




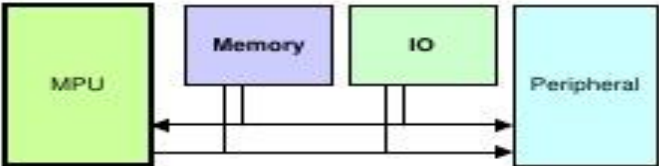
EMBEDDED SYSTEMS TRAINING

Khóa học : “Basic Embedded Linux”

❖ Email : training.laptrinhnhung@gmail.com

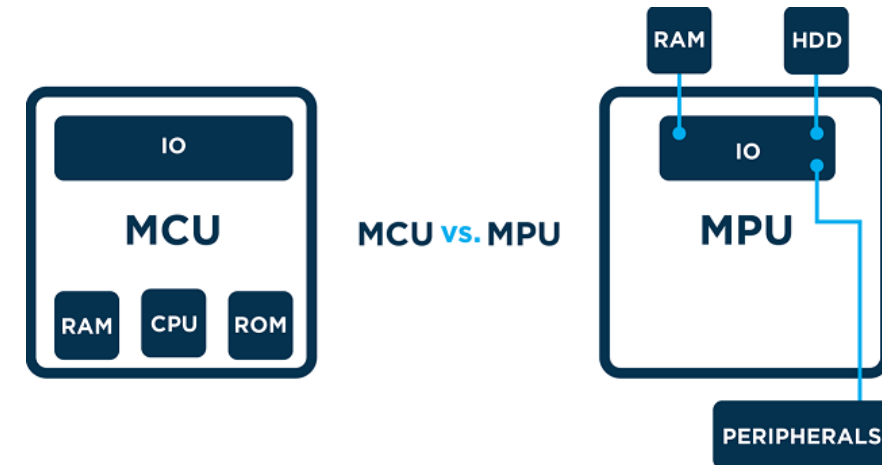
❖ Website : <http://hethongnhung.com> – <http://laptrinhnhung.com>

MCU and MPU

Classification	MCU	MPU
Chip	Embedded A CPU core, memory, peripherals, IO ... into a single chip.	Contain only the main processor (CPU core)
Block Diagram of a system		
General Application Area	<ul style="list-style-type: none"> - Self contained to complete a task. - Targeted for small, compact, and low cost system 	<ul style="list-style-type: none"> -Need external memory, peripheral to accomplish a task - Targeted for complex, high performance and expandable system
Processor (CPU) Core	4, 8, 16 bit	32 bit or above
Examples	8051,pic16f887a,M16, H8, SH1/2	SH3/4,8085
Application	Washing machine, car side mirror, air con	Handphone, PDA

difference between MPU and MCU

Attributes	Microcontrollers	Microprocessors
Application	Are application specific and are designed to perform certain limited tasks.	Have generic application and are capable of executing big and complicated tasks.
One Solution	Have inbuilt processor, RAM, ROM and I/O Ports. Like a small stand-alone computer in a single Integrate Chip.	Generally don't have inbuilt RAM, ROM and I/O ports. The pins are used to interface with external RAM, ROM and ports.
Performance	Limited performance.	Very high performance.
Speed	Generally operate at speeds from 8 MHz – 200 MHz.	Generally operate at speeds above 1 GHz.
Power Consumption	Are embedded inside other devices, so are designed to consume less power.	Consume relative more power. As performance is the given higher weightage over power.
Cost	Affordable and cheap. Can get started with a minimum external circuit.	Very expensive and requires other peripherals to work along.



