

Find the available serial ports

Since Linux only support 4 serial ports (ttyS0, ttyS1, ttyS2, ttyS3) under the default condition. Most likely, ttyS0 & ttyS1 are supported by mother board's built-in serial controllers and ttyS2 & ttyS3 are free for additional I/O card.

Serial ports could be checked by the following commands.

```
setserial /dev/ttyS0 -a          (COM-1)
setserial /dev/ttyS1 -a          (COM-2)
setserial /dev/ttyS2 -a          (COM-3)
setserial /dev/ttyS3 -a          (COM-4)
```

If COM-1 is used by mouse, the response is similar to
/dev/ttyS0 : Device or resource busy

If the COM-1 does not attach any device, the response is similar to
/dev/ttyS0, Line 0, UART: 16550A, Port: 0x3f8, irq: 4
Baud_base: 115200, clos_delay: 50, divisor: 0
closing_wait: 3000, closing_wait2: infinite
Flags: spd_normal skip_test

In case ttyS2 (COM-3) is free, the response for command setserial /dev/ttyS2 -a is shown below.
/dev/ttyS2, Line 2, UART: unknown, Port: 0x3e8, irq: 4
Baud_base: 115200, clos_delay: 50, divisor: 0
closing_wait: 3000, closing_wait2: infinite
Flags: spd_normal skip_test

Find the PCI card resource (I/O port address & IRQ) for the serial ports

Enter the command
more /proc/pci

The response is similar to the following
Bus 0, Device 11, function 0:
Serial controller : Unknown vendor Unknown device (rev 01).
Vendor id=9710, Device id=9820

Medium devsel. Fast back-to-back capable. IRQ 11 (IRQ depends on PCI)

```
Board with Nm9835CV part
I/O at 0xc000 [0xc001]    serial port 1
I/O at 0xc400 [0xc401]    not used
I/O at 0xc800 [0xc801]    not used
I/O at 0xd000 [0xd001]    not used
I/O at 0xd400 [0xd401]    not used
```

I/O at 0xd800 [0xd801] not used

From the `/proc/pci` file, it is possible to find the PCI card's I/O port address and IRQ. Especially, the NetMos parts always shows:

Vendor id=9710, Device id=9820

Configure the parameters for **ttyS2** or **ttyS3**

Enter (if **ttyS2** or **ttyS3** are free):

```
setserial /dev/ttyS2 port 0xc000 UART 16550A
irq 11 Baud_base 115200
setserial /dev/ttyS3 port 0xc008 UART 16550A
irq 11 Baud_base 115200
```

Check the setting for **ttyS2** or **ttyS3**

```
Type setserial /dev/ttyS2 -a :
/dev/ttyS2, Line 2, UART: 16550A, Port: 0xc000, irq: 11
Baud_base: 115200, clos_delay: 50, divisor: 0
closing_wait: 3000, closing_wait2: infinite
Flags: spd_normal skip_test
```

The **ttyS2** or **ttyS3** are ready for application

GPRS settings

To be able to use GPRS, you should configure your dialer to initialize the modem with following init strings (replace 'your.apn.here' with appropriate APN):

```
ATZ
AT&FV1&D2&S0&C1S0=0
AT+CGDCONT=1,"IP","your.apn.here"
```

PPP Internet connection can be established by dialing to the phone number '*99#'.

Example

To connect to the internet over GPRS using wvdial dialer you can use the following configuration file (/etc/wvdial.conf):

Launch terminal in Linux, you must enter as root then launch:

```
wvdialconf /etc/wvdial.conf
```

It will create wvdial.conf file in /etc/ folder, when edit this file and enter such parameters

```
[Dialer Defaults]
```

```
Modem = /dev/ttyS2
```

```
;(Modem= /dev/ttyS2)
```

```
;(Modem= /dev/ttyS3)
```

```
Baud = 115200
```

```
Init1 = ATZ
```

```
Init2 = AT&FV1&D2&S0&C1S0=0
```

```
Username= your login
```

```
Password= your password
```

```
[Dialer gprs]
```

```
Phone = *99#
```

```
Init3= AT+CGDCONT=1,"IP","your.apn.here"
```

Note: You need to replace 'your login', 'your password' and 'your.apn.here' with values appropriate for your GSM operator.

After that, enter in terminal such command:

```
wvdial --config /etc/wvdial.conf gprs
```

Modem will connect to the internet, to stop connection press in the terminal CTRL+C.