Study on PCI Bus and ISA Bus Conversion Design

Jinbiao Hou

Department of Computer Science and Technology, Dezhou University, Dezhou 253023, China, houjinb@126.com

Abstract

At present there are still a large number of ISA bus boards in use. Providing an interface to the existing ISA products is a problem faced by the majority of developers and users. To resolve this problem, a PCI-to-ISA adapter is designed. It includes basic architecture design and adapter hardware circuit design. Thereinto hardware circuit design includes PCI bus interface circuit design, ISA bus interface circuit design and I/O detection module design. Then the serial EEPROM 93LC46B interface design and configuration is introduced. Finally, the I/O detection module and the device based on the ISA interface are debugged. The debugging results and practice show that the design is feasible, reasonable and useful.

Keywords: PCI Bus, ISA Bus, EEPROM

1. Introduction

In 1999, companies such as Microsoft and Intel introduced the personal computer hardware and software specification which was the PC99 specification. It provided that the production of personal computers would no longer use the ISA bus which was replaced by the new PCI bus standard. However, after more than ten years of accumulation and precipitation, mature products based on the ISA bus abound, especially in the field of industrial control there are still a large number of ISA interface equipments. Therefore how to convert PCI bus into ISA bus and to provide an interface to the existing ISA products is a problem faced by the majority of developers and users. The study on PCI-to-ISA conversion technology emerged in this context.

2. Interface design technology based on the PCI bus

Due to the excellent performance of the PCI bus and its status, it makes PCI devices become one of the most widely used personal computer external devices, and one of the key technologies of design of PCI devices is the implementation of PCI local bus interface [1]. Combined with the status of domestic and international research, the implementation of PCI local bus interface design mainly has the following two methods [2].

(1) Use a dedicated PCI interface chip to achieve PCI local bus interface. The main function of the dedicated PCI interface chip is to achieve the PCI local bus interface and to provide easily used and extensible user interface. The main manufacturers have PLX Co. and AMCC Co., common models have AMCC's S5920, S5930, S5933, PLX’s PCI9052, PCI9054, etc.. The advantages of using such special chips to develop PCI devices are short development cycle, no complex interface logic design and no need to fully understand and grasp PCI specification details. Its disadvantage is that users may only use part of the special chip functions. It will cause some waste of resources.

(2) Achieve PCI local bus interface based on CPLD or FPGA technology. Compared with a dedicated interface chip this method requires designers to fully understand and grasp PCI specification details and requires self-realization of the complex interface logic. Designing PCI interface in this way is very cost-effective. It can save board area and reduce power consumption, but will extend the development cycle.

Designers can also implement the program through purchase of IP (intellectual property) cores which can achieve PCI interface. However this method is expensive. An IP core is generally thousands of dollars and the cost of supporting software tools is also very expensive. This is more suitable in high volume applications. If the number is small it is best not to apply this method.
3. The PCI-to-ISA adapter design

3.1. The basic architecture of the PCI-to-ISA adapter

PCI9052 is an interface chip which is launched for the expansion of an adapter board by PLX Technology Co. and which can provide a mixed high-performance PCI bus target (slave) mode. Its power consumption is low. It uses the PQFP160 pins package and accords with the PCI2.1 specification. The local bus can be set to 8/16/32 bits (non-) multiplexed bus by programming and the data transfer rate can reach 132Mb/s [4].

The ISA bus mode of PCI9052 provides an ISA logical interface. Users can directly convert PCI bus and ISA bus. They can very easily convert ISA design to PCI design and make ISA adapter rapidly and cost-effectively converted to the PCI bus. It is this feature that makes PCI9052 be the best choice in achieving PCI and ISA bus conversions [5].

The basic hardware architecture of the PCI-to-ISA adapter is shown in Figure 1. PCI9052 serves as the intermediate bridge in the PCI bus to ISA bus conversion.

