

My experience of building AMBED server by RaspberryPI

Shu JA3GQJ/XLX595/<http://hamradio.dip.jp/index.php>

1. Things necessary

Two ThumbDVs
One RaspberryPI3

2. Structure of procedure manual

- 1) Change ThumbDV Product Description
- 2) Obtain and install FTDI D2XX Drivers
- 3) Installation of AMBED server
- 4) AMBED server restart procedure

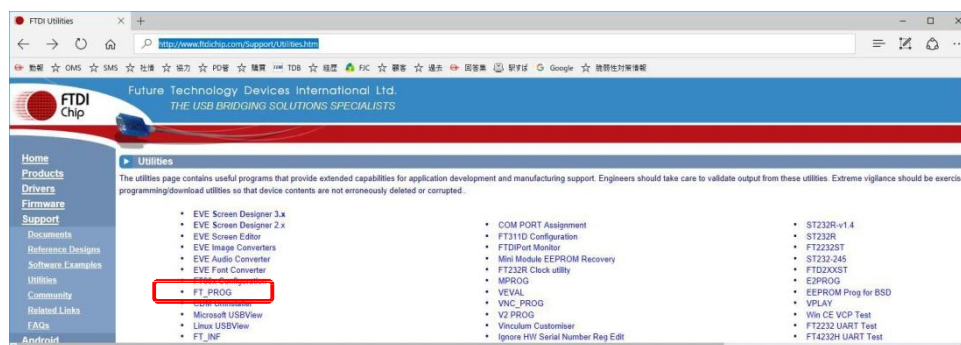
3. Precondition

- 1) In this procedure manual, AMBED server platform is RaspberryPI 3.
- 2) This manual assumes that you are using ThumbDV.
- 3) In the description of this procedure manual, "#" is the root login prompt and "\$" is the login prompt of the user ID (eg pi).

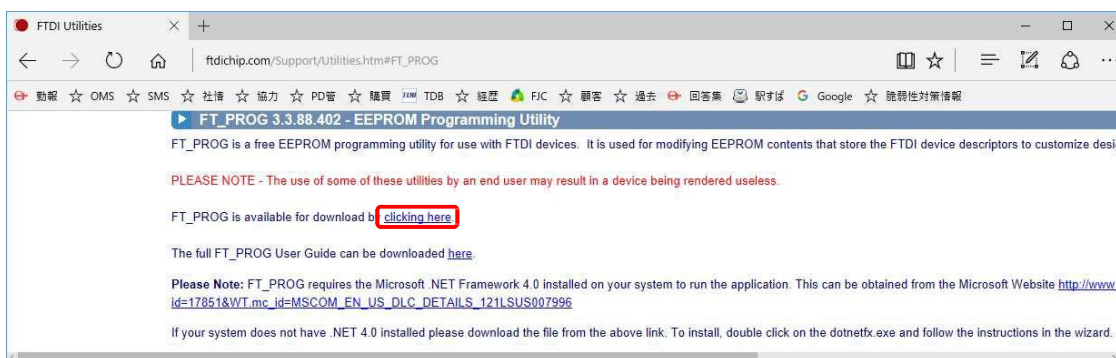
4. Change ThumbDV Product Description

- 1) Devices connected to ambd server via USB are based on USB-3000.
ThumbDV with similar functions can be substituted, but in order to substitute, it is necessary to change "Product Description" of ThumbDV to "USB - 3000" character string.
- 2) In order to change "Product Description" of ThumbDV, it is necessary to install "FT Prog" tool on Windows PC.
- 3) Obtaining "FT Prog" tool

