Subpart - STM32MP15 - Starter Package

Contents

1 Article purpose

This article gathers the information that applies to any STM32MP15 board.

> ⚠️ **To start efficiently the board, it's recommended to go through the Starter Package article relative to your board: Category:Starter Package**

That's the reason why the steps 2, 3 and 6 referenced in this article depend on the considered board:

- **Step 2** is either STM32MP15 Evaluation boards - Starter Package#step2 or STM32MP15 Discovery kits - Starter Package#step2
- **Step 3** is either STM32MP15 Evaluation boards - Starter Package#step3 or STM32MP15 Discovery kits - Starter Package#step3
- **Step 6** is either STM32MP15 Evaluation boards - Starter Package#step6 or STM32MP15 Discovery kits - Starter Package#step6

2 Checking the boot sequence

Assuming you have performed the optional step 6 above, the information (coming successively from the U-Boot and the Linux operating system that is booting on the board), should be displayed on the host PC Terminal.

> (i) The information displayed below might differ from that on your own application, and should just be considered an example.

```
File Edit View Search Terminal Help


Welcome to minicom 2.7

OPTIONS: I18n
Compiled on Jan  1 2014, 17:13:19.
Port /dev/ttyACM0, 16:19:42

Press CTRL-A Z for help on special keys


root@stm32mp1:~#
INFO:    Reset reason (0x4):
INFO:       Pad Reset from NRST
INFO:    Using SDMMC
INFO:       Instance 1
INFO:    Boot used partition fsbl1
NOTICE:  BL2: v1.4(debug):v1.4-441-g83df7ce-dirty
NOTICE:  BL2: Built : 08:05:45, Jan 23 2018
INFO:    BL2: Doing platform setup
INFO:    RAM: DDR3 2x4Gb 528MHz v1.32 +
INFO:    Memory size = 0x40000000 (1024 MB)
INFO:    BL2: Loading image id 4
INFO:    Loading image id=4 at address 0x2ffe3000
INFO:    Image id=4 loaded: 0x2ffe3000 - 0x2fff6000
INFO:    BL2: Loading image id 5
INFO:    Loading image id=5 at address 0xc0100000
INFO:    GPT partition size : 659968
WARNING: Skip signature check (header option)
INFO:    Image id=5 loaded: 0xc0100000 - 0xc01a1200
INFO:    read version 0 current version 0

NOTICE:  SP_MIN: v1.4(debug):v1.4-441-g83df7ce-dirty
NOTICE:  SP_MIN: Built : 08:05:45, Jan 23 2018
INFO:    ARM GICv2 driver initialized
INFO:    SP_MIN: Initializing runtime services
INFO:    SP_MIN: Preparing exit to normal world

U-Boot 2017.11-stm32mp-r2 (Jan 23 2018 - 08:06:48 -0500)
```

First information scrolled on the remote Terminal during boot

Last information scrolled on the remote Terminal during boot

In parallel, a U-Boot splash screen picture is displayed on the DSI display (if one is connected to the board).

A user space graphical boot splash screen (PSplash) picture is then briefly displayed on the HDMI monitor if one is connected

to the board, or otherwise on the DSI display (if one is connected to the board).

When the boot process is complete, the launcher application is shown on the HDMI monitor if one is connected to the board, or otherwise on the DSI display (if one is connected to the board).

