# Digi Connect ME 9210
# Linux: serial port 2 for JTAG modules

# Document History

| Date | Version | Change Description |
|---|---|---|
| 08/05/2009 | V1.0 | Initial entry/outline |
| | | |
| | | |

# Table of Contents

# 1 Problem Description

- The Digi Connect ME 9210 module with JTAG has only one serial line (port A of the NS9210 processer) connected.
- If you want to activate the second serial line on the development board "P2 Serial Port 2", follow the instructions below. Click here for source files.
- Note: a kernel compiled for the modules with JTAG connectors, using second serial line on board, might not startup on modules without JTAG connector.

# 2 Requirements

To try the example in this document you need:

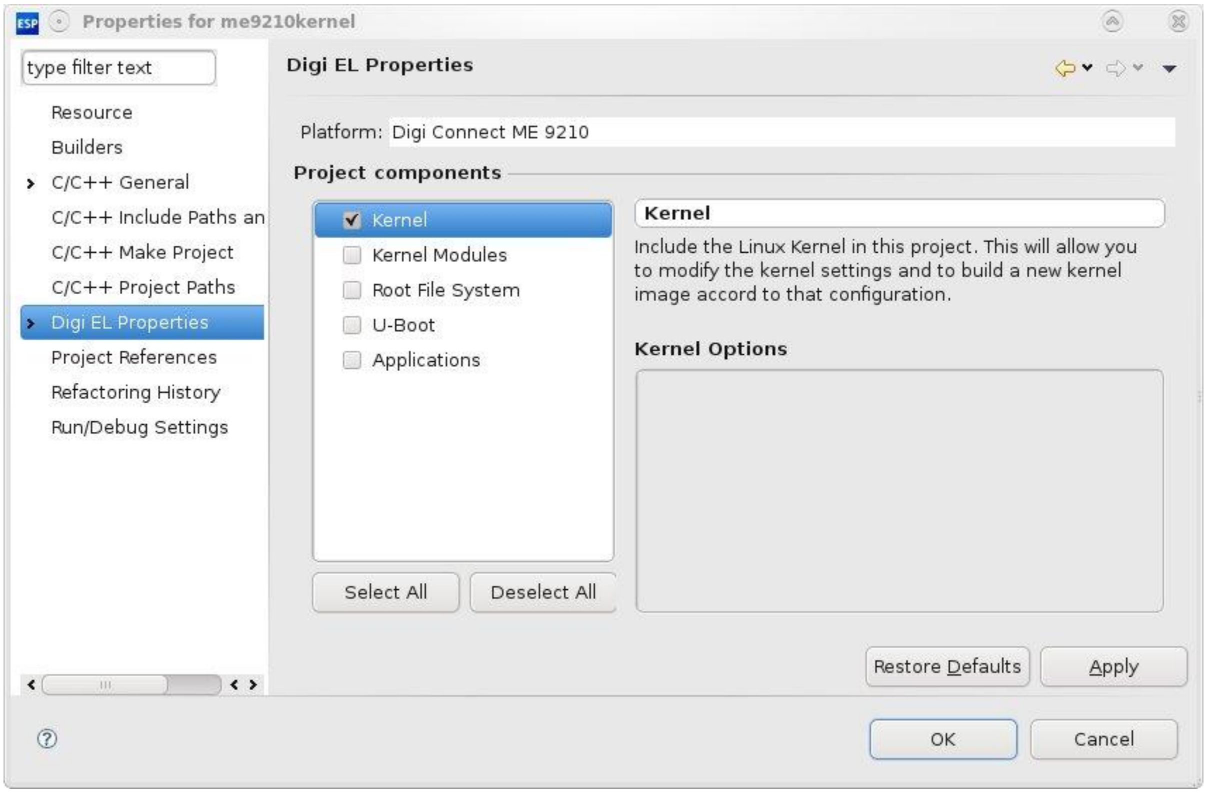
- Digi Connect ME 9210 module with Linux (DC-ME-Y402-LX).
- Digi Embedded Linux (DEL) 5.0 or above development environment.
  For DEL 5.0 you need to have installed the latest patches via the package manager.
  With DEL 5.1 it should work out of the box.
- The Digi Connect ME development board.

  You can get everything together in a Digi Connect ME 9210 Linux JumpStart Kit: DC-ME-9210-LX

# 3 Software Setup

- Install Digi Embedded Linux (DEL) 5.0 or higher, apply latest patches with the Package Manager.