Figure 1. The basic hardware architecture of the PCI-to-ISA adapter

The PCI bus interface provided by PCI9052 integrates a bus interface which is compatible with the PCI2.1 specification. As one end of the bridge it is connected with the PCI bus (implemented through inserting the adapter into a slot on the PC motherboard). This is also a interface which a PCI device must have. At the same time PCI9052 need also be connected with EEPROM to complete the PCI device initialization and the configuration of some properties of PCI9052.

When PCI9052 is set to the ISA interface operating mode, the ISA bus interface provided by PCI9052 provides a bus interface which is compatible with the standard ISA specification. As the other end of the bridge it can be directly connected with ISA devices. Thereby it achieves a smooth transition of the PCI bus to ISA bus.

The "bridge" role of PCI9052 is achieved through the address space mapping mechanism. The mapping mechanism of the address space maps the PCI address space to the address space of ISA devices. Thus accessing PCI devices will be converted to accessing ISA devices.

As the original ISA cards have passed years of market test, their circuit design programs have been very mature and they can meet actual needs. Therefore, this design does not change ISA card circuit design, but according to the above principle of the bridge uses the ISA mode of PCI9052 to achieve PCI-to-ISA adapter. Thus ISA devices are directly converted into PCI devices.

From a general perspective, this design is applicable in most of ISA interface cards. As long as ISA interface signals provided by PCI9052 contain interface signals of ISA cards and targeted changes are made to EEPROM initialization content and drivers, this design can be applied.

Due to the unique electrical specifications of the PCI bus devices, in the design of the PCI-to-ISA adapter we need to pay special attention to the following aspects for PCI9052 (signal name suffix # indicates that the signal is active low).

1. The bus mode selection signal MODE needs to be grounded. PCI9052 works in ISA mode, so it can be only the non-multiplexed mode, that is that the address and data bus are used separately.

2. In PCI signals, prs1 and prs2 must at least one be grounded, otherwise the system can not find the designed board. Microcomputer system is to rely on these two signals to determine whether there are cards in slots. Their connections are related to the power used by PCI cards. The specific meanings are shown in Table 1 (where 0 is vacant, 1 means grounding).
### Table 1. Meanings of different connections of prsnt1 and prsnt2

<table>
<thead>
<tr>
<th>prsnt1</th>
<th>prsnt2</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>no card</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>15W</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>25W</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>7.5W</td>
</tr>
</tbody>
</table>

(3) PCI9052 does not support the EEPROM which does not have continuous reading data function. And it can not use 8-bit EEPROM chips. For some serial EEPROM, PLX does not support them write, such as 93LC46 of ATMEL, but 93LC46 of MICROCHIP can. For 93LC46 of MICROCHIP, its ORG pin does not have to connect Vcc. If it connects Vcc, it supports the write operation.

(4) Special pins connection. Pin LRDYi# already has an 80kΩ internal pull-up resistor and pin TEST has a 50kΩ internal pull-down resistor, so pin LRDYi# and TEST can be vacant. Pin LHOLD locally holds request signals, it should be grounded through a 10kΩ resistor. If this signal is not dealt with, reading and writing chips will crash.

(5) The use of decoupling capacitors. The role of decoupling capacitors is to eliminate interference on the power cord and to stabilize power input. A circuit board may require different levels of decoupling capacitors. The greater the capacitance value is, the greater interference ripples which are filtered out are. For the boards designed with PCI9052 chips, at least 8 0.1μF capacitors are usually taken for decoupling. We propose to add a 0.1μF capacitor as possible for the Vcc pin of each chip on the plug-in board. For the power supply provided by PCI slots, the capacitors that the capacitance value is greater should be used to filter. The selection of 22μF capacitors meets the requirements.

(6) The use of the drive. To increase the external output data of the PCI card and the address signal current, we generally use 74244 as the address signal driver and use 74245 as the driver of the data signal. When users design circuits, according to the situation they can also do without them.