Cortex[®]-M processors

MCU + DSP

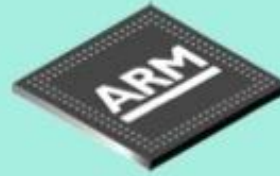


RTOS

Smallest footprint / lowest power



Cortex[®]-R processors



Highest performance / real-time



Cortex[®]-A processors

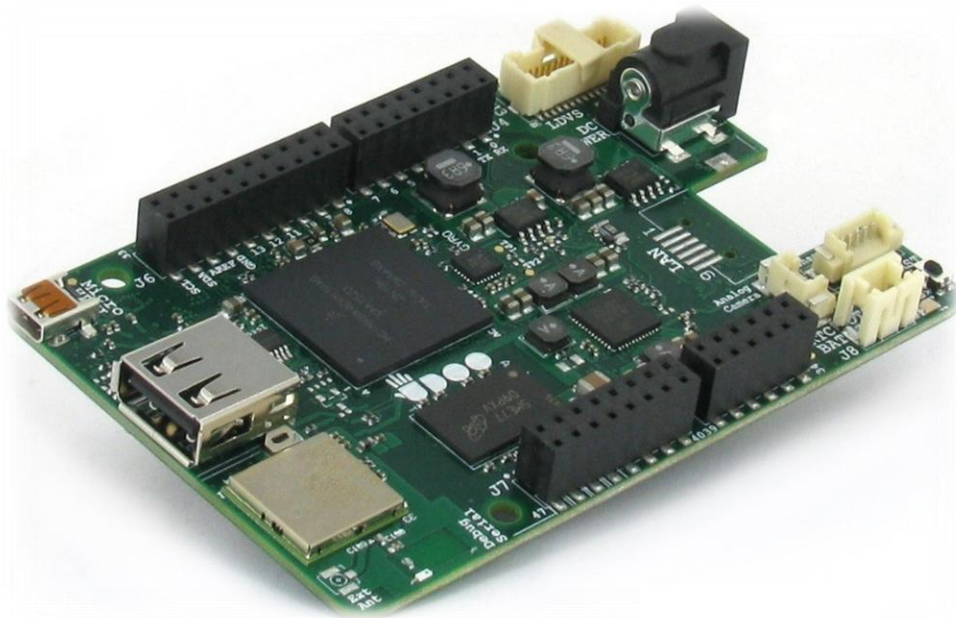


Rich OS

Highest performance



Competition summary – Standard competition set (cont.)



(*) Display only for graphics-related projects



[http://www.udoo.org/docs-neo/Hardware Reference/Overview.htm](http://www.udoo.org/docs-neo/Hardware_Reference/Overview.htm)

“UDOO boards have no internal storage or built-in boot code, OS and storage are on microSD so you need to plug a pre-loaded microSD to boot.”

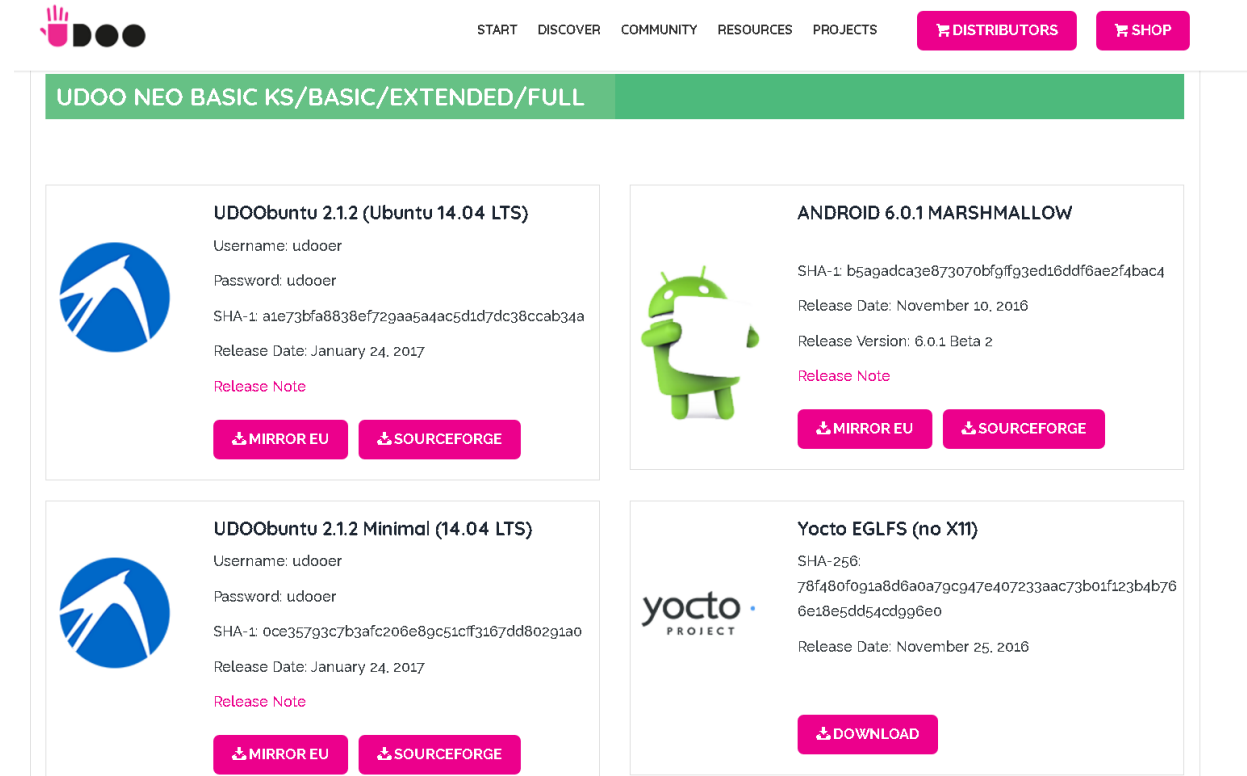
The steps to start the board are:

- 1 - Download UDOO NEO's official Operating system
- 2 - Prepare the Micro SD Card
- 3 - Insert the Micro SD Card
- 4 - Power up via DC connector or micro USB if no display is connected to board

Download the official image from Site Download section to your PC and unzip it
<http://www.udoo.org/downloads/>

