Embedded Linux USB

Universal Serial Bus

Norman McEntire

Reference

- https://en.wikipedia.org/wiki/USB

Introduction to USB

- Many Embedded Linux Projects Use USB
- Two Options
  - USB Host
    - Other USB Device (e.g. Keyboard, Mouse, Ethernet Adapter, etc.) plug into Embedded Linux
  - USB Device
    - Embedded Linux System appears as USB Device (to connect to USB Host)
      - Example: Embedded Linux appears as Ethernet Device to another USB Host
- Here we focus on USB Host (as provided by RPi)

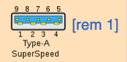
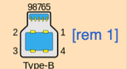
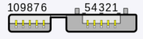
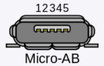
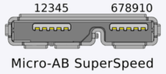
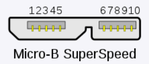
USB on RPi

- RPi has two “stacks” of USB Host Ports
  - Each stack as two USB Host A Connectors
    - USB 2: Up to 480 Mbits/s
    - USB 3 (blue connector): Up to 5Gbits/s

USB Connectors

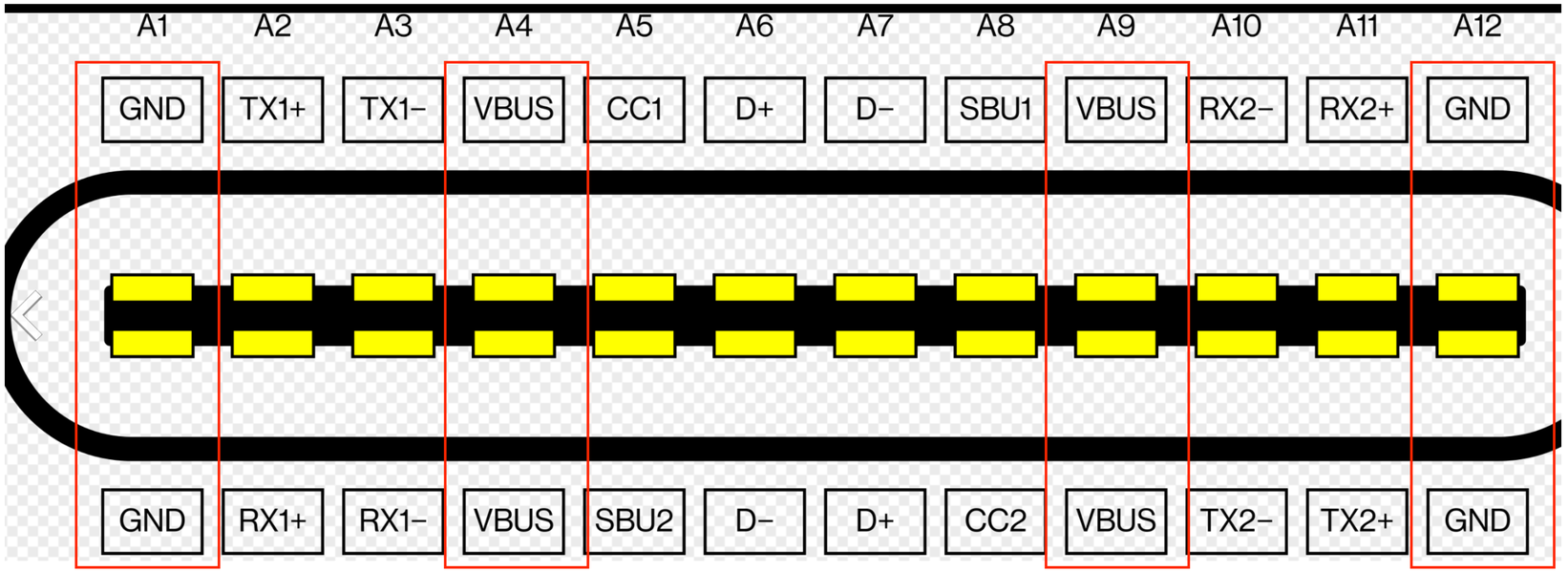
(Table from Wikipedia)