- Create a new Digi EL Kernel/Rootfs/U-Boot Project for Platform Digi Connect ME 9210, but select only Kernel as project components:



Digi EL Properties

Platform: Digi Connect ME 9210

Project components

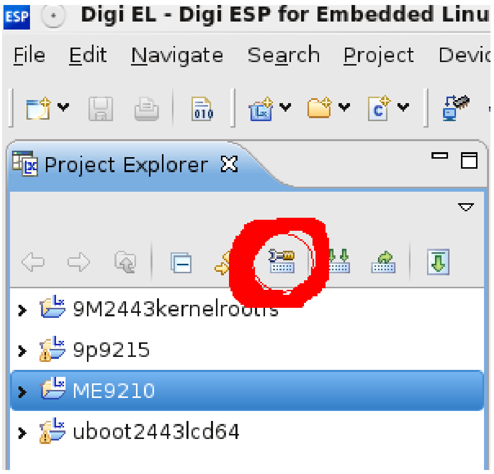
- ☑ Kernel
- ☐ Kernel Modules
- ☐ Root File System
- ☐ U-Boot
- ☐ Applications

Kernel

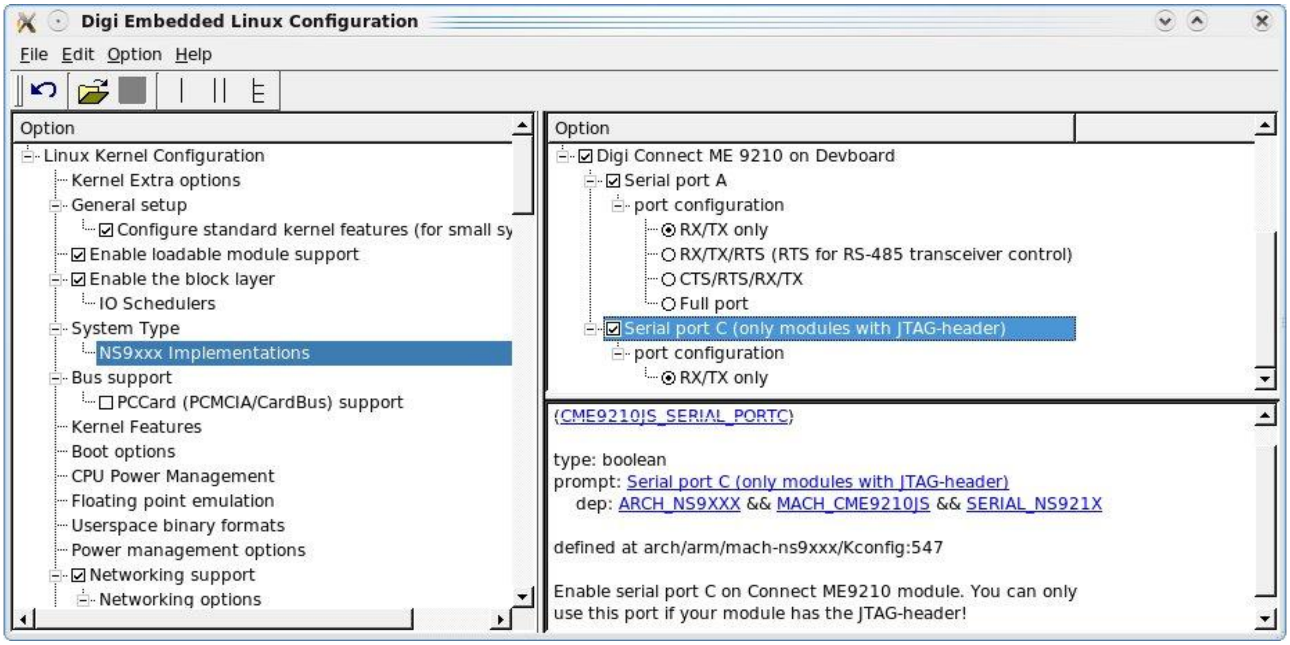
Include the Linux Kernel in this project. This will allow you to modify the kernel settings and to build a new kernel image accord to that configuration.

