PCI S5920 Developer's Kit
User Manual
And
Technical Reference Manual

Revision 1.3
April 1998

For Marketing and Application Information Contact:

Applied Micro Circuits Corporation
6290 Sequence Drive
San Diego, CA 92121-4358
(619) 450-9333
http://www.amcc.com

The material in this document supersedes all previous documentation issued for any of the products included herein.

AMCC reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

AMCC does not assume any liability arising out of the application or use of any product or circuit described herein, neither does it convey any license under its patents rights nor the rights of others.

AMCC reserves the right to ship devices of higher grade in place of those of lower grade.

AMCC SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Contents

CHAPTER 1

INTRODUCTION

Introduction

The AMCC PCI Developer's Kit contains everything needed for the *PCI* developer to immediately begin operating and experimenting with a S5920 based *PCI* design. For software engineers, the Developer's Kit is a fully functional PCI to Add-On bus test card. The programmer can immediately begin testing and operating numerous aspects of PCI Bus to Add-On Bus data transfers, timings, control and overall operation. The programmer can also test and become familiar with the various aspects of PCI BIOS functions and PCI Configuration Space operation. A set of DOS based development programs allow the programmer to view and change device register contents from the PCI Bus as well as view and change PCI configurations. Additional development software provides downloading, editing, configuring and programming capability to the optional serial boot load nvRAM contained on the main Developer Kit PCI card.

For the hardware designer, the Developer's Kit provides fully functional PCI to Add-On bus design examples. The main S5920 PCI card shows Add-On bus connection to onboard SRAM. The ISA Adapter card allows the designer to plug in an existent ISA card to the Add-On bus to begin logic optimization and reduction.

The Developer's Kit comes complete with schematics, PCB artwork, EPLD equation source code for each application example and development software source code. The designer is able to implement portions or all of an Add-On bus design using a supplied bread board. Extra headers and EPLD sockets are available in the Developer Kit to further assist in proto-typing and general experimentation. Dedicated Hewlett Packard PCI logic analyzer headers are provided for directly cable connection.

Developer's Kit Overview

The PCI Developer's kit contains two printed circuit boards plus a software tools CD-ROM. The S5920 PCI card contains an S5920, SRAM and a pre-programmed EPLD containing Add-On bus control functions. This card was developed to demonstrate interconnection of the S5920 PCI interface chip to the PCI Bus and interconnection of the S5920's Add-On Bus to a basic SRAM design. The onboard EPLD is specifically programmed to control the Add-On bus for Active Mode data transfers for burst or single cycle data reads and writes to the SRAM.

The Add-On bus signals are also routed to a set of four external application connectors. These connectors provide the designer with additional Add-On bus connection capability. The designer can utilize these for attaching his/her own application PCB to the PCI card's Add-On bus. Two of these connectors are designed to provide simultaneous connection of the user's PCB and a logic analyzer. Although many logic analyzers may be connected, the are designed specifically for connection directly with Hewlett Packards PCI logic analyzer pod cabling.

The second PCB is an ISA Adapter interface card designed specifically to mate with the S5920 PCI card. The adapter card was developed to provide direct connect of many existent ISA cards to the S5920 Add-On bus. An adapter card EPLD is programmed to convert Add-On bus signals to ISA card signals and vise versa. The adapter card provides the designer with a basic functioning interface example to the PCI bus allowing the designer to start design optimization and logic reduction. The programmer can immediately begin reading and writing data from the PCI bus to ISA card addresses.

It is important for the designer to remember, the developer's kit was designed to demonstrate various aspects of S5920 user design. The specific EPLDs, Add-On logic components and software was chosen to support multiple application illustrations. Therefore, the device costs and complexity is more than will be necessary for many applications.

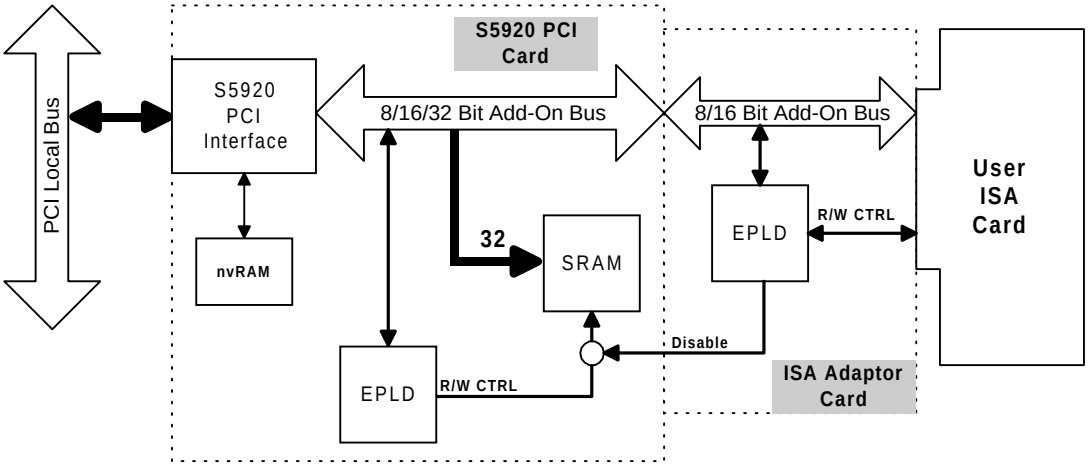


Figure 1-1 Developer Kit Block Diagram

The Developer's Kit Goal

The S5920DK1 was designed to help both hardware and software engineers go into production with a new design as quickly as possible. Hence, AMCC has provided the following:

A fully functional hardware design example of an SRAM interface and an ISA bus card interface.

Documentation text files to help come up to speed quickly on all parts of the 5920DK1.

Hardware -- all source files to re-create the 5920DK1 boards and use them as the basis for your design. Also included are all source files for the EPLDs.

Software -- program examples and utility tool source code to help develop your new software and debug hardware.

Win95 and WinNT software device drivers for the S5920 and S5933 are currently available through our Development Partners. Please visit our web site at www.amcc.com for up-to-date links.

Developer's Kit Features

The S5920 PCI Card

The Primary design aid to the Developer's Kit is the main PCI Developer Card. This board contains the S5920 device interfaced to the PCI Bus giving the developer a functional example of device location, trace lengths/routing and decoupling. The Add-On Bus of the S5920 is interfaced to board signal headers, SRAM and an EPLD device. The EPLD supplied serves as an example of Add-On Local Bus interface control to SRAM.

The ISA Adapter Card

The ISA Adapter Card supplied with the Developer's Kit may be interconnected to the signal headers on the main PCI Developer Card. This interconnection provides the developer with a data and control signal path to many existent PC ISA Bus designs to begin transferring data and examining S5920 register data.

The Software

For the software developer, the 5920DK provides a fully functional PCI bus to Add-On bus test environment. Much software development can start immediately, without waiting for your new hardware to be built. The software developer can become familiar with the PCI BIOS and PCI Configuration Space Registers. Additional supplied software supports downloading, editing, and programming data to an optional serial boot load nvRAM on the 5920DK. Source code is included for all programs supplied with the 5920DK, allowing easily customization of the programs to your application.

Notation Conventions

Low-Active Signals Signals which are asserted (or active) in the low voltage state are defined with a trailing number/pound (#) sign within the schematics; or with a leading exclamation (!) for EPLD equations.

The following designations are used throughout this book when referring to the size of data objects.

A BYTE is an 8-bit object.
A WORD is a 16-bit, or 2 byte object.
A DWORD is a double word and is a 32-bit or 4-byte object.

Hexadecimal notations are indicated with a trailing “h” or a leading 0x.

9A4Fh
0110h

Binary notations are indicated with a trailing “b”.

1010b
0110b

Developer Recommended Documentation

- AMCC Data Book: S5920 PCI Interface (supplied)
- PCI Local Bus Specification, Revision 2.1 (PCI SIG)
- PCI BIOS Specification, Revision 2.1 (PCI SIG)
- Other related Applications Notes and Design Notes can be downloaded from the AMCC website at:

http://www.amcc.com

To obtain listed documentation from the PCI SIG, contact:

PCI Special Interest Group
P.O. Box 14070
Portland, OR 97214
(800) 433-5177
(503) 797-4207
FAX (503) 234-6762

CHAPTER 2

DEVELOPER KIT HARDWARE INSTALLATION

Developer's Kit Contents

The S5920 Developer Kit contains the following hardware, software and documentation:

- **Primary Developer S5920 PCI Card**

  This card is the main design aid of the S5920 Developer Kit. We recommend designers follow this design as an example of correct device location, trace lengths, routing and decoupling.

  Contains:

  - Serial nvRAM
  - 32K DWORDs SRAM (128K optional)
  - Pre-programmed EPLD
  - Four Add-On bus to logic analyzer connectors
  - Four user design to Add-On bus interface connectors

- **ISA Adapter Card**

  This card is designed to help convert existent ISA based cards to PCI based designs. When connected to the main PCI card, virtually any ISA card may be inserted. Hardware and software engineers may the begin accessing the ISA design and start converting.

- **Developer Software Tool CD-ROM**
- **S5920 Data Book**
- **Developer Kit User Manual (this manual)**

System Requirements

The minimum system requirements are:

- 386 processor
- 512K system RAM memory
- 10 Meg Hard Disk Space
- CD-ROM Drive
- DOS 5.0 or Higher with ANSI.SYS
- 256-color VGA Display
- PCI bus motherboard slots

The recommended system requirements are:

- 486DX processor or better
- 1 MB system RAM memory
- 10 Meg Hard Disk
- 3.5 Floppy Disk Drive
- CD-ROM Drive
- DOS 5.0 or Higher with ANSI.SYS
- 256-color VGA Display
- Keyboard

Installing Hardware

The following section details the installation procedure for hardware components contained in the Developer's Kit. This developers kit is intended and designed for an electronics laboratory environment in which the PC containing the DK will remain open. This allows access to special connectors for logic analyzers and physical space to insert the ISA adapter card and the ISA card under evaluation. Be sure that all AC power has been removed from your computer before proceeding. AMCC recommends all installation work be done at a static free workstation. If one is not available, ensure that you have removed the static charge from your cloths by touching an object made of metal on the computer before proceeding.

Installing the PCI Card

- Remove the cover mounting screws on your computer and carefully remove the cover. Store the cover in a safe place.
- Ensure the Developer Kit's main S5920 PCI Card is jumpered as shown in one of the two figures below for either SRAM or ISA Adapter card operation. See the jumper description section of this manual to configure for other required options.

- Hold the Developer PCI Card by its top corners and insert into any available PCI slot. Press down gently but firmly until the card is seated.
- Follow all suggested safety guidelines in your computer manufacturer's manual.