- ① Click "FT Prog" at <http://www.ftdichip.com/Support/Utilities.htm>

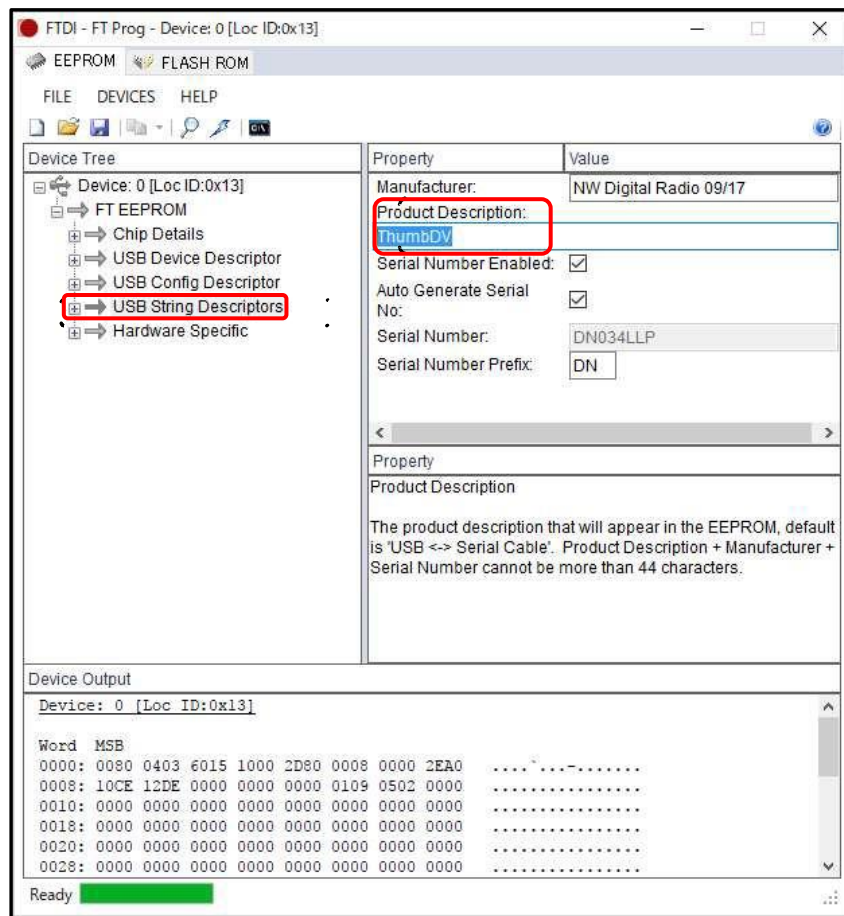


- ② Click "clicking here" to download "FT Prog".

