xpad - Linux USB driver for X-Box gamepads

This is the very first release of a driver for X-Box gamepads.
Basically, this was hacked away in just a few hours, so don't expect
miracles.

In particular, there is currently NO support for the rumble pack.
You won't find many ff-aware linux applications anyway.

0. Notes

Driver updated for kernel 2.6.17.11. (Based on a patch for 2.6.11.4.)

The number of buttons/axes reported varies based on 3 things:
- if you are using a known controller
- if you are using a known dance pad
- if using an unknown device (one not listed below), what you set in the module configuration for "Map D-PAD to buttons rather than axes for unknown pads" (module option dpad_to_buttons)

If you set dpad_to_buttons to 0 and you are using an unknown device (one not listed below), the driver will map the directional pad to axes (X/Y), if you said N it will map the d-pad to buttons, which is needed for dance style games to function correctly. The default is Y.

dpad_to_buttons has no effect for known pads.

0.1 Normal Controllers

With a normal controller, the directional pad is mapped to its own X/Y axes.
The jstest-program from joystick-1.2.15 (jstest-version 2.1.0) will report 8
axes and 10 buttons.

All 8 axes work, though they all have the same range (-32768..32767)
and the zero-setting is not correct for the triggers (I don't know if that
is some limitation of jstest, since the input device setup should be fine. I
didn't have a look at jstest itself yet).

All of the 10 buttons work (in digital mode). The six buttons on the
right side (A, B, X, Y, black, white) are said to be "analog" and
report their values as 8 bit unsigned, not sure what this is good for.

I tested the controller with quake3, and configuration and
in game functionality were OK. However, I find it rather difficult to
play first person shooters with a pad. Your mileage may vary.

0.2 Xbox Dance Pads

When using a known dance pad, jstest will report 6 axes and 14 buttons.

For dance style pads (like the redoctane pad) several changes
have been made. The old driver would map the d-pad to axes, resulting
in the driver being unable to report when the user was pressing both

left+right or up+down, making DDR style games unplayable.

Known dance pads automatically map the d-pad to buttons and will work correctly out of the box.

If your dance pad is recognized by the driver but is using axes instead of buttons, see section 0.3 - Unknown Controllers

I've tested this with Stepmania, and it works quite well.

0.3 Unknown Controllers

If you have an unknown xbox controller, it should work just fine with the default settings.

HOWEVER if you have an unknown dance pad not listed below, it will not work UNLESS you set "dpad_to_buttons" to 1 in the module configuration.

PLEASE, if you have an unknown controller, email Dom <binary1230@yahoo.com> with a dump from /proc/bus/usb and a description of the pad (manufacturer, country, whether it is a dance pad or normal controller) so that we can add your pad to the list of supported devices, ensuring that it will work out of the box in the future.

1. USB adapter

Before you can actually use the driver, you need to get yourself an adapter cable to connect the X-Box controller to your Linux-Box. You can buy these online fairly cheap, or build your own.

Such a cable is pretty easy to build. The Controller itself is a USB compound device (a hub with three ports for two expansion slots and the controller device) with the only difference in a nonstandard connector (5 pins vs. 4 on standard USB connector).

You just need to solder a USB connector onto the cable and keep the yellow wire unconnected. The other pins have the same order on both connectors so there is no magic to it. Detailed info on these matters can be found on the net ([1], [2], [3]).

Thanks to the trip splitter found on the cable you don't even need to cut the original one. You can buy an extension cable and cut that instead. That way, you can still use the controller with your X-Box, if you have one ;)

2. Driver Installation

Once you have the adapter cable and the controller is connected, you need to load your USB subsystem and should cat /proc/bus/usb/devices. There should be an entry like the one at the end [4].

Currently (as of version 0.0.6), the following devices are included:

 original Microsoft XBOX controller (US), vendor=0x045e, product=0x0202
 smaller Microsoft XBOX controller (US), vendor=0x045e, product=0x0289
 original Microsoft XBOX controller (Japan), vendor=0x045e, product=0x0285
 InterAct PowerPad Pro (Germany), vendor=0x05fd, product=0x107a
 RedOctane Xbox Dance Pad (US), vendor=0x0c12, product=0x8809

The driver should work with xbox pads not listed above as well, however
you will need to do something extra for dance pads to work.

If you have a controller not listed above, see 0.3 - Unknown Controllers

If you compiled and installed the driver, test the functionality:
> modprobe xpad
> modprobe joydev
> jstest /dev/js0

If you're using a normal controller, there should be a single line showing
18 inputs (8 axes, 10 buttons), and its values should change if you move
the sticks and push the buttons. If you're using a dance pad, it should
show 20 inputs (6 axes, 14 buttons).

It works? Voila, you're done ;)

3. Thanks

I have to thank ITO Takayuki for the detailed info on his site
 http://euc.jp/periphs/xbox-controller.ja.html.

His useful info and both the usb-skeleton as well as the iforce input driver
(Greg Kroah-Hartmann; Vojtech Pavlik) helped a lot in rapid prototyping
the basic functionality.

4. References
----

1. http://euc.jp/periphs/xbox-controller.ja.html (ITO Takayuki)
2. http://xpad.xbox-scene.com/
3. http://www.xboxhackz.com/Hackz-Reference.htm

4. /proc/bus/usb/devices - dump from InterAct PowerPad Pro (Germany):

T: Bus=01 Lev=03 Prnt=04 Port=00 Cnt=01 Dev#= 5 Spd=12 MxCh= 0
D: Ver= 1.10 Cls=00(>ifc) Sub=00 Prot=00 MxPS=32 #Cfgs= 1
P: Vendor=05fd ProdID=107a Rev= 1.00
C:* #Ifs= 1 Cfg#= 1 Atr=80 MxPwr=100mA
I: If#= 0 Alt= 0 #EPs= 2 Cls=58(unk.) Sub=42 Prot=00 Driver=(none)
E: Ad=81(I) Atr=03(Int.) MxPS= 32 Ivl= 10ms
E: Ad=02(O) Atr=03(Int.) MxPS= 32 Ivl= 10ms

5. /proc/bus/usb/devices - dump from Redoctane Xbox Dance Pad (US):

T: Bus=01 Lev=02 Prnt=09 Port=00 Cnt=01 Dev#= 10 Spd=12 MxCh= 0
D: Ver= 1.10 Cls=00(>ifc) Sub=00 Prot=00 MxPS= 8 #Cfgs= 1

```
P:  Vendor=0c12 ProdID=8809 Rev= 0.01
S:  Product=XBOX DDR
C:* #Ifs= 1 Cfg#= 1 Atr=80 MxPwr=100mA
I:  If#= 0 Alt= 0 #EPs= 2 Cls=58(unk. ) Sub=42 Prot=00 Driver=xpad
E:  Ad=82(I) Atr=03(Int.) MxPS=  32 Ivl=4ms
E:  Ad=02(O) Atr=03(Int.) MxPS=  32 Ivl=4ms
```

--

Marko Friedemann <mfr@bmx-chemnitz.de>
2002-07-16
 - original doc

Dominic Cerquetti <binary1230@yahoo.com>
2005-03-19
 - added stuff for dance pads, new d-pad->axes mappings