Seven-Color automatic flashing LED



Overview

The seven-color LED is an example of a fully-automated display device. It integrates both an RGB bulb and a small controller that flashes through seven different color settings at a variety of speeds. Compared to a simple bulb (such as the RGB LED module), the programming model is simple: all one has to do is tell it to start its cycle. Automated bulbs like this frequently feature in ornamental or decorative lighting displays. Because its pin configuration is identical to many simpler LEDs, it can replace a single-color LED in many circuits with a more decorative output. This experiment uses the Raspberry Pi to initiate the 7-color LED automated flash sequence.

Experimental Materials

Raspberry Pi x1

Breadboard x1

Seven-color LED x1

Dupont jumper wires

Experimental Procedure

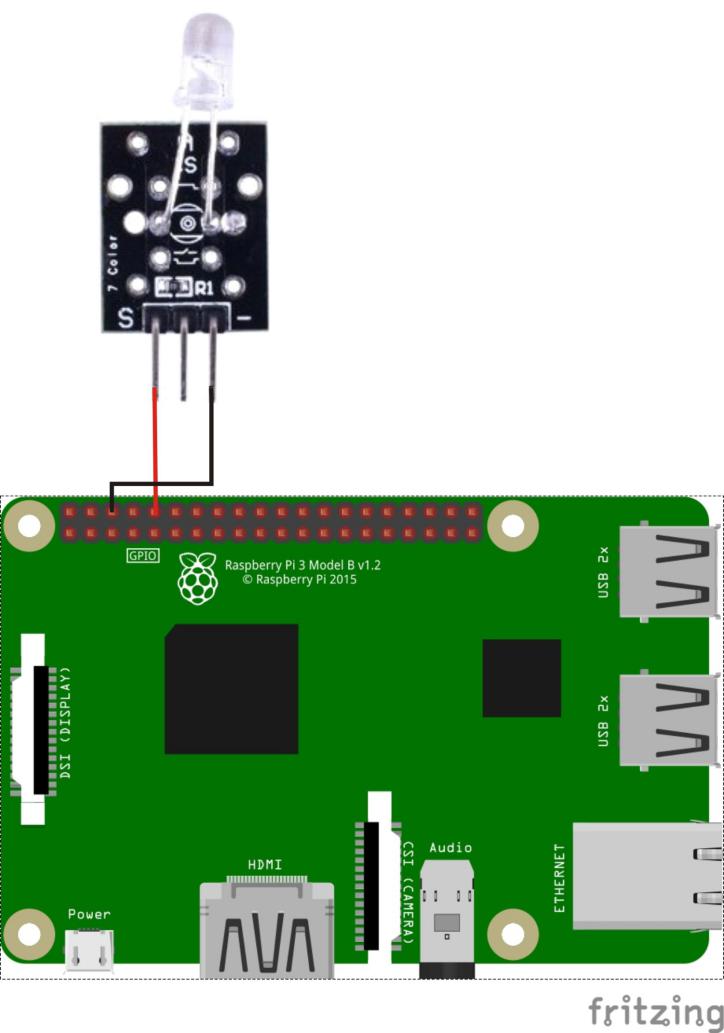
1. If you have not done so already, prepare your development system by installing the Python interpreter, RPi.GIO library, and wiringPi library as described in READ\_ME.TXT.
2. Install the seven-color LED in your breadboard and use Dupont jumper wires to connect it to your Raspberry Pi as illustrated in the Wiring Diagram below. (The seven-color LED module in this kit includes onboard series resistors, so no additional resistors are needed.)
3. Execute the sample stored in this experiment’s subfolder.

If using C, compile and execute the C code:  
cd Code/C  
gcc 7colorLED.c -o 7colorLED.out –lwiringPi  
./7colorLED.out

If using Python, launch the Python script:  
cd Code/Python  
python 7colorLED.py

1. Make experimental observations. The seven-color LED should flash through its color cycle, alternating slow, then fast, then slow again.

Wiring diagram



Seven-color LED pin position:

S ↔ Raspberry Pi pin 10

“-“ ↔ Raspberry Pi GND

Sample code

Python code

#!/usr/bin/env python

import RPi.GPIO as GPIO

import time

pin\_R = 10

def setup():

GPIO.setmode(GPIO.BOARD) # Numbers GPIOs by physical location

GPIO.setup(pin\_R, GPIO.OUT)

try:

setup()

while True:

GPIO.output(pin\_R, GPIO.HIGH)

except KeyboardInterrupt:

GPIO.cleanup()

C code

#include <wiringPi.h>

#include <softPwm.h>

#include <stdio.h>

typedef unsigned char uchar;

#define LedPin 16

int main(void)

{

int i;

if(wiringPiSetup() == -1)

{

printf("setup wiringPi failed !");

return -1;

}

pinMode(LedPin, OUTPUT);

while(1)

{

digitalWrite(LedPin, HIGH);

}

return 0;

}

Technical Background

◆ Forward voltage: 3.0~4.5V