**Experiment 10**

Rishabh Shah J058

**Part 1:**

Button controlled robot

from gpiozero import Button, Robot

from time import sleep

from signal import pause

robot = Robot((17, 18), (22, 23))

left = Button(2)

right = Button(3)

forward = Button(4)

backward = Button(5)

go = Button(6)

instructions = []

def add\_instruction(btn):

instructions.append({

left: (-1, 1),

right: (1, -1),

forward: (1, 1),

backward: (-1, -1),

}[btn])

def do\_instructions():

instructions.append((0, 0))

robot.source\_delay = 0.5

robot.source = instructions

sleep(robot.source\_delay \* len(instructions))

del instructions[:]

go.when\_pressed = do\_instructions

for button in (left, right, forward, backward):

button.when\_pressed = add\_instruction

pause()

**Part 2:**

Make a robot drive forward when it detects motion

from gpiozero import Robot, MotionSensor

from signal import pause

robot = Robot(left=(4, 14), right=(17, 18))

pir = MotionSensor(5)

pir.when\_motion = robot.forward

pir.when\_no\_motion = robot.stop

pause()

**Part 3:**

Present the value of a potentiometer on an LED bar graph using PWM

from gpiozero import LEDBarGraph

from time import sleep

from \_\_future\_\_ import division # required for python 2

graph = LEDBarGraph(5, 6, 13, 19, 26, pwm=True)

graph.value = 1/10 # (0.5, 0, 0, 0, 0)

sleep(1)

graph.value = 3/10 # (1, 0.5, 0, 0, 0)

sleep(1)

graph.value = -3/10 # (0, 0, 0, 0.5, 1)

sleep(1)

graph.value = 9/10 # (1, 1, 1, 1, 0.5)

sleep(1)

**Part 4:**

Internet connection status indicator

from gpiozero import LED, PingServer

from gpiozero.tools import negated

from signal import pause

green=LED(1)

red=LED(2)

internet = PingServer(‘google.com’)

green.source=internet

red.source=negated(green)

pause()

**Part 5:**

You can read the Raspberry Pi’s own CPU temperature using the built-in CPUTemperature class, and display this on a “bar graph” of LEDs

from gpiozero import CPUTemperature, LED

cpu=CPUTemperature(min\_temp=50, max\_temp=90)

led=LED(1)

led.source=cpu

**Part 6:**

Control LED using TimeofDay i.e. LED should be on between 7-8am

from gpiozero import LED, TimeOfDay

from datetime import time

from signal import pause

led = LED(2)

tod = TimeOfDay(time(7), time(8))

tod.when\_activated = led.on

tod.when\_deactivated = led.off

pause()

**Part 7:**

DiskUsage

//Extends PolledInternalDevice to provide a device which is active when the disk space used exceeds the threshold value. The following example plots the disk usage on an LED bar graph:

from gpiozero import LEDBarGraph, DiskUsage

from signal import pause

disk = DiskUsage()

print('Current disk usage: {}%'.format(disk.usage))

graph = LEDBarGraph(5, 6, 13, 19, 25, pwm=True)

graph.source = disk

pause()

**Part 8:**

8. LoadAverage

//Extends PolledInternalDevice to provide a device which is active when the CPU load average exceeds the threshold value. The following example plots the load average on an LED bar graph:

from gpiozero import LEDBarGraph, LoadAverage

from signal import pause

la = LoadAverage(min\_load\_average=0, max\_load\_average=2)

graph = LEDBarGraph(5, 6, 13, 19, 25, pwm=True)

graph.source = la

pause()