

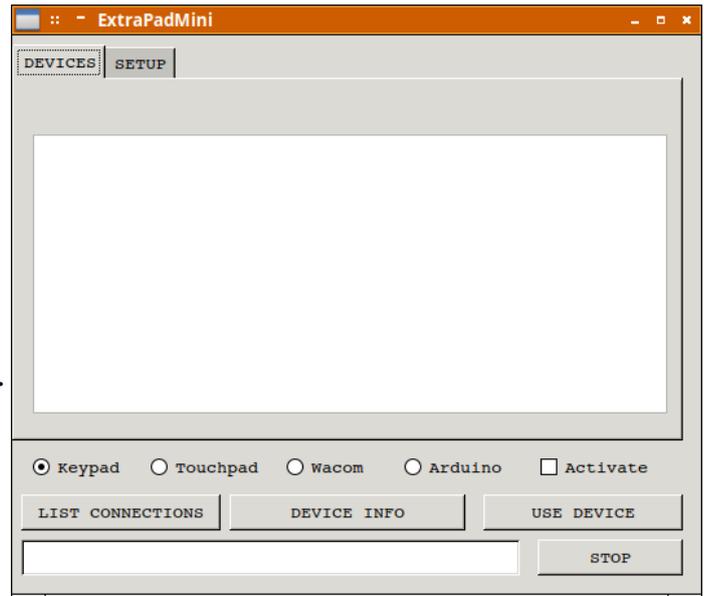
ExtraPadMini

ExtraPadMini is written with Gambas3 runs on most 64bit Linux OS.

To be able to use some of the devices there is root permission needed.

This program is a minimal setup for using external pads or serial connection through usb ports on your pc. The generated data by your external device can be used to execute program action code in the <choosendevic>_action sub. This action is only performed when the 'Activate' checkbox is checked.

Otherwise you get the generated data from the chosen device in the textbox field.



Before you can create code and use the touchpad or wacompad or Arduino you need to put values into the 'SETUP' tab.

Touchpad Settings and Wacompad Settings have the maximum x-axes value (horizontal upper left corner to the right) have the maximum y-axes value (vertical upper left corner to lower left corner)

With the COLS and ROWS you can divide your pad into a number of squares to imitate as if you were using keypad buttons.

Each new button can be connected to some code you write in the <selecteddevice>_action sub.

Serial Settings contain the classic settings for a Arduino Uno serial connection.

You have to change to your serial board communication protocol values.

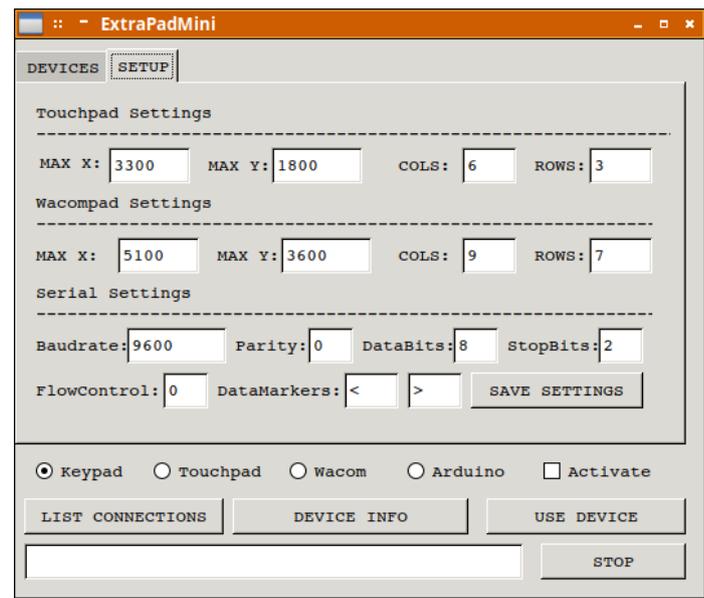
I have written an Arduino program which sends data which is marked with a "<" to start and ">" to end data. (Arduino IR to read from remote control)

I have put values in the settings which work for my devices on my laptop touchpad and the Wacom Volito 2 pad and Arduino Uno & Nano.

New settings have to be saved by clicking on the 'SAVE SETTINGS' button. These will be saved in the *home/<user>/config/gambas3/ExtraPadMini.conf*

At first use you'll need to save your settings first once as the config file does not exist yet.

