Project Title: Device Driver for BH1750VI a digital ambient light Sensor on Raspberry Pi.

BH1750 is a Light Intensity Module which works using I2C protocol. The work here presents an I2C device driver to interface this sensor to Raspberry pi.

Summary:

A kernel module is written for BH1750VI a digital ambient light sensor IC for I2C bus interface on Raspberry Pi. This IC is the most suitable to obtain the ambient light data for adjusting LCD and Keypad backlight power of Mobile phone. In our project we are writing a kernel space code which uses i2c bus to read values from the sensor and it will be displayed on the kernel log. Although there are several Techniques to handle an i2c device (Ex: IIO, bus_register_info, sysfs), in this code we have tried to implement using IIO Subsystem and i2c device instantiation.

The method we are using for I2C instantiation is "instantiate the device explicitly". This method is appropriate when a larger device uses an I2C bus for internal communication. This is done by filling a struct i2c_board_info and calling i2c_new_device(). Suppose we don't know for sure whether I2C device is present or not, or it may have different addresses from one board to the next. So in this case, we are calling i2c_new_probed_device() instead of i2c_new_device().

The Industrial I/O core offers a unified framework for writing drivers for many different types of embedded sensors. An IIO device usually corresponds to a single hardware sensor and it provides all the information needed by a driver handling a device. A typical IIO driver will register itself as an I2C or SPI driver and will create two routines, probe and remove.

Hardware Design:

Please Note that we have used Raspberry pi 2 model B V1.1 with Raspbian installed.

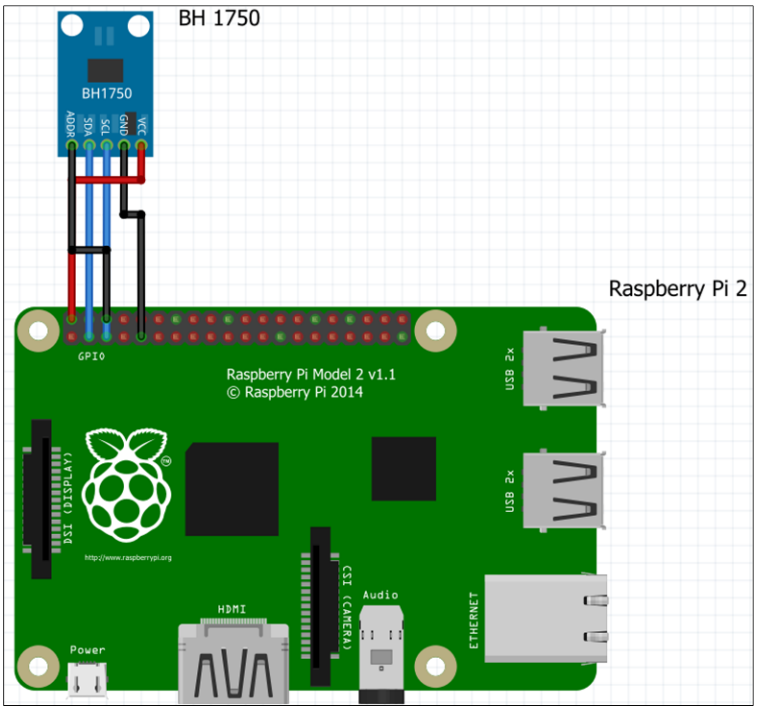
Requirements:

- Raspberry pi 2 model B Board
- Memory Card
- Raspbian Jessie (Preferred)
- Linux Header installed and compatible with the kernel version
- BH1750VI Light Intensity Module
- Connecting Wires
- Raspberry pi Datasheet

