Xbox 360 controller demo

Generated by Doxygen 1.8.6

Wed Jun 24 2015 15:24:44

Contents

Chapter 1

Course: DES (EL32), Assignment 10

Author

A.W. Janisse

Assignment description

The main goal of this application is to explore how specific USB hardware can be controlled. For this goal a Datel Xbox 360 controller will be used. Although it is not an original Microsoft one it should be 100% compatible. Below an image of this controller is given.

The software developed for this assignment is using the **libusb** library. This library provides an abstraction for communication with USB devices. The assignment has the following requirements:

- The program can show the states of all the buttons;
- The program can controll the LED's;
- The program can controll the rumble actuator;
- The program can be build with a Makefile;
- The program can run on the Raspberry Pi;
- The software is documentated.

Building the source

Please refer to the **Functional description of the xbcweb module** (p. **??**) page for more information on this topic.

How it all works

On the page3 page a detailed description is given on this topic.

Extra's

During this assignment I got excited about **Doxygen**. Dogygen is a tool which can be used to produce documentation in broad sense. One of the possibilities is to generate documentation from comments in source file. The webpage you are reading right now is created with Doxygen.

Chapter 2

Bug List

File buttons.h (p. **??**)

No known bugs.

Chapter 3

Module Index

3.1 Modules

Here is a list of all modules:

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 5

File Index

5.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 6

Module Documentation

6.1 Functional description of the xbcdeamon module

6.2 Functional description of the xbcweb module

6.2.0.1 Functional description of the xbcweb module

6.3 Functions common for xbcdeamon and xbcweb

Data Structures

- struct **button**

Typedefs

- typedef **button buttons** [MAX_DEVS]

6.3.1 Detailed Description

6.3.2 Typedef Documentation

6.3.2.1 typedef button buttons[MAX_DEVS]

An array representing the ....

6.4 Functions used by the xbcdeamon module

6.5 Functions used by the xbcweb module

Chapter 7

Data Structure Documentation

7.1 button Struct Reference

```
#include <buttons.h>
```

Data Fields

- bool **avail**

  Bit to track if controller is availlable.
- bool **D_UP**

  D-Pad up.
- bool **D_DN**

  D-Pad down.
- bool **D_LEFT**

  D-Pad left.
- bool **D_RIGHT**

  D-pad right.
- bool **START**

  Start button.
- bool **BACK**

  Back button.
- bool **LS_PRESS**

  Left stick press.
- bool **RS_PRESS**

  Right stick press.
- bool **LB**

  Button LB.
- bool **RB**

  Button RB.
- bool **LOGO**

  Xbox logo button.
- bool **SPARE**

  Unused.
- bool **A**

  Button A.
- bool **B**

  Button B.

- bool **X**

  Button X.
- bool **Y**

  Button Y.
- uint8_t **Left_trigger**

  Left trigger. Produces a value from 0 to 255.
- uint8_t **Right_trigger**

  Right trigger. Produces a value from 0 to 255.
- int16_t **Left_stick_X**

  Left joystick x-value. Produces a value from -32768 to 32767.
- int16_t **Left_stick_Y**

  Left joystick y-value. Produces a value from -32768 to 32767.
- int16_t **Right_stick_X**

  Right joystick x-value. Produces a value from -32768 to 32767.
- int16_t **Right_stick_Y**

  Right joystick y-value. Produces a value from -32768 to 32767.

7.1.1 Detailed Description

A structure representing the ....

The documentation for this struct was generated from the following file:

- common/**buttons.h**

7.2 command Struct Reference

Data Fields

- uint8_t **id**
- char **cmd** [MAX_CMD_LEN]
- int16_t **val**

The documentation for this struct was generated from the following file:

- common/commands.h

7.3 device Struct Reference

Data Fields

- int8_t **bus**
- int8_t **address**
- libusb_device_handle ∗ **handle**

The documentation for this struct was generated from the following file:

- xbcdaemon/devices.c

Chapter 8

File Documentation

8.1 common/buttons.h File Reference

Implementation for the Devices.

```
#include <stdbool.h>
#include "common.h"
```

Data Structures

- struct **button**

Typedefs

- typedef **button buttons** [MAX_DEVS]

8.1.1 Detailed Description

Implementation for the Devices.

Author

A.W Janisse

Bug No known bugs.

Version

1.0 First release.