launcher application layer shown on the HDMI monitor or on the DSI display
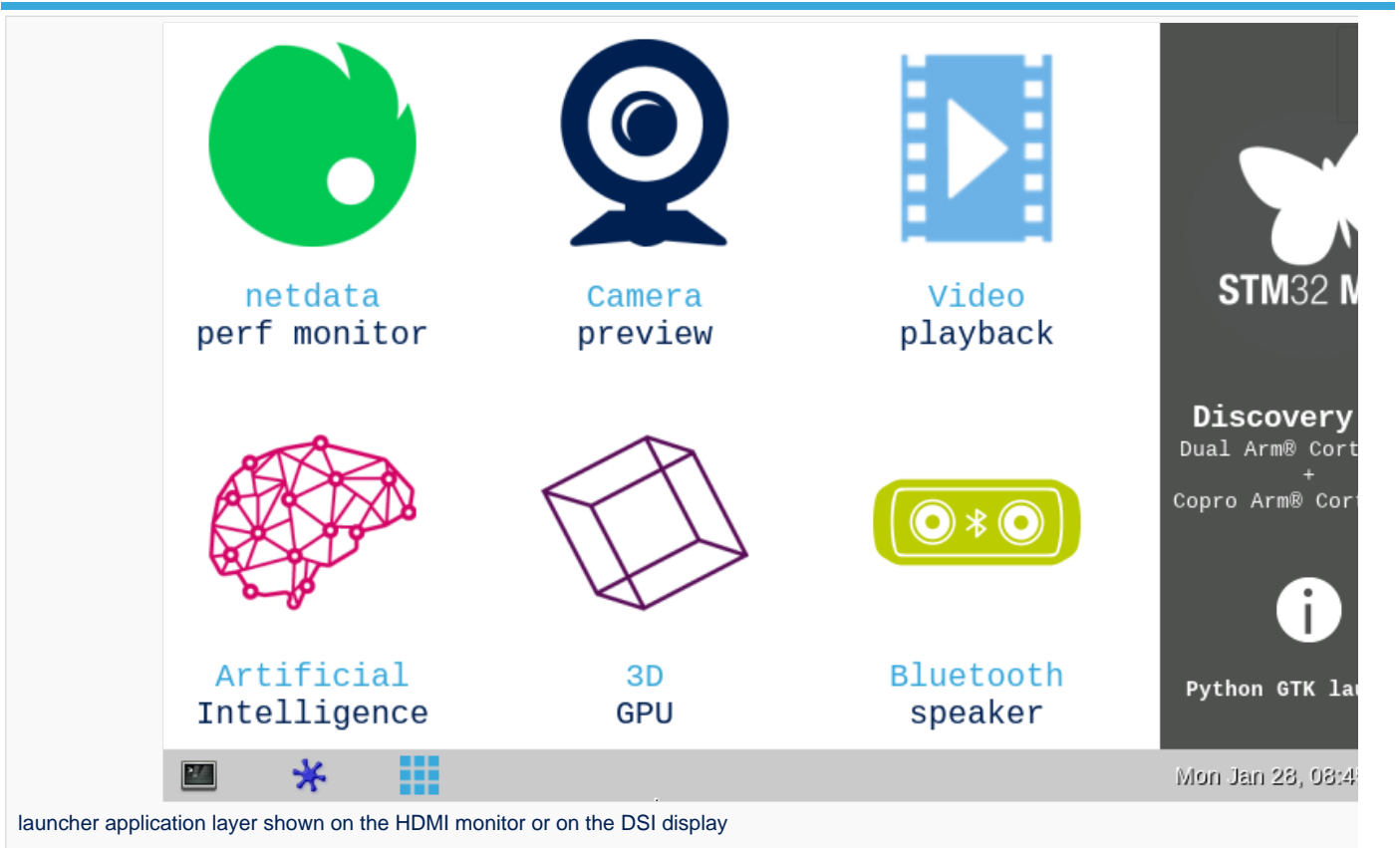
More detail about this launcher application in GTK demo launcher page.

3 Mouse, keyboard and Ethernet hot-plugs

Let's assume that the optional step 2 and step 3 were not achieved when setting up the system above.

When connecting a USB mouse, the following information is displayed by the Terminal program:

```
[  926.786326] usb 2-1.1: new low-speed USB device number 3 using ehci-platform
[  926.961413] input: Logitech Optical USB Mouse as /devices/platform/soc/5800d000.usbh-
ehci/usb2/2-1/2-1.1/2-1.1:1.0/0003:046D:C016.0001/input/input2
[  926.975098] hid-generic 0003:046D:C016.0001: input: USB HID v1.10 Mouse [Logitech
Optical USB Mouse] on usb-5800d000.usbh-ehci-1.1/input0
```

When connecting a USB keyboard, the following information is displayed by the Terminal program:

```
[ 1009.026567] usb 2-1.3: new low-speed USB device number 4 using ehci-platform
[ 1009.193990] input: Dell Dell USB Keyboard as /devices/platform/soc/5800d000.usbh-ehci
/usb2/2-1/2-1.3/2-1.3:1.0/0003:413C:2003.0002/input/input3
[ 1009.280101] hid-generic 0003:413C:2003.0002: input: USB HID v1.10 Keyboard [Dell Dell
USB Keyboard] on usb-5800d000.usbh-ehci-1.3/input0
```

When connecting an Ethernet cable, the following information is displayed by the Terminal program:

```
[ 1215.356377] stm32-dwmac 5800a000.ethernet eth0: Link is Up - 1Gbps/Full - flow control
rx/tx
[ 1215.363377] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
[ 1215.391068] Link is Up - 1000/Half
```

4   Remote and local Terminal programs

As already explained in the step 6 above, a **remote Terminal** program can be installed and configured on your host PC in order to communicate with the board through a serial link or an Ethernet link: see How to get Terminal.