Available connectors by USB standard

Standard		USB 1.0 1996	USB 1.1 1998	USB 2.0 2001	USB 2.0 Revised	USB 3.0 2008	USB 3.1 2013	USB 3.2 2017	USB4 2019	USB4 v2.0 2022
Max Speed	***Marketing name*** **(operation mode)**	*Low-Speed* & *Full-Speed*		*High-Speed*		*SuperSpeed USB 5Gbps*, original: *SuperSpeed* (Gen 1)	*SuperSpeed USB 10Gbps*, original: *SuperSpeed+* (Gen 2)	*SuperSpeed USB 20Gbps* (USB 3.2 Gen 2×2)	*USB4 40Gbps* (USB4 Gen 3×2)	*USB4 80Gbps* (USB4 Gen 4)
	Signaling rate	1.5 Mbit/s & 12 Mbit/s			480 Mbit/s	5 Gbit/s	10 Gbit/s	20 Gbit/s	40 Gbit/s	80 Gbit/s
Connector	**Standard-A**	 Type-A 1.0 - 1.1			 Type-A 2.0	 Type-A SuperSpeed		 [rem 1]	—	
	Standard-B	 Type-B				 Type-B SuperSpeed		 [rem 1]	—	
	Mini-A	[rem 2]		 Mini-A		—				
	Mini-AB[rem 3][rem 4]			 Mini-AB		—				
	Mini-B			 Mini-B		—				
	Micro-A[rem 5]				 Micro-A	 Micro-A SuperSpeed		 [rem 1]	—	
	Micro-AB[rem 3][rem 7]	[rem 2] [rem 6]			 Micro-AB	 Micro-AB SuperSpeed		 [rem 1]	—	
	Micro-B				 Micro-B	 Micro-B SuperSpeed		 [rem 1]	—	
	Type-C (USB-C)	[rem 6]				 (Enlarged to show detail)				

USB C

(From Wikipedia)



dmesg and USB

```
$ dmesg | grep usb
[    0.131429] usbcore: registered new interface driver usbfs
[    0.131478] usbcore: registered new interface driver hub
[    0.131528] usbcore: registered new device driver usb
[    0.131766] usb_phy_generic phy: supply vcc not found, using dummy regulator
[    0.131911] usb_phy_generic phy: dummy supplies not allowed for exclusive requests
[    1.734769] usbcore: registered new interface driver r8152
[    1.734827] usbcore: registered new interface driver lan78xx
[    1.734876] usbcore: registered new interface driver smsc95xx
[    1.785829] usb usb1: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 6.01
[    1.785848] usb usb1: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[    1.785862] usb usb1: Product: xHCI Host Controller
[    1.785874] usb usb1: Manufacturer: Linux 6.1.21-v7l+ xhci-hcd
[    1.785885] usb usb1: SerialNumber: 0000:01:00.0
[    1.787157] usb usb2: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice= 6.01
[    1.787176] usb usb2: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[    1.787190] usb usb2: Product: xHCI Host Controller
[    1.787202] usb usb2: Manufacturer: Linux 6.1.21-v7l+ xhci-hcd
[    1.787214] usb usb2: SerialNumber: 0000:01:00.0
[    1.789582] usbcore: registered new interface driver uas
[    1.789667] usbcore: registered new interface driver usb-storage
[    1.797587] usbcore: registered new interface driver usbhid
[    1.797599] usbhid: USB HID core driver
[    2.073764] usb 1-1: new high-speed USB device number 2 using xhci_hcd
[    2.266369] usb 1-1: New USB device found, idVendor=2109, idProduct=3431, bcdDevice= 4.21
[    2.266396] usb 1-1: New USB device strings: Mfr=0, Product=1, SerialNumber=0
[    2.266411] usb 1-1: Product: USB2.0 Hub
[    7.307181] usbcore: registered new interface driver brcmfmac
```

lsusb and USB

```
$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

```
$ lsusb --tree
/:  Bus 02.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/4p, 5000M
/:  Bus 01.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/1p, 480M
    |__ Port 1: Dev 2, If 0, Class=Hub, Driver=hub/4p, 480M
```

lsusb --verbose

```
$ sudo lsusb --verbose

Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB               3.00
  bDeviceClass            9 Hub
  bDeviceSubClass         0
  bDeviceProtocol         3
  bMaxPacketSize0         9
  idVendor           0x1d6b Linux Foundation
  idProduct          0x0003 3.0 root hub
  bcdDevice            6.01
  iManufacturer           3 Linux 6.1.21-v7l+ xhci-hcd
  iProduct                2 xHCI Host Controller
  iSerial                 1 0000:01:00.0
  bNumConfigurations      1
  Configuration Descriptor:
. . .
```

Demo: Plug in USB Camera

```
$ ls /dev/ > /tmp/a.txt

$ ## Plug in USB device

$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 003: ID 046d:0825 Logitech, Inc. Webcam C270
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub

$ ls /dev > /tmp/b.txt
$ diff /tmp/a.txt /tmp/b.txt
41a42
> media4
179a181,182
> video0
> video1

$ ls -l /dev/media4
crw-rw---- 1 root video 239, 4 Nov  1 05:07 /dev/media4
$ ls -l /dev/video?
crw-rw----+ 1 root video 81, 14 Nov  1 05:07 /dev/video0
crw-rw----+ 1 root video 81, 15 Nov  1 05:07 /dev/video1
```

dmesg results from plugging in Camera

```
$ dmesg | tail

[  975.841138] usb 1-1.4: new high-speed USB device number 3 using xhci_hcd
[  976.183232] usb 1-1.4: New USB device found, idVendor=046d, idProduct=0825, bcdDevice= 0.12
[  976.183258] usb 1-1.4: New USB device strings: Mfr=0, Product=0, SerialNumber=2
[  976.183275] usb 1-1.4: SerialNumber: 19226FD0
[  976.306434] usb 1-1.4: Found UVC 1.00 device <unnamed> (046d:0825)
[  976.419130] input: UVC Camera (046d:0825) as /devices/platform/scb/fd500000.pcie/
pci0000:00/0000:00:00.0/0000:01:00.0/usb1/1-1/1-1.4/1-1.4:1.0/input/input2
[  976.419753] usbcore: registered new interface driver uvcvideo
[  976.488477] usb 1-1.4: set resolution quirk: cval->res = 384
[  976.488814] usbcore: registered new interface driver snd-usb-audio
```

Options for Accessing USB Devices

- Option 1 (most common): As char, block, or network device
  - The USB device looks like any other char, block, or network device
- Option 2: libusb
  - User-mode library for access to USB devices
- Option 3: Kernel-mode usb driver

Demo: libusb - Part 1

```
$ find /usr/include | grep libusb

$ sudo apt-get install libusb-1.0-0-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libirs-export161 libisccfg-export163 policycoreutils selinux-utils
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libusb-1.0-doc
The following NEW packages will be installed:
  libusb-1.0-0-dev libusb-1.0-doc
. . .

$ dpkg -L libusb-1.0-0-dev
/.
. . .
/usr/include/libusb-1.0/libusb.h
/usr/lib
/usr/lib/arm-linux-gnueabihf
/usr/lib/arm-linux-gnueabihf/libusb-1.0.a
/usr/lib/arm-linux-gnueabihf/pkgconfig
/usr/lib/arm-linux-gnueabihf/pkgconfig/libusb-1.0.pc
. . .
/usr/lib/arm-linux-gnueabihf/libusb-1.0.so
```

Demo: libusb - Part 2

```
$ find /usr/include | grep libusb

$ sudo apt-get install libusb-1.0-0-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libirs-export161 libisccfg-export163 policycoreutils selinux-utils
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libusb-1.0-doc
The following NEW packages will be installed:
  libusb-1.0-0-dev libusb-1.0-doc
. . .

$ dpkg -L libusb-1.0-0-dev
/.
. . .
/usr/include/libusb-1.0/libusb.h
/usr/lib
/usr/lib/arm-linux-gnueabihf
/usr/lib/arm-linux-gnueabihf/libusb-1.0.a
/usr/lib/arm-linux-gnueabihf/pkgconfig
/usr/lib/arm-linux-gnueabihf/pkgconfig/libusb-1.0.pc
. . .
/usr/lib/arm-linux-gnueabihf/libusb-1.0.so
```

hello-usb.c - Part 1

```
#include <stdio.h>
#include <libusb-1.0/libusb.h>

int main() {
    libusb_context *ctx = NULL;
    libusb_device **devs;
    ssize_t cnt;

    // Initialize the libusb library
    if (libusb_init(&ctx) < 0) {
        fprintf(stderr, "Failed to initialize libusb\n");
        return 1;
    }
```

hello-usb.c - Part 2

```
// Get the list of USB devices
cnt = libusb_get_device_list(ctx, &devs);
if (cnt < 0) {
    fprintf(stderr, "Failed to get device list\n");
    libusb_exit(ctx);
    return 1;
}

printf("Found %ld USB devices\n", cnt);

// Print information about each connected USB device
for (ssize_t i = 0; i < cnt; i++) {
    libusb_device *dev = devs[i];
    struct libusb_device_descriptor desc;

    if (libusb_get_device_descriptor(dev, &desc) < 0) {
        fprintf(stderr, "Failed to get device descriptor\n");
    } else {
        printf("Device %ld: VID=0x%04x, PID=0x%04x\n", i, desc.idVendor, desc.idProduct);
    }
}
```

hello-usb.c - Part 3