The program will run without the config file which let's you figure out what max x,y values are given by the touchpad or wacompad so you can adjust the settings values and save them.



SELECTING A DEVICE

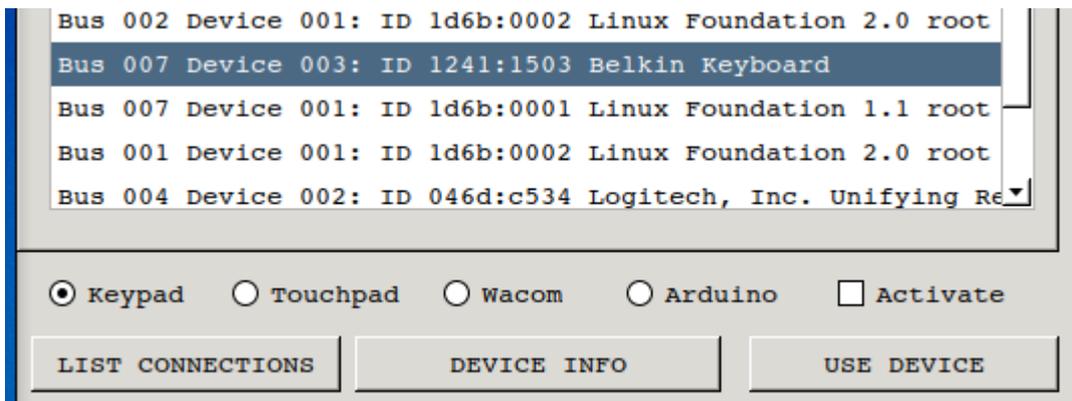
1. Select a device you want to use

As you see there are four devices you can select from. Only one can be selected. Note that you always have to connect before starting the program. If you want to change, better stop the program and disconnect and reconnect if necessary. Select the **radiobutton** for your device.

The Arduino is connected through usb but emulates an serial connection.

If you do not have the Arduino Ide installed you will need to add the 'dialout' group and also add your 'user' to the group once.

2. Select 'LIST CONNECTIONS'

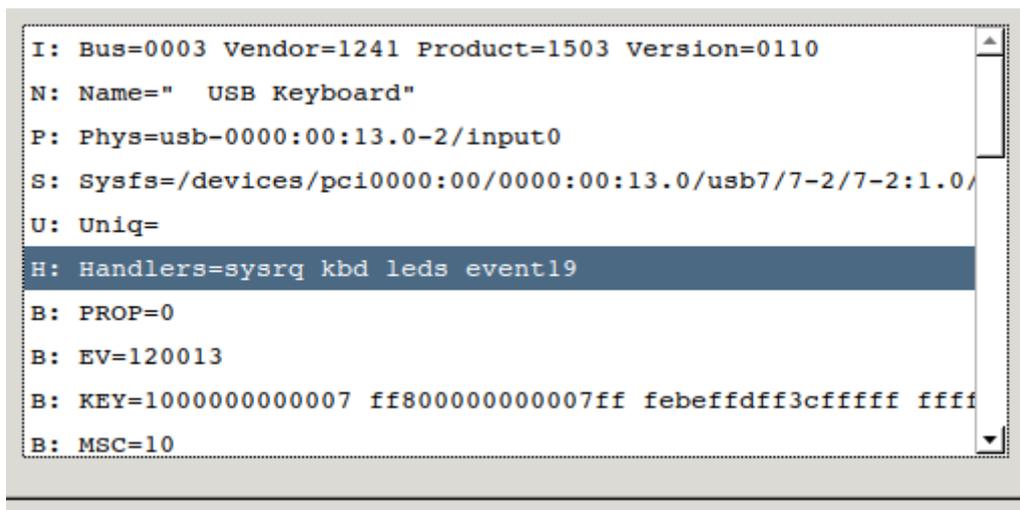


A list with the usb devices is generated in the lstINFO listbox.

Select your device in that list

- e.g An external Keypad is connected: Belkin Keyboard. In that case this was given by the internal 'lsusb' shell command.

3. Select 'DEVICE INFO'



Information about the device, name and what eventfile is associated with your selected device .