Kernel Options

- Configure the project (right click on the project an select configure)



- Select Linux Kernel Configuration, Type->NS9xxx Implementations->Digi Connect ME 9210 on Devboard->Serial Port C (CONFIG_CME9210JS_SERIAL_PORTC):



```
Linux Kernel Configuration
    Kernel Extra options
    General setup
        ☑ Configure standard kernel features (for small sy
    ☑ Enable loadable module support
    ☑ Enable the block layer
        IO Schedulers
    System Type
        NS9xxx Implementations
    Bus support
        ☐ PCCard (PCMCIA/CardBus) support
    Kernel Features
    Boot options
    CPU Power Management
    Floating point emulation
    Userspace binary formats
    Power management options
    ☑ Networking support
        Networking options
```

```
☑ Digi Connect ME 9210 on Devboard
    ☑ Serial port A
        port configuration
            ● RX/TX only
            ○ RX/TX/RTS (RTS for RS-485 transceiver control)
            ○ CTS/RTS/RX/TX
            ○ Full port
    ☑ Serial port C (only modules with JTAG-header)
        port configuration
            ● RX/TX only
```

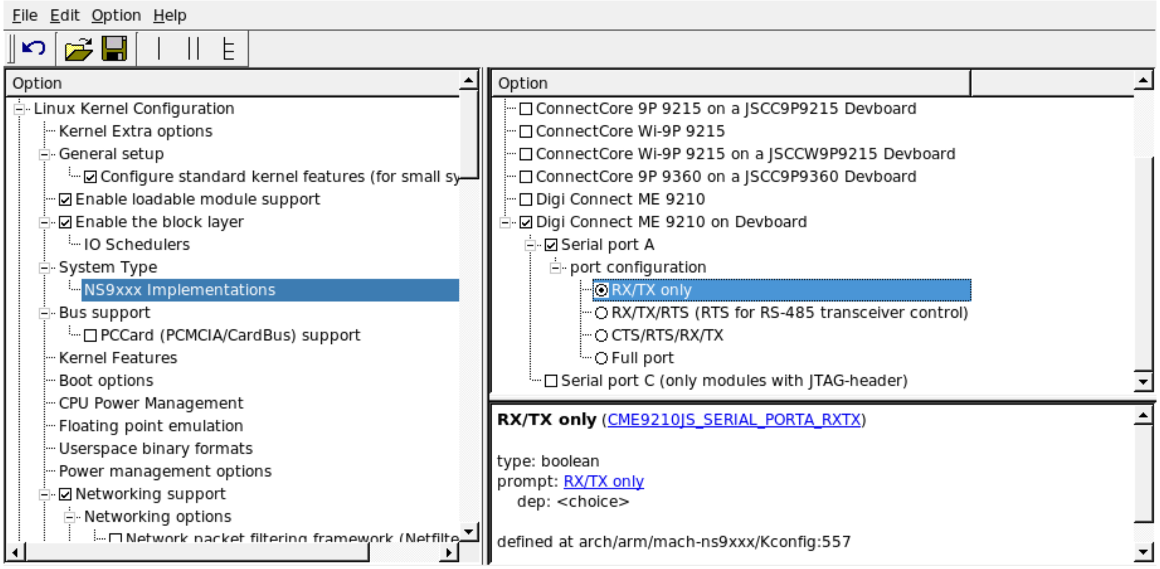
(CME9210JS_SERIAL_PORTC)

type: boolean
prompt: Serial port C (only modules with JTAG-header)
dep: ARCH_NS9XXX && MACH_CME9210JS && SERIAL_NS921X

defined at arch/arm/mach-ns9xxx/Kconfig:547

Enable serial port C on Connect ME9210 module. You can only use this port if your module has the JTAG-header!

- Select Digi Connect ME 9210 on Devboard in the Linux Kernel Configuration System Type. Enable Serial port A with RX/TX only (CONFIG_CME9210JS_SERIAL_PORTA_RXTX):



```
Linux Kernel Configuration
    Kernel Extra options
    General setup
        ☑ Configure standard kernel features (for small sy
    ☑ Enable loadable module support
    ☑ Enable the block layer
        IO Schedulers
    System Type
        NS9xxx Implementations
    Bus support
        ☐ PCCard (PCMCIA/CardBus) support
    Kernel Features
    Boot options
    CPU Power Management
    Floating point emulation
    Userspace binary formats
    Power management options
    ☑ Networking support
        Networking options
            ☐ Network packet filtering framework (Netfilte
```