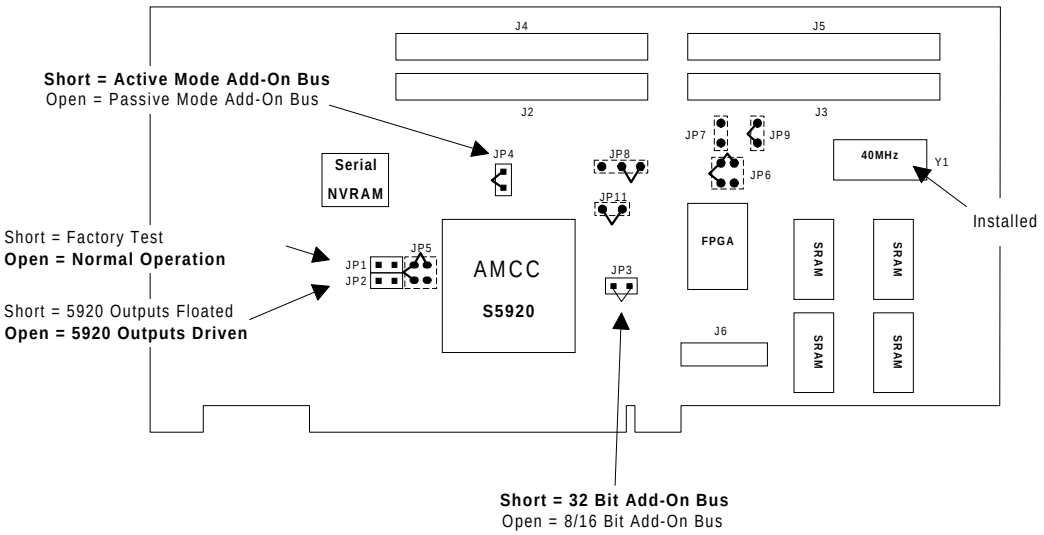


Figure 2-1 PCI Card Jumpers for Active Mode SRAM Operation

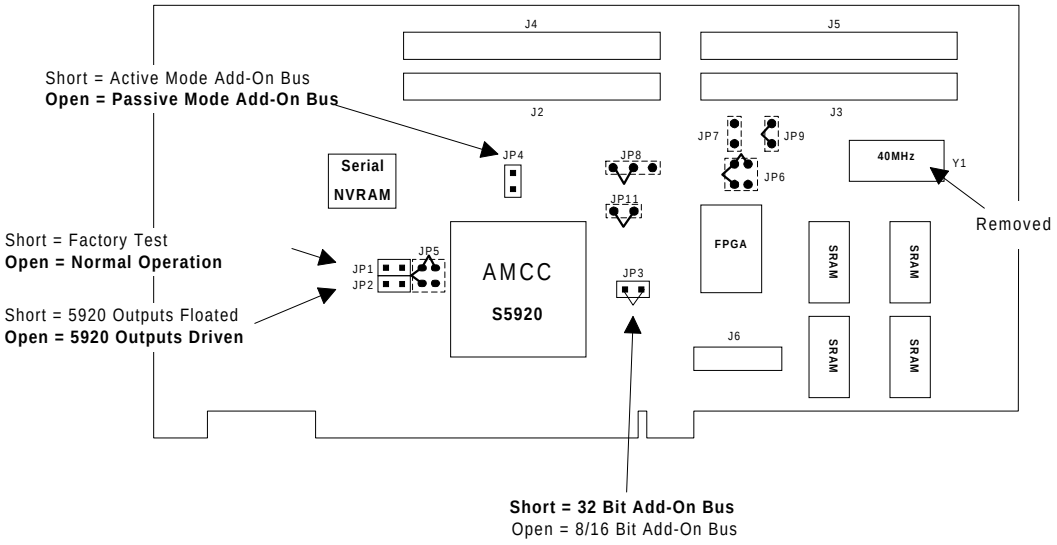


Figure 2-2 PCI Card Jumpers for Passive Mode ISA Adapter Operation

Installing the ISA Adapter Card

- No jumpering is necessary prior to installing the Developer Kit's ISA Adapter Card.
- Hold the Developer ISA Adapter Card by its sides align with connectors J2 through J5 on the main S5920 PCI card. Press down gently but firmly until the card is seated.
- Carefully install the main S5920 PCI card into the PC.

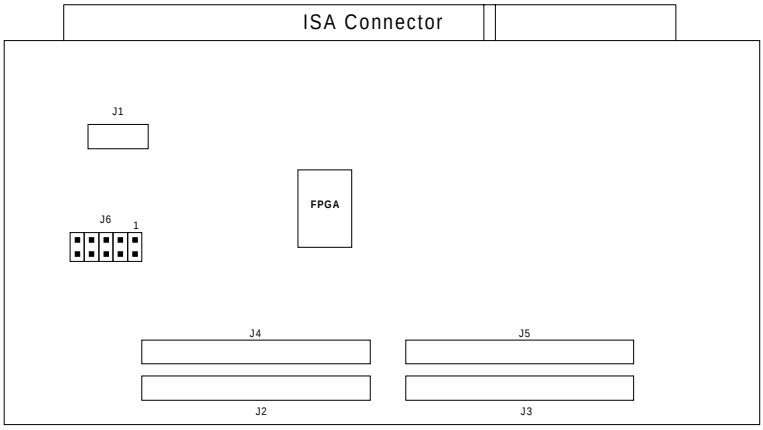


Figure 2-2 ISA Adapter Card

Connecting The Hewlett Packard Logic Analyzer

The S5920 PCI card was developed with a set of conveniently located logic analyzer connectors. These connectors were specifically designed to allow the ISA Adapter card or a new proto-type design to be connected simultaneously with a logic analyzer. This allows the developer to operate and test Add-On bus circuits while examining setup and hold times along with data transfers. The ISA Adapter card and developer proto-types connect to the S5920 Add-On bus through connectors J2 to J5 on the component side of the S5920 PCI card. The logic analyzer connects to the Add-On bus through the same connectors from the solder side of the S5920 PCI card. The four connectors contain all the signals of the Add-On bus and ground references. These connectors are pin designated for direct pod cable connection to the Hewlett Packard 16500B or C logic analyzer. Up to four cables may be connected to cover the entire Add-On bus signal set. Refer to the schematics for signal location before connecting the HP or any other logic analyzer.

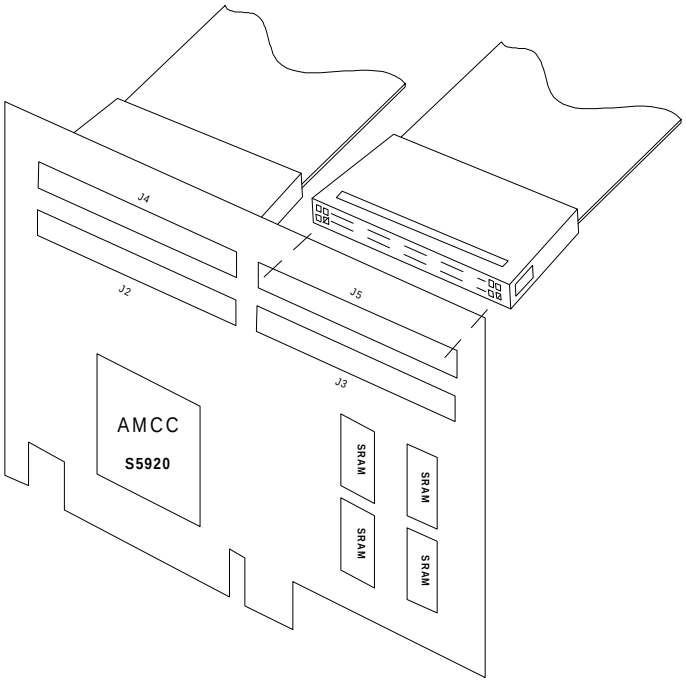


Figure 2-3 Logic Analyzer Connection

Installing The Software

Shown below are the basic directory folders for the CD-ROM supplied with the developer kit. The content description of each folder and sub-folder is also listed. For normal hardware development, it is only necessary to copy the utility '.exe' programs to the hard drive for easy access and execution.

S5933 Hardware Folder

Misc - Miscellaneous drawings and PCB assembly files
PCBs - The Gerber files for the Developer Kit PCBs
PLDs - The CUPL source files for the PLDs and description docs
Sch - The OrCAD schematics for building the S5933DK1

S5933 Software Folder

Contains the assembly source code for the AMCCDIAG utility program. Also contains example h, library and include C files.

S5920 Hardware Folder

PCI_Card
- Altera - The Altera EPLD code for the PCI card
- DXF - The DXF format files for building the PCI card PCB
- Gerbers - The Gerber plotter files for the PCI card PCB
- Sch - The Protel schematic files for the PCI card
- OrCADlibrary - The 5920 and 5933 OrCAD schematic library files
- PCB - The Protel PCB files for the PCI card PCB
- NC - The NC drill files for the PCI card PCB

ISA_Card
- Altera - The Altera EPLD code for the PCI card
- DXF - The DXF format files for building the PCI card PCB
- Gerbers - The Gerber plotter files for the PCI card PCB
- Sch - The Protel schematic files for the PCI card
- PCB - The Protel PCB files for the PCI card PCB
- NC - The NC drill files for the PCI card PCB

S5920 Software Folder

Contains the Utility programs for the S5920DK1 and the C source code, library and include folders for the programs Also contains example h, library and include C files.

nvRAM_Tool Folder

Contains the nvRAM utility program, AMCCPCI.EXE, used or both the S5920 and S5933 Dks. Also includes the C source code, library and include folders for the program.

Books Folder

The S5920 data book PDF file
The S5933 data book PDF file
The S5920 DK manual

Applications Folder

Contains various application and design notes for the S5933 and S5920 PCI devices. Also are the device summary files for device history.

Licence.doc - The Developer's Kit user license agreement
readme.txt - Latest manual and software updates.

System Checkout

After installing the hardware and software as described in the previous sections, an operational test should be run to ensure proper system to Developer's Kit function. This will ensure the motherboard, system BIOS, PCIBIOS, utility programs, DOS, Win95 and the Developer's Kit hardware are all in sync. The following steps will operate various aspects of the DK to verify correct operation.

STEP 1 Test the Presence of the S5920DK1

Run the SCAN.EXE program. Examine the display and verify a VID=10E8 and DID=5920 entry is present. This indicates a PCIBIOS is present and has located and recognized the S5920 Developer Kit main card. If this line is not present, verify the settings in the system BIOS menus and the correct installation of the DK main card.

STEP 2 Test the state of the S5920 PCI Configuration space

Run the CFG.EXE program. Select the DK by entering the number beginning the line containing the VID and DID of the DK as indicated in step 1 above. Examine the Command register. The two least significant bits should be a one. This indicates the BIOS has enabled the S5920 on the PCI bus to respond as either a memory or I/O device. There should also be values other than FF or 00 in Base Address registers B0, B1, B3 and B4. Write down the address for B1 for the next step. Note: the values place in the Base Address registers are assigned by the system BIOS during power-up. They can be changed through the utility program but will result in the "Blue Screen of Death" if re-located over other software.

STEP 3 Test the SRAM Read and Write Operation

While in DOS, run the MEMRW.EXE program. Enter the following:

memrw /a<B1 address> /o44332211↵
memrw /a<B1 address+4> /o88776655↵

Next type the following and verify the response is the same as indicated:

memrw /a<B1 address> ↵	Response: B1 address = 44332211
memrw /a<B1 address+4> ↵	Response: B1 address+4 = 88776655
memrw /a<B1 address+1> ↵	Response: B1 address+1 = 55443322

This correct responses indicates successful reads and writes to the onboard SRAM through the S5920 Pass-Thru data channel and proper operation of the EPLD state machine.

STEP 4 Test the nvRAM

Run the AMCCPCI program. Select the S5920 non-volatile memory builder. Select to load a memory image from the 24C16. A successful load verifies proper reading operation through the S5920. Next select 'write to device'. A successful nvRAM write verifies complete nvRAM operation.

The operational tests are complete.

Chapter 3

Developer Kit Software

Introduction

The software utility programs supplied with the Developer's Kit provides PCI card diagnostics and a developer interface based on the C++ programming language and x86 style systems. These programs are the interface through which the developer can access S5920 operation and configuration registers on the Developer's Kit PCI card or a newly developed PCI card. The following table lists each utility program and describes their function.

Utility Programs

Program	Function	Platform
AMCCPCI.EXE	Reads and writes to the S5920 nvRAM to change power-up configurations/options.	DOS, Win95
SCAN.EXE	Performs a PCI Bus scan for devices and lists by VID, DID, SVID, Bus and Index.	DOS, Win95
MEMRW.EXE	Reads and writes to PCI memory or I/O space to move data through the S5920 Pass-Thru data channels.	DOS
OPR.EXE	Reads and writes to the S5920 operation registers from the PCI bus.	DOS
CFG.EXE	Reads and writes to the S5920 configuration registers from the PCI bus.	DOS, Win95

Each program's source code is supplied in either machine or C++. Applied Micro Circuits Corporation provides the source to programmers for use, in all or part, for the development of

All programs (except AMCCPCI) have been built in a Borland C++ IDE environment. To modify the programs, search for the project file *5920.ide*. Open it in Windows Explorer to launch the Borland IDE.

NOTE: References to DOS platforms indicates the system must boot in DOS. Not a DOS window or DOS prompt under Win95. Win95 indicates will run in a DOS prompt in Win95.

Win95 and WinNT Software Device Drivers

Currently Applied Micro Circuits Corporation maintains links on its web site (www.amcc.com) to third party software companies having device drivers for the AMCC S5933 and S5920 PCI devices. AMCC works closely with vendors, making sure their products enhance your development process. However, we leave all aspects of development, marketing and support to these development partners. See the above web sites for the latest information on these device drivers.

Software Tools for the AMCCPCI Program

Microsoft Visual C++ 5.0

For device drivers and 32-bit development (Win32, VxDs, WinNT Kernel Mode Device Drivers, WDM Device Drivers, etc.).

CXL

A menu support library included on the CD-ROM. Both source code and documentation are included. Used extensively in AMCCPCI.

Software Tools for All Other Supplied Programs

Borland C++ 4.51 or higher

Borland Turbo Assembler (TASM) 4.0 (or higher)

io.c has been supplied in assembly language.

General Use Software Tools

SoftICE 3.2

Although normally used for kernel-mode work such as device drivers, we have found this debugger to be useful for many other debugging tasks, such as working in Win95 DOS boxes.

PCI SIG ID Policy

The PCI Special Interest Group has developed a device and card identification system to ensure all PCI Bus devices are uniquely identified. This identification system allows software operating systems to load appropriate software drivers based on the ID numbers. Use the following table as a reference guide for temporary PCI identification numbers for use in the developer's kit. The indicated numbers are the factory defaults pre-programmed into the onboard nvRAM and are loaded into the S5920 PCI Configuration Registers during power-up initialization.

Configuration Register	Name	Value
Vendor Identification	VID	10E8h
Device Identification	DID	5920h
Revision Identification	RID	00h
Subsystem Vendor ID	SVID	10E8h
Subsystem Identification	SID	00EEh

The PCI SIG has divided identification numbers into two groups. Group one is dedicated to the chip manufacturer to uniquely identify the silicon device on the PCI bus. Group two is dedicated to the end user or board manufacturer to uniquely identify the end product on the PCI bus.

- **VID** The vendor identification number is assigned by the PCI SIG to the IC manufacturer. In this case, 10E8h has been registered to the name Applied Micro Circuits Corporation for identifying AMCC as a PCI chip device manufacturer.
- **DID** The device identification number is assigned by AMCC under it's rights of VID assignment. AMCC assigns a unique DID to each of it's PCI chip devices. In this case, AMCC has assigned 5920h to uniquely identify the S5920 PCI interface chip.
- **Revision** The revision number is also assigned by AMCC. This number is assigned and programmed to identify the revision level of the silicon die within the device package. In this case, the register is hardwired to the silicon's revision.
- **SVID** The sub-vendor identification number is assigned by the PCI SIG to the end board manufacturer to uniquely identify the manufacturer's name. All developers need to acquire a unique SID number from the PCI SIG for their company name. In this case, AMCC has assigned 10E8h to identify AMCC as the manufacturer of the S5920 developer kit.

- **SID** The system identification number is assigned by the end product manufacturer under the rights of their SVID assignment. This will uniquely identify the end product within the market for software operating systems. In this case, AMCC has assigned a unique SID of 00EEh as the S5920 developer kit.

AMCCPCI.EXE: Utility Program

The AMCCPCI.EXE utility program provides the user with a menu driven display to change the S5920 configuration space and device power-up options. The configuration space values and power-up options are contained in an nvRAM connected to the S5920. This program contains the necessary software routines to read, change and write the contents of this serial nvRAM. The following is a list of the program menu tree. Typing *amccpci* in either DOS or a DOS window under Win95 will start the utility program..

1) S5920 Non-volatile memory builder

Load Memory Image
- PCI Device
  - Which PCI device?
    - Which serial nvRAM?
- File
  - Which File?
- Exit Menu

Save Memory Image
- Save to File
  - File name?
- Merge with File
  - Which File?
- Exit Menu

Edit Memory Image
- Edit Base Addresses
  - Lists Base Address Registers
- Edit Other Configuration Registers
  - Lists Configuration Registers
- Edit Location 45h Configuration Bits
  - Lists location 45 hex Bits
- Exit Menu

Write to Device
- Which PCI Device?
  - Which nvRAM?

Display Memory Image
- Displays Memory Image in hex

Exit Menu

2) S5933 Non-volatile memory builder

Load Memory Image
- PCI Device
  - Which PCI device?
    - Which Serial nvRAM?
- File
  - Which File?
- Exit Menu

Save Memory Image
- Save to File
  - File name?
- Merge with File
  - Which File?
- Exit Menu

Edit Memory Image
- Edit Base Addresses
  - Lists Base Address Registers
- Edit Other Configuration Registers
  - Lists Configuration Registers
- Edit Location 45 Configuration Bits
  - Lists location 45 hex Bits
- Exit Menu

Write to Device
- Which PCI Device?
  - Which nvRAM?

Display Memory Image
- Displays Memory Image in hex

Exit Menu

3) Exit Program

Note: The program modifies data from it's own memory space. This space defaults to set variables upon start up. To modify what's in the nvRAM, the contents must be loaded into the program space by choosing 'Load Memory Image' from 'PCI device' and select the serial nvRAM type from your board. The S5920 DK is shipped with a 24C16 serial nvRAM device.

Developer's Kit nvRAM Factory Settings

The following are the factory programmed settings for the nvRAM to run the SRAM and ISA Adapter card design examples.

Base Address Registers

	Type	Size	Pass-Thru Width	Memory Location	Prefetchable
Base Address 0	I/O	128 bytes			
Base Address 1	Memory	128 bytes	32 Bits	Anywhere	No
Base Address 2	Disabled				
Base Address 3	Memory	1 Mbytes	16 Bits	Anywhere	No
Base Address 4	Memory	1 Mbytes	8 Bits	Anywhere	No

Other Configuration Registers

Vendor ID	10E8	Device ID	5920	
Subsystem Vendor ID	10E8	Subsystem ID	00EE	
Revision ID	00			
Base Class Code	04			
Sub Class Code	00			
Programming I/F	00	BIST Capable?	N	
Latency Timer	00			
Interrupt Line	00	Interrupt Pin	1	INTA#
Maximum Grant	FF	Max. Latency	00	

Location 45 Hex

Readretry#, RD# Operation	1
WRmode#, WR# Operation	1
Target Latency Timer Control	1

IMPORTANT NOTE: The developer's Kit hardware and software has been designed to operate using the Base Address and Configuration Register values indicated above. Altering these values may cause improper operation. The ISA Adaptor Card was designed to function in memory mapped mode and not I/O. Designers may change settings with the appropiate software and hardware design changes.

OPR.EXE Utility Program

The opr.exe utility program allows the user to display and read or write to the S5920 Operation registers addressed in either memory or I/O space. This tool can be used for reading and writing to; outgoing mailbox, incoming mailbox, mailbox status, Interrupt, reset control and the Pass-Thru configuration registers. The following is the option menu. Typing *opr* followed by a *?* will display the option list.

COMMAND SYNTAX: opr [↵] [R] [33] [A] [?] [W:<register>=<data>]

Option Menu:

↵ = Display the S5920 Operation registers
R = Displays the S5920 Base Address 0 space in a hex table format
W:<register>=<data> Writes to the register named with the hex data
Example: opr w:omb=103
33 = Displays the S5933 Operation registers if installed. (Used for S5933DK1)
A = Displays the Add-On Operation registers through the S5933DK1 ISA card.
? = Displays help menu

Example of Operation register display:

5920 Operation registers....

Out Mailbox	[OMB]:	00000020
In Mailbox	[IMB]:	00000f0f
Mailbox Flags	[MBEF]:	0000F00C
Interrupt Reg	[INTCSR]:	00000C0C
Reset Control	[RCR]:	00000000
Pass-Thru Cfg	[PTCR]:	80808080

NOTE: Some of the utility programs perform a PCI Bus scan and list all found devices each time the program is run with a task. The user is then required to select which PCI device in the list the program task is directed to before performing it. The opr.exe program is an example of this. By using the DOS SET command to assign data to variables, the PCI device selection menu can be skipped. Example: by typing *SET AMCC_DID=5920* will assign 5920 to the device ID. The next time opr.exe is run, it will use the 5920 selection from the PCI Bus scan and perform the task without asking. Only one variable needs to be set to identify a PCI device. The variables that can be set for all the utility programs are:

AMCC_VID AMCC_DID AMCC_SVID AMCC_SID AMCC_INDEX

CFG.EXE Utility Program

The cfg.exe utility program allows the user to display and modify the S5920 Configuration registers. At startup, a brief display shows all PCI devices within the host system. Bridges, I/O cards, video cards and the S5920 developer's kit are displayed by vendor ID, Device ID, etc. Once the S5920 developer's kit has been selected, cfg.exe is used to read and write to the S5920's configuration registers. The following is the option menu. Typing *cfg* followed by a *?* will display the option list.

