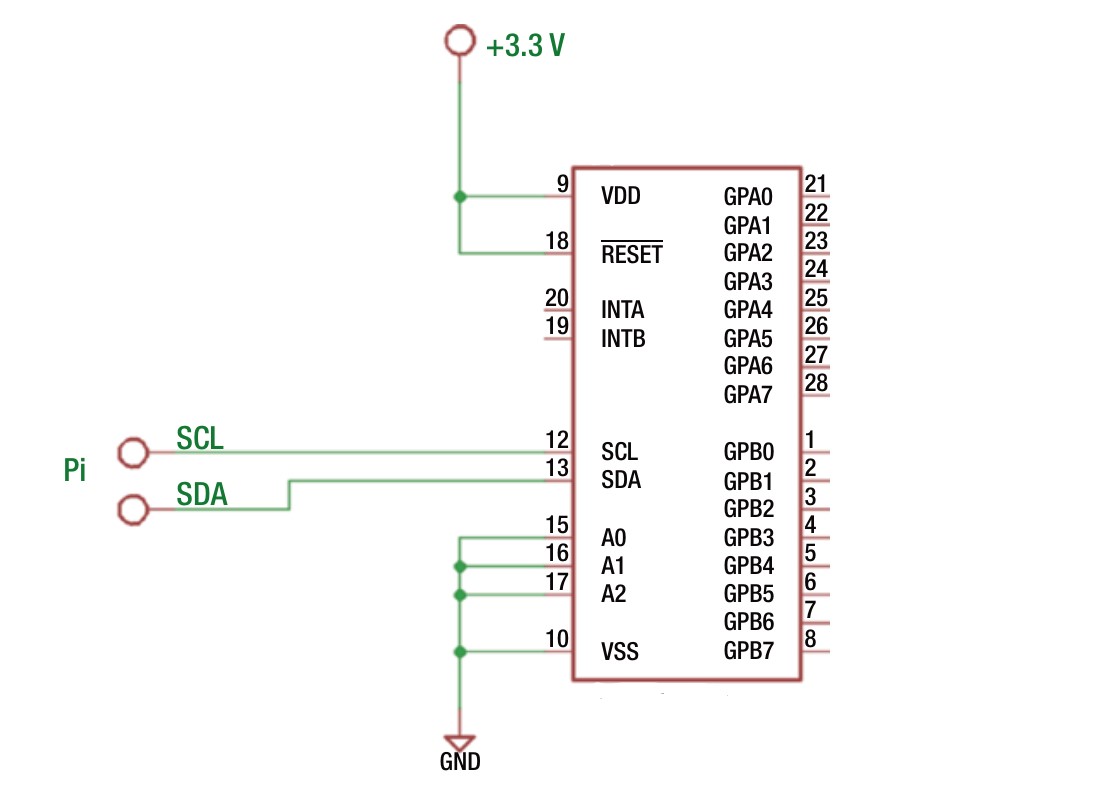
* **Title :** expansion of GPIOs of Raspberry pi using MCP23017