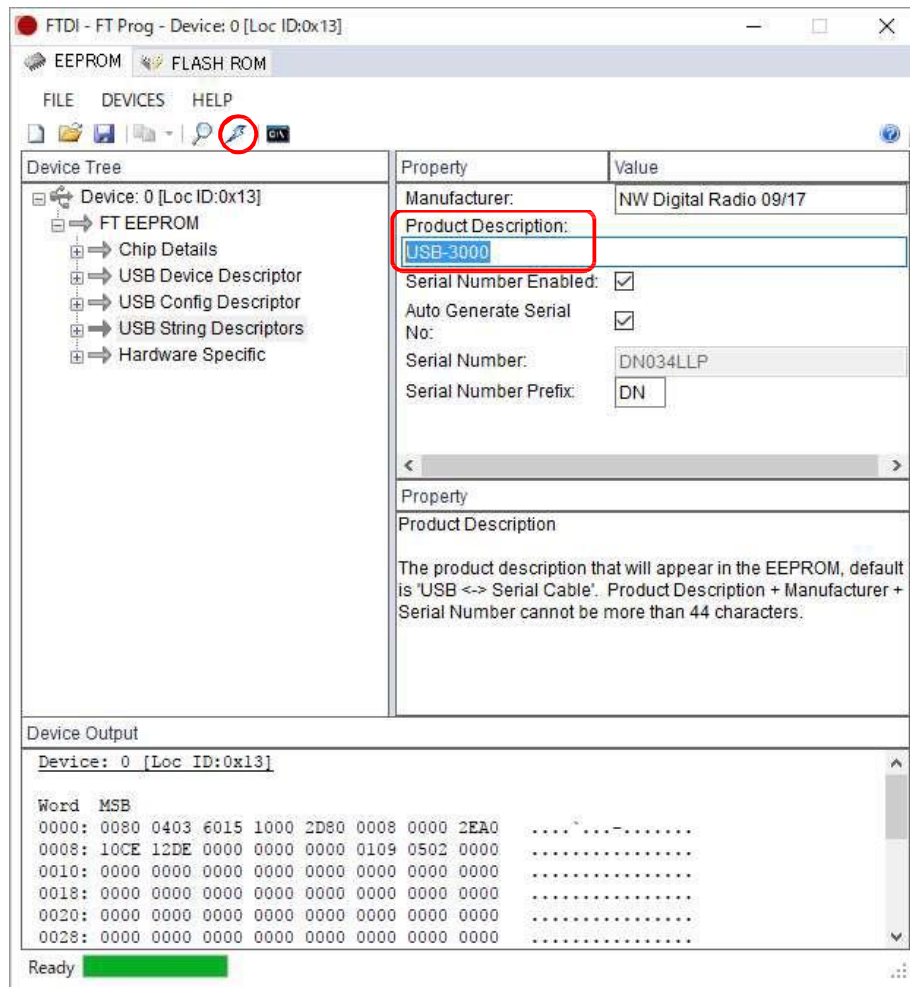


http://www.ftdichip.com/Support/Utilities/FT_Prog_v3.6.88.402%20Installer.exe

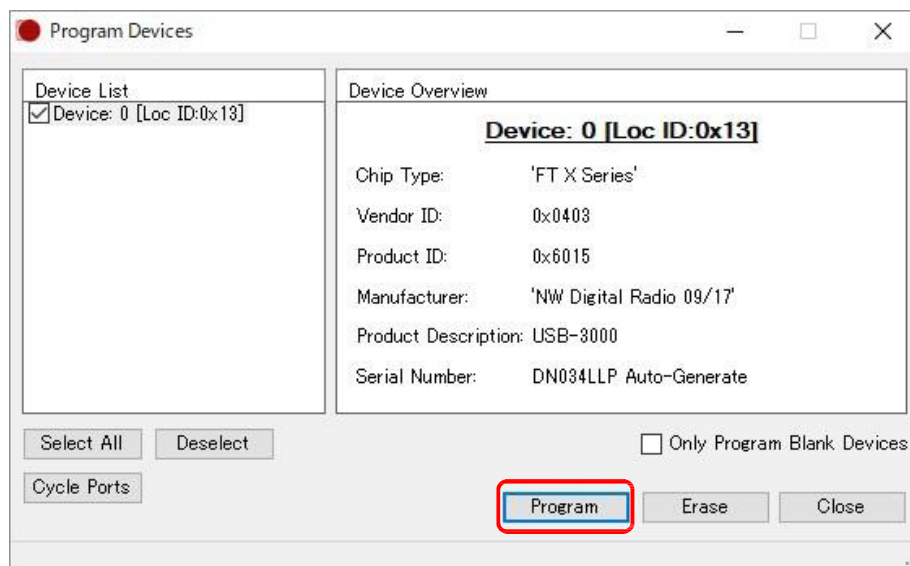
- 4) Installation of "FT Prog" tool
Excute the downloaded "FT_Prog_v3.6.88.402 Installer.exe" and install it.
- 5) Change of "Product Description"
 - ① Connect ThumbDV to USB port of Windows PC.
 - ② Excute the "FT Prog" tool.
 - ③ Press the "F5" key of keyboard to read the DEVICE information.
 - ④ Click "USB String Descriptors" in "Device Tree".



- ⑤ "Product Description" shows "ThumbDV". Change it here.
- ⑥ Overwrite "USB String Descriptors" from "ThumbDV" to "USB-3000".
- ⑦ Click the icon of the lightning mark (⚡ in the figure below).



⑧ Click the "Program" button to write to ThumbDV.



6. Obtain and install FTDI D2XX Drivers

"FTDI D2XX Drivers" must be prepared to recognize ThumbDV connected by USB to RaspberryPI.

- ① Check the type of "FTDI D 2 XX Drivers" of Raspberry PI 3.

```
# lscpu
Architecture: armv7l
Byte Order:  Little Endian
CPU(s):      4
On-line CPU(s) list:  0-3
Thread(s) per core:   1
Core(s) per socket:   4
Socket(s):    1
Model:       4
Model name:  ARMv7 Processor rev 4 (v7l)
```

The other terms are omitted.

With this command, you can confirm that the type of RaspberryPI 3 is ARMv7 ".

【Reference】 <http://senyoltw.hatenablog.jp/entry/2016/04/24/043244>

- ② Get "FTDI D2XX Drivers" and install in advance.

English installation procedure is posted in the following

URL. <http://www.ftdichip.com/Drivers/D2XX/Linux/ReadMe-linux.txt>

- 3) Before starting work(Important)

It will not work unless ThumbDV is reset.

So just after rebooting without loading ThumbDV in RaspberryPI 3 you will start installing FTDI D2XX

Drivers below.

- 4) Using the browser of RaspberryPI, download FTDI D2XX Drivers.