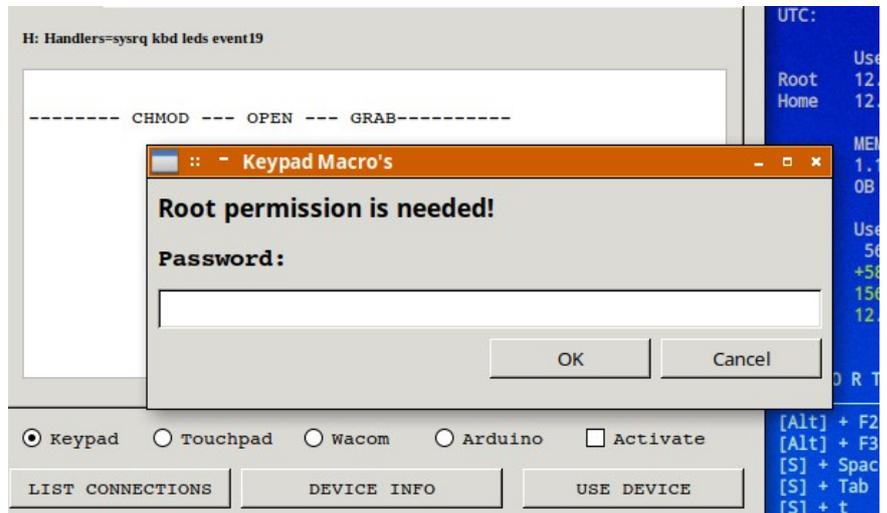
If the Arduino is selected this button is disabled as all info needed to connect is given in the setup.

If an external keypad is used it is necessary to select corresponding H: Handlers event in the list. e.g Belkin Keyboard the N: Name is not the same as shown in the image. It is also possible that multiple keyboards are shown. In most cases the 'kbd leds eventX' is the one to use for an external keypad.

4. Select 'USE DEVICE'

Root permission is needed to be able to use the usb `dev/input/event` -file. Commit your password with 'OK'

The Arduino does not need this as the <user> belongs to the dialout group and has permission.

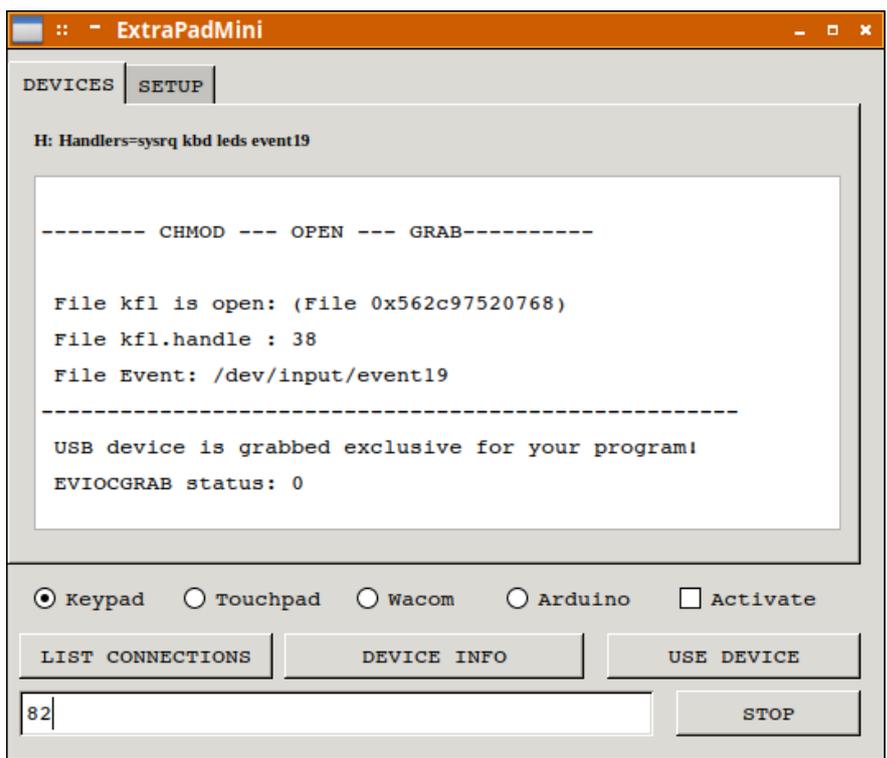


This is the message you get with the information about eventfile, handler, and grab status.

Your device is now grabbed to be used only by your program.

All actions from the device will be caught by the ExtraPadMini program .

