

Re: PCI device driver question

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.development.system/2006-12/msg00268.html>

- *From:* "elliotng.ee@xxxxxxxxxx" <elliotng.ee@xxxxxxxxxx>
 - *Date:* 26 Dec 2006 07:52:08 -0800
-

Thanks for your help. I followed this advice:

Having said all that, if this is just a one-off or occasional special-purpose administrative operation, you may not even need to write a driver (on most platforms at least). You can use the `lspci` program to find the physical address of the pci device's memory. Then you open `"/dev/mem"`, `mmap` the desired area and read / write data directly to or from it in your user program.

Following GH's advice, I looked at `lspci`, and found that the device Region 0 memory is mapped at `a0000000` (32-bit, prefetch, size=512 M)

I wrote the following program:

```
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/mman.h>
#include <asm/io.h>

int main(){

int memory;
unsigned int data;
unsigned char *address;

memory = open("/dev/mem", O_RDWR);
address = mmap((void*)0, 0xffffffff, PORT_WRITE | PORT_READ, MAP_FIXED,
memory, 0xa0000000);

data = readl(address);
printf("Data: %x", data);
writel(0xFF, address);
data = readl(address);
printf("Data: %x", data);

munmap((void*)address, 0xffffffff);
close(memory);
```

Re: PCI device driver question

```
return 0;  
  
}
```

I just want to make sure that my syntax is correct. When I went to run the makefile, however, it did not work. It was having problems recognizing `readl` and `writel` even though both functions should be in `<asm/io.h>`. My makefile is the following:

```
testIO: testIO.c  
gcc -o $@ $<
```

I have the following errors/warnings:

```
In file included from /usr/include/asm/io.h:11 from testIO.c:6:  
/usr/include/asm-i386/io.h:1:2: warning: #warning "You should include  
<sys/io.h>. This time I will do it for you."
```

```
testIO.c: (.text+0x59): undefined reference to 'readl'  
testIO.c: (.text+0x82): undefined reference to 'writel'  
testIO.c: (.text+0x90): undefined reference to 'readl'
```

gil_hamilton@xxxxxxxxxxx wrote:

elliottng.ee@xxxxxxxxxxx wrote:

OK, after doing some research, it appears that I can use the char device `read` and `write` functions in the pci device driver. To summarize what I have, I have written a pci device driver. In userspace, I want to be able to access the pci device driver, read from memory, and write to memory. I have recently found out that I can try to read/write from using file operations.

(1) You need to map the device memory into kernel virtual address space with `ioremap`. Then, in the `test_write` function, you copy data from user space into kernel space (possibly to a small temporary holding area), then use `writel` to put it into the (mapped) device memory (adding in the `f_pos` value).

`test_read` is simply the inverse, `readl` to read the device memory space, then copy it to user space.

`copy_to_user` and `copy_from_user` can be used for data transfer between user space and kernel space. Look for the book "Linux Device Drivers" (3rd edition) -- which is available in pdf format for free online -- for much detailed info.

(2) You create a device node (e.g. "`mknod /dev/mydevice c xxx 0`" where `xxx` is your device's major number from `/proc/devices`). Your program then opens `/dev/mydevice`, `lseek`'s to the proper offset and does `read` or `write` calls.

Re: PCI device driver question

Having said all that, if this is just a one-off or occasional special-purpose administrative operation, you may not even need to write a driver (on most platforms at least). You can use the `lspci` program to find the physical address of the pci device's memory. Then you open `"/dev/mem"`, `mmap` the desired area and read / write data directly to or from it in your user program.

GH