You can choose from :

- UD00buntu minimal (console mode)
- UD00buntu full (recommended)
- Android
- Yocto

A screenshot of the UDOO NEO website's download section. The page has a green header with the UDOO logo and navigation links: START, DISCOVER, COMMUNITY, RESOURCES, PROJECTS, DISTRIBUTORS, and SHOP. Below the header is a green bar with the text "UD00 NEO BASIC KS/BASIC/EXTENDED/FULL". The main content area is divided into four cards. The first card is for "UD00buntu 2.1.2 (Ubuntu 14.04 LTS)" with a blue bird logo, listing username, password, SHA-1, release date, and a release note link, with buttons for "MIRROR EU" and "SOURCEFORGE". The second card is for "UD00buntu 2.1.2 Minimal (14.04 LTS)" with the same blue bird logo and details, also with "MIRROR EU" and "SOURCEFORGE" buttons. The third card is for "ANDROID 6.0.1 MARSHMALLOW" with a green Android robot logo, listing SHA-1, release date, release version, and a release note link, with "MIRROR EU" and "SOURCEFORGE" buttons. The fourth card is for "Yocto EGLFS (no X11)" with the Yocto Project logo, listing SHA-1 and release date, with a "DOWNLOAD" button.

UD00 NEO BASIC KS/BASIC/EXTENDED/FULL

UD00buntu 2.1.2 (Ubuntu 14.04 LTS)
Username: udooer
Password: udooer
SHA-1: a1e73bfa8838ef729aa5a4ac5d1d7dc38ccab34a
Release Date: January 24, 2017
[Release Note](#)
[MIRROR EU](#) [SOURCEFORGE](#)

UD00buntu 2.1.2 Minimal (14.04 LTS)
Username: udooer
Password: udooer
SHA-1: 0ce35793c7b3afc206e89c51cff3167dd80291a0
Release Date: January 24, 2017
[Release Note](#)
[MIRROR EU](#) [SOURCEFORGE](#)

ANDROID 6.0.1 MARSHMALLOW
SHA-1: b5agadca3e873070bf9ff93ed16ddf6ae2f4bac4
Release Date: November 10, 2016
Release Version: 6.0.1 Beta 2
[Release Note](#)
[MIRROR EU](#) [SOURCEFORGE](#)

Yocto EGLFS (no X11)
SHA-256: 78f480f091a8d6a0a79c947e40723aac73b01f123b4b766e18e5dd54cd996e0
Release Date: November 25, 2016
[DOWNLOAD](#)

Create a bootable uSD card with the downloaded image.

WindowsOS, MacOS and **LinuxOS** procedures are described in documentation section on official site.

http://www.udoo.org/docs/Getting_Started/Create_A_Bootable_MicroSD_card_for_UDOO_QUAD-DUAL.html

For WindowsOS the application

Win32DiskImager.exe

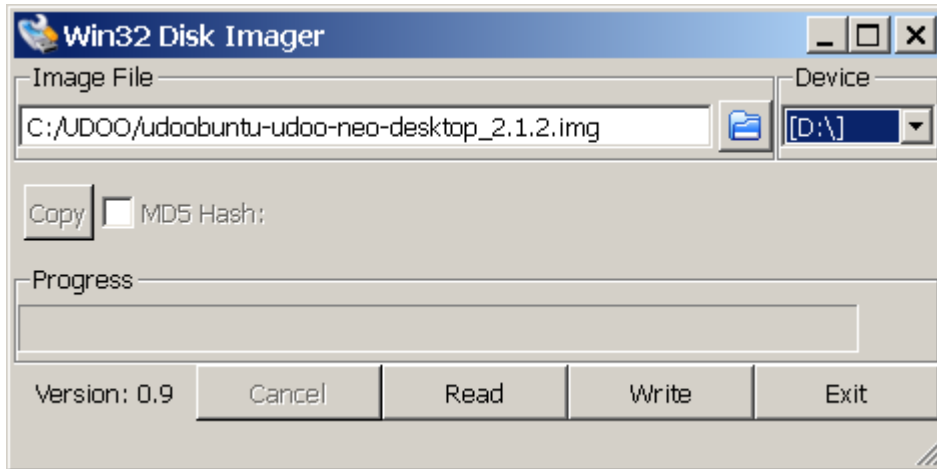
can be used to write the image

For LinuxOS the command

dd

can be used to write the image.

