**LAB 1-2 DISPLAYING TEXTS ON I2C LCD DISPLAYS**

|  |  |
| --- | --- |
| Description | In this lab you will display various information on the I2C LCD Display. |
| What you will learn | * How to communicate with I2C LCD Libraries * How to display the time and IP Address of your Pi on the LCD display * How to get live stock market quotes and display it |
| What you need | * I2C LCD Display |
| Duration | 45 minutes |

**Connecting your LCD Display to your Pi**

* Connect VCC of LCD Display to 5V on Power Supply Channel on Breadboard
* Connect GND of LCD Display to GND on Power Supply Channel on Breadboard
* Connect SDA of LCD Display to #SDA on Pi Cobbler
* Connect SCL of LCD Display to #SCL on Pi Cobbler

**Enabling the SPI interface**

1. Open **Terminal** and enter the following command:

sudo raspi-config

1. Select **5. Interfacing Options**
2. Select **P4 SPI**:
3. Select **Yes** for the SPI interface to be enabled.
4. Press **Ok** for message showing that The SPI interface is enabled.
5. You will see the main menu of the raspi-config utility. Select **Finish** to go back to command prompt.
6. **Reboot** the pi at the command prompt for the changes to take effect.

**Setting up the lcd directory**

1. Download the compressed lcd directory from MEL and expand it in **/home/pi/**
2. Open **Terminal** and enter the following command:

$ cd lcd

$ sudo sh install.sh

1. Note that your Raspberry Pi will reboot instantly after LCD drivers are installed
2. Open the file named **hellolcd.py**
3. To run this program, perform the following command in **Terminal**

$ sudo python hellolcd.py

import lcddriver

import time

display = lcddriver.lcd()

try:

while True:

print("Writing to display")

display.lcd\_display\_string("This is my new", 1)

display.lcd\_display\_string("LCD Program!", 2)

time.sleep(2)

except KeyboardInterrupt:

print("Cleaning up!")

display.lcd\_clear()

**Notes on this program**

* import lcddriver allows us to make use of the LCD libraries
* display=lcdsdriver.lcd() creates a new instance of the LCD class
* display.lcd\_clear() clears the contents and buffer on the LCD display
* display.lcd\_display\_string() allows us to display a string on the LCD display
* time.sleep(2) allows us to display the text for 2 secs before it refreshes

**Displaying the current time and IP Address**

Open and execute the file called **timeandip.py** in the **/home/pi/lcd directory**

import lcddriver

import time

from subprocess import \*

from datetime import datetime

#Initialise the display

display = lcddriver.lcd()

try:

cmd = "ip addr show eth0 | grep inet | awk '{print $2}' | cut -d/ -f1"

def run\_cmd(cmd):

p = Popen(cmd, shell=True, stdout=PIPE)

output = p.communicate()[0]

return output

while True:

print("Writing to display")

ipaddr = run\_cmd(cmd)

print(ipaddr)

display.lcd\_display\_string(datetime.now().strftime('%b %d %H:%M:%S\n'), 1)

display.lcd\_display\_string('IP %s' % (ipaddr), 2)

time.sleep(2)

except KeyboardInterrupt:

print("Cleaning up!")

display.lcd\_clear()

**Notes on this program**

* The led directory contains all the libraries and sample codes for using the LCD panel
* The Popen() function executes a child program in a new process using the specified command (cmd)
* The communicate() method sends data to stdin and returns a tuple (stdoutdata, stderrdata)
* The command below displays the Ethernet (eth0) IP address
  + cmd = "ip addr show eth0 | grep inet | awk '{print $2}' | cut -d/ -f1"
  + if you are on wireless, you may want to replace eth0 with wlan0 or wlan1

**Additional Exercise**

Use your 3 LEDs to simulate a typical traffic light system at a traffic intersection. In this order, light up

* The green LED for 10 secs
* The white/yellow LED for 1 sec
* The red LED for 10 secs

When the green and white/yellow LED is on, display “DO NOT WALK” on the LCD display and when the red LED is on, display “WALK”.