```
    // Free the list of devices, unreference each device to allow them to be freed
    libusb_free_device_list(devs, 1);

    // Close the libusb context
    libusb_exit(ctx);

    printf("Program finished successfully!\n");
    return 0;
}
```

Demo: libusb - Building and Running

```
$ gcc -o usb-demo usb-demo.c -lusb-1.0

$ ./usb-demo
Found 4 USB devices
Device 0: VID=0x1d6b, PID=0x0003
Device 1: VID=0x046d, PID=0x0825
Device 2: VID=0x2109, PID=0x3431
Device 3: VID=0x1d6b, PID=0x0002
Program finished successfully!
```

Next Demo
Plug USB/Ethernet Adapter Into RPi

RPi dmsg: Plugging In USB/Ethernet Adapter

```
$ dmesg | tail
[   16.460172] Bluetooth: BNEP filters: protocol multicast
[   16.460184] Bluetooth: BNEP socket layer initialized
[   16.466860] Bluetooth: MGMT ver 1.22
[   16.490294] NET: Registered PF_ALG protocol family
[   16.529029] cryptd: max_cpu_qlen set to 1000
[   17.004796] Bluetooth: RFCOMM TTY layer initialized
[   17.004821] Bluetooth: RFCOMM socket layer initialized
[   17.004839] Bluetooth: RFCOMM ver 1.11
```

```
$ dmesg | tail
[11453.194478] usb 1-1.2: New USB device found, idVendor=2357, idProduct=0601, bcdDevice=30.00
[11453.194503] usb 1-1.2: New USB device strings: Mfr=1, Product=2, SerialNumber=6
[11453.194518] usb 1-1.2: Product: USB 10/100/1000 LAN
[11453.194531] usb 1-1.2: Manufacturer: TP-LINK
[11453.194544] usb 1-1.2: SerialNumber: 000001000000
[11457.533519] usb usb2-port2: Cannot enable. Maybe the USB cable is bad?
[11461.873562] usb usb2-port2: Cannot enable. Maybe the USB cable is bad?
[11461.873930] usb usb2-port2: attempt power cycle
[11462.223698] usb 1-1.2: reset high-speed USB device number 5 using xhci_hcd
[11462.393320] r8152 1-1.2:1.0: load rtl8153a-3 v2 02/07/20 successfully
[11462.434662] r8152 1-1.2:1.0 eth1: v1.12.13
[11462.498005] usbcore: registered new interface driver cdc_ether
```

RPi - lsmod

cdc_ appears

```
$ lsmod | head
Module                  Size  Used by
cdc_ether              16384  0
```

RPi - ifconfig

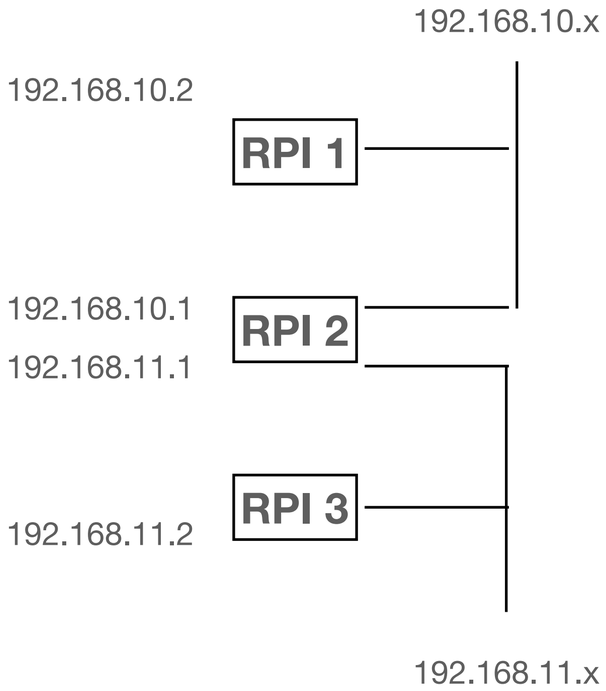
New eth1 appears

```
$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether e4:5f:01:7d:fc:ea  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether f4:f2:6d:14:01:80  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Example: Use RPI as Router

With Two eth interfaces (eth0, eth1)



Summary

- Many Embedded Linux Projects Use USB
- Two Options
  - USB Host
    - Other USB Device (e.g. Keyboard, Mouse, Ethernet Adapter, etc.) plug into Embedded Linux
  - USB Device
    - Embedded Linux System appears as USB Device (to connect to USB Host)
      - Example: Embedded Linux appears as Ethernet Device to another USB Host
- Here we focus on USB Host (as provided by RPi)