COMMAND SYNTAX: cfg [R] [SCAN] [W:,register name>=<data>]

Option Menu:

R = Displays the PCI configuration space in a hex table format
W:<register>=<data> Writes to the R/W register named with the hex data
Example: cfg w:pcicmd=103
SCAN = Scans and displays PCI bus devices located (used if set
? = Displays help menu

Example of opening PCI bus scan

0) VID=8086, DID=1250, SVID=0000, SID=0000, BUS=0, INDEX=0, Intel
1) VID=8086, DID=7000, SVID=0000, SID=0000, BUS=0, INDEX=0, Intel
2) VID=102B, DID=0519, SVID=0000, SID=0000, BUS=0, INDEX=0, Unknown User
3) VID=10E8, DID=5920, SVID=10E8, SID=00EE, BUS=0, INDEX=0, AMCC
4) VID=8086, DID=1229, SVID=8086, SID=0009, BUS=0, INDEX=0, Intel

Note: Number 3 is the S5920 Developer's Kit.

Example of the configuration space display for the S5920:

Vendor ID	[VID]:	10E8	RO
Device ID	[DID]:	5920	RO
Command	[PCICMD]:	0103	R/W
Status	[PCISTS]:	0280	RO
Revision ID	[RID]:	00	RO
Class Code	[CLCD]:	000004	RO
SVID	[SVID]:	10E8	RO
SID	[SID]:	00EE	RO
Cache Line	[CALN]:	00	RO
Latency Timer	[LAT]:	00	RO
Header Type	[HDR]:	00	RO
BIST	[BIST]:	00	RO
Base 0	[B0]:	0000FC81	R/W
Base 1	[B1]:	FFF80000	R/W
Base 2	[B2]:	00000000	R/W
Base 3	[B3]:	FFC00000	R/W
Base 4	[B4]:	FF800000	R/W
Base 5	[B5]:	Not Implemented	
Exp. ROM Addr	[XROM]:	00000000	RO
Interrupt Line	[INTLN]:	09	R/W
Interrupt Pin	[INTPIN]:	01	RO
Min. Grant	[MING]:	00	RO
Max. Latency	[MAXL]:	00	RO

MEMRW.EXE Utility Program

The memrw.exe utility program allows the user to read or write 8, 16 or 32-bit data to a PCI bus address located in memory or I/O space. This tool can be used for reading and writing to; SRAM, through the ISA Adapter card to registers on an ISA card inserted or to the S5920 operation registers. The following is the option menu. Typing *memrw* followed by a *?* will display the option list.

COMMAND SYNTAX: memrw /A<address> [/O<data>] [C] [L] [I] [B] [W] [D] [L]

Option Menu:

/I = Indicates the /O command below will be to I/O space over memory
/O = Cause a write data to memory address indicated
/A<address> The hex address to R/W data

/B = The data field is a BYTE
/W = The data field is a WORD
/D = The data field is a DWORD (default if not otherwise specified)

/C = Read and display data, repeat read and display only if data changes
/L = Continually repeat the read or write

Examples: (Note: if you type it wrong, the option menu will be displayed automatically)

Writing DWORDs to memory addresses type:

memrw /afff80000 /o44332211 ↵	Response: 44332211 => fff80000
memrw /afff80004 /o88776655 ↵	Response: 88776655 => fff80000

Reading DWORDs from memory addresses type:

memrw /afff80000 ↵	Response: fff80000 = 44332211
memrw /afff80004 ↵	Response: fff80004 = 88776655
memrw /afff80001 ↵	Response: fff80000 = 55443322

Writing DWORDs to I/O addresses type:

memrw /afff8 /o44332211 /I↵	Response: 44332211 => fff8

The /C and /L options are designed to aid the hardware engineer capture scope or logic analyzer PCI or Add-On bus signals. The /C option shown in the example below will read data from the address indicated and display it. The program will then re-display the address's data only if the value changes. A *^C* will terminate the sampling. This option is used for testing for a register status bit change or changing data on an input port. The /L option will continue to loop and execute a read or write of data to the address specified.

memrw /afff80000 /c ↵	Response: fff80000 = 44332211 (and waits for a change)
memrw /afff80000 /o10e85920 /L ↵	Response: fff80000 = 44332211 (writes and repeats writing)
memrw /afff80000 /L ↵	Response: fff80000 = 44332211 (reads and reads)

SCAN.EXE Utility Program

The scan.exe utility program allows the user to see a brief display of all PCI devices within the host system. Bridges, I/O cards, video cards and the S5920 developer's kit are displayed by vendor ID, Device ID, etc. The following is a display example of the utility. Typing *scan*.

Example of PCI bus scan

```
0)  VID=8086,   DID=1250,   SVID=0000,   SID=0000,    BUS=0,   INDEX=0,   Intel
1)  VID=8086,   DID=7000,   SVID=0000,   SID=0000,    BUS=0,   INDEX=0,   Intel
2)  VID=102B,   DID=0519,   SVID=0000,   SID=0000,    BUS=0,   INDEX=0,   Unknown User
3)  VID=10E8,   DID=5920,   SVID=10E8,   SID=00EE,    BUS=0,   INDEX=0,   AMCC
4)  VID=8086,    DID=1229,   SVID=8086,   SID=0009,    BUS=0,   INDEX=0,   Intel
```

Note: Number 3 is the S5920 Developer's Kit.

DKTEST.EXE Utility Program

The dktest.exe utility program provides the user with the ability to test and exercise various circuits of the S5920 Developer's Kit. The program tests the read and write capabilities of the onboard nvRAM, SRAM, S5920 operation registers and the Pass-Thru data channels. The following is the option menu. Typing *dktest* followed by a *?* will display the option list.

COMMAND SYNTAX: dktest [NV] NVSIZE NOSAVE MEM OP PT PTSIZE NOCHK

Option Menu:

NV = Saves the contents of the nvRAM. Reads and writes 2048 byte test pattern to the nvRAM. Restores the contents of the nvRAM when complete.
NVSIZE = Specifies the size of the nvRAM read/write range (Default is 2048)
NOSAVE = Do not save or restore the nvRAM contents.
MEM = Reads and writes to SRAM on the PCI card
OP = Reads and writes to all the S5920 operation reisters and bits.
PT=<region> = Specifies the Pass-Thru region to use in the memory test
PTSIZE=<size> = Defines the size of the Pass-Thru region for testing (Default = entire region)
NOCHK = Do not check if the Pass-Thru regions are enabled.

Example of nvRAM test:

dktest nv ↵

Testing 2048 bytes of nvRAM, should take about 50 seconds....
saving current nvRAM contents....
running data test....
restoring nvRAM contents....
All tests passed

Example of SRAM test of 20000h DWORDS:

dktest mem ↵

testing memory address 0xfff80000, size 0x20000.....
All tests passed.

Example of operation register tests:

dktest op ↵

testing operation registers.....
All tests passed.

Chapter 4

Developer Kit Hardware

S5920 PCI Card

The following section describes various aspects of the hardware design for the S5920 PCI card.

Jumper Descriptions

JP1 (TEST) The TEST signal is a reserved input to the S5920 and must always be left open or in the logic “one” state. For factory use only.

JP2 (FLT#) The FLT# signal floats all S5920 output signals when asserted. Leave JP2 open for normal operation.

JP3 (DQMODE) DQMODE defines the Add-On bus DQ width. JP3 shorted configures the DQ bus for 32 bits and open configures a 16 bit DQ bus. The default is open for developer kit SRAM operation. For ISA adaptor operation, set according to the ISA card installed.

JP4 (PTMODE) Pass-Thru mode configures how the Add-On bus will function when using the Pass-Thru data channel. JP4 open will configure the Add-On bus to function in passive mode. This mode allows other devices to share the Add-On bus and requires these devices to drive S5920 bus control signals. JP4 shorted will configure the Add-On bus for active mode. In this mode, data reads or writes to the Pass-Thru channel will cause the S5920 to drive the DQ bus and bus control signals through an internal state machine. The S5920 PCI card SRAM application uses an EPLD programmed to operate in active mode. Short JP4 to use the onboard EPLD for the SRAM application. The ISA Adaptor card application is provided with two example EPLDs. One EPLD operates in active mode and the other in passive mode. Jumper according to which EPLD is installed in the ISA Ada[ptor card.

JP5 (ADCLK/BPCLK) The developer kit is designed to offer the designer all possible input and output clock jumpering configurations. The S5920 PCI card is pre-jumpered through extra fine traces for normal operation of the SRAM and ISA adaptor applications. These traces may be cut and solder bridged between pads to configure for user designs. Due to the high frequency of the clock line, AMCC highly recommends the technique of solder bridging between pads with no sharp points left after soldering be utilized. Refer to the S5920 PCI card schematic before altering the default configuration.

JP6 (EPLD CLK) JP6 is provided to allow the designer alternate clock input and output configurations to the onboard EPLD. JP6 is pre-jumpered through extra fine traces for normal operation of the SRAM and ISA adaptor applications. Should the designer desire to change the example EPLD equations to function differently or implement other clock frequencies, these traces may be cut and wire wrap soldered between pads as desired. Refer to the S5920 PCI card schematic before altering the default configuration.

JP7 (Termination) An optional clock input or output R/C termination is provided for the clock signal. The designer may install components and jumper JP7 as needed. Refere to the S5920 PCI card schematic for details.

JP8 (ADCLK3) JP8 is provided to allow the designer alternate clock input and output configurations to his development card when attached to the S5920 external connectors. JP8 is pre-jumpered through extra fine traces for normal operation of the ISA Adaptor application. Refer to the S5920 PCI card schematic before altering the default configuration.

JP9 (Termination) JP9 is provided to allow an optional series resistor termination to be installed when using crystal Y1. The designer can install any value resistor and jumper accordingly. Refer to the S5920 PCI card schematic before altering the default configuration.

JP11 (Termination) JP11 is provided to allow an optional series resistor termination to be installed in the ADCLK1. The designer can install any value resistor and jumper accordingly. Refer to the S5920 PCI card schematic before altering the default configuration.

Test Point Description

TP1 - PCI CLK. General purpose test point of the PCI clock. To be temporarly used with a low impedeance oscilloscope probe for examine PCI clock signal integrity.

TP2 - SDA. General purpose test point of nvRAM's serial data line. To be used to verify signal integrity and data transmission during reads and write to the serial nvRAM.

TP3 - SCL. General purpose test point of nvRAM's serial clock line. To be used to verify signal integrity and clock transmission during reads and write to the serial nvRAM.

TP4 - ADCLK1. General purpose test point of the S5920's Add-On bus clock input. To be temporarly used with a low impedeance oscilloscope probe for examine ADCLK clock's input signal integrity.

TP5 - BPCLK. General purpose test point of the S5920's Add-On bus BPCLK's synchronous clock output. To be temporarly used with a low impedeance oscilloscope probe for examine BPCLK's output signal integrity.

TP6 - IRQ#. General purpose test point of the Add-On bus interrupt output from the PCI bus. To be used to examine by an oscilloscope or logic analyzer to see the presence of an inetrrupt from the PCI bus to the Add-On bus.

PCB Connector Description

J1 - This is the primary PCI edge connector. All PCI communications and handshaking take place through this connector. Refer to the PCI SIG specification for signal names and definations.

J2 tru J5 - The Add-On bus from the S5920 is wired to an on board EPLD and SRAM application design. The Add-On bus is also paralleled to these four connectors. When the ISA Adaptor card is in place, these connectors become the Add-On bus interface by removing the on board EPLD from the design. This is accomplished thru grounding pin 40 of J2 which is a disable signal to the EPLD. This applies to the ISA Adaptor card as well as any user proto design connected to the S5920 PCI card.