When the Activate checkbox is checked then all actions will be performed in the window that is active or mouse pointer is pointed at accordingly your written code in the selecteddevice_action sub.



e.g Belkin Keyboard : the '0' button has generated the code : '82'

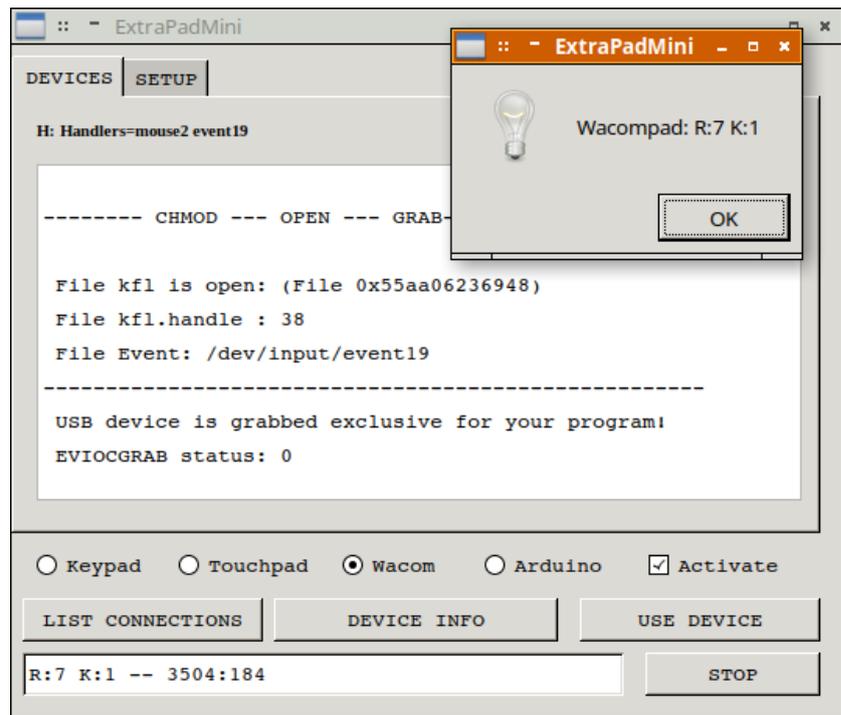
As the checkbox 'Activate' is true the code is performed in 'keypad_action' sub. The code in the program only gives a message box with the data from the keypad but can be exploited for each possible data – you can call with the shell command, gambas scripts , xdotool commands, bash scripts, system calls etc... and redirect output into the active programs you want. This gives you the opportunity to automate a series of handlings with a single input.

Wacom pad Volito2

Here you see the result of the 'Activate'

In the messagebox the row and col values as calculated from the settings .

In the textbox field you see also the Y-axis value and X-axis value.



Arduino Nano Clone

You need to select the device in the listbox.