WindowsOS



LinuxOS

```
udoouer@udooneo: ~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        7.4G  2.4G  4.6G  35% /
devtmpfs         338M  4.0K  338M   1% /dev
none            4.0K   0    4.0K   0% /sys/fs/cgroup
none           100M  208K  100M   1% /run
none            5.0M   0    5.0M   0% /run/lock
none           498M   0   498M   0% /run/shm
none           100M  16K   100M   1% /run/user
/dev/mmcblk0p1   32M   25M   7.9M  76% /boot
tmpfs            20M   0    20M   0% /gpio
tmpfs            20M   0    20M   0% /sensors

udoouer@udooneo: ~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        7.4G  2.4G  4.6G  35% /
devtmpfs         338M  4.0K  338M   1% /dev
none            4.0K   0    4.0K   0% /sys/fs/cgroup
none           100M  316K  100M   1% /run
none            5.0M   0    5.0M   0% /run/lock
none           498M   0   498M   0% /run/shm
none           100M  16K   100M   1% /run/user
/dev/mmcblk0p1   32M   25M   7.9M  76% /boot
tmpfs            20M   0    20M   0% /gpio
tmpfs            20M   0    20M   0% /sensors
/dev/sde1        9.0M  2.9M  6.2M  32% /media/udoouer/SD_KERNEL
/dev/sde2        3.6G  334M  3.1G  10% /media/udoouer/SD_RFS
```

```
udoouer@udooneo: ~$ umount /dev/sde1
udoouer@udooneo: ~$ umount /dev/sde2
umount: /dev/sde2 is not mounted (according to mtab)
udoouer@udooneo: ~$ sudo dd bs=1M if=/home/udoouer/udoobuntu-udoo-neo-minimal 2.1.2.img of=/dev/sde
810+0 records in
810+0 records out
849346560 bytes (849 MB) copied, 200.895 s, 4.2 MB/s
udoouer@udooneo: ~$
```

Starting up the board - Insert the Micro SD Card

uSD card should be inserted when board is NOT powered (USB or DC power).



There are several ways you may choose to work/connect with your board. They are described in UDOO NEO documentation site on Getting Started section.

You may choose to use it as

- a desktop (monitor/lvds)

<http://www.udoo.org/docs-neo/Getting Started/Use as a Lightweight Desktop PC.html>

Recommendation for this case is to power up the board by DC adapter.

- an headless device (no display attached to board), case in which you may choose to connect by USB as
 - an SSH connection
 - an serial terminal connection
 - an VNC connection

<http://www.udoo.org/docs-neo/Getting Started/Use as a headless IoT Device.html>

You may power up the board by **USB only** if the LVDS is not attached to the board.

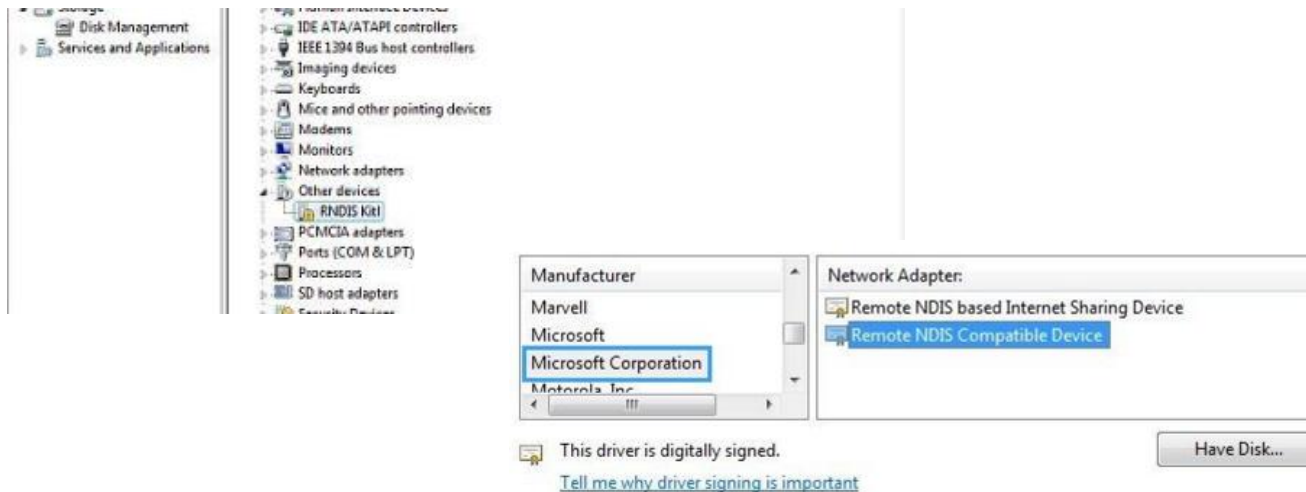
You can access the UDOO board by a remote connection and not using LVDS or monitor.

http://www.udoo.org/docs-neo/Basic_Setup/Usb_Direct_Connection.html

The USB connection provides ETH connection over **RNDIS** in Windows and over **usbnet** in Linux.