The remote Terminal on your host PC can be used to enter command lines, as shown below with the *ifconfig* command to query the network interface parameters:

Example of command (*ifconfig*) executed on the remote Terminal program

A **local Terminal** program can be launched directly on the board. Click on the small icon at the top left corner of the display (see the red arrow on the figure below):

Icon to launch a local Terminal program

Wayland Terminal program
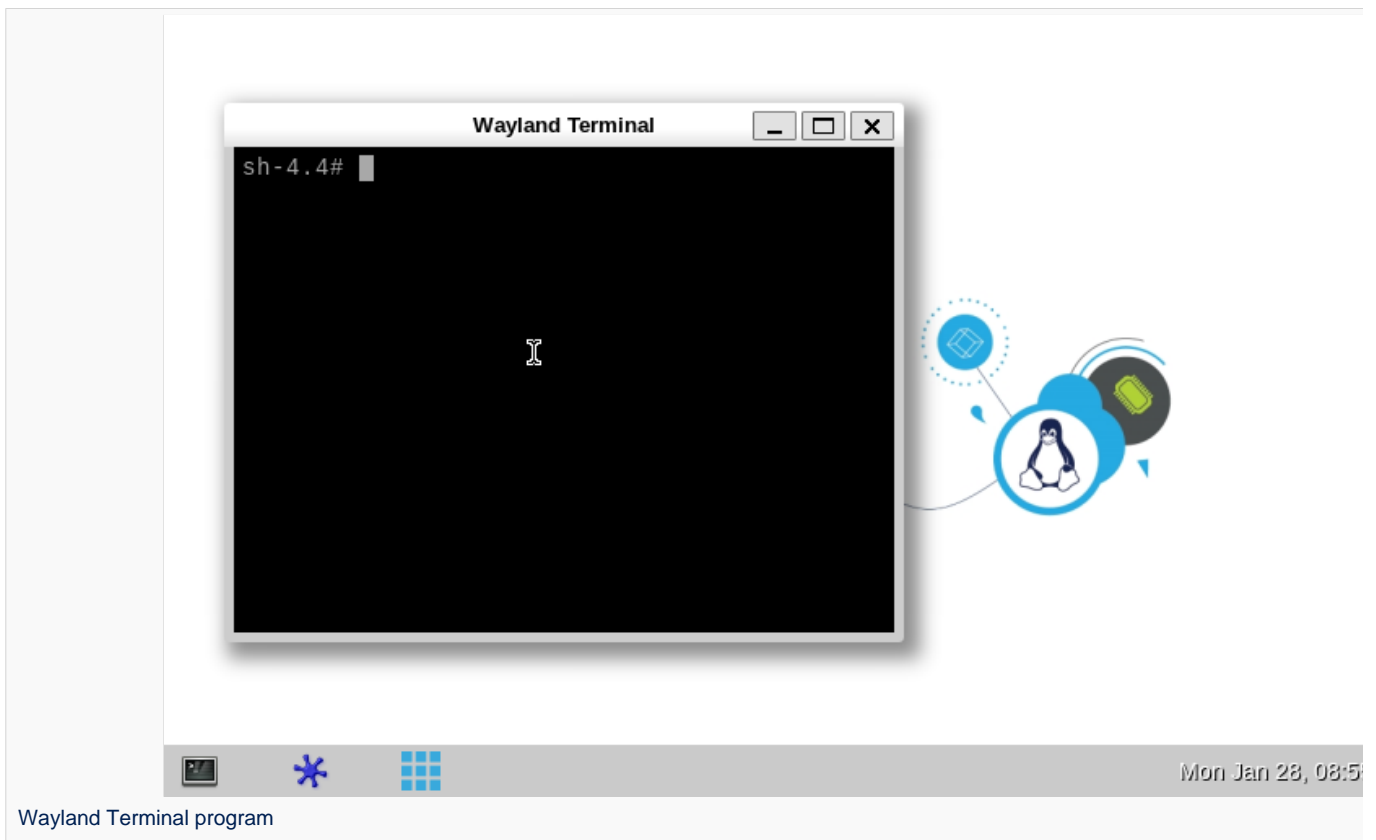
Then the on-board Wayland Terminal can be used to directly enter command lines as shown below, still with the *ip addr* command to query the network interface parameters:

Example of command (*ip addr*) executed on the Wayland Terminal program

Consequently, unless an explicit restriction is mentioned, command lines can be run from both Terminals.

Note: in this article, any command executed on the board (through the remote Terminal or the Wayland Terminal) starts with **Board $>**, while any command executed on the host PC starts with **PC $>**.

5 Executing basic commands

> ℹ️ The outputs of the commands executed below are examples; the outputs obtained might differ.

5.1 Having a look at the OpenSTLinux directory structure

The directory structure and directory contents in the OpenSTLinux distribution is standard. Some details are available in the OpenSTLinux directory structure article.

5.2 Identifying the software running on board

> ℹ️ Obviously, what you obtain might differ from what is displayed on the *Output example* column.

5.2.1 For ecosystem release v2.1.0

Software	How to get its version	Output example
TF-A	TF-A Version number	NOTICE: BL2: v2.2-r2.0(debug):v2.2-dirty
U-Boot	See the version displayed in the console	2020.01-stm32mp-r2 [...]
Linux kernel	`Board $> cat /proc/version`	Linux version 5.4.56 (xxxx@yyyy) (gcc version 9.3.0 (GCC)) [...]
GCC	`Board $> cat /proc/version`	Linux version 5.4.56 (xxxx@yyyy) (gcc version 9.3.0 (GCC)) [...]
Yocto Project	`Board $> grep DISTRO_CODENAME /etc/build`	DISTRO_CODENAME = dunfell
Weston	`Board $> weston --version`	weston 8.0.0
GStreamer	`Board $> gst-play-1.0 --version`	GStreamer 1.16.2

Software	How to get its version	Output example
| GPU | ```
Board $> cat /sys/kernel
/debug/gc/version
``` | 6.4.3.279124 built [...] |

5.2.2 For ecosystem release v2.0.0

Software	How to get its version	Output example
TF-A	TF-A Version number	NOTICE: BL2: v2.2-r1.0(debug):v2.0-r3.0
U-Boot	See the version displayed in the console	2020.01-stm32mp-r1 [...]
| Linux kernel | ```
Board $> cat /proc/version
``` | Linux version 5.4.31 (xxxx@yyyy) (gcc version 9.3.0 (GCC)) [...] |
| GCC | ```
Board $> cat /proc/version
``` | Linux version 5.4.31 (xxxx@yyyy) (gcc version 9.3.0 (GCC)) [...] |
| Yocto Project | ```
Board $> grep
DISTRO_CODENAME /etc/build
``` | DISTRO_CODENAME = dunfell |
| Weston | ```
Board $> weston --version
``` | weston 8.0.0 |
| GStreamer | ```
Board $> gst-play-1.0 --
version
``` | GStreamer 1.16.2 |
| GPU | ```
Board $> cat /sys/kernel
/debug/gc/version
``` | 6.4.1.244507 built [...] |

5.3   Configuration tips

- Configure the keyboard layout (qwerty, azerty...)

5.4   Getting board IP address

Prerequisite: your board is connected to your local network through the Ethernet connector (see step 3).

- Get the IP address of your board with the *ip* Linux command line (**recommended method**):

```
Board $> ip addr show
eth0
3: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen
1000
    link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff
    inet xx.xx.xx.xx/xx brd xx.xx.xx.xx scope global dynamic eth0
       valid_lft 159045sec preferred_lft 159045sec
    inet6 xxxx::xx:xx:xx:xx/xx scope link
       valid_lft forever preferred_lft forever
```

- Get the IP address of your board with the *ifconfig* Linux command line (a deprecated but well-known command):