### 3.2. Hardware circuit design of the PCI-to-ISA adapter

This PCI-to-ISA adapter can convert PCI bus operation into 8/16 bits ISA bus memory operation and I/O operation. It sets the memory space address range from 0x100 to 0x10F and I/O space address range from 0x200 to 0x20F. It does not support interrupts and DMA. 8 light-emitting diodes are used for the I/O output detection, that is to display low 8-bit data of the ISA bus. A switch is used for the I/O input detection, that is to set the lowest position of the ISA bus data to 0 or 1. Two jumpers allow users to choose whether prsnt1 and prsnt2 are grounded, that is to choose the maximum power consumption of the PCI-to-ISA adapter.

#### 3.2.1. PCI bus interface circuit design

This PCI-to-ISA adapter is designed to a PCI card which is 32-bit and which works in 5V power supply signal environment. Because of using dedicated interface chip PCI9052, complex PCI bus protocol is avoided. Thus the signal connection in the PCI side is made easier. Figure 2 is the connection circuit diagram of the PCI-to-ISA adapter with PCI bus.

(1) prsnt1 and prsnt2 are represently connected with jumper CON2, that is that the right to select the maximum power consumption of the adapter is handed to users. It is selected by users.

(2) The adapter works in a 5V signal environment, so all 3.3V signals provided by PCI bus should be vacant processed.

(3) The adapter is a 32-bit PCI board, so 64-bit transmission request signal REQ64# and 64-bit transmission response signal ACK64# should be vacant processed.

(4) INTA#, INTB#, INTC# and INTD# are used to implement the interrupt request. Generally only INTA# is used, the next three can only be used for multi-function devices or cascade devices. This design does not use them and makes vacant processing.

(5) The adapter does not implement JTAG boundary scan, so pins TDI and TDO must be connected in order to make scan not disconnect. Test clock TCK, test mode selection TMS and test reset TRST# pin are vacanty processed.
(6) The adapter is not connected with other PCI devices and arbitration pins are not used, so bus occupancy request signal REQ# and the signal GNT# which allows master devices to get the bus use right are vacantly processed.

(7) The adapter does not use the PCI memory which has the caching feature, so compensation monitoring signal SBO# of cache pins and monitoring signal SDONE can be left vacant.

(8) 5V signal provided by the PCI bus provides power for all chips of the PCI-to-ISA adapter.

(9) 12V signal and -12V signal provided by the PCI bus respectively connect to 12V and -12V power supply pins in an ISA slot.

(10) 32-bit address and data multiplexed signal AD[31:1] are directly connected to the corresponding pins of PCI9052.

(11) The control signals provided by the PCI bus, such as CLK, C/BE[3:1], IRDY#, DEVSEL#, LOCK#, PERR#, SERR#, INTA#, RST#, IDSEL, FRAME, TRDY#, STOP#, PAR, are directly connected to the corresponding pins of PCI9052.

3.2.2. ISA bus interface circuit design

In a different mode, some of the pins of PCI9052 have different definitions and functions, so functions and connections of each pin of PCI9052 are required to be well understood [6]. For example, PCI9052 is set in the ISA mode, which pins need connect the power supply through pull-up resistors, which pins need be grounded through pull-down resistors, which pins are left vacant, which pins must be connected directly to the ISA slot.

ISA bus interface circuit connection is not so simple as PCI bus interface circuit (PCI9052 and PCI bus signals can directly connect). It requires the designer fully understand PCI9052 and ISA bus signals in order to ensure that the connection is right. This example will in turn introduce all kinds of signals in the ISA side how to be connected. Figure 2(b) and 3 are connection circuit diagrams of the PCI-to-ISA adapter with ISA bus.

(a) The PCI bus pins connection diagram
(b) The PCI9052 with PCI and ISA bus pins connection diagram

Figure 2. Connection circuit diagrams of the PCI-to-ISA adapter with PCI bus

Figure 3. The ISA bus pins connection diagram

(1) The adapter does not support DMA, so all DMA channel request signals DRQ# are vacant, DMA channel response signal DACK# connects the power and is set high, but count end signal TC has to ground.

(2) PCI9052 provides high byte enable signal SBHE# for ISA slots, so memory 16-bit chip-select signal MEM CS16# on ISA bus and I/O 16-bit chip-select signal IO CS16# can be vacant.

