

Ubuntu 10.04 Desktop (32-bit and 64-bit)

D-1. Install the MIport serial card in the computer.

D-2. Open a terminal window. This can be found on the desktop menu under Application | Accessories | Terminal.

D-3. **[optional]** Start recording everything entered into, and output to, the terminal. This step is only required if there is a problem installing the driver.

```
username@computer$ script -t 2> ~/MIport.time -a ~/MIport.script ↵
```

```
Script started on Tue 31 Aug 2010 03:56:11 AM CDT
```

D-4. Extract the archive that stores the MIport driver.

```
username@computer$ tar -zxvf /media/MIport\ 3.0/linux/drivers/2.6/MIport/MIport-3.0.0.tar.gz ↵
```

```
MIport-3.0.0/  
MIport-3.0.0/rc.local  
MIport-3.0.0/bbmknod.sh  
MIport-3.0.0/README  
MIport-3.0.0/README.pdf  
MIport-3.0.0/Makefile  
MIport-3.0.0/MIport.c  
MIport-3.0.0/rc.MIport
```

D-5. Change the current working directory to where the MIport driver source code is located.

```
username@computer$ cd MIport-3.0.0 ↵
```

D-6. Compile the MIport device driver.

```
username@computer$ make ↵
```

```
make -C /lib/modules/2.6.32-24-generic/build M=/home/username/MIport-3.0.0 modules  
make[1]: Entering directory `/usr/src/linux-headers-2.6.32-24-generic'  
CC [M] /home/username/MIport-3.0.0/MIport.o  
Building modules, stage 2.  
MODPOST 1 modules  
CC /home/username/MIport-3.0.0/MIport.mod.o  
LD [M] /home/username/MIport-3.0.0/MIport.ko  
make[1]: Leaving directory `/usr/src/linux-headers-2.6.32-24-generic'
```

D-7. Determine the present working directory. The output may be different than shown below.

```
username@computer$ pwd ↵
```

```
/home/username/MIport-3.0.0
```

D-8. Install the MIport device driver. The part after “PWD=” of the command below should match the output from the command above.

```
username@computer$ sudo make install PWD=/home/username/MIport-3.0.0 ↵
```

Installing MIport 3.0 Driver on Ubuntu 10.04

```
make -C /lib/modules/2.6.32-24-generic/build M=/home/username/MIport-3.0.0 modules_install
make[1]: Entering directory `/usr/src/linux-headers-2.6.32-24-generic'
INSTALL /home/username/MIport-3.0.0/MIport.ko
DEPMOD 2.6.32-24-generic
make[1]: Leaving directory `/usr/src/linux-headers-2.6.32-24-generic'
cp ./rc.MIport /etc/rc.MIport
chmod 755 /etc/rc.MIport
```

D-9. Start the MIport driver.

```
username@computer$ sudo /etc/rc.MIport ↓
```

```
Searching for MIport driver in: ...
The MIport driver was found in: ...
The MIport driver is loaded.
Making special nodes for MIport.

Making /dev/ttyM0 /dev/cum0
Making /dev/ttyM1 /dev/cum1
Making /dev/ttyM2 /dev/cum2
Making /dev/ttyM3 /dev/cum3
Making /dev/ttyM4 /dev/cum4
Making /dev/ttyM5 /dev/cum5
Making /dev/ttyM6 /dev/cum6
Making /dev/ttyM7 /dev/cum7
```

D-10. Verify that the device driver is running. If the output is similar to what is shown below, then the MIport driver is running. If the output is blank, then the MIport driver is not running.

```
username@computer$ dmesg | grep MIport ↓
```

```
[ 271.828182] B&B Electronics PCI MIport serial driver Revision: 4.5
[ 314.159265] MIport 0000:07:05:05.0: PCI INT A -> GSI 26 (level, low) -> IRQ 26
```