- ① Using the browser of RaspberryPI, download FTDI D2XX Drivers.

<http://www.ftdichip.com/Drivers/D2XX.htm>

- ② Select the following from the table of "Currently Supported D2XX Drivers".

Operating System	:	Linux
Processor Architecture	:	ARM

- ③ Click "1.4.6 ARMv7 hard-float" to download it.

Download file name : 「libftd2xx-arm-v7-hf-1.4.6.tgz」

【Reference】 "Install FTDI Drivers in Raspai"

<http://android.ohwada.jp/archives/6253>

D2XX Drivers

D2XX drivers allow direct access to the USB device through a DLL. Application software can access the USB device through a series of DLL function calls. The functions available are listed in the [D2XX Programmer's Guide](#) document which is available from the [Documents](#) section of this site.

Programming examples using the D2XX drivers and DLL can be found in the [Projects](#) section of this site.

This software is provided by Future Technology Devices International Limited "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall future technology devices international limited be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services, loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

FTDI drivers may be used only in conjunction with products based on FTDI parts.

FTDI drivers may be distributed in any form as long as license information is not modified.

If a custom vendor ID and/or product ID or description string are used, it is the responsibility of the product manufacturer to maintain any changes and subsequent WHCK re-certification as a result of making these changes.

For more detail on FTDI Chip Driver licence terms, please [click here](#).

Currently Supported D2XX Drivers:

Operating System	Release Date	Processor Architecture					Comments
		x86 (32-bit)	x64 (64-bit)	ARM	MIPS	SH4	
Windows*	2017-08-30	2.12.28	2.12.28	-	-	-	WHQL Certified. Includes VCP and D2XX. Available as a setup executable. Please read the Release Notes and Installation Guides.
Windows RT	2014-07-04	1.0.2	-	1.0.2	-	-	A guide to support the driver (AN_271) is available here
Linux	2017-06-29	1.4.6	1.4.6	1.4.6 ARMv5 soft-float 1.4.6 ARMv5 soft-float uClibc 1.4.6 ARMv6 hard-float (suits Raspberry Pi) 1.4.6 ARMv7 hard-float 1.4.6 ARMv8 hard-float	1.4.6 MIPS32 soft-float 1.4.6 MIPS32 hard-float 1.4.6 MIPS openwrt-uclibc		If unsure which ARM version to use, compare the output of <code>readelf</code> and <code>file</code> commands on a system binary with the content of <code>release/build/libftd2xx.txt</code> in each package. ReadMe

- ④ Move the download file to the working folder.

The following is the procedure when the working folder is under "/".

```
# cd /home/pi/Downloads
# mv libftd2xx-arm-v7-hf-1.4.6.tgz /
# cd /
```

- ⑤ Unzip the download file.

```
# tar xfvz libftd2xx-arm-v7-hf-1.4.6.tgz
```

- ⑥ Confirm the unzipped folder structure.

```
# cd release
# ls -l
```

```
build
libftd2xx  (re-linkable objects)
libusb     (re-linkable objects)
libftd2xx.a (static library)
libftd2xx.so.1.4.6 (dynamic library)
libftd2xx.txt (platform-specific information)
examples
libusb     (source code)
ftd2xx.h
WinTypes.h
```

⑦ Install shared library and static library D2XX.

```
<1> Copy library to /usr.
    # cp /release/build/libftd2xx.* /usr/local/lib
<2> Grant the library access privilege.
    # chmod 0755 /usr/local/lib/libftd2xx.so.1.4.6
<3> Create a symbolic link.
    # ln -sf /usr/local/lib/libftd2xx.so.1.4.6 /usr/local/lib/libftd2xx.so
    ----- Link source ----- Registered name -----
```

5) Connect ThumbDV to Raspberry PI and test if EEPROM calling is possible.

① Compile the EEPROM calling program.

```
# cd /release/examples/EEPROM/read

# make -B      ※If it is normal, the following line is displayed.

cc main.c -o bitmode -Wall -Wextra -lftd2xx -lpthread -lrt -Wl,-rpath /usr/local/lib -L/usr/local/lib
```

② Attach the first ThumbDV to the USB port of RaspberryPI3.