(3) In the design the adapter does not support interrupts. But taking into account that the users with ISA equipments may use interrupts, LINTi1 and LINTi2 are respectively connected to two pins with the highest priority in 6 external interrupt request signals on ISA bus. That is IRQ3 and IRQ4.

(4) The system does not require -5V power supply signal, so corresponding pins on the ISA bus are vacant.

(5) Taking into account ISA devices not as bus master devices, the master signal pin MASTER# is vacantly processed.

(6) Power signals 5V, 12V, -12V are provided directly by the computer system.

(7) The designed adapter does not have to refresh dynamic memory, so REFRESH# should be connected to the power supply and be set high.

(8) CHRDY is a local channel ready signal and is generally made pull-up processing, so it is connected to the power supply through a 10kΩ pull-up resistor, but at the same time it is still connected to a corresponding pin of PCI9052.

(9) CHCHK# is a local channel check signal and is generally made pull-up processing, so it is connected to the power supply through a 10kΩ pull-up resistor.
(10) On the ISA bus there are two memory writing and two memory reading signals. They are respectively SMEMW# and MEMW# as well as SMEMR# and MEMR#. The difference is that SMEMW# and SMEMR# are valid only when the memory addressing range is less than 1MB and that MEMW# and MEMR# are valid within the entire 16MB range. However, in the ISA bus protocol time sequences of SMEMW# and MEMW# are same and time sequences of SMEMR# and MEMR# are same too. Therefore SMEMW# and MEMW# can be simultaneously connected to PCI9052 memory write signals, SMEMR# and MEMR# can be simultaneously connected to PCI9052 memory reading signals.

(11) LAD[15:0] is a 16-bit data bus.

(12) LA[23:2] and ISAA[1:0] together form the ISA address bus. For the 8-bit data bus, ISAA[1:0] is equivalent to LA[1:0], they participate in the address decoding together. For the 16-bit data bus, they read and write two bytes at a time. Then ISAA[0] is a low byte valid signal and it does not participate in the address decoding. ISAA[1] and LA[23:2] together make the address decoding and ISAA[1] is the lowest address bit. Note that not all of the address lines must be decoded and that we should select the number of decoded address lines according to the actual on-board memory (I/O) space size.

(13) LCLK is an ISA end clock signal, according to chip requirements it is connected to an external 8MHz active crystal oscillator.

(14) OSC is an oscillation signal, the master clock provides a square wave that the duty cycle is 50%. The typical used frequency of the PC/XT machine is 14.31818MHz, but now the PC has not provided such frequency oscillation signals. This design connects the signal to the PCI9052's BCLKO pin.

(15) For other signals of ISA end, such as LRESET, NOWS#, IOWR#, IORD#, BALE, SBHE#, they can be directly connected to corresponding pins of PCI9052.

3.2.3. I/O detection module design of the adapter

In order to easily debug the adapter and quickly check the address space mapping correctness after the completion of the adapter, this design increases an I/O ports detection circuit on the adapter. As shown in Figure 4, in the I/O detection module, decoding of ISA bus low address LA[9:2] outputed by PCI9052 which works in ISA mode makes chip selection to the detection circuit, output data is displayed by the light-emitting diodes and the switch inputs binary data.

![Figure 4. The I/O detection circuit](image)

(1) The output detection circuit

The output detection circuit consists of light-emitting diodes, a latch, a drive and current limiting resistors. Light-emitting diodes in the industrial control system are the most common display device. When certain current flows, light-emitting diodes can emit light. The current size generally requires about 10mA. Therefore, the current limiting resistors take 510Ω. The former level of the light-emitting diode uses a reverse drive 74240. It mainly plays a driving role. 74273 is a latch and it outputs stable data for the post-stage circuit. The chip-select address is 200H to 203H, that is to say that only when the address of the ISA bus is 200H to 203H, the light emitting diodes display the output data.