```
Board $> ifconfig eth0
eth0      Link encap:Ethernet  HWaddr xx:xx:xx:xx:xx:xx
          inet addr:xx.xx.xx.xx  Bcast:xx.xx.xx.xx  Mask:255.255.252.0
          inet6 addr: xxxx::xx:xx:xx:xx/xx Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2619 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1311 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:353250 (344.9 KiB)  TX bytes:118305 (115.5 KiB)
          Interrupt:247
```

5.5   Copying a file from your host PC to your board (and reciprocally)

- How to transfer a file over network (**recommended method**)
- How to transfer a file over serial console

5.6   Miscellaneous commands

> The outputs of the commands executed below are examples; the outputs that you obtain might differ.

5.6.1   Printing distribution specific information

```
Board $> cat /etc/os-release
```

ID="openstlinux-weston" NAME="ST OpenSTLinux - Weston - (A Yocto Project Based Distro)" VERSION="3.1-openstlinux-20-06-10-internal (dunfell)" VERSION_ID="3.1-openstlinux-20-06-10-internal" PRETTY_NAME="ST OpenSTLinux - Weston - (A Yocto Project Based Distro) 3.1-openstlinux-20-06-10-internal (dunfell)"

Where:

ID	A lower-case string identifying the operating system
NAME	A string identifying the operating system

VERSION	A string identifying the operating system version, including a release code name
VERSION_ID	A lower case string identifying the operating system version
PRETTY_NAME	A full pretty name of the release

5.6.2   Printing system information

```
Board $> uname -a
Linux stm32mp1 5.4.56 #1 SMP PREEMPT Wed Aug 5 07:59:52 UTC 2020 armv7l armv7l armv7l GNU/
Linux
```

Where:

Linux	Kernel name
stm32mp1	Network node hostname
5.4.56	Kernel release
#1 SMP PREEMPT Wed Aug 5 07:59:52 UTC 2020	Kernel version
armv7l	Machine hardware name
GNU/Linux	Operating system

5.6.3   Printing Linux kernel and GCC versions

```
Board $> cat /proc/version
Linux version 5.4.31 (xxxx@yyyy) (gcc version 9.3.0 (GCC)) #1 SMP PREEMPT Wed Apr 8 07:08:
47 UTC 2020
```

Where:

Linux	Kernel name
version 5.4.31	Kernel release
(xxxx@yyyy)	Person (xxxx) who compiled the kernel, and machine (yyyy) where it happened
(gcc version 9.3.0 (GCC))	Version of the GCC compiler used to compile the kernel
#1 SMP PREEMPT Wed Apr 8 07:08: 47 UTC 2020	Kernel version; type of kernel (SMP) and date and time of the kernel compilation

5.6.4   Printing the amount of disk space available on all mounted file systems

```
Board $> df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       719M  342M  339M  51% /
```

```
devtmpfs        372M     0   372M    0% /dev
tmpfs           436M     0   436M    0% /dev/shm
tmpfs           436M    14M   423M    4% /run
tmpfs           436M     0   436M    0% /sys/fs/cgroup
tmpfs           436M   4.0K   436M    1% /tmp
/dev/mmcblk0p4   58M   8.2M    46M   16% /boot
/dev/mmcblk0p7  6.4G    68M   6.1G    2% /usr/local
tmpfs           436M   128K   436M    1% /var/volatile
/dev/mmcblk0p5   15M   8.2M   5.3M   61% /vendor
tmpfs            88M     0    88M    0% /run/user/0
```

Where:

Filesystem	Source of the mount point, usually a device
Size	Total size in human readable format (e.g. 1K, 234M, 2G)
Used	Used size in human readable format
Available	Available size in human readable format
Use%	Percentage of used size divided by the total size
Mounted on	Mount point

Note: the user file system (userfs) and the boot file system (bootfs) are accessible respectively through the */usr/local* mounting point, and the */boot* mounting point (see Flash partitions for a description of the file systems).

Das U-Boot -- the Universal Boot Loader (see U-Boot_overview)

Linux® is a registered trademark of Linus Torvalds.

Display Serial Interface (MIPI® Alliance standard)

High-Definition Multimedia Interface (HDMI standard)

Human Interface Device (for USB, Bluetooth...)

Trusted Firmware for Arm® Cortex®-A

Boot Loader stage 2

Graphics Processing Units

uniprocessor

Receive

Transmit

symetric multiprocessing