③ If the OS registers the driver module, delete it once.

```
<1> Confirm whether OS registers driver module.
    # lsmod
```

```
xrf111.dip.jp - Tera Term VT
root@XLX111:/release/examples/EEPROM/read# lsmod
Module              Size  Used by
ftdi_sio            32963  0
usbserial           29943  1 ftdi_sio
fuse                99603  3
rfcomm              37723  6
cmac                 3239  1
bnep                12051  2
hci_uart            20020  1
btbcm               7916  1 hci_uart
bluetooth           365511  29 hci_uart,bnep,btbcm,rfcomm
brcmfmac            223048  0
brcmutil            9092  1 brcmfmac
cfg80211            543091  1 brcmfmac
snd_bcm2835         24427  1
snd_pcm             98501  1 snd_bcm2835
rfkill              20851  6 bluetooth,cfg80211
snd_timer           23968  1 snd_pcm
snd                 70032  5 snd_timer,snd_bcm2835,snd_pcm
bcm2835_gpiomem      3940  0
fixed               3285  0
uio_pdrv_genirq      3923  0
uio                 10204  1 uio_pdrv_genirq
evdev               12423  0
sg                  20825  0
i2c_dev             6913  0
ip_tables           13161  0
x_tables            20578  1 ip_tables
ipv6                409035  64
```

Red borders indicate that ftdi_sio and usbserial are already registered.

<2> If ftdi_sio or usbserial is registered, delete it with the following command.

```
# rmmod ftdi_sio
# rmmod usbserial
```

<3> The deleted results are as follows. In the figure below ftdi_sio and usbserial are deleted.

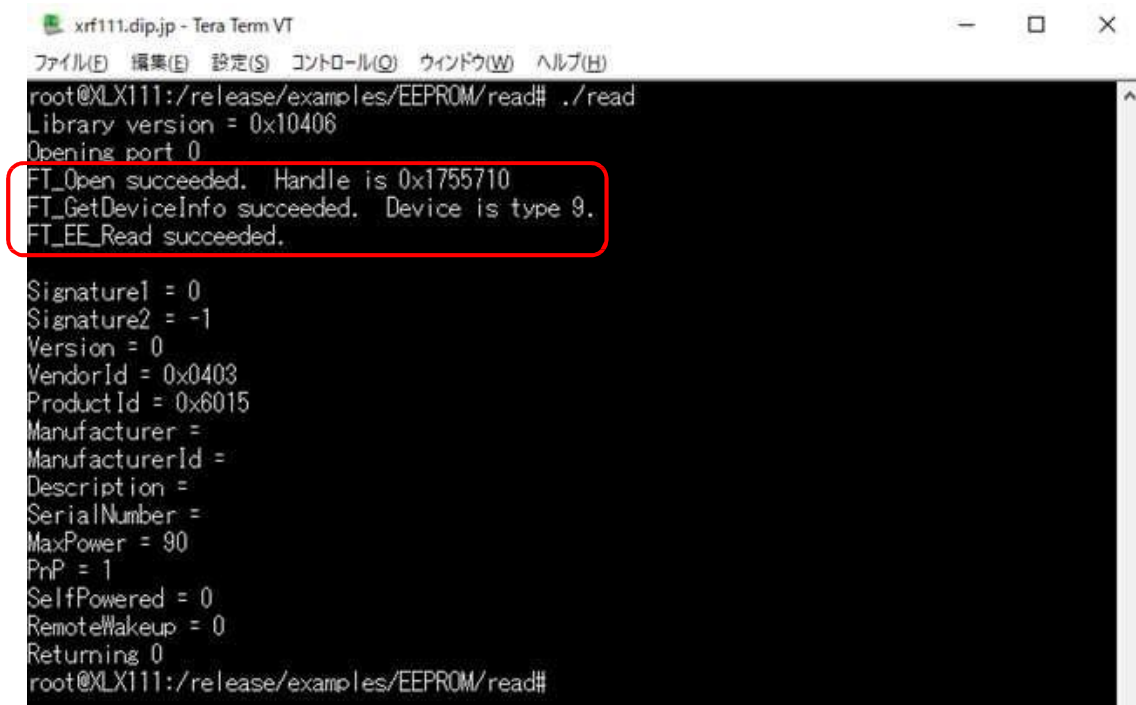


```
xrf111.dip.jp - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(C) ウィンドウ(W) ヘルプ(H)
root@XLX111:/release/examples/EEPROM/read# lsmod
Module                Size  Used by
fuse                  99603  3
rfcomm                37723  6
cmac                   3239   1
bnep                  12051  2
hci_uart              20020  1
btbcm                  7916   1 hci_uart
bluetooth             365511 29 hci_uart, bnep, btbcm, rfcomm
```