(2) The input detection circuit

The data input is achieved by using a tri-state gate to connect a switch. The switch can select connecting power supply or grounding. Thus it inputs data to the lowest bit on ISA data bus. The tri-state gate selection is 74125. The circuit can connect up to four. The tri-state gate selection can also be...
In few applications of a switch, this tri-state gate interface is very convenient. The chip-select address of the tri-state gate is 204H to 207H, that is to say that only when the address of the ISA bus is 204H to 207H, the switch can work.

4. The serial EEPROM 93LC46B interface design and configuration

4.1. EEPROM interface design

Using PCI9052 requires an external EEPROM to provide configuration information. This PCI-to-ISA adapter uses Microchip Technology's product 93LC46B (64×16 bit EEPROM). In practical applications, 93LC46B is word-addressable. It has a total of 84 words [8]. PCI9052 uses serial transmission interface lines EECS, EESK, EEDI, EEDO to complete data exchange with 93LC46B. The most important function of 93LC46B is to save PCI9052 work after power down (determined by the register). After the RSTB pin effective process or the 9346CR register automatically load command, the PCI9052 achieve a series of EEPROM read operations. It reads out the content in the 93LC46B [9]. Figure 5 is the interface circuit diagram of the serial EEPROM 93LC46B.

![Figure 5. The interface circuit diagram of EEPROM 93LC46B](image)

93LC46B effectively starts work through the chip-select signal and makes access through three serial interface signal lines [10]. CS in the figure represents a chip-select signal. It is active high. SK (Serial Data Clock) is a clock input signal, data reading and writing is synchronized with SK. The automatically timed writing clock does not require the SK signal. DI (Serial Data Input) is a serial data input signal, accepts the commands, the address and data from the outside. DO (Serial Data Output) is a serial data output signal and is used to output the data of the selected address. For 93LC46B, both 6 feet and 7 feet should be vacant.

4.2. The configuration information of the EEPROM

For the special interface chip PCI9052, its configuration information is through the serial EEPROM to store and to load in the chip reset. The EEPROM content is directly related to the PCI9052 ability to work correctly. Table 2 is the contents of the EEPROM when the memory space address range is from 0x100 to 0x10F, the I/O space address range is from 0x200 to 0x20f and it only makes 16-bit memory or I/O operations (the addresses and the contents are expressed in hexadecimal). The EEPROM specific configuration information is as follows.

1. The device number DID is 0x9050. The manufacturer VID is 0x10B5. The subsystem number SDID is 0x0000. The subsystem manufacturer SVID is 0x0000. The device type number is 0x06800000. Thereinto DID and VID generally use the default value. This is the flag of the PLX 9052 chip. But SDID and SVID can set up by yourself and we can also free apply for a globally unique ID to the PLX. For the device type number we can query a manual and we can also specify one to it. This case is 0x06800000. When the computer starts, it will prompt to find a PCI bridge device. In ISA mode, the local space 1 must be mapped to I/O space, the local space 0 must be mapped to memory space.

2. The local space 0 range register value is 0xFFFFFFFF. It indicates that the memory space size is 16 bytes. That bit0 is 0 indicates that this space is mapped to memory space.

3. The local space 0 base address register value is 0x00000101. It indicates that the base address of the space 0 is 0x100. That bit0 is 1 indicates that the space 0 is enabled.

4. The local space 0 descriptor register value is 0x00000022. Thereinto that bit[23:22] is 01 indicates that the local space 0 data bus width is 16 bits.

5. The local space 1 range register value is 0xFFFFFFFF. It indicates that the I/O space size is 16 bytes. That bit0 is 1 indicates that this space is mapped to the I/O space.
(6) The local space 1 base address register value is 0x00000102. It indicates that the base address of the space 0 is 0x200. That bit0 is 1 indicates that the space 1 is enabled.

(7) The local space 0 descriptor register value is 0x00400022. Thereinto that bit[23:22] is 01 indicates that the local space 1 data bus width is 16 bits.

(8) The chip-select 0 base register value is 0x00000109. It indicates that when the local address falls 0x100 to 0x10F, chip selection is in effect. In ISA mode, although there is no chip-select signal 0 you must set it and make its value match with the local space 0 base address and range, otherwise the local space is unable to respond to the PCI memory control command. Similarly, The chip-select 1 base register value also need match with the local space 1 base address and range, otherwise the local space is also unable to respond to the PCI I/O control command. The value of this register is very important.