J6 - This connector is supplied for programming the EPLD (U6) on the S5920 PCI card. For programming, use the Altera programming cable (P/N PL-BYTEBLASTER) available for approximately $150 from your local Altera distributor.

ISA Adaptor Card

The following section describes various aspects of the hardware design for the ISA Adaptor card.

Jumper Descriptions

JP1 to JP5 - (Options) These optional jumpers are available for future and user application implementation. This jumper block allows for five undefined options to be defined and implemented into the onboard EPLD. See the schematic for further details.

Test Point Description

There are no test point incorporated on the ISA Adaptor card.

PCB Connector Description

J1 - None.

J2 tru J5 - The Add-On bus from the S5920 is wired to an on board EPLD and SRAM application design. The Add-On bus is also paralleled to these four connectors. When the ISA Adaptor card is in place, these connectors become the Add-On bus interface by removing the on board EPLD from the design. This is accomplished thru grounding pin 40 of J2 which is a disable signal to the EPLD. This applies to the ISA Adaptor card as well as any user proto deign connected to the S5920 PCI card.

J6 - This connector is supplied for programming the EPLD (U1) on the ISA Adaptor card. For programming, use the Altera programming cable (P/N PL-BYTEBLASTER) available for approximately $150 from your local Altera distributor.

Chapter 5

Developer Kit Design Aids

The AMCC PCI Developer's Kit was designed to provide everything needed for the *PCI* developer to immediately begin operating and experimenting with a S5920 based *PCI* design. Additionally, the kit includes many design and development aids intended to help reduce the design time for both software and hardware designers. Software source code, schematic and PCB source files and EPLD equation files are all included as aids to new S5920 based PCI designs. The following section details these design aids.

Schematics

All schematics source files for the S5920 PCI card and the ISA Adapter card are shown in Appendix A. The source files and library files are contained on the CD-ROM. The schematics were developed under Protel rev 3.1 for Windows 95. These files can be imported into other software development tools using EDIF or DXF file formats. Some software packages are capable of directly importing. The bill of materials are also located in the CD-ROM.

PCB Artwork

All PCB artwork source files for the S5920 PCI card and the ISA Adapter card are contained on the CD-ROM. The PCBs were developed under Protel PCB rev 3.1 for Windows 95. These files can be imported into other software development tools using direct or DXF file formats.

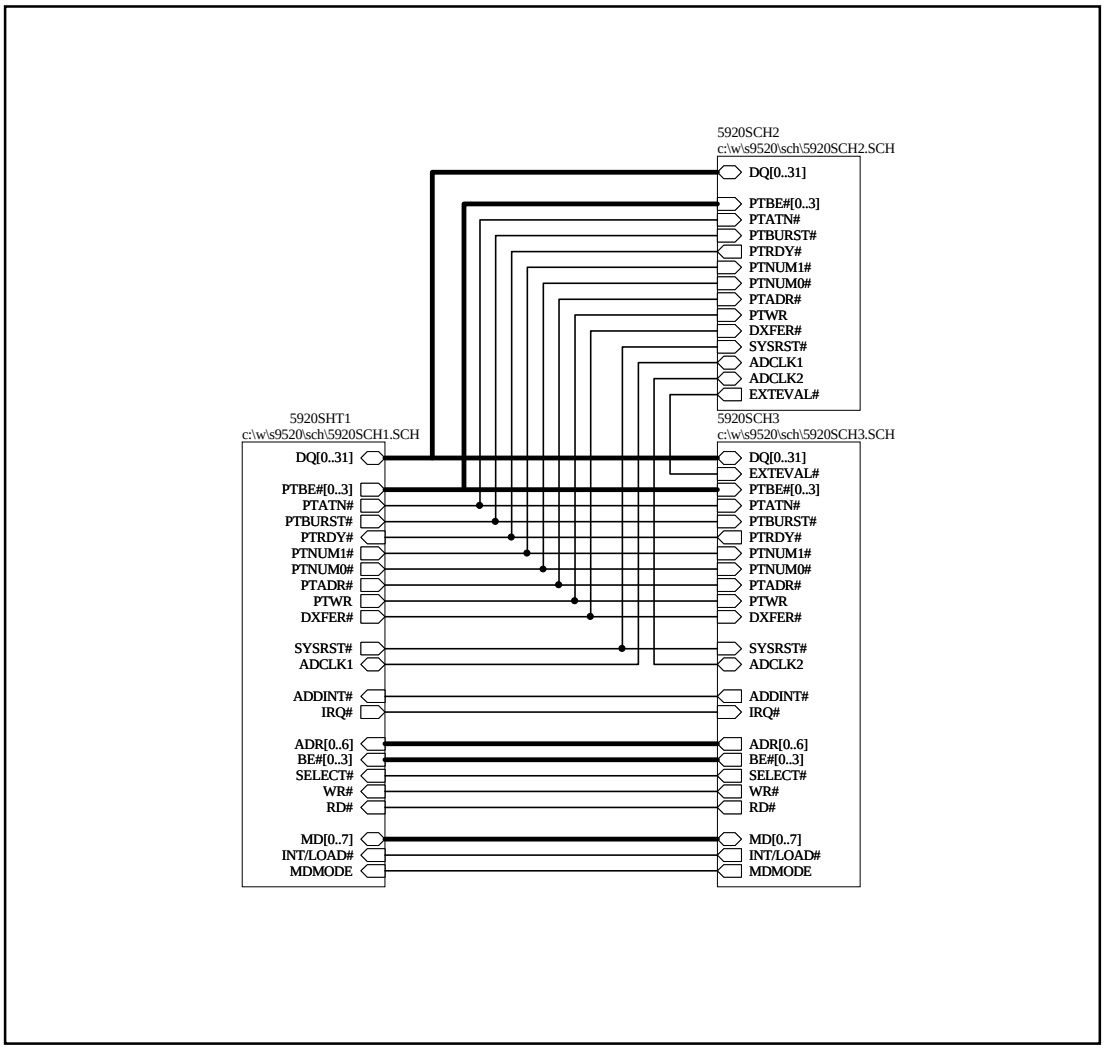
S5920 PCI Card and ISA Adapter Card EPLD Equations

The PCI Card and ISA Adapter Card application example EPLD equation files are contained on the CD-ROM. The files give a basic implementation for interfacing the Add-On bus to SRAM and also to adapt the Add-On bus to an ISA controller. These examples give the designer a basic start from which to begin logic optimization to reduce component cost thru a smaller and slower PLDs. AMCC expects the final solution for many ISA designs to be less than $5 for final glue. These files were created with Altera's "MAX + plus II" version 8.0.6.

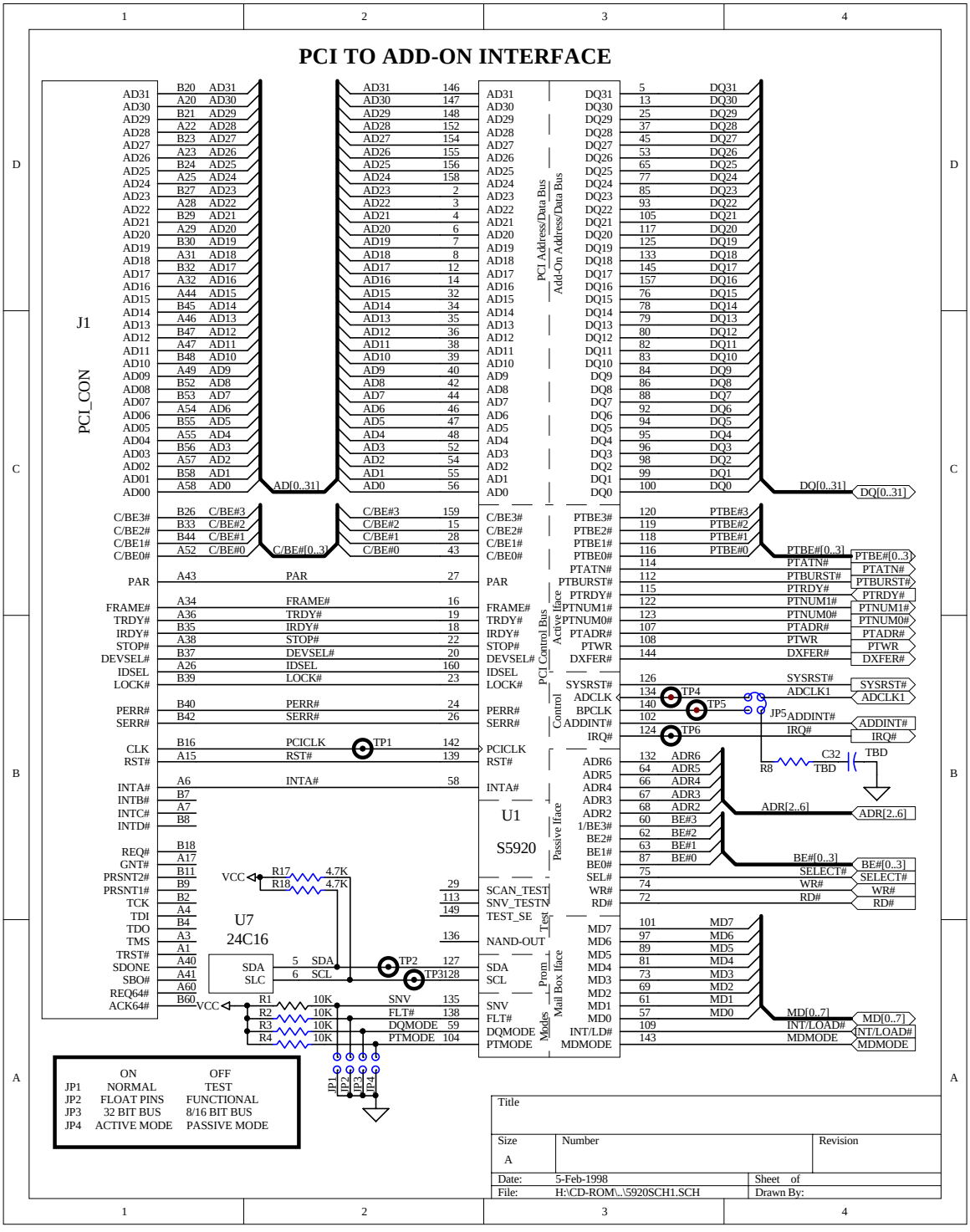
Software Source Code

All software source code files are supplied on the CD-ROM. These files were created and compiled with Microsoft's Visual C++ version 5.0, Borland's C++ version 4.51 and Borland's Turbo Assembler version 4.0. These files give the programmer a basic example of S5920 and PC BIOS calls, in standard C, for data transfer and status operation.