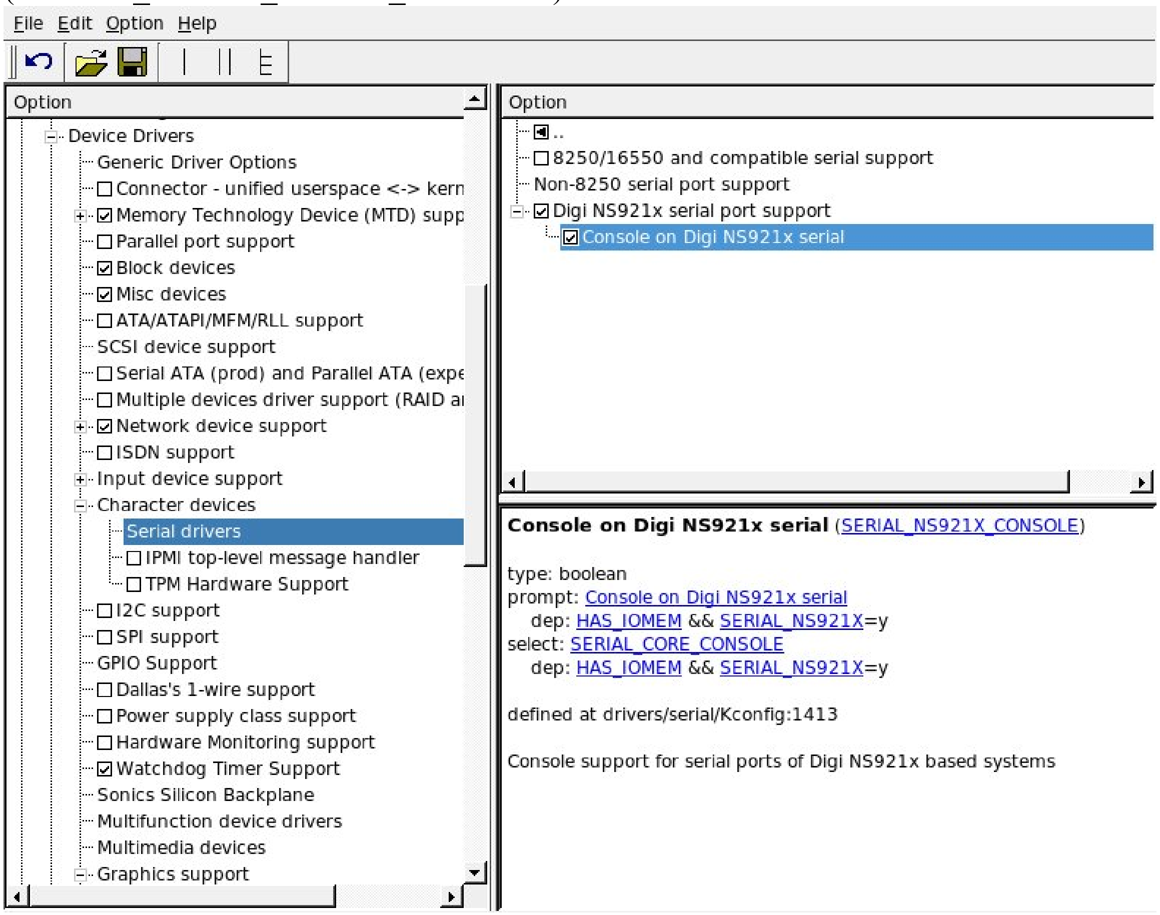
```
☐ ConnectCore 9P 9215 on a JSCC9P9215 Devboard
☐ ConnectCore Wi-9P 9215
☐ ConnectCore Wi-9P 9215 on a JSCCW9P9215 Devboard
☐ ConnectCore 9P 9360 on a JSCC9P9360 Devboard
☐ Digi Connect ME 9210
☑ Digi Connect ME 9210 on Devboard
    ☑ Serial port A
        port configuration
            ● RX/TX only
            ○ RX/TX/RTS (RTS for RS-485 transceiver control)
            ○ CTS/RTS/RX/TX
            ○ Full port
    ☐ Serial port C (only modules with JTAG-header)
```

**RX/TX only** (CME9210JS_SERIAL_PORTA_RXTX)

type: boolean
prompt: RX/TX only
dep: <choice>

defined at arch/arm/mach-ns9xxx/Kconfig:557

- Select Device Drivers, Character devices, Serial drivers, Digi NS921x serial port support, Console on Digi NS921x serial (CONFIG_SERIAL_NS921X_CONSOLE):