(9) The interrupt control status register value is 0x00001000. It indicates not to use the interrupt. Thereinto that bit12 is 1 indicates that the ISA mode is enabled. If it is not 1, then the PCI9052 will work in C mode and will cause errors.

<table>
<thead>
<tr>
<th>Offset address</th>
<th>00 01 02 03</th>
<th>04 05 06 07</th>
<th>08 09 0A 0B</th>
<th>0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000H</td>
<td>50 90</td>
<td>B5 10</td>
<td>80 06</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000010H</td>
<td>FF FF</td>
<td>F0 FF</td>
<td>FF FF</td>
<td>F1 FF</td>
</tr>
<tr>
<td>00000020H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>01 01 00</td>
</tr>
<tr>
<td>00000030H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000040H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000050H</td>
<td>00 00 00</td>
<td>09 02 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000060H</td>
<td>45 00 92 44</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
</tbody>
</table>

Table 2: The EEPROM configuration information

In order to compare with Table 2, the EEPROM configuration informations that the adapter works in other cases are respectively given below.

Table 3 indicates that the base address of the I/O space changes from 0x200 to 0x320, but the space size is unchanged. This requires making corresponding modifications in the local space 1 base address register and chip-select 1 base register.

<table>
<thead>
<tr>
<th>Offset address</th>
<th>00 01 02 03</th>
<th>04 05 06 07</th>
<th>08 09 0A 0B</th>
<th>0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000H</td>
<td>50 90</td>
<td>B5 10</td>
<td>80 06</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000010H</td>
<td>FF FF</td>
<td>F0 FF</td>
<td>FF FF</td>
<td>F1 FF</td>
</tr>
<tr>
<td>00000020H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>01 01 00</td>
</tr>
<tr>
<td>00000030H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000040H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000050H</td>
<td>00 00 00</td>
<td>09 23 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000060H</td>
<td>45 00 92 44</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
</tbody>
</table>

Table 3: The EEPROM Configuration information after changing the chip-select base address

Table 4 shows that the I/O operation to the ISA bus is from 16 bits to 8 bits. This requires making corresponding modifications in the local space 1 description register.

<table>
<thead>
<tr>
<th>Offset address</th>
<th>00 01 02 03</th>
<th>04 05 06 07</th>
<th>08 09 0A 0B</th>
<th>0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000H</td>
<td>50 90</td>
<td>B5 10</td>
<td>80 06</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000010H</td>
<td>FF FF</td>
<td>F0 FF</td>
<td>FF FF</td>
<td>F1 FF</td>
</tr>
<tr>
<td>00000020H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>01 01 00</td>
</tr>
<tr>
<td>00000030H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000040H</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000050H</td>
<td>00 00 00</td>
<td>09 02 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
<tr>
<td>00000060H</td>
<td>45 00 92 44</td>
<td>00 00 00</td>
<td>00 00 00</td>
<td>00 00 00</td>
</tr>
</tbody>
</table>
5. The PCI-to-ISA adapter debugging

5.1. The I/O detection module debugging

The working principle of the I/O detection module is by address chip selection either to let light-emitting diodes display output data, or to read the data given by the switch. The I/O detection module debugging is made in "debug" of DOS. Debugging methods are respectively introduced below in output detection and input detection.

Figure 6 shows the detection command of the data output in “debug” of DOS. “o” indicates output, that is output command. “cfa0” is a chip-select address. The range of BAR3 is from 0xcfa0 to 0xcfaf. The address of BAR3 is in one-to-one correspondence with the ISA bus I/O space address. Therefore the PCI bus address 0xcfa0 also corresponds to the ISA bus address 0x200. The ISA bus address from 0x200 to 0x203 is the latch 74LS273 chip-select address which provides displaying output data. It includes 0x200. Therefore by "o" command the data 00H, FFH, 55H, etc. are output to the latch and are loaded into the light-emitting diodes. Thus it can detect the correctness of the output operation. After three output commands execution, 8 light-emitting diodes display all out, all on and out between appearance.

