

EET 2812

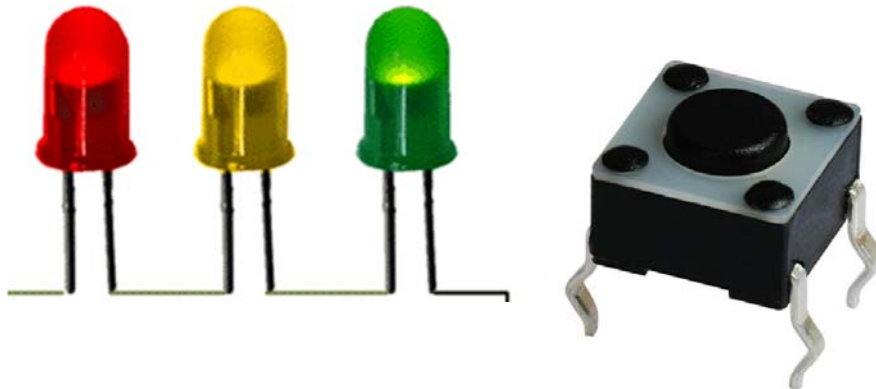
Activity 9

Single Board Computer (RPI)

Touch Sensor, LED's

Cuyahoga Community College

Youth Technology Academy

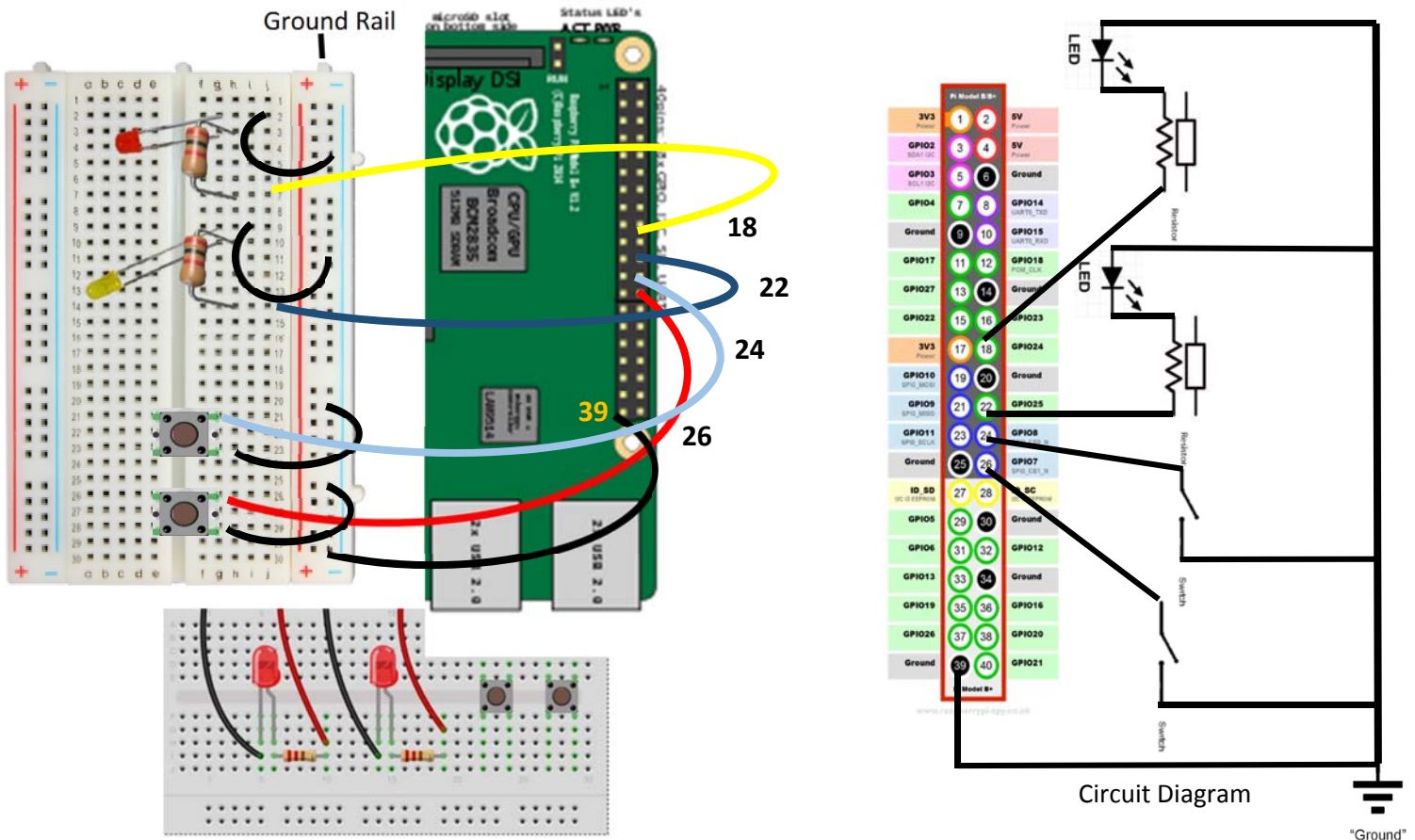


# ACTIVITY

## RPi Sensors Programming – SwitchLED

**Overview:** Build a circuit with the RPi, 2 buttons and 2 LEDs. Program the RPi to turn the lights on and off when a button is pressed.

**Vocabulary:** RPi, circuit, LED, function library, pull-up resistor, pull-down resistor



Create the circuit above using the RPi, breadboard, 2 buttons and 2 LEDs connected to pins GND39, 26, 24, 22, and 18.

Boot up the RPi and change your directory to my\_python:

```
$cd my_python
```

Open nano and create a program called buttonLED.py:

```
$nano buttonLED.py
```

Write the code in nano to display a message and turn on a light when a button is pressed:  
**Remember: Everything after the # is a comment and will not be read by the program.**

```
from time import sleep          # Import time function from the sleep Library
import RPi.GPIO as GPIO        # Import GPIO Library
GPIO.setmode(GPIO.BOARD)      # Use Physical Pin Numbering Scheme
button1=26                     # Button 1 is connected to physical pin 26
button2=24                     # Button 2 is connected to physical pin 24
LED1=22                        # LED 1 is connected to physical pin 22
LED2=18                        # LED 2 is connected to physical pin 18
GPIO.setup(button1,GPIO.IN,pull_up_down=GPIO.PUD_UP)