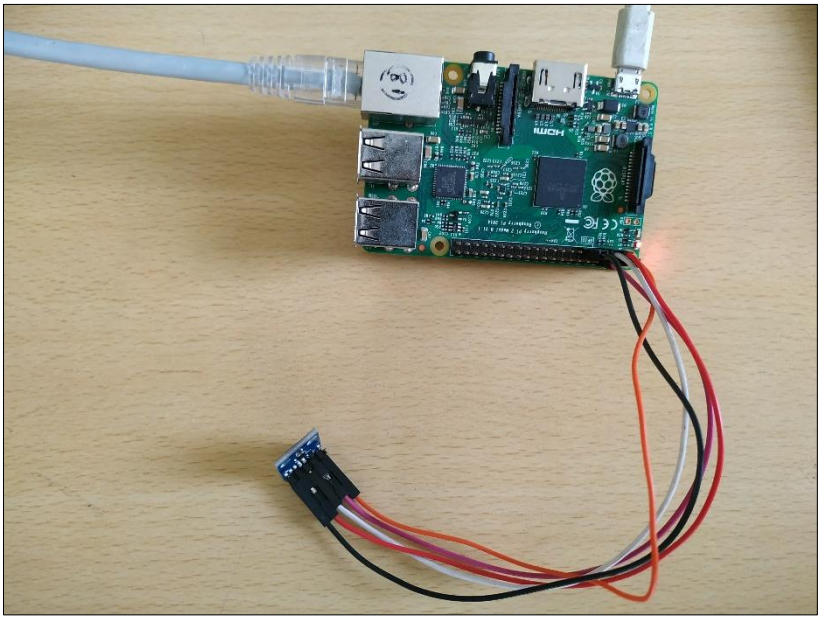
Connections:

- Connect Vcc of Sensor to 5V pin on Raspberry pi
- Connect SCL to SCL pin on board
- Connect GND to Ground pin on board
- Connect SDA to SDA pin on board
- Connect ADDR to Ground pin on board

Schematic:



Actual Photograph:



Initial Testing of Sensor:

Once the Connections are complete, initial testing involves detecting the address of this Sensor. For that you may have to go through some Steps:

- On the Terminal run $ lsmod
- Please do $sudo apt-get install raspberrypi-kernel-headers which will create a build directory inside kernel header. (Important for make)
- If i2c_dev appears in the list go to step 4

- If it does not appear run $modprobe i2c_dev and check again using lsmod command
- Assuming that i2c_dev driver is present run $ i2cdetect –y 1
- This command should display the slave address of this sensor which is 0x23.

Code Compilation Procedure:

- Download the Kernel Level code "i2c.c", "sensor.c" and Makefile .
- Go to the Directory where these files are downloaded.
- Run $ make all on terminal
- This should generate i2c.ko file successfully
- Run the command $sudo make install
- Run the command $sudo depmod –a which thoroughly probes and examines all modules in the kernel
- Insert the module using $ sudo insmod i2c.ko
- Run the command $sudo modprobe sensor to insert loadable kernel module into linux kernel as an IIO device
- Now the sensor device file should be created under /sys/bus/iio/iio:device*X*/ and /dev/iio:device*X*
- Run the command $cat /sys/bus/iio/devices/iio:device0/in_illuminance_raw to read the contents of the file in_illuminance_raw on the terminal which is nothing but the light intensity sensor reading.
- To Remove the module run $rmmod i2c.ko

Possible Errors:

- **Error while running i2cdetect –y 1 "could not open /dev/i2c-1" No such file or Directory**

Possible Cause: i2c_Dev not in Kernel Module
Solution: Modprobe should be able to insert the module, However if that does not happen you may have to check with the blacklisted device .
Follow this link: http://www.runeaudio.com/forum/how-to-enable-i2c-t1287.html
Another reference can be Exploring Raspberry Pi by Derek Molloy chapter -8

- **Error while Insmod ,Build directory not found**

Cause: Linux headers are not installed
Solution: Check the version you are running on by $uname –r
Download the Respective Header file from this link: https://www.niksula.hut.fi/~mhiienka/Rpi/linux-headers-rpi/
Once headers are downloaded install it using command:
sudo dpkg -i linux-headers-"Depending on ur version"armhf.deb
eg sudo dpkg -i linux-headers-4.1.19+_4.1.19+-2_armhf.deb
you may have to check with the errors and possible solutions provided in terminal itself and run those commands .