https://en.wikipedia.org/wiki/Ethernet_over_USB

WindowsOS



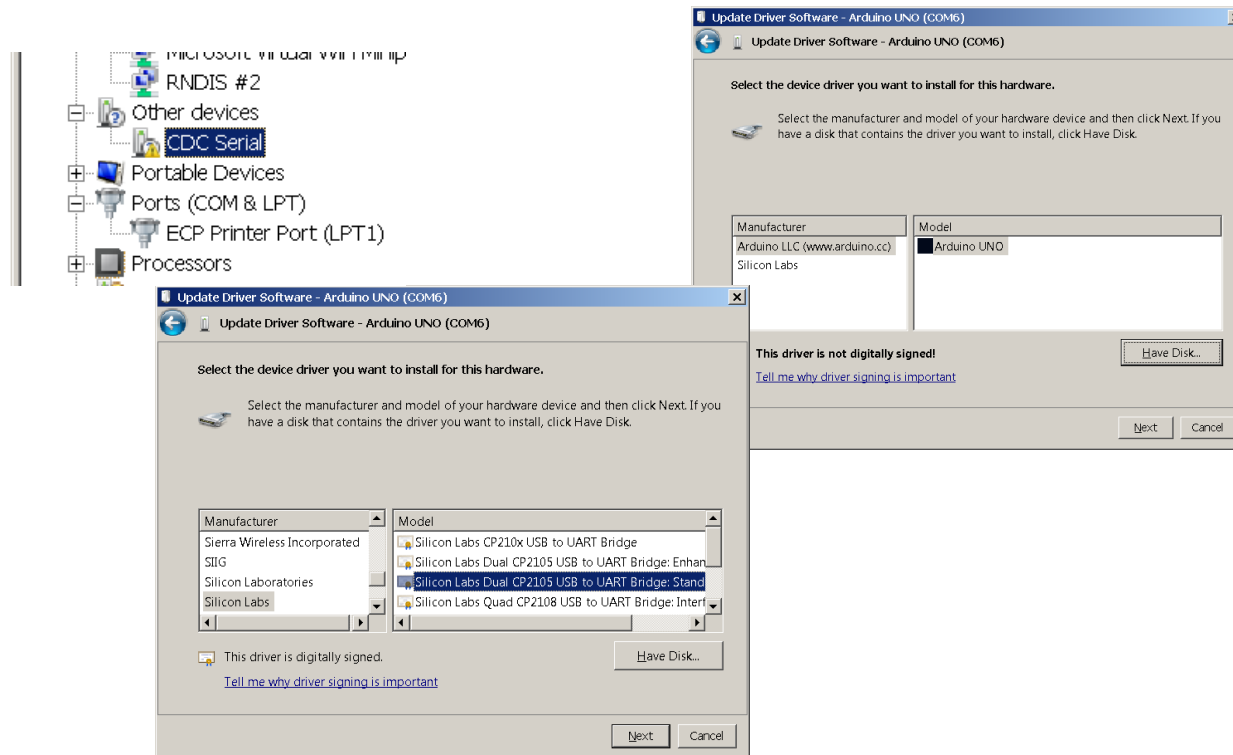
LinuxOS

Linux doesn't need a specific driver installation to make USB connection work properly. Everything should work out-of-the-box

- For virtual serial over USB you will need to install a driver in WindowsOS:

<http://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>

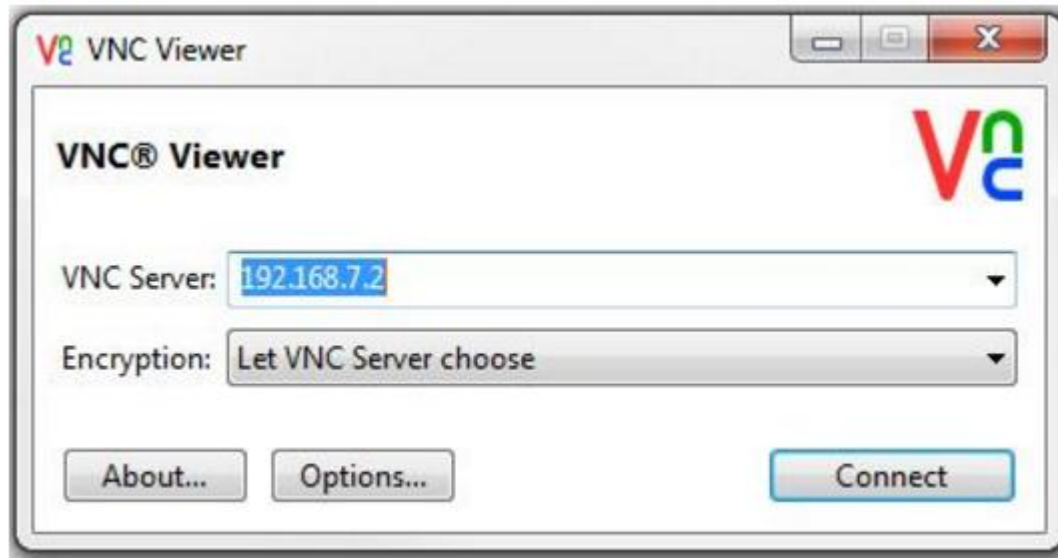
In LinuxOS you will find it as *cpc210x converter*.



```
mi@ubuntu: ~
File Edit View Terminal Help
[ 4778.059189] end request: I/O error, dev fd0, sector 0
[ 5830.351536] usb 2-2.1: new full speed USB device using uhci_hcd and address 4
[ 5830.633003] usb 2-2.1: configuration #1 chosen from 1 choice
[ 5830.669648] usbcore: registered new interface driver usbserial
[ 5830.669876] USB Serial support registered for generic
[ 5830.669937] usbcore: registered new interface driver usbserial_generic
[ 5830.669939] usbserial: USB Serial Driver core
[ 5830.672413] USB Serial support registered for cp210x
[ 5830.672469] cp210x 2-2.1:1.0: cp210x converter detected
[ 5830.908495] usb 2-2.1: reset full speed USB device using uhci_hcd and address
4
[ 5831.097456] usb 2-2.1: cp210x converter now attached to ttyUSB0
[ 5831.097475] usbcore: registered new interface driver cp210x
[ 5831.097477] cp210x: v0.09:Silicon Labs CP210x RS232 serial adaptor driver
[ 5831.100385] usb 2-2.1: USB disconnect, address 4
[ 5831.100600] cp210x ttyUSB0: cp210x converter now disconnected from ttyUSB0
[ 5831.100614] cp210x 2-2.1:1.0: device disconnected
[ 5886.825011] usb 2-2.1: new full speed USB device using uhci_hcd and address 5
[ 5887.103602] usb 2-2.1: configuration #1 chosen from 1 choice
[ 5887.108625] cp210x 2-2.1:1.0: cp210x converter detected
[ 5887.346136] usb 2-2.1: reset full speed USB device using uhci_hcd and address
5
[ 5887.684335] usb 2-2.1: cp210x converter now attached to ttyUSB0
mi@ubuntu:~$
```