D-11. **[optional]** If recording was started in Step 2, stop recording. The file ~/MIport.script contains all output recorded from the terminal.

```
username@computer$ exit ↓
```

```
exit
```

```
Script done on Tue 31 Aug 2010 03:58:48 AM CDT
```

D-12. Test serial communications using gterm. Change the serial port under Configuration | Ports and set the port to /dev/ttyM0 (for each serial port, increment the number at the end of the name; e.g., /dev/ttyM1).

```
username@computer$ gterm & ↓
```

D-13. If the driver works correctly, modify /etc/rc.local to automatically load the driver when the operating system starts.

```
username@computer$ sudo gedit /etc/rc.local
```

Add the following code to the file (before the "exit 0" statement):

```
if [ -f /etc/rc.MIport ] ; then
    exec /etc/rc.MIport
fi
```

An example of the code to add to `/etc/rc.local` is in the MIport driver folder in the file `rc.local`.

D-14. Restart the computer, and then test serial communications again with `gtkterm`.

Ubuntu 10.04 Server (32-bit and 64-bit)

S-1. Determine which version of GCC built the current Linux kernel.

```
username@computer$ cat /proc/version ↵
```

```
Linux version 2.6.32-21-server (buildd@yellow) (gcc version 4.4.3 (Ubuntu 4.4.3-4ubuntu5) ) #32-Ubuntu SMP Fri Apr 16 09:17:34 UTC 2010
```

The current Linux kernel was built using GCC v4.4.3.

S-2. Install the version of GCC that built the current Linux kernel.

```
username@computer$ sudo apt-get update ↵
```

```
:
:
:
Fetched 12.0MB in 50s (237kB/s)
Reading package lists... Done
```

```
username@computer$ sudo apt-get install gcc-4.4 gcc ↓
```

```

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  binutils gcc-4.4 libc-bin libc-dev-bin libc6 libc6-dev libgomp1 linux-libc-dev manpages-dev
:
:
After this operation, 35.3MB of additional disk space will be used.
Do you want to continue [Y/n]? y
:
:
Setting up binutils (2.20.1-3ubuntu7) ...
Setting up libgomp1 (4.4.3-4ubuntu5) ...
Setting up gcc-4.4 (4.4.3-4ubuntu5) ...
Setting up gcc (4:$4.3-1ubuntu1) ...
Setting up libcdev-bin (2.11.1-0ubuntu7.5) ...
Setting up linux-libc-dev (2.6.32-25.45) ...
Setting up libc6-dev (2.11.1-0ubuntu7.5) ...

```

S-3. Get the version of the Linux kernel.

```
username@computer$ uname -r ↵
```

2.6.32-21-server

S-4. Install the Linux kernel headers for the current Linux kernel. In the command below, the part after “linux-headers-” should match the output of the command above.

Installing MIport 3.0 Driver on Ubuntu 10.04

```
username@computer$ sudo apt-get install linux-headers-2.6.32-21-server ↵
```

```
:  
:  
Setting up linux-headers-2.6.32-25 (2.6.32-25.45)  
Setting up linux-headers-2.6.32-25-server (2.6.32-25.45)
```

S-5. Create a folder for mounting the CD-ROM.

```
username@computer$ sudo mkdir -p /media/cdrom ↵
```

S-6. Check that the folder was created for mounting the CD-ROM.

```
username@computer$ ls /media ↵
```

```
cdrom floppy floppy0
```

If cdrom is not listed above, then the folder was not create correctly.

S-7. Insert the CD-ROM into the CD-ROM drive and mount the CD-ROM.

```
username@computer$ sudo mount -t iso9660 -r /dev/cdrom /media/cdrom ↵
```

S-8. Check that the CD-ROM was mounted.

```
username@computer$ ls /media/cdrom ↵
```

```
autorun.inf linux manuals MIport.ico windows
```

If the files listed above are not shown, then the CD-ROM was not mounted.