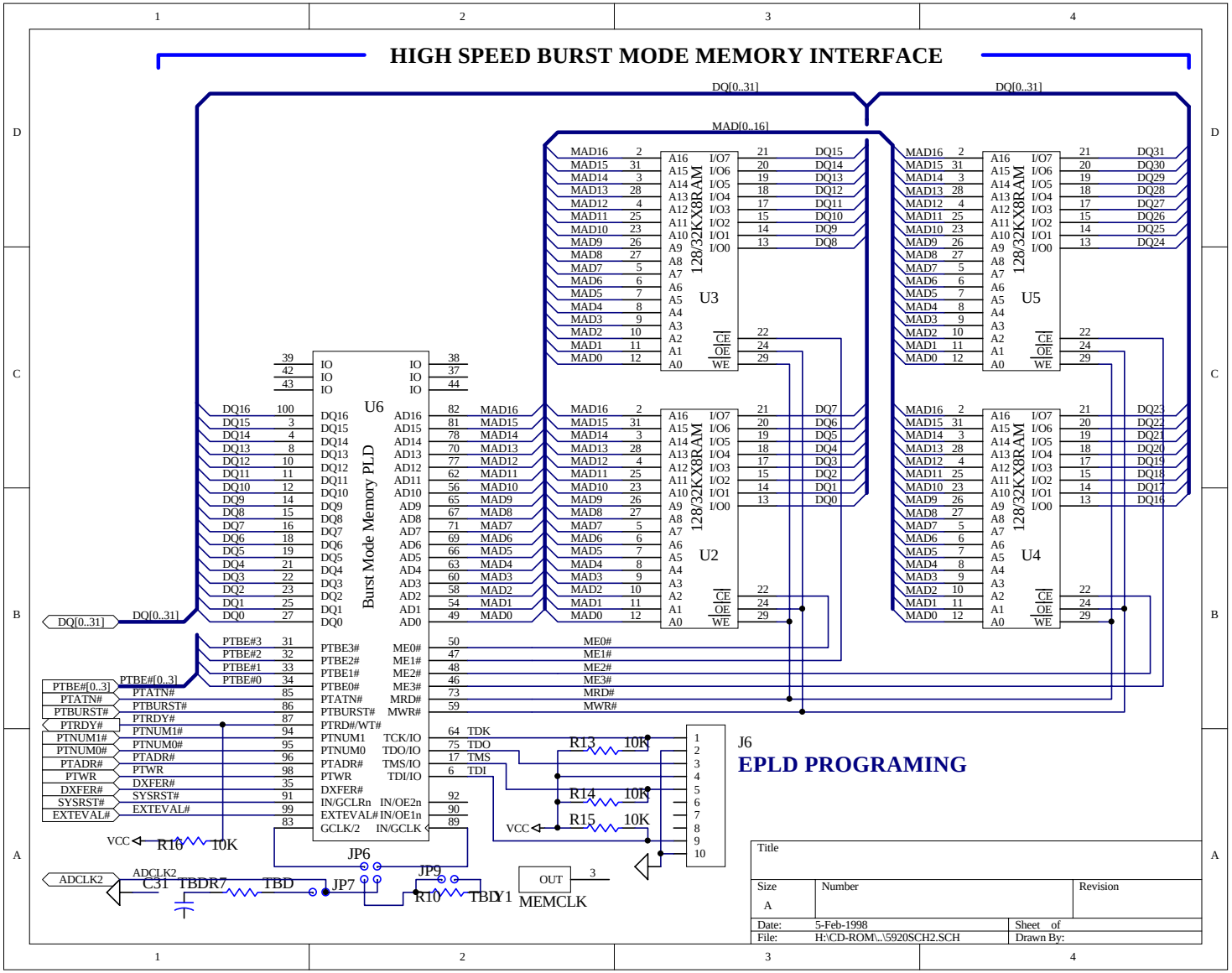
Appendix A

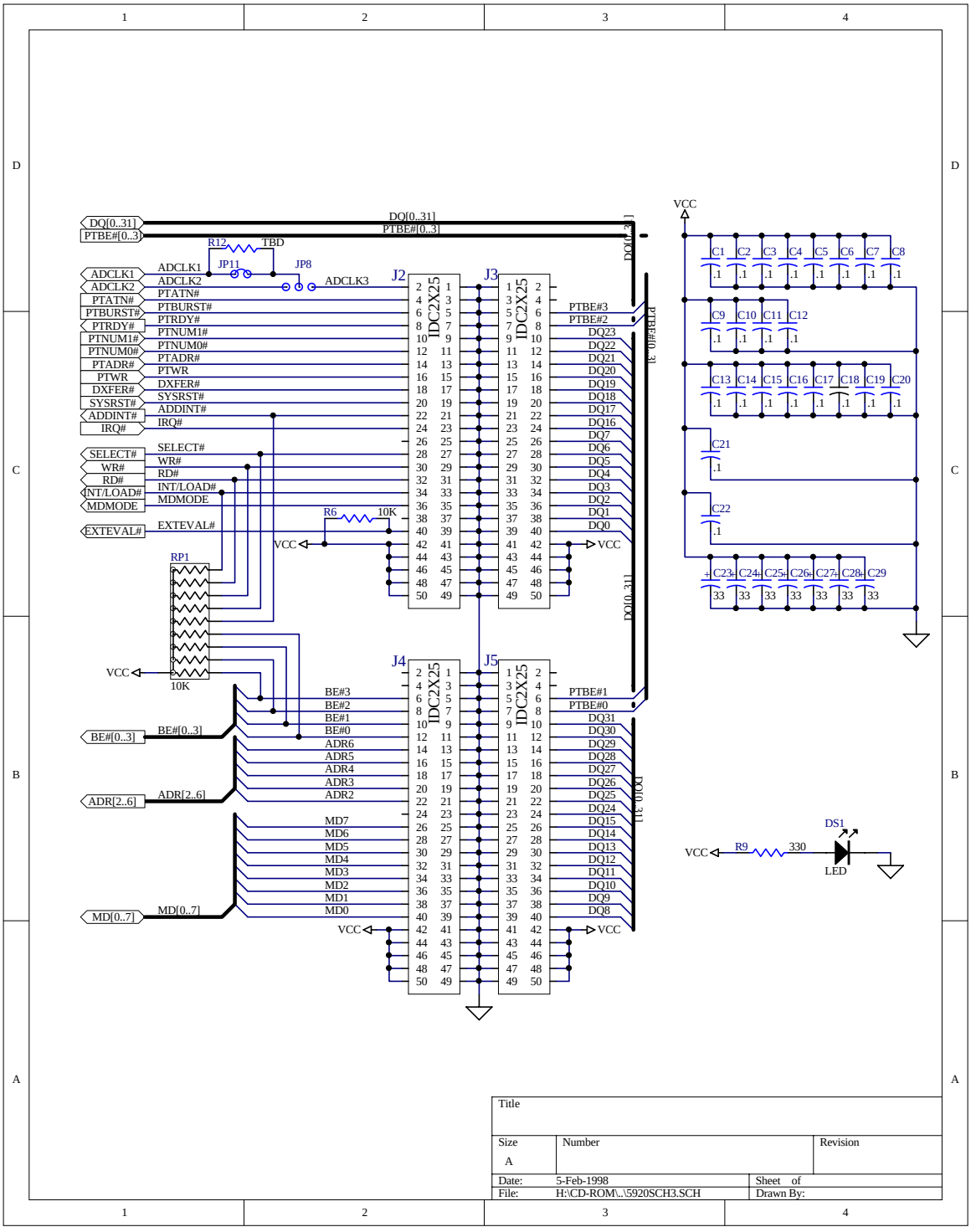


S5920 PCI Card Block Diagram

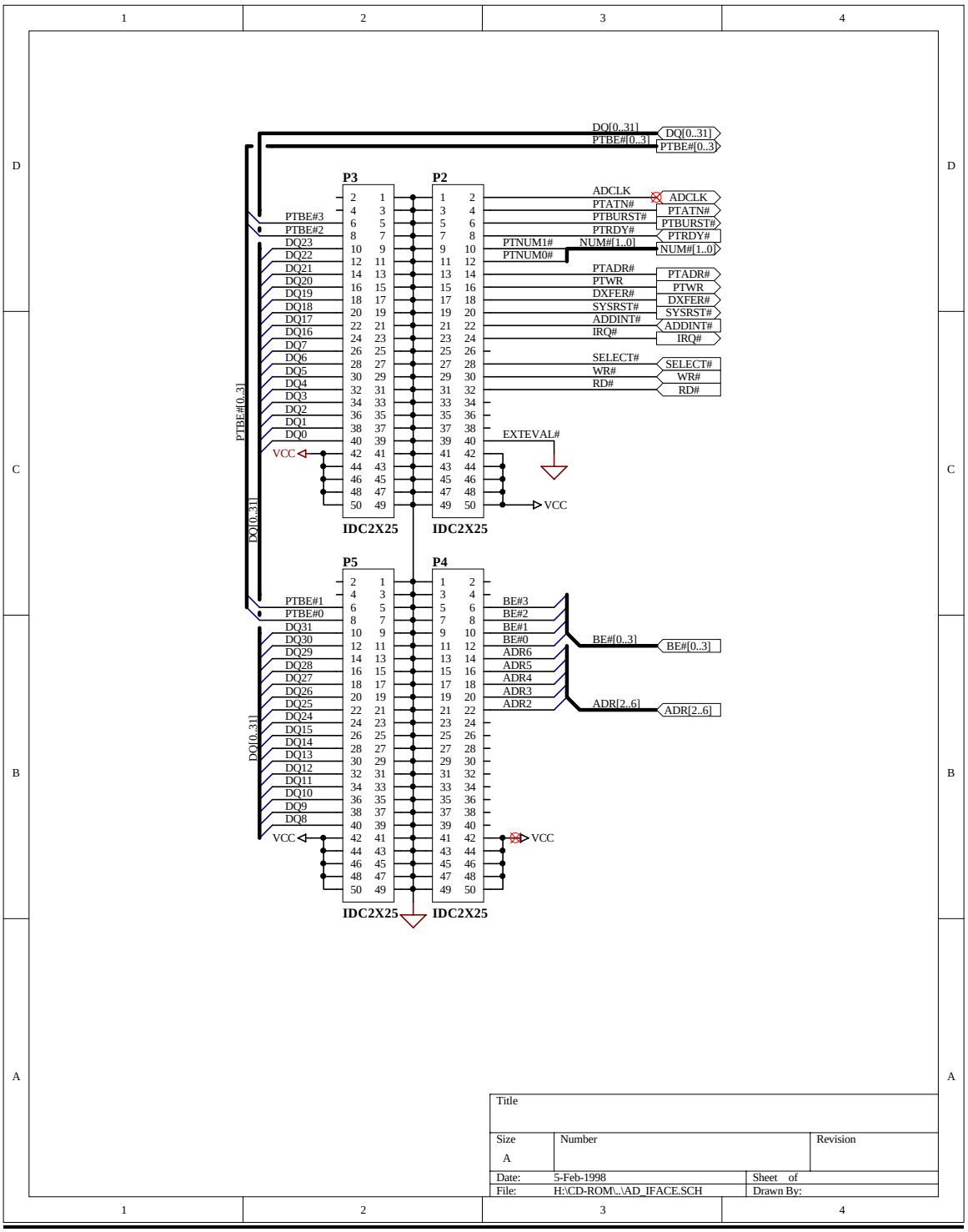


S5920 PCI Card Sheet 1

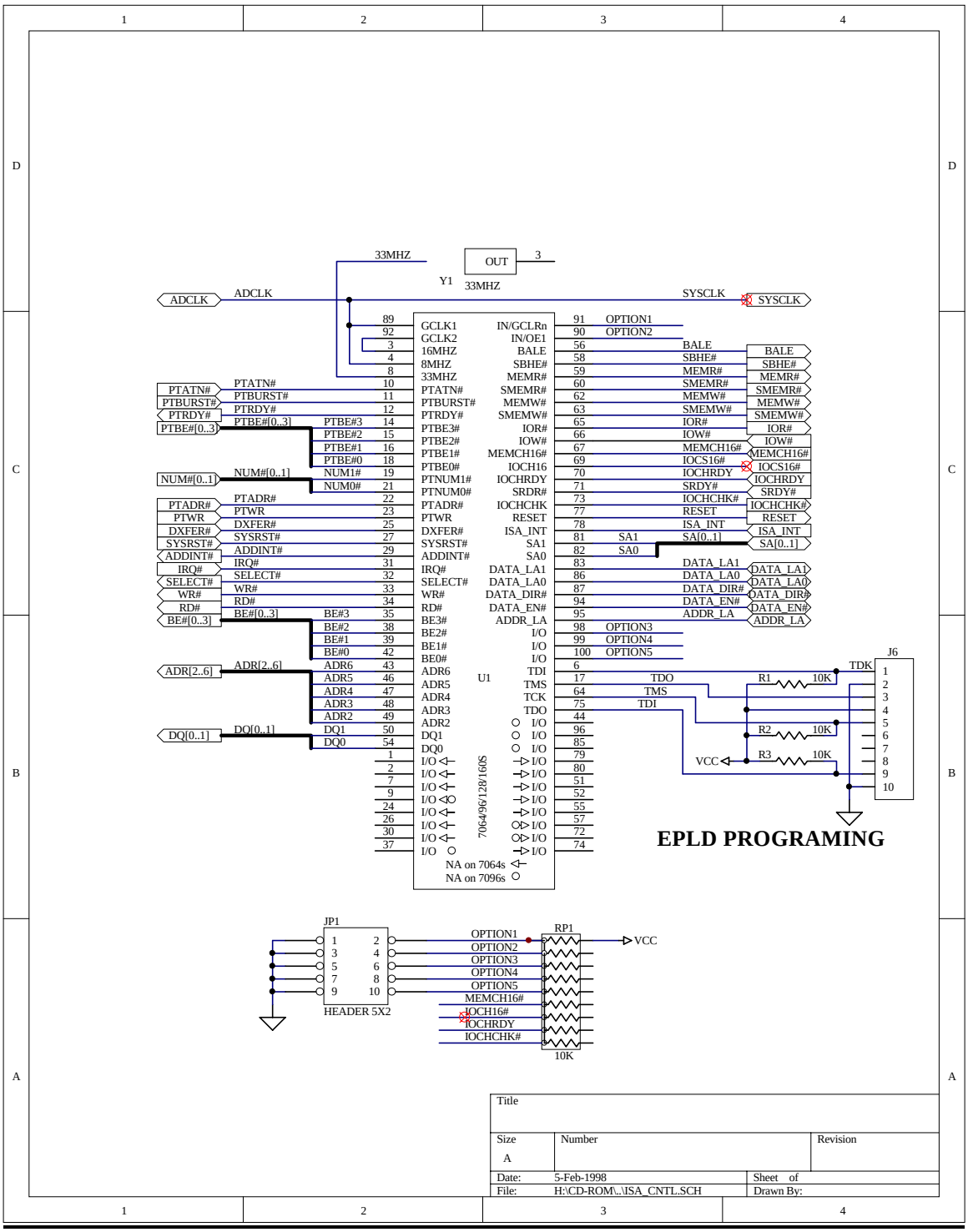

HIGH SPEED BURST MODE MEMORY INTERFACE
DQ[0..31]
MAD[0..16]
U6
Burst Mode Memory PLD
U3
U2
U5
U4
128/32KX8RAM
J6
EPLD PROGRAMING
JP6
JP7
JP9
R10 10K
R13 10K
R14 10K
R15 10K
C31 TBDR7
TBD
TBDY1
MEMCLK
ADCLK2
VCC
Title
Size A
Number
Revision
Date: 5-Feb-1998
Sheet of
File: H:\CD-ROM\..\5920SCH2.SCH
Drawn By:



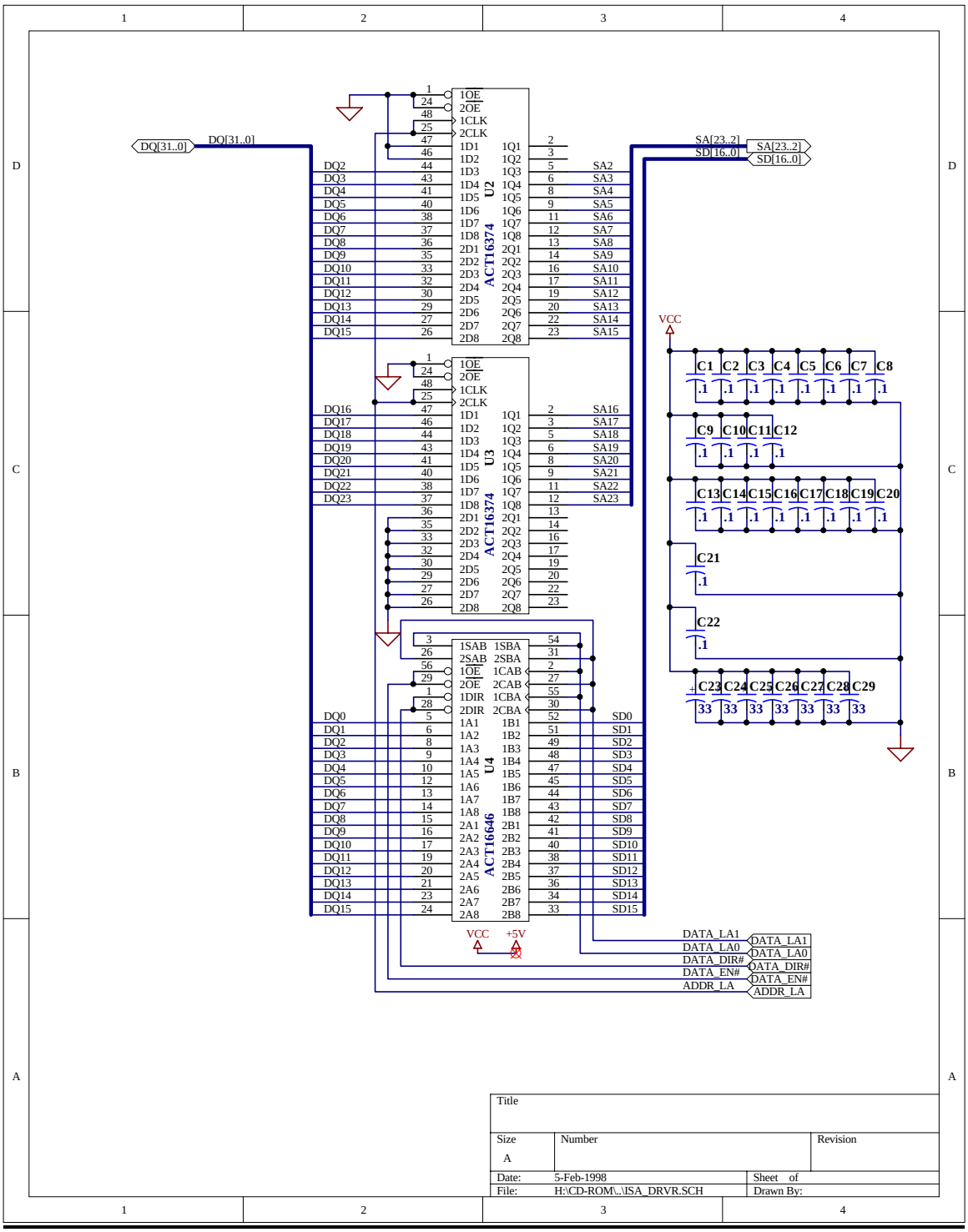
S5920 PCI Card Sheet 3



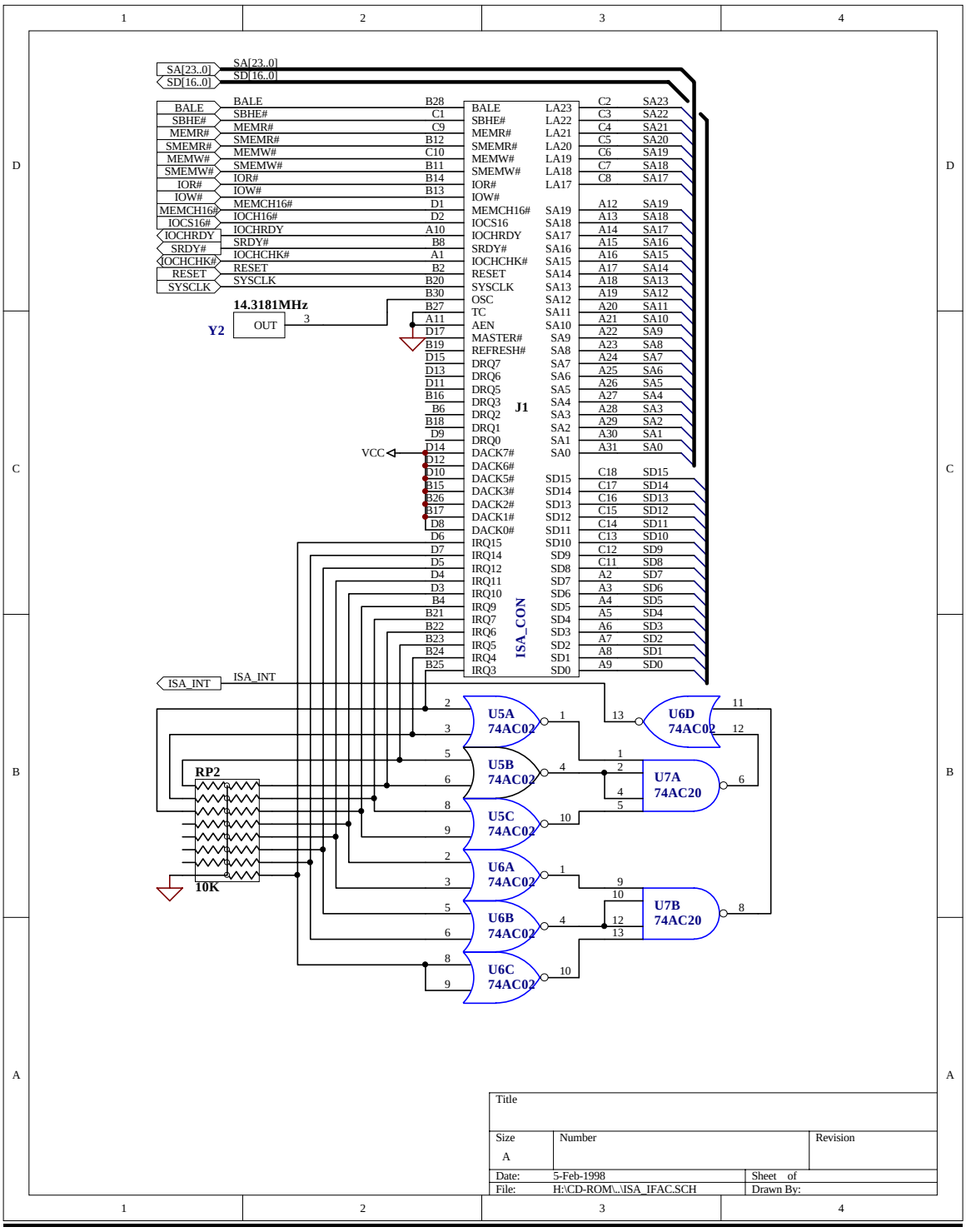
S5920 ISA Adaptor Card Sheet 1



S5920 ISA Adaptor Card Sheet 2



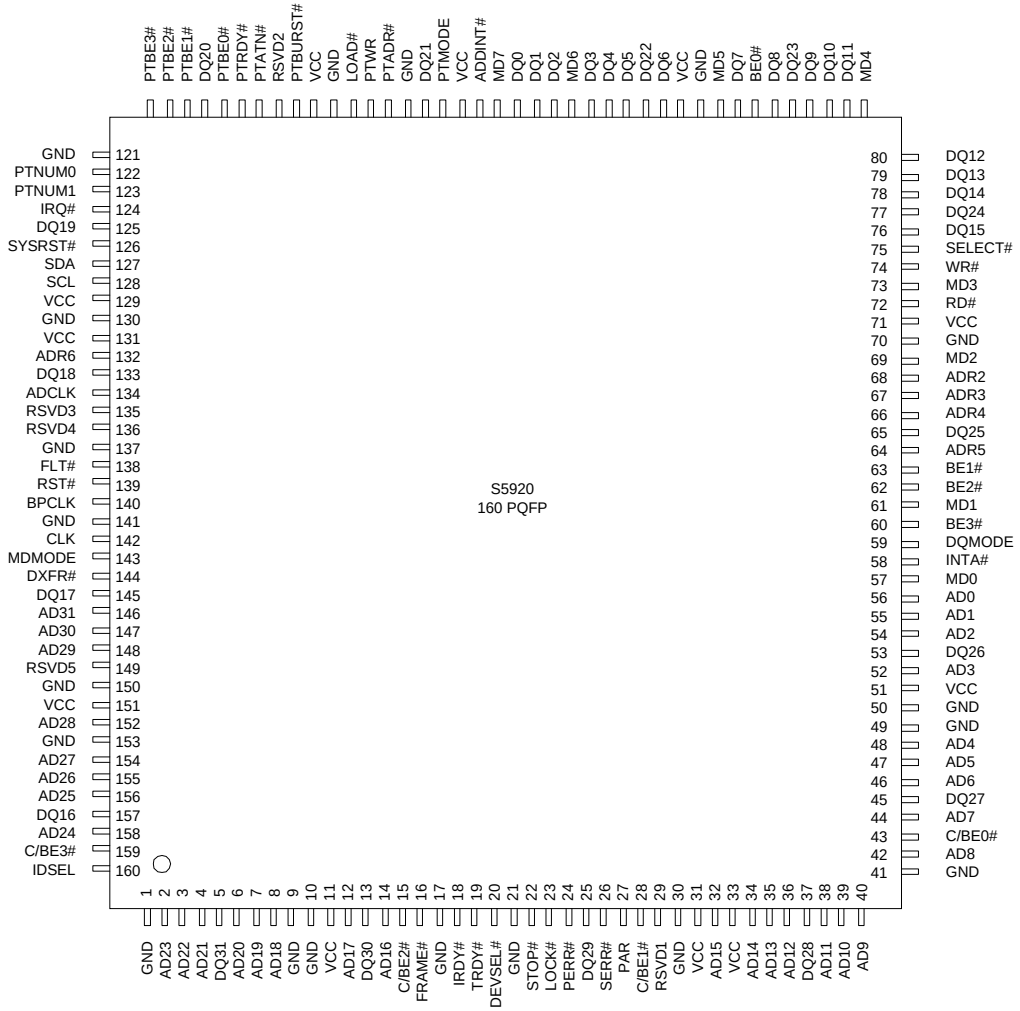
S5920 ISA Adaptor Card Sheet 3

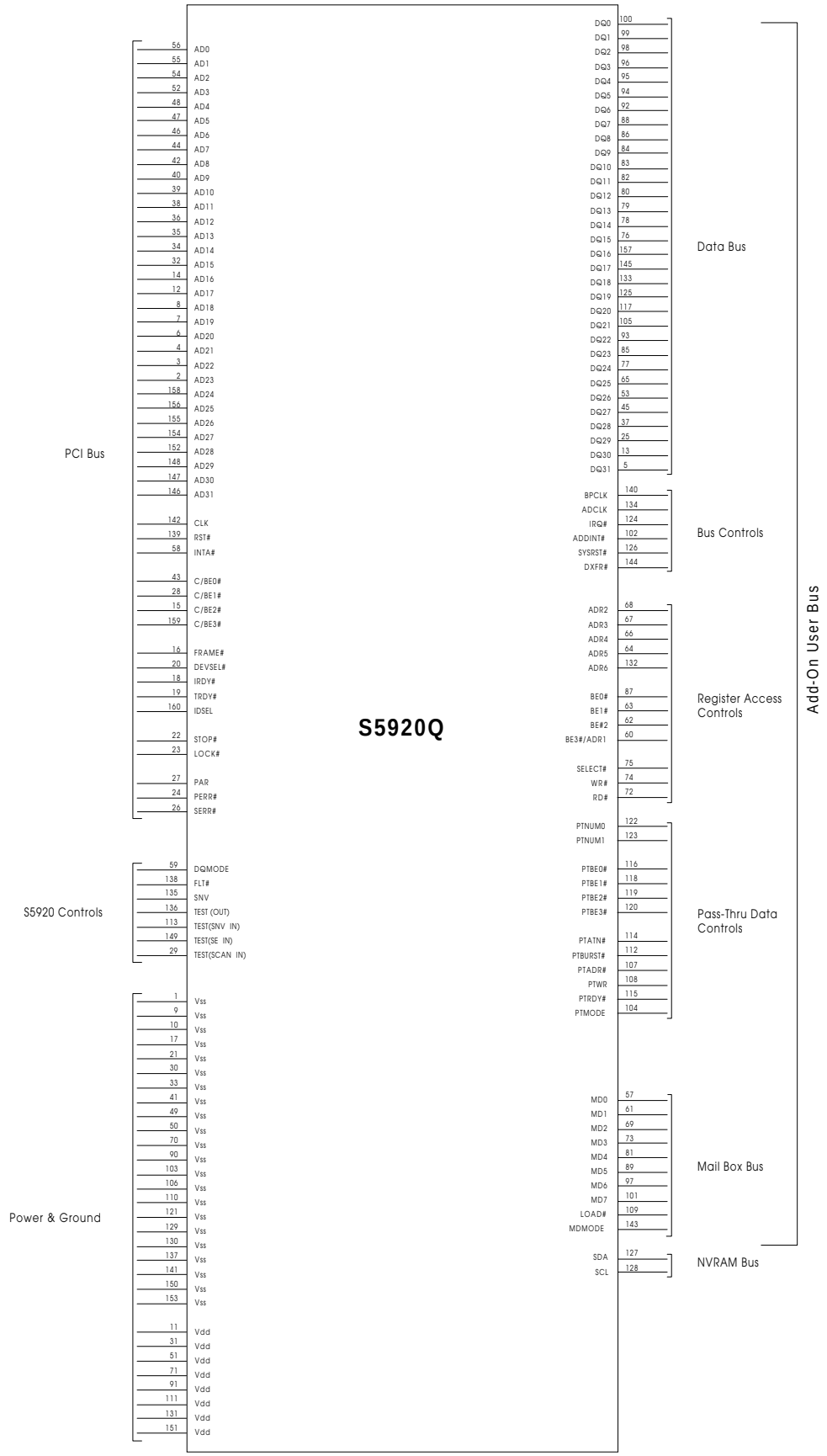


S5920 ISA Adaptor Card Sheet 4

Appendix B

S5920 Pinout and Pin Assignment




S5920Q
PCI Bus
56 AD0
55 AD1
54 AD2
52 AD3
48 AD4
47 AD5
46 AD6
44 AD7
42 AD8
40 AD9
39 AD10
38 AD11
36 AD12
35 AD13
34 AD14
32 AD15
14 AD16
12 AD17
8 AD18
7 AD19
6 AD20
4 AD21
3 AD22
2 AD23
158 AD24
156 AD25
155 AD26
154 AD27
152 AD28
148 AD29
147 AD30
146 AD31
142 CLK
139 RST#
58 INTA#
43 C/BE0#
28 C/BE1#
15 C/BE2#
159 C/BE3#
16 FRAME#
20 DEVSEL#
18 IRDY#
19 TRDY#
160 IDSEL
22 STOP#
23 LOCK#
27 PAR
24 PERR#
26 SERR#
S5920 Controls
59 DQMODE
138 FLT#
135 SNV
136 TEST (OUT)
113 TEST(SNV IN)
149 TEST(SE IN)
29 TEST(SCAN IN)
Power & Ground
1 Vss
9 Vss
10 Vss
17 Vss