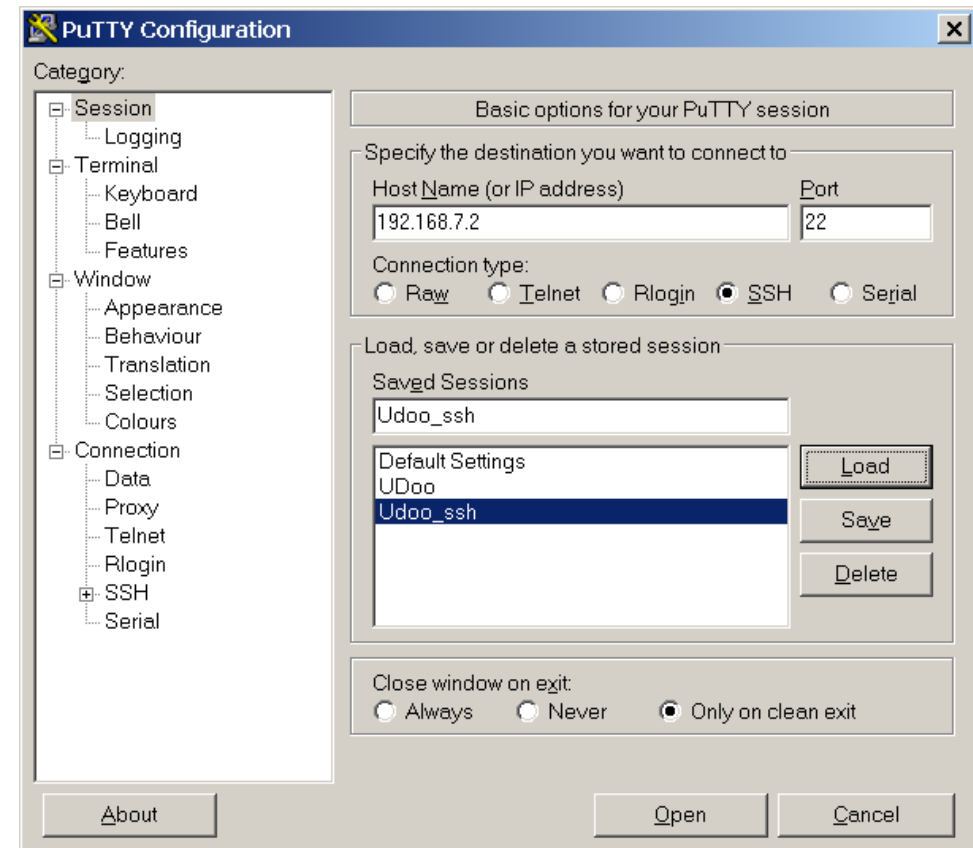

VNC connection

- Use VNC viewer to connect to remote UDOO
<https://www.realvnc.com/download/viewer/>



SSH connection

Use putty.exe to connect over SSH to remote UDOO



Serial connection

This is provided as a virtual serial over the USB connection.