Figure 6. The detection command of the data output

Figure 7 shows the detection command of the data input in “debug” of DOS. “i” indicates input, that is input command. “cfa4” is a chip-select address. Similar to the testing situation of data output, we can infer that the address 0xcfa4 of the PCI bus corresponds to the ISA bus I/O space address 0x204. The ISA bus address 0x204 to 0x207 is the switch input data chip-select address. It includes 0x204. Therefore data input can be detected. The first command in the figure is executed after the switch is set high, so the lowest bit of reading data displayed on the computer is 1; the second command is executed after the switch is set low, so the lowest bit of reading data is 0. With data output detection, if the ISA bus is set to 16 bits, then only two addresses can make the data input detection circuit work; if the ISA bus is set to 8 bits, then four addresses can all make the data input detection circuit work.

Figure 7. The detection command of the data input

5.2. The debugging based on ISA interface devices

The design chooses an ISA interface D/A board which can output multiple waveforms to debug. The purpose of debugging is to see whether the D/A board still can work normally after it is plugged into the PCI-to-ISA adapter. If it can work properly, it shows that the PCI-to-ISA adapter circuit design as well as the configuration of the EEPROM are correct; otherwise it shows that the PCI-to-ISA adapter circuit design as well as the configuration of the EEPROM have problems, they need to be further refined and improved.

The preparatory work before debugging and debugging process are briefly introduced below.

(1) The preparatory work

The D/A board interfaces are based on ISA bus. It needs to be plugged into the ISA slot. Its data width is 8 bits. It is controlled by the corresponding signal generating software. It can output the
desired signal. The chip-select address of the D/A board is 0x32f and is not within the address range which is set for the I/O detection module by the aforementioned EEPROM, so before using the D/A board the information in the EEPROM need be re-configured. The EEPROM chip-select base address should be changed to 0x320, the I/O space size can be unchanged. The data bus of the D/A board only has 8 bits, so we need to change 16-bit I/O operations to ISA bus to 8-bit I/O operations in the EEPROM corresponding place. Please see Table 3 and Table 4.

The D/A board is inserted in the ISA slot before, so the chip-select address operated by the signal generating software which controls their work is 0x32F. However, now the D/A board is first plugged into the PCI-to-ISA adapter and then is plugged into a PCI slot through the adapter, so the chip-select address operated by the signal generating software correspondingly becomes the PCI address, that is that the address value of the software operation should be changed to 0xcfaf.

(2) The board debugging

After a series of preparatory work, we start joint debugging of the entire system. First the D/A board is plugged into the ISA slot of the PCI-to-ISA adapter, then the PCI-to-ISA adapter is plugged into the PCI slot on the computer's motherboard. After that, the output of the D/A board is connected to the oscilloscope. The connection of the entire system is completed.

Start the computer and enter the operating system, then execute the signal generating application software and enter into the control interface of the application software. Carrier and modulation wave parameters can be set on the control interface. After setting parameters, we click on "output signal", the corresponding modulation wave will be output. Figure 8 shows the waveform of the modulated wave displayed in the oscilloscope.

![Figure 8. The waveform of the modulated wave displayed in the oscilloscope](image)

The actual waveform observed from the oscilloscope compares with the theoretical waveform generated by the signal generating software. The two are completely consistent. This shows that the D/A board is in good working condition. Thus it can be inferred that the PCI-to-ISA adapter circuit design and EEPROM configuration have no problem. We successfully implement the PCI bus to the ISA bus conversion.

6. Conclusion

With the rapid development of computer application technology, the demand for high-performance and high-efficiency panels becomes increasingly large, ISA bus of low performance in microcomputers is bound to be eliminated. However, in practical applications, there are still a large number of boards which support the ISA bus. This design supports ISA devices to access the PCI bus. It is an effective solution to this problem. Debugging results and practice show that the design is feasible, reasonable and useful.

References


