D2XX for Linux

--------------

As Linux distributions vary these instructions are a guide to installation

and use. FTDI has tested the driver with Ubuntu 14.04 (kernel version 3.13)

for i386 and x86\_64, and Raspbian 7 (kernel version 3.18) for arm-v6-hf.

FTDI developed libftd2xx primarily to aid porting Windows applications

written with D2XX to Linux. We intend the APIs to behave the same on

Windows and Linux so if you notice any differences, please contact us

(see http://www.ftdichip.com/FTSupport.htm).

FTDI do not release the source code for libftd2xx. If you prefer to work

with source code and are starting a project from scratch, consider using

the open-source libFTDI.

libftd2xx uses an unmodified version of libusb (http://libusb.info) which

is distributed under the terms of the GNU Lesser General Public License

(see libusb/COPYING or http://www.gnu.org/licenses). Source code for

libusb is included in this distribution.

Installing the D2XX shared library and static library.

------------------------------------------------------

1. tar xfvz libftd2xx-<platform>-1.3.6.tgz

This unpacks the archive, creating the following directory structure:

build

libftd2xx (re-linkable objects)

libusb (re-linkable objects)

libftd2xx.a (static library)

libftd2xx.so.1.3.6 (dynamic library)

libftd2xx.txt (platform-specific information)

examples

libusb (source code)

ftd2xx.h

WinTypes.h

2. cd build

3. sudo -s

or, if sudo is not available on your system:

su

Promotes you to super-user, with installation privileges. If you're

already root, then step 3 (and step 7) is not necessary.

4. cp libftd2xx.\* /usr/local/lib

Copies the libraries to a central location.

5. chmod 0755 /usr/local/lib/libftd2xx.so.1.3.6

Allows non-root access to the shared object.

6. ln -sf /usr/local/lib/libftd2xx.so.1.3.6 /usr/local/lib/libftd2xx.so

Creates a symbolic link to the 1.3.6 version of the shared object.

7. exit

Ends your super-user session.

Building the shared-object examples.

------------------------------------

1. cd examples

2. make -B

This builds all the shared-object examples in subdirectories.

With an FTDI device connected to a USB port, try one of the

examples, e.g. reading EEPROM.

3. cd EEPROM/read

4. sudo ./read

If the message "FT\_Open failed" appears:

Perhaps the kernel automatically loaded another driver for the

FTDI USB device.

sudo lsmod

If "ftdi\_sio" is listed:

Unload it (and its helper module, usbserial), as follows.

sudo rmmod ftdi\_sio

sudo rmmod usbserial

Otherwise, it's possible that libftd2xx does not recognise your

device's Vendor and Product Identifiers. Call FT\_SetVIDPID before

calling FT\_Open/FT\_OpenEx/FT\_ListDevices.

Building the static-library example.

------------------------------------

1. cd examples/static

2. rm lib\*

Cleans out any existing libraries built for another target.

3. cp /usr/local/lib/libftd2xx.a .

4. make -B

5. sudo ./static\_link

This example demonstrates writing to, and reading from, a device with

a loop-back connector attached.

The examples show how to call a small subset of the D2XX API. The full

API is available here:

http://www.ftdichip.com/Support/Documents/ProgramGuides/D2XX\_Programmer%27s\_Guide(FT\_000071).pdf