                                # Make button1 an input, Activate Pull UP Resistor
GPIO.setup(button2,GPIO.IN,pull_up_down=GPIO.PUD_UP)
                                # Make button 2 an input, Activate Pull Up Resistor
GPIO.setup(LED1,GPIO.OUT)      # Make LED 1 an Output
GPIO.setup(LED2,GPIO.OUT)      # Make LED 2 an Output
BS1=False                      # Set Flag BS1 to indicate LED is initially off
BS2=False                      # Set Flag BS2 to indicate LED is initially off
while(1):                      # Create an infinite Loop
    if GPIO.input(button1)==0:  # Look for button 1 press
        print "Button 1 Was Pressed:"
        if BS1==False:        # If the LED is off
            GPIO.output(LED1,True) # turn it on
            BS1=True           # Set Flag to show LED1 is now On
            sleep(.5)         # Delay
        else:                  # If the LED is on
            GPIO.output(LED1,False) # Turn LED off
            BS1=False          # Set Flag to show LED1 is now Off
            sleep(.5)
    if GPIO.input(button2)==0:  #Repeat above for LED 2 and button 2
        print "Button 2 Was Pressed:"
        if BS2==False:
            GPIO.output(LED2,True)
            BS2=True
            sleep(.5)
        else:
            GPIO.output(LED2,False)
            BS2=False
            sleep(.5)
GPIO.cleanup()                 #reset pins for next program
```

Save the program pressing CTRL O and ENTER. Then exit nano by pressing CTRL X.

Run the program as a superuser: **\$sudo python buttonLED.py**

**To exit the program press CTRL Z**