21 Vss
30 Vss
33 Vss
41 Vss
49 Vss
50 Vss
70 Vss
90 Vss
103 Vss
106 Vss
110 Vss
121 Vss
129 Vss
130 Vss
137 Vss
141 Vss
150 Vss
153 Vss
11 Vdd
31 Vdd
51 Vdd
71 Vdd
91 Vdd
111 Vdd
131 Vdd
151 Vdd
DQ0 100
DQ1 99
DQ2 98
DQ3 96
DQ4 95
DQ5 94
DQ6 92
DQ7 88
DQ8 86
DQ9 84
DQ10 83
DQ11 82
DQ12 80
DQ13 79
DQ14 78
DQ15 76
DQ16 157
DQ17 145
DQ18 133
DQ19 125
DQ20 117
DQ21 105
DQ22 93
DQ23 85
DQ24 77
DQ25 65
DQ26 53
DQ27 45
DQ28 37
DQ29 25
DQ30 13
DQ31 5
Data Bus
BPCLK 140
ADCLK 134
IRQ# 124
ADDINT# 102
SYSRST# 126
DXFR# 144
Bus Controls
ADR2 68
ADR3 67
ADR4 66
ADR5 64
ADR6 132
BE0# 87
BE1# 63
BE#2 62
BE3#/ADR1 60
SELECT# 75
WR# 74
RD# 72
Register Access Controls
PTNUM0 122
PTNUM1 123
PTBE0# 116
PTBE1# 118
PTBE2# 119
PTBE3# 120
PTATN# 114
PTBURST# 112
PTADR# 107
PTWR 108
PTRDY# 115
PTMODE 104
Pass-Thru Data Controls
MD0 57
MD1 61
MD2 69
MD3 73
MD4 81
MD5 89
MD6 97
MD7 101
LOAD# 109
MDMODE 143
Mail Box Bus
SDA 127
SCL 128
NVRAM Bus
Add-On User Bus