Print Date: 30.03.2011

Linux Driver support for Brainboxes US/VX products

Brainboxes Limited, 18 Hurricane Drive, Liverpool International Business Park, Speke, Liverpool, L24 8RL, UK
Tel: +44 (0)151 220 2500 Fax: +44 (0)151 252 0446 Web: www.brainboxes.com Email: sales@brainboxes.com

Contents

1. Installation

There are 2 methods to install US/VX devices in a Linux system. Start from kernel 2.6.14, it includes a module called ftdi_sio which supports our US/VX chipset. We can use build in ftdi_sio (if standard distro) (method 1) or we can rebuild kernel or kernel module (method 2).

Method 1 is an easy method, but it's restricted to only one device (1 and 2 port device only) at a time unless you restart or unload/reload the ftdi_sio module.

Method 2 is more technical because it involves re-building the kernel module or kernel. It will also support multiple devices in the system as well as 4 and 8 port US/VX device.

1.1. Method 1 (easy method – 1 and 2 port devices only)

Use this method if you have only one US or VX device and want to use with Linux (kernel version 2.6.14 or later).

1. Open a terminal window in Linux

2. Copy "99-bbftdisio.rules" files to /etc/udev/rules.d directory.

   e.g: **sudo cp /home/TestUser/Desktop/diskimage/99-bbftdisio.rules /etc/udev/rules.d**

   where *"/home/TestUser/Desktop/diskimage/*" is the location of "99-bbftdisio.rules" file.

3. Restart your system.

4. After system restart, plug in your US or VX device.

5. Open a terminal window in Linux again.

6. Use **lsusb <*CR*>** to confirm Linux recognize the system

1.2. Method 2 (support all US/VX devices)

This method is a bit difficult but it will work with multiple devices plugged into the Linux system.

Prerequisite: Need kernel build tools on your system to build the Linux kernel or module.

To build the patch kernel module, generally you need to perform the following steps:

1. Setup Kernel development package and tools
2. Download kernel source
3. Patch ftdi_sio files. 'ftdi_sio.patch' and 'ftdi_sio_ids.patch' are to patch the Linux kernel 2.6.x ftdi_sio module source.
4. Build the kernel module.

5. Copy Module to the Linux model system and load the module.

Please follow the steps below to build the module (the following examples use Ubuntu 10.10)

1. Get the Linux kernel source.

sudo apt-get install aptitude<*CR*>

sudo apt-get install patch<*CR*>

export KERNEL_VERSION=`uname -r|cut -d '-' -f 1`<*CR*>

sudo aptitude install linux-source-$KERNEL_VERSION<*CR*>

cd /usr/src<*CR*>

sudo tar jxf linux-source-$KERNEL_VERSION.tar.bz2<*CR*>

sudo ln -s linux-source-$KERNEL_VERSION linux<*CR*>

sudo ln -s /usr/src/linux/lib/modules/`uname -r`/build<*CR*>

cd linux<*CR*>

2. Prepare the build system

sudo cp /boot/config-`uname -r` /usr/src/linux/.config<*CR*>

cd linux<*CR*>

sudo make oldconfig<*CR*>

sudo make prepare0<*CR*>

sudo make scripts<*CR*>

sudo make modules_prepare<*CR*>

sudo cp /lib/modules/2.6.35-22-generic/kernel/drivers/usb/serial/ftdi_sio.ko /lib/modules/2.6.35-22-generic/kernel/drivers/usb/serial/ftdi_sio.ko_backup<*CR*>

3. Patch and Build the 'ftdi_sio' module. 'ftdi_sio.patch' and 'ftdi_sio_ids.patch' are to patch the Linux kernel 2.6.x ftdi_sio module source. The following example show building a single 'ftdi_sio' module. You can also rebuild the whole kernel system.

sudo patch /usr/src/linux/drivers/usb/serial/ftdi_sio.c < /home/bandu/Downloads/ftdi_sio.patch<*CR*>

sudo patch /usr/src/linux/drivers/usb/serial/ftdi_sio_ids.h < /home/bandu/Downloads/ftdi_sio_ids.patch<*CR*>

sudo make -C /lib/modules/`uname -r`/build M=`pwd` drivers/usb/serial/ftdi_sio.ko<*CR*>

4. Copy Module to the Linux model system and load the module.

sudo cp /usr/src/linux/drivers/usb/serial/ftdi_sio.ko /lib/modules/2.6.35-22-generic/kernel/drivers/usb/serial/<*CR*>

sudo rmmod ftdi_sio<*CR*>

sudo modprobe ftdi_sio<*CR*>

5. (Optional) Prepare module to load every time at start up. You need to append 'ftdi_sio' (exclude "") string at /etc/modules file.

2. Checking US/VX in Linux System

We used ftdi_sio kernel module as US/VX card in Linux system.

To check the device load up use **lsusb<*CR*>** command. Check the device ID against device ID section.

Use **dmesg | grep FTDI<*CR*>** to check ftdi_sio module has loaded up properly.

Use **ls /dev/ttyUSB*<*CR*>** to list serial ports. Serial port will load up as ***ttyUSBx*** where x is the index of the serial port.

3. 422/485 Duplex Mode Setting

RS422/485 US and VX devices use DTR line to switch between duplex mode. To change the duplex mode settings in RS422/485 serial port, you need to set the DTR line when opening the RS422/485 serial port.

Full duplex => DTR = true or high or 1

Half duplex => DTR = false or low or 0

4. US/VX Device ID List

Brainboxes Vendor ID is **0x05D1**.

Code	Description	Device ID
US-101	USB 1xRS232	0x1011
US-324	USB 1xRS422/485 1Mbaud	0x1013
US_357	USB 1xRS232/422/485	0x7001
US-257	USB 2xRS232 1Mbaud	0x5001
US-313	2xRS422/485 1Mbaud	0x6001

US-701	USB 4xRS232 1Mbaud	0x2011 0x2012
US-346	USB 4xRS422/485 1Mbaud	0x3011 0x3012
US-279	USB 8xRS232 1Mbaud	0x2021 0x2022 0x2023 0x2024
US-842	USB 8xRS422/485 1Mbaud	0x8001 0x8002 0x8003 0x8004
US-160	USB 16xRS232 1Mbaud	0x9001 0x9002 0x9003 0x9004 0x9005 0x9006 0x9007 0x9008
US-606	6 Port RS232 USB to Serial	0x2001 0x2002 0x2003
VX-001	ExpressCard 1 Port RS232	0x1001
VX-012	ExpressCard 2 Port RS232	0x1002
VX-023	ExpressCard 1 Port RS422/485	0x1003
VX-034	ExpressCard 2 Port RS422/485	0x1004