Device Drivers
- Generic Driver Options
- □ Connector - unified userspace <-> kern
- ☑ Memory Technology Device (MTD) supp
- □ Parallel port support
- ☑ Block devices
- ☑ Misc devices
- □ ATA/ATAPI/MFM/RLL support
- SCSI device support
- □ Serial ATA (prod) and Parallel ATA (expe
- □ Multiple devices driver support (RAID ar
- ☑ Network device support
- □ ISDN support
- Input device support
- Character devices
  - Serial drivers
  - □ IPMI top-level message handler
  - □ TPM Hardware Support
- □ I2C support
- □ SPI support
- GPIO Support
- □ Dallas's 1-wire support
- □ Power supply class support
- □ Hardware Monitoring support
- ☑ Watchdog Timer Support
- Sonics Silicon Backplane
- Multifunction device drivers
- Multimedia devices
- Graphics support

..
- □ 8250/16550 and compatible serial support
- Non-8250 serial port support
- ☑ Digi NS921x serial port support
  - ☑ Console on Digi NS921x serial

**Console on Digi NS921x serial** (SERIAL_NS921X_CONSOLE)

```
type: boolean
prompt: Console on Digi NS921x serial
   dep: HAS_IOMEM && SERIAL_NS921X=y
select: SERIAL_CORE_CONSOLE
   dep: HAS_IOMEM && SERIAL_NS921X=y

defined at drivers/serial/Kconfig:1413

Console support for serial ports of Digi NS921x based systems
```

- Build and install the project

# 4 Hardware Setup

- Turn off the development board.
- Connect the power cable.
- Plug Connect ME 9210 module to the development board.
- Connect Ethernet cable to Connect ME 9210 to your development PC (for updating firmware).
- Connect a serial null modem cable (pins 2 and 3 crossed) to your host computer (e.g. COMA is the CONSOLE). Plug the cable into Serial Port A of the Digi development board.
- Connect a serial null modem cable (pins 2 and 3 crossed) to your host computer (e.g. COMB is the serial Port 2). Plug the cable into Serial Port B of the Digi development board.
- Set the devboard SW3 dip switches to any hardware handshake you have enabled in the kernel config

# 5 Testing

Run the new build kernel on the Digi Connect ME 9210 module (e.g. update the image in flash build in section 3).

Start a terminal program with 38400 8N1 on your development host on COM1 (console) and COM2 (serial port 2). Boot Linux on the Digi Connect ME 9210 module. On the serial console (COM1), configure the serial interface:

```
# stty -F /dev/ttyNS2 38400
```

Send some test string from the ConnectME 9210 serial port A (console) to the development host (COM2 serial port B):

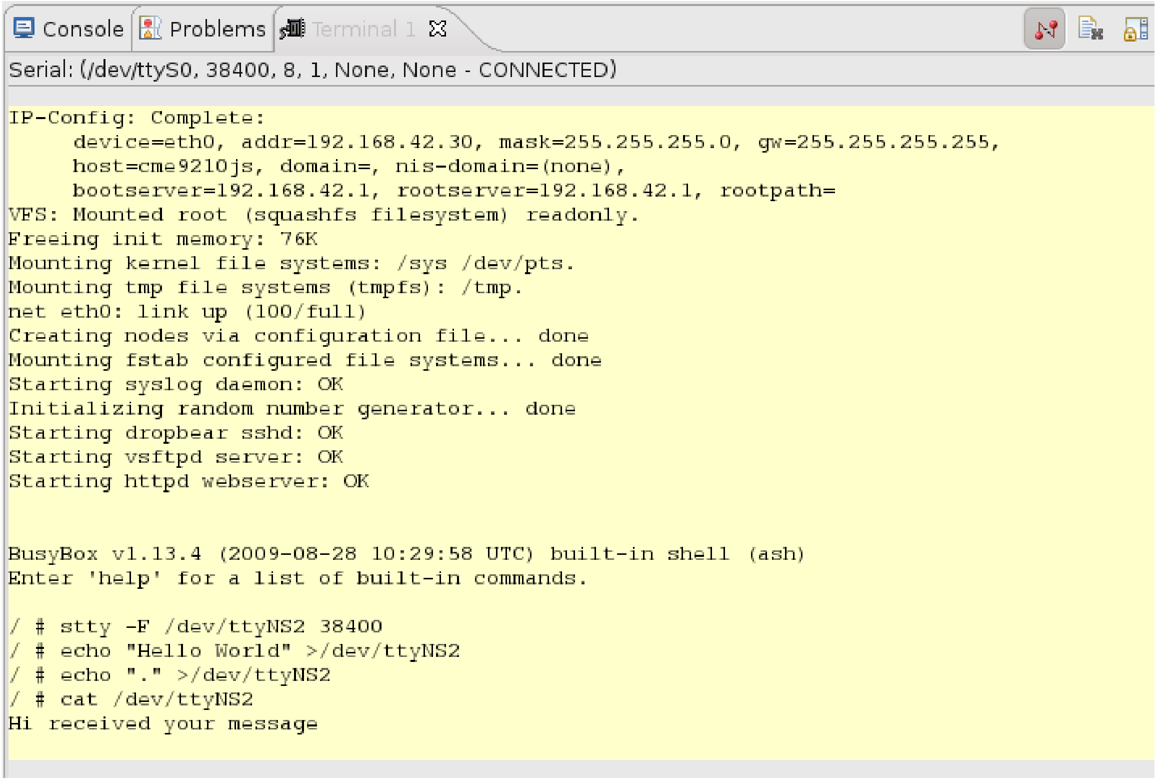
```
# echo "Hello World" >/dev/ttyNS2
```