In this case it uses the *dev/ttyACM0* port

Some Arduino devices use *dev/ttyUSBx* port

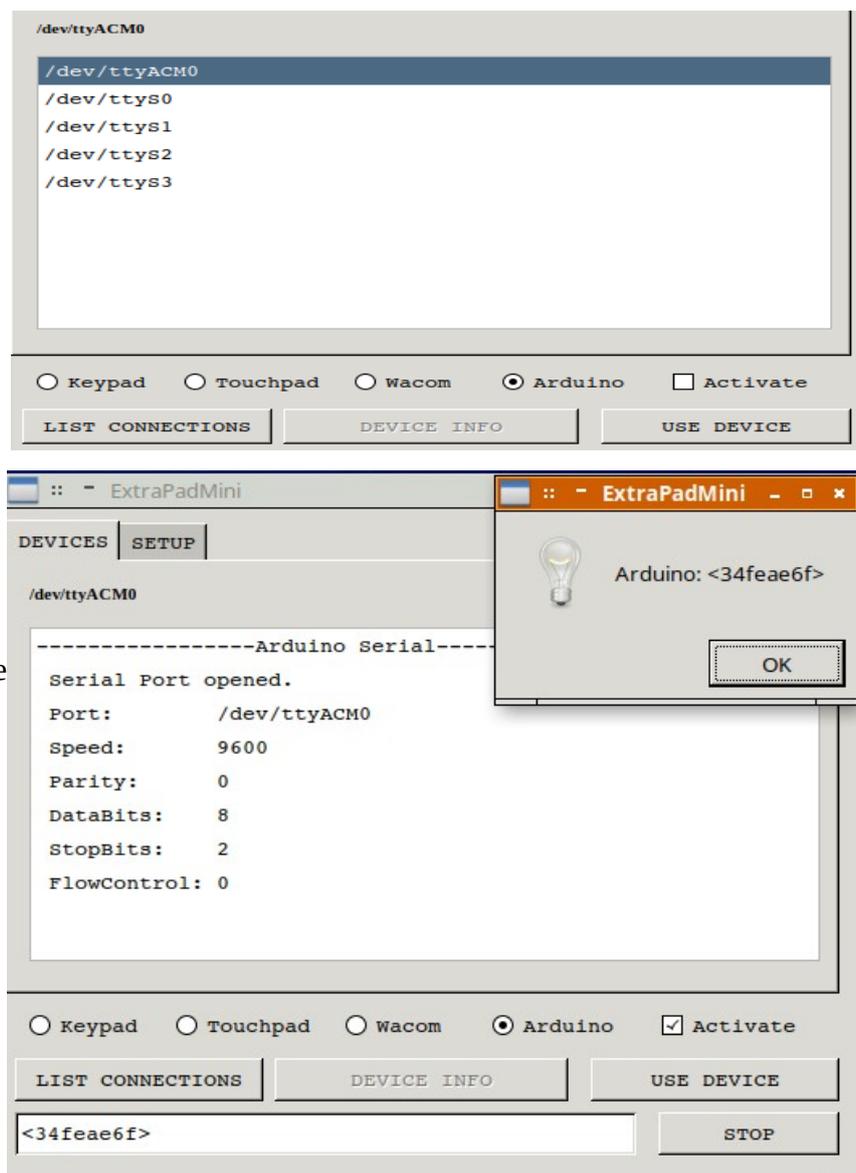
Here you see the output from the 'OK' button on the IR remote control and Arduino.

I put the IR data pulse between two markers as given in my SETUP setting, "<" and ">" to mark 8 byte data

I have a universal remote control and some others at home who send 'FFFFFFF' as end pulse.

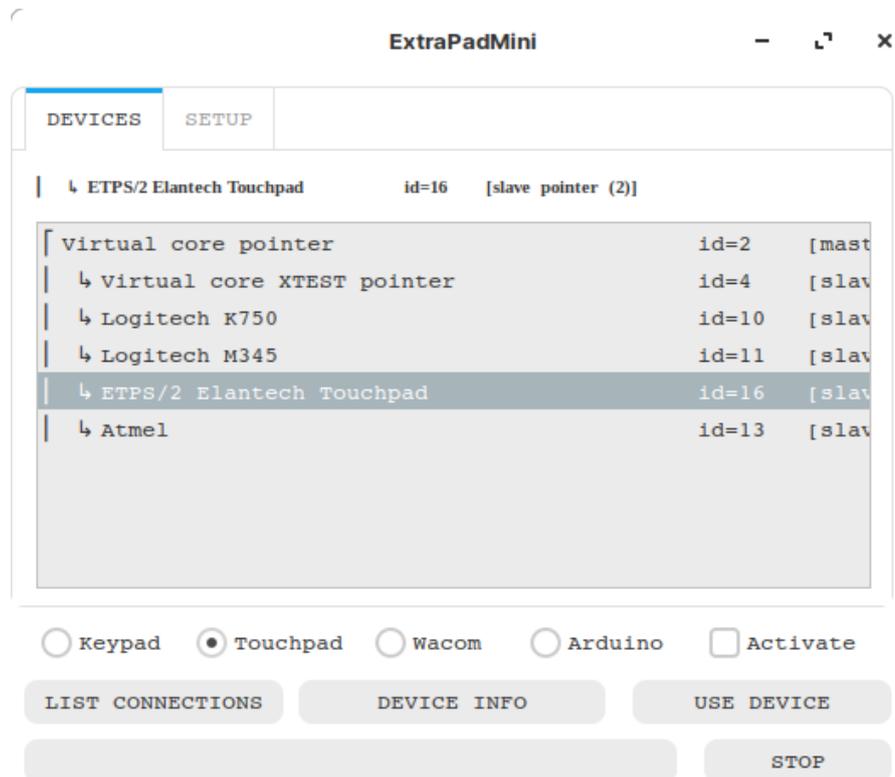
Some remote controls send more than one code before sending a finish sending pulse. In this case a change in the coding will be necessary.