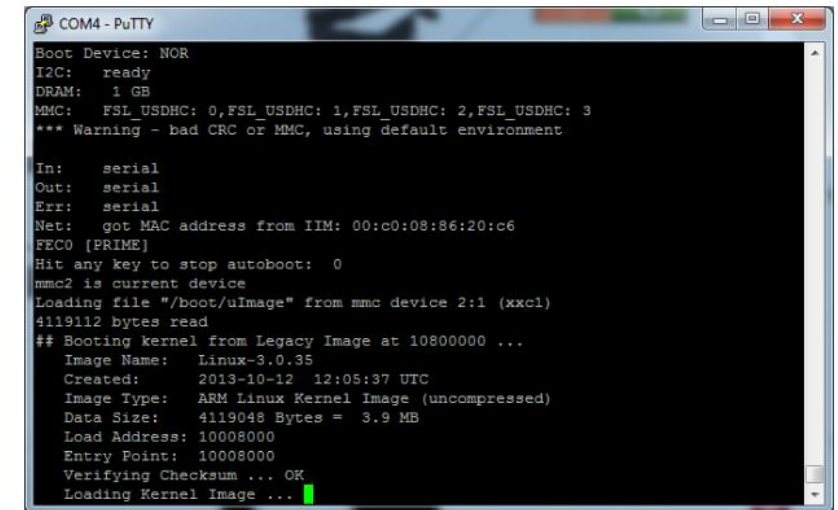
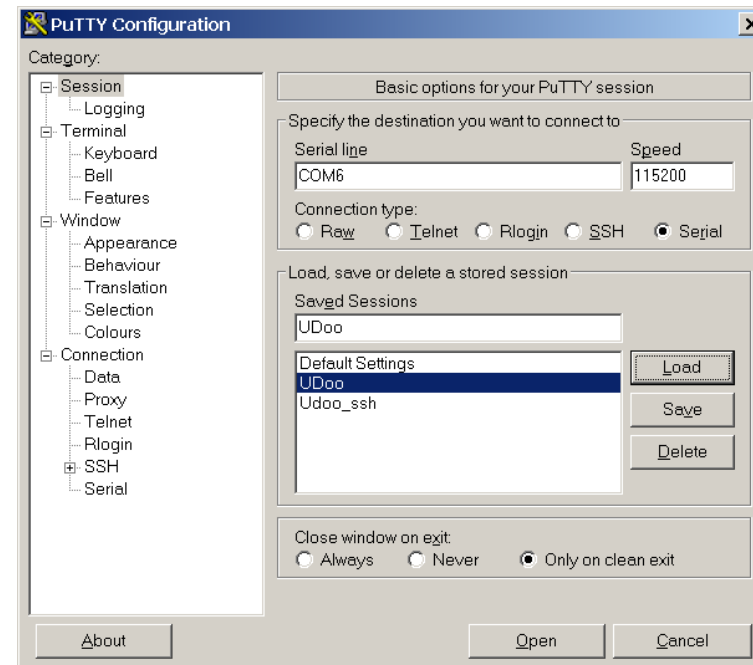
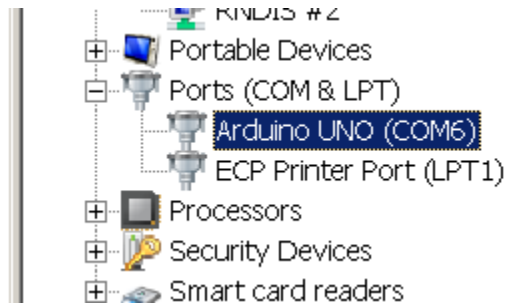
It is available only after UDOO boot process

