ISACSR, registers are written in this example. These registers will have to be programmed. This can occur as part of this sequence or at a later time, but before the transmitter or receiver is turned on.

The Finale

Congratulations, the controller is now active and should be responding to I/O cycles directed to the I/O address programmed in PnP registers 0x60 and 0x61. Your driver software can now read and write the CSR and ISACSR registers.

At this point, the Am79C961/A is capable of being initialized for normal receive and transmit operations. This initialization can be accomplished by using the system memory resident Initialization Block described on page 48[pg.50] or the alternative Initialization method described on the same page.

In the system memory resident Initialization Block, the user writes the system memory location of the Initialization Block into registers CSR1 and CSR2 by performing slave write operations to the Am79C961/A. Registers CSR0 and CSR3 should also be written with the appropriate values at this time. The initialization operation is started and the Initialization Block is fetched by the Am79C961/A when the INIT bit in CSR0 is set, again by a slave write to the Am79C961/A. This operation causes the Am79C961/A to become a bus master and automatically write the data contained in the Initialization Block into the appropriate registers of the Am79C961/A. The Initialization Block contents are defined on page 95[pg.102]. At this point, assuming all the CSR and ISACSR registers have valid contents, the transmitter and receiver can be turned on.

In the alternative Initialization method, the internal registers are programmed with direct software writes. These write operations are applied to the Am79C961/A as slave write operations. The registers to be written in this alternative Initialization method are described in Appendix D of the 79C961/A Data Sheets. The transmitter and receiver can be turned on once all the appropriate CSR and ISACSR registers are programmed.

REFERENCE MATERIALS:

Am79C961 Data Sheet (PID#18183, Rev.B, Issue Date April 1994)

Am79C961A Data Sheet (PID#19364, Rev.A, Issue Date October 1994)

Am79C961A Data Sheet Amendment (PID#19364, Rev A, Amendment 2, Issue Date June 1995)

AN012001 Application Note: “How to perform Plug and Play Initiation Key with PCnet-ISA+,” (Am79C961, Rev.1.01, Issue Date March 29, 1995) [Can be found the ***Embedded Networking Applications Design Guide Kit***, PID#20397.]

AN013001 Application Note: “How to perform Plug and Play Initiation Key with PCnet- ISA II,” (Am79C961A, Rev.1.01, Issue Date March 29, 1995) [Can be found the ***Embedded Networking Applications Design Guide Kit***, PID#20397.]

EESETUP Ver 4.1 Software

Source program “Wake.c,” found in EE_DUMP.ZIP Software

SCOPE Reference Document and Software Ver 1.0

ACKNOWLEDGMENTS:

This note would not have been possible without the detailed discussions held with Gene Gonzales. Much of the code described in this note came directly from the software routines created by Gene and the other engineers in the AMD Networking Division. These routines are available to customers and are well annotated, making them easy to understand. I would also like to acknowledge Harand Gaspar, Carl Ching and David Stoenner for their review of the original draft of this note.

Appendix A

Summary:

The following describes a new bit that was added to the PnP Register 0xF0 that would allow a user to turn off the standard PnP mechanism.

A reminder: The addendum works ONLY for those PCnet-ISA II devices that have the "FD" marking following the datecode. (See description below.)

This addendum will be incorporated into the next edition of the PCnet-ISA II datasheet and/or Ethernet Family Databook (1997).

Addendum:

Am79C961A (FD) PCnet-ISA II Jumperless, Full Duplex Single-Chip Ethernet Controller for ISA

This information references the data sheet identified as PID#19364, Rev.A, Amendment/0, Issue Date: October 1994.

This addendum applies to all Am79C961A controllers with a new date code marking that contains an "FD" following the date code (e.g. 9629-FD).

PCnet-ISA II Legacy bit feature description

The current PCnet-ISA II chip is designed such that it always responds to Plug and Play configuration software. There are situations where this response to the Plug and Play software is undesirable. An example of this is when a fixed configuration is required, or when the only possible resource available for the PCnet-ISA II conflicts with a present but not used resource such as an IRQ, or when the chip is used is a system with a buggy PnP BIOS.

To function in the situations above, a new feature has been added to the PCnet-ISA II chip. This new feature makes the chip ignore the PnP software's special initiation key sequence (6A,...). This will effectively turn the chip into "Legacy" mode operation, where it will be visible in the I/O space, and only special set-up programs will be able to reconfigure it, but standard PnP software will have no effect on the chip. In case the EEPROM is missing, empty, or corrupted, the chip will still recognize AMD's special initiation key sequence (6B,...).

To enable this feature, a one has to be written into the LGCY_EN bit, which is bit 6 of Plug and Play Register 0xF0. A preferred method would be to set this bit in the Vendor Byte (PnP 0xF0) field of the EEPROM located in word offset 0x1A.

Data Sheet Changes:

Page 46 of the data sheet shows a chart containing the bit descriptions of the internal registers associated with Plug and Play operation. PnP register 0xF0, bit 6, which was previously defined as Reserved, is now defined as follows:

Bit	Name	
7	Reserved	
6	LGCY_EN	**<==(New Bit)**
5	DXCVRP	
4	FL_SEL	
3	BP_CS	
2	APROM_EN	
1	AEN_CS	
0	IO_MODE	

PnP register 0xF0, bit 6 (LGCY_EN) - Legacy mode enable.

When written with a one, the PCnet-ISA II will not respond to the Plug and Play initiation key sequence (6A) but will respond to the AMD key sequence (6B). Therefore, it cannot be reconfigured by the Plug and Play software. When set to zero (default), the PCnet-ISA II will respond to the 6A key sequence if the EEPROM read was successful, otherwise it will respond to the 6B key sequence.

End of Addendum

Trademarks