You also see the data in the messagebox as 'Activate' is checked.



Touchpad

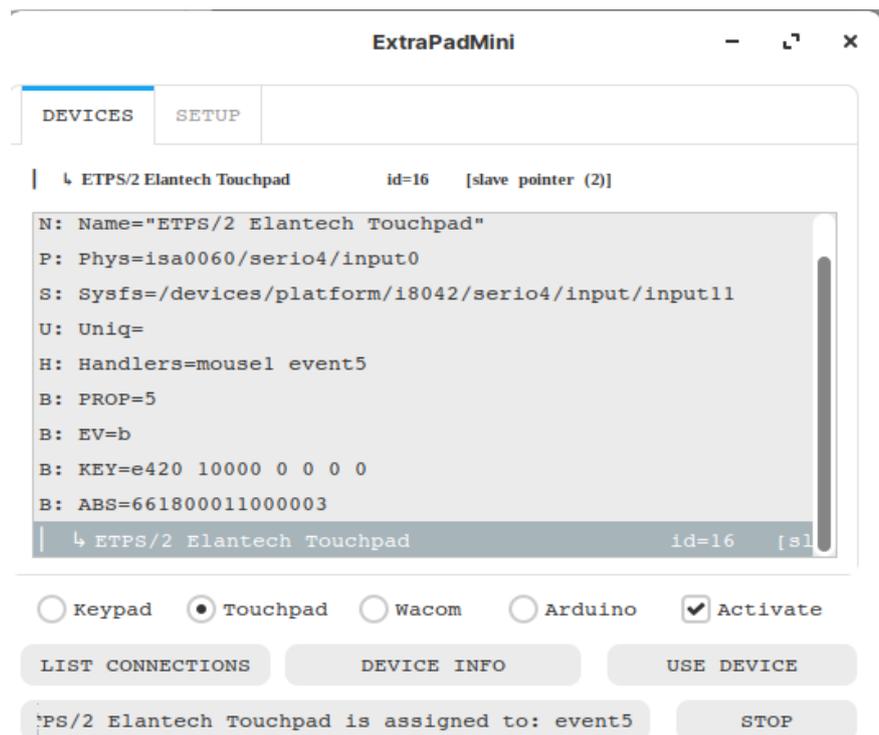
Here is my touchpad on the laptop known as a Elantech Touchpad.
Select this pad in the listbox.



DEVICE INFO shows in the listbox the Handlers which is a mouse1 event5 file assigned to the touchpad usb port. At the bottom of the list you see the name again . Select that name again en continue with 'USE DEVICE'

Your password for root permisson will be asked.

After that you can use the touchpad and you'll get the row and column and y,x coordinates too. On Activate a messagebox from in File_read() call to touchpad_action sub will be opened





Here you see how I adapted my touchpad with a cardboard and tape on one side. The cardboard holes makes it easy , the thickness allows to touch and not generate a signal this gives a clear separation and distincting and feel where you can tap/touch.

I use this to subtitle small videos with Subtitle Composer, some key shortcuts and multiple keystrokes I have now in one fingerinput , without having to lift my arms or hands away from the laptop. Makes it much more productive and time saving.

Next to it on the right a Arduino clone with a IR sensor stuck on it. Wrote the code to intercept the ir remote pulses and use to browse and look at VLC or Youtube browser videos, start,stop,pauze, full screen all with my remote control (from my bed) – and shutdown pc.

Next to the right also a Arduino Nano in a vitamin tube with a keyboard attached and also IR sensor, use it as an extension and put remote control in front above keyboard so I have again extra programmable buttons in keyboard reach.

Next to it on the right the Wacompad with a paper table grid taped on it . The Wacom needs to be tapped with the pen . This is my latest reuse , you can put words, symbols or icons in the table with your texteditor accordingly the program you want to automate.

Hope you can make good use of the little program in your daily routines.

A executable is also in the zip file.

This program runs with Gambas 3.9, 3.12 and 3.15 64 bit linux OS

Executable runs on 32 bit LinuxOs but the ioctl commands do not function. (8bit <> 16bit I suppose)

Have fun coding

Greetings

dinge