④ Execute the EEPROM calling program.

```
# cd /release/examples/EEPROM/read
# ./read
```

⑤ The following contents are displayed for the execution result.



```
xrf111.dip.jp - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(C) ウィンドウ(W) ヘルプ(H)
root@XLX111:/release/examples/EEPROM/read# ./read
Library version = 0x10406
Opening port 0
FT_Open succeeded. Handle is 0x1755710
FT_GetDeviceInfo succeeded. Device is type 9.
FT_EE_Read succeeded.

Signature1 = 0
Signature2 = -1
Version = 0
VendorId = 0x0403
ProductId = 0x6015
Manufacturer =
ManufacturerId =
Description =
SerialNumber =
MaxPower = 90
PnP = 1
SelfPowered = 0
RemoteWakeup = 0
Returning 0
root@XLX111:/release/examples/EEPROM/read#
```

The red frame indicates that the reading is successful.

6) Confirmation test of the static library is carried out.

① Copy the library under /usr.

```
# cd /release/examples/static
# cp /usr/local/lib/libftd2xx.a /release/examples/static
```

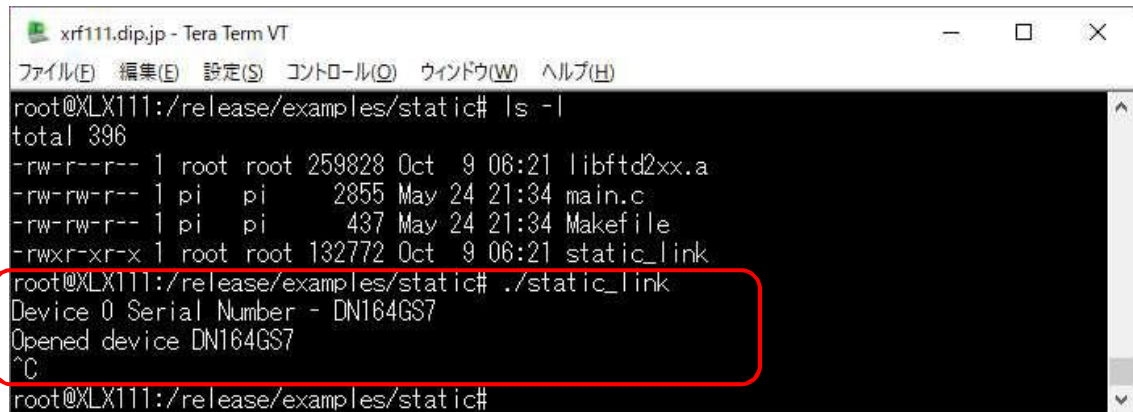
② Compile the program for checking the static library.

```
# make -B ※ If it is normal, the following line is displayed.
gcc main.c -o static_link libftd2xx.a -lpthread -lrt -Wall -Wextra
```

③ Execute the static library check program.

```
# ./static_link
```

④ Make sure that you have successfully read the serial number of ThumbDV.



```
xrf111.dip.jp - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(C) ウィンドウ(W) ヘルプ(H)
root@LX111:/release/examples/static# ls -l
total 396
-rw-r--r-- 1 root root 259828 Oct  9 06:21 libftd2xx.a
-rw-rw-r-- 1 pi  pi    2855 May 24 21:34 main.c
-rw-rw-r-- 1 pi  pi    437 May 24 21:34 Makefile
-rwxr-xr-x 1 root root 132772 Oct  9 06:21 static_link
root@LX111:/release/examples/static# ./static_link
Device 0 Serial Number - DN164GS7
Opened device DN164GS7
^C
root@LX111:/release/examples/static#
```

⑤ Ctrl + C to end it.

⑥ To reset ThumbDV, shut down Raspberry PI3.

```
# shutdown now
```

7. Installation of AMBED server

1) The storage location of the developer (Mr. Luc) installation procedure is the following URL.
Please refer as necessary. <https://github.com/LX3JL/xlxd/tree/master/AMBED>

2) With RaspberryPI 3 powered off, disconnect ThumbDV from USB and turn on the power. (Important)

3) Installation of the g ++ compiler (Skip this if installed.)

```
# apt-get install build-essential
# apt-get install g++-4.7
```

4) Download the AMBED source code and compile it.

```
# cd /root
# git clone https://github.com/LX3JL/xlxd.git
# cd xlxd/AMBED
# make
# make clean
# make install
# chmod 755 /root/xlxd/AMBED/run
```

5) AMBED startup script settings

① Edit the following line to match the IP address of the AMBED server.

```
# nano /root/xlxd/AMBED/run
```

② If AMBED is running on the same machine as xlxd, use **the default 127.0.0.1**.
Otherwise, set the machine's own IP address.