Check if the test string is received by your terminal program on COM2.

Send some test string from the development host via the serial port 2 to the Digi Connect ME 9210. First start a program on the Connect ME 9210 which is able to receive the chars, e.g.:

```
# cat /dev/ttyNS2
```

Enter some string on the COM2 serial port 2 terminal program of your development host (press Enter, if everything is configured in line mode which is the default) and check if it is received on the Connect ME 9210 console.

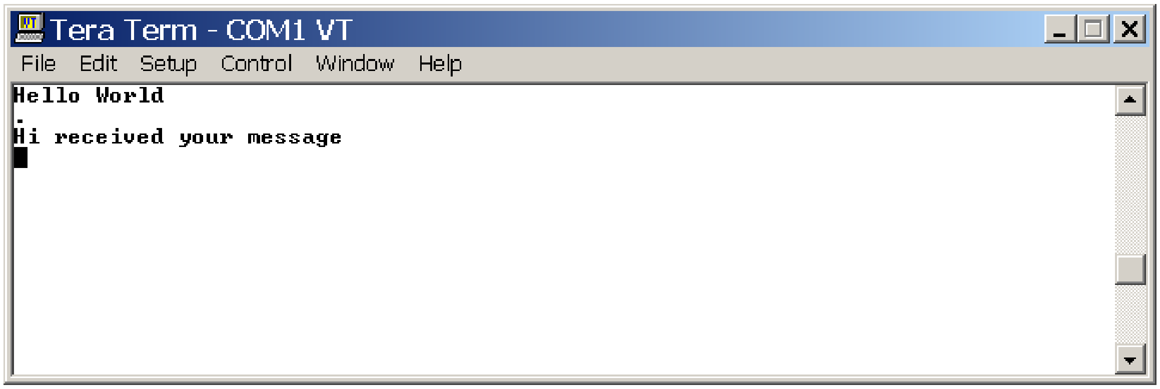


Serial: (/dev/ttyS0, 38400, 8, 1, None, None - CONNECTED)

```
IP-Config: Complete:
     device=eth0, addr=192.168.42.30, mask=255.255.255.0, gw=255.255.255.255,
     host=cme9210js, domain=, nis-domain=(none),
     bootserver=192.168.42.1, rootserver=192.168.42.1, rootpath=
VFS: Mounted root (squashfs filesystem) readonly.
Freeing init memory: 76K
Mounting kernel file systems: /sys /dev/pts.
Mounting tmp file systems (tmpfs): /tmp.
net eth0: link up (100/full)
Creating nodes via configuration file... done
Mounting fstab configured file systems... done
Starting syslog daemon: OK
Initializing random number generator... done
Starting dropbear sshd: OK
Starting vsftpd server: OK
Starting httpd webserver: OK


BusyBox v1.13.4 (2009-08-28 10:29:58 UTC) built-in shell (ash)
Enter 'help' for a list of built-in commands.

/ # stty -F /dev/ttyNS2 38400
/ # echo "Hello World" >/dev/ttyNS2
/ # echo "." >/dev/ttyNS2
/ # cat /dev/ttyNS2
Hi received your message
```



```
Hello World
.
Hi received your message
```