Use putty.exe to connect using serial to UDOO

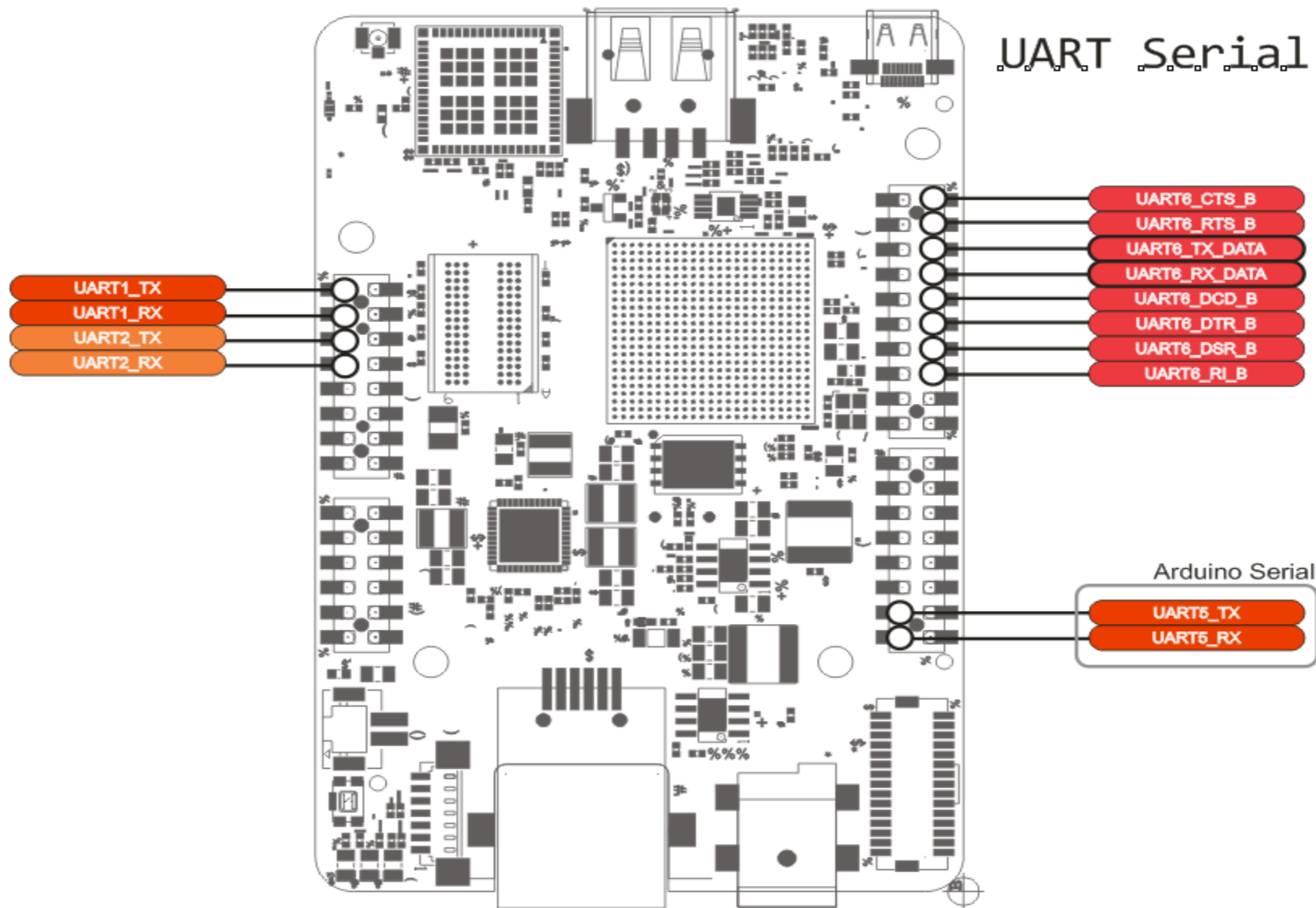
Debug serial connection

This is available immediately the UDOO board is powered up.

It provide view of the booting process.
Use putty.exe to connect using serial to UDOO



Headless UD00 – Reference UART1 as debug console



When the board is powered on, the CPU executes code in its internal ROM, loading the first sectors of the SD card. In this way the [U-Boot boot-loader](#) is loaded and executed.

The boot-loader

This thin layer of software takes care of initialize some registers, devices (like the PMIC) and RAM time settings. It is composed of two stages, the first is called SPL (secondary program loader) which [initializes several things](#):

- [arch_cpu_init\(\)](#) initializes some registers, the watchdog, the DMA, etc;
- [ccgr_init](#) initializes CCGR registers in the CCM (Clock Controller Module);
- [board_early_init_f](#) initializes the M4 core and the pads of the UART1;
- [timer_init](#) initializes CPU timers and clock sources;
- [preloader_console_init](#) initializes serial port communications and prints the message "*U-Boot SPL 2015.04-00267-gd781468 (Dec 16 2015 - 14:44:56)*";
- [spl_dram_init](#) sets board-specific DRAM configuration (UDOO Neo Basic has 512MB of RAM and different timings);
- [memset](#) zeros BSS memory;
- [board_init_r](#) continues the boot, loading the second stage of the boot-loader.

In the [second stage](#), more devices and registers are initialized. I2C buses, LVDS, Ethernet, Wireless and motions sensors pads are initialized. The PFUZE3000 power regulator is [setup](#) and MMC is [initialized](#) so files can be read from it.

Linux kernel boot

The last step is to load the Linux kernel zImage and the device tree file, both from the /boot partition:

```
reading /zImage
4376112 bytes read in 232 ms (18 MiB/s)
Booting from mmc ...
reading dts-overlay/imx6sx-udoo-neo-full-hdmi-m4.dtb
45210 bytes read in 35 ms (1.2 MiB/s)
Kernel image @ 0x80800000 [ 0x0000000 - 0x42c630 ]
## Flattened Device Tree blob at 83000000
   Booting using the fdt blob at 0x83000000
   Using Device Tree in place at 83000000, end 8300e099
Switched to ldo_bypass mode!

Starting kernel ...

[ 0.000000] Booting Linux on physical CPU 0x0
[ 0.000000] Linux version 3.14.56-udooneo-01989-.....
```




Questions?