6) Starting AMBED

① Since logs are always output to the screen during startup, use Remote Desktop such as VNC, or on the monitor connected to the HDMI terminal of RaspberryPI 3 display Desktop screen.

② Open LXTerminal on the Desktop screen.

③ Make root privilege.

```
# su -
```

※ Enter a password if necessary.

- ④ You will install both ThumbDV in RaspberryPI3. **(Important)**
- ⑤ Start AMBED.

```
# cd /root/xlxd/AMBED
# ./run
```
- ⑥ The normal screen after starting AMBED is as follows.

```
xrf111.dip.jp - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
root@XLX111:~/xlxd/ambed# ./run
root@XLX111:~/xlxd/ambed# Starting AMBEd
Initializing vocodecs:
Detected 2 USB-FTDI devices
Description : USB-3000 Serial : DN164GS7
Description : USB-3000 Serial : DN2179BE
Opening USB-3000:DN164GS7 device
ReadDeviceVersion : 0AMBE3000R1V120.E100.XXXX.C106.G514.R009.B0010411.C0020208
Opening USB-3000:DN2179BE device
ReadDeviceVersion : 0AMBE3000R1V120.E100.XXXX.C106.G514.R009.B0010411.C0020208
Codec interfaces initialized successfully : 2 channels availables
Initializing controller
AMBEd started and listening on 127.0.0.1
```

(Supplement)

- In the early stage, AMBED scans all USB ports for detection.
- They are setting the status of all available USB-3xxx devices.

8.AMBED server restart procedure

- 1) The source code of the AMBED server as of August 2017 does not incorporate the process of resetting the USB connected ThumbDV with software. Therefore, this procedure describes how to restart the AMBED server in the procedure to reset by hardware.
- 2) Shut down RaspberryPI3.

```
# shutdown now
```
- 3) Unplug the power cord from RaspberryPI3.
- 4) Also remove both of two ThumbDV from RaspberryPI3.**(Important)**
- 5) Supply power to RaspberryPI3.
- 6) Starting AMBED

- ① Open the Remote DeskTop screen such as VNC or display the DeskTop screen while watching the monitor connected to the HDMI terminal of RaspberryPI 3.

Open LXTerminal on the DeskTop screen.

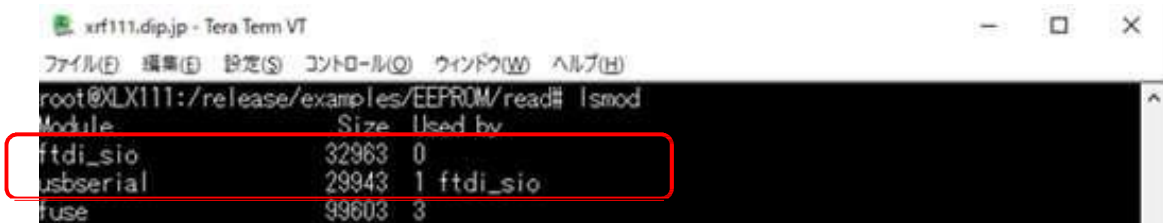
③ Make root privilege

su - ※ Enter a password if necessary

④ You will install both twoThumbDV in RaspberryPI3. (Important)

⑤ Make sure that it is initialized.

lsmod



```
xrf111.dip.jp - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(C) ウィンドウ(W) ヘルプ(H)
root@XLX111:/release/examples/EEPROM/read# lsmod
Module                Size  Used by
ftdi_sio               32963  0
usbserial              29943  1 ftdi_sio
fuse                   99603  3
```

Make sure that the standard USB driver is displayed like a red frame.

⑥ Start AMBED.

```
# cd /root/xlxd/AMBED
# ./run
```

The file that sets the IP address of the AMBED server on the XLX v2.0.0 side is as follows.

```
# nano /etc/init.d/xlxd
```

As of November 2, 2017