S-9. Extract the archive that stores the MIport driver.

```
username@computer$ tar -zxvf /media/cdrom/linux/drivers/2.6/MIport/MIport-3.0.0.tar.gz ↵
```

```
MIport-3.0.0/  
MIport-3.0.0/rc.local  
MIport-3.0.0/bbmknod.sh  
MIport-3.0.0/README  
MIport-3.0.0/README.pdf  
MIport-3.0.0/Makefile  
MIport-3.0.0/MIport.c  
MIport-3.0.0/rc.MIport
```

S-10. Unmount the CD-ROM.

```
username@computer$ sudo umount /media/cdrom ↵
```

S-11. Change the current working directory to where the MIport driver source code is located.

```
username@computer$ cd MIport-3.0.0 ↵
```

S-12. Compile the MIport device driver.

Installing MIport 3.0 Driver on Ubuntu 10.04

```
username@computer$ make ↵
```

```
make -C /lib/modules/2.6.32-24-generic/build M=/home/username/MIport-3.0.0 modules
make[1]: Entering directory `/usr/src/linux-headers-2.6.32-24-generic'
  CC [M]  /home/username/MIport-3.0.0/MIport.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC      /home/username/MIport-3.0.0/MIport.mod.o
  LD [M]  /home/username/MIport-3.0.0/MIport.ko
make[1]: Leaving directory `/usr/src/linux-headers-2.6.32-24-generic'
```

S-13. Determine the present working directory. The output may be different than shown below.

```
username@computer$ pwd ↵
```

```
/home/username/MIport-3.0.0
```

S-14. Install the MIport device driver. The part after “PWD=” of the command below should match the output from the command above.

```
username@computer$ sudo make install PWD=/home/username/MIport-3.0.0 ↵
```

```
make -C /lib/modules/2.6.32-24-generic/build M=/home/username/MIport-3.0.0 modules_install
make[1]: Entering directory `/usr/src/linux-headers-2.6.32-24-generic'
  INSTALL /home/username/MIport-3.0.0/MIport.ko
  DEPMOD  2.6.32-24-generic
make[1]: Leaving directory `/usr/src/linux-headers-2.6.32-24-generic'
cp ./rc.MIport /etc/rc.MIport
chmod 755 /etc/rc.MIport
```

S-15. Start the MIport driver.

```
username@computer$ sudo /etc/rc.MIport ↵
```

```
Searching for MIport driver in: ...
The MIport driver was found in: ...
The MIport driver is loaded.
Making special nodes for MIport.
```

```
Making /dev/ttyM0 /dev/cum0
Making /dev/ttyM1 /dev/cum1
Making /dev/ttyM2 /dev/cum2
Making /dev/ttyM3 /dev/cum3
Making /dev/ttyM4 /dev/cum4
Making /dev/ttyM5 /dev/cum5
Making /dev/ttyM6 /dev/cum6
Making /dev/ttyM7 /dev/cum7
```

S-16. Verify that the device driver is running. If the output is similar to what is shown below, then the MIport driver is running. If the output is blank, then the MIport driver is not running.

```
username@computer$ dmesg | grep MIport ↵
```

```
[ 271.828182] B&B Electronics PCI MIport serial driver Revision: 4.5
[ 314.159265] MIport 0000:07:05:05.0: PCI INT A -> GSI 26 (level, low) -> IRQ 26
```

S-17. Use a serial communications program to verify that the driver works. The serial ports will be “/dev/ttyM0” through “/dev/ttyM7”.

S-18. If the driver works correctly, modify `/etc/rc.local` to automatically load the driver when the operating system starts.

```
username@computer$ sudo vim /etc/rc.local ↵
```

Add the following code to the file (before the "exit 0" statement):

```
if [ -f /etc/rc.MIport ] ; then  
  exec /etc/rc.MIport  
fi
```

An example of the code to add to `/etc/rc.local` is in the MIport driver folder in the file `rc.local`.

S-19. Restart the computer, and then test serial communications again.