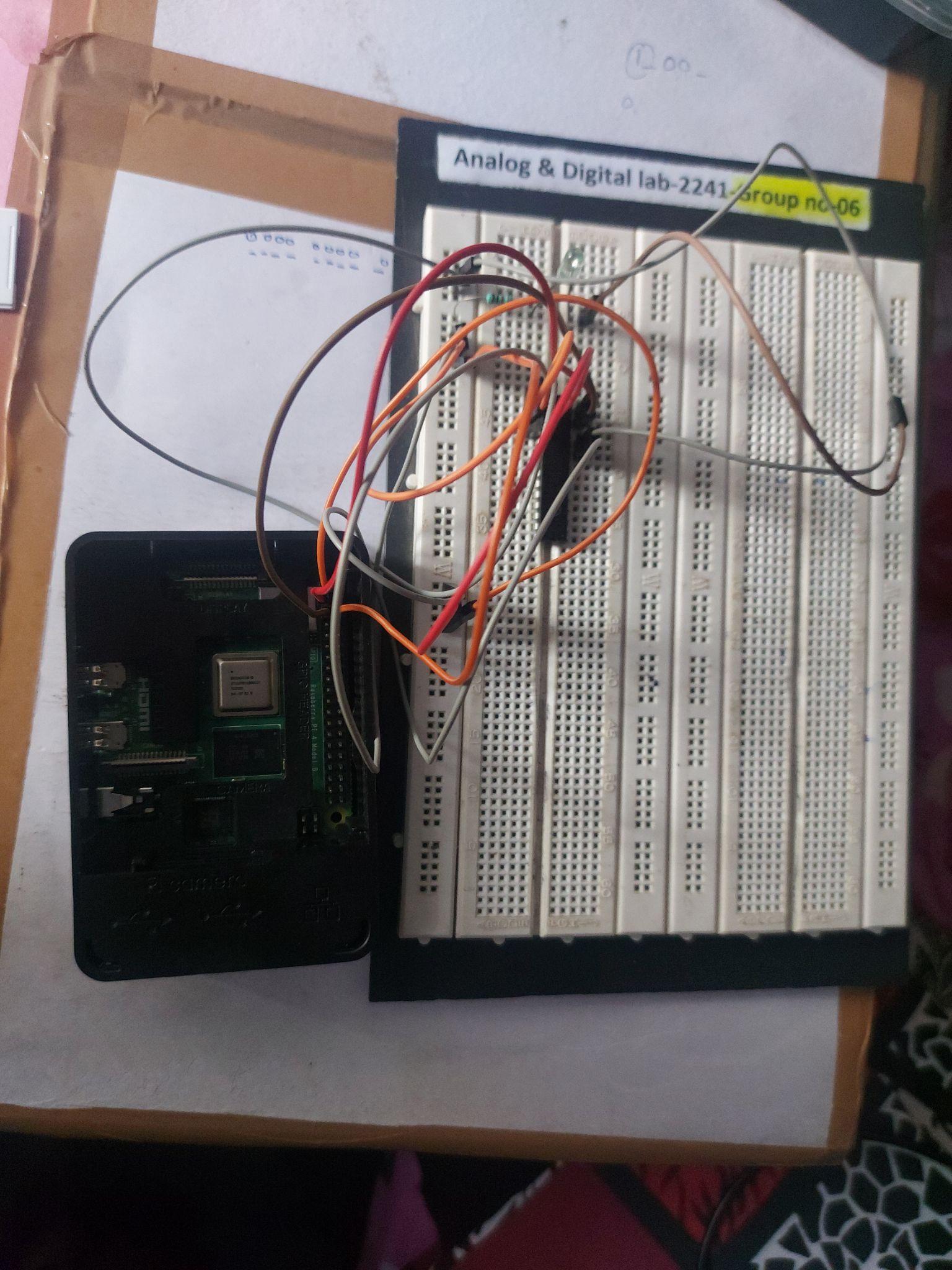
**Summary :** MCP23017 is a 16-Bit I/O Expander with Serial Interface. It has 16 Gpio pins which can be read or written by using i2c interface. These pins can be accessed by i2c standard commands directly. This driver lets the linux kernel find the mcp chip as GPIO pin. The pins can be exported from user space itself like a normal gpio pin of Rpi.

* **Hardware :**

**Schematic**

****

**Actual Hardware :**

****

**Kernel Space Driver and it’s build process**

1. **$ make**
2. **$ sudo insmod final\_proc.ko**

1. **$ echo 12 >/sys/class/gpio/export**

We will now see a gpio12 device appear under /sys/class gpio that was not there before:

1. **$ ls /sys/class/gpio**

export\* gpio24@ gpiochip0@ gpiochip100@ gpiochip504@ unexport\*

This is a symbolic link to a directory which has a number of files in it:

1. **$ ls /sys/class/gpio/gpio12/**

active\_low\* device@ direction\* edge\* power/ subsystem@ uevent\* value\*

Step 4 is to set the pin to be either an input or an output. This is done by writing either "in", or "out" to the direction file we saw above. For example, to set gpio24 as an input we would do:

1. **$ echo in >/sys/class/gpio/gpio12/direction**

Or to set it as an output:

1. **$ echo out >/sys/class/gpio/gpio12/direction**

The file can be read back if you want to check the current status:

1. **$ cat /sys/class/gpio/gpio12/direction**

Output : out

While this may not be true for all hardware, a GPIO pin will generally default to being an input as this is always safe to do at the hardware level.

If we are configuring an output pin, we can now set it's level. You write the value 0 or 1 (corresponding to low or high) to the value file for the pin. Continuing our example this could be done with:

1. **$ echo 1 >/sys/class/gpio/gpio12/value**

to set it high

1. **$ cat /sys/class/gpio/gpio12/value**
2. **$ echo 12 >/sys/class/gpio/unexport**

After doing this, the entry for gpio12 will no longer appear in sysfs: