LED bar graph

Overview

In this project, we use a number of LED to make a flowing water light

Experimental Materials:

Raspberry Pi *1

LED bar graph *1
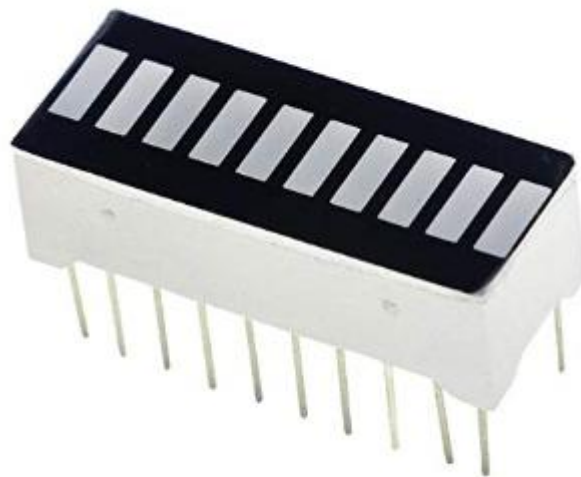
T-type expansion board *1

220 ohm resistor *10

Breadboard*1

Some DuPont lines
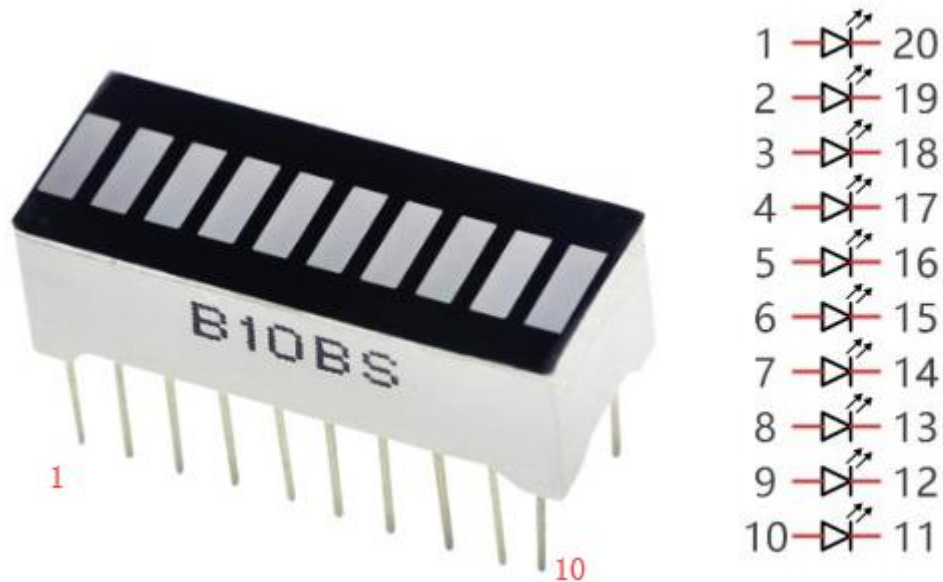
Product description：

Let us learn about the basic features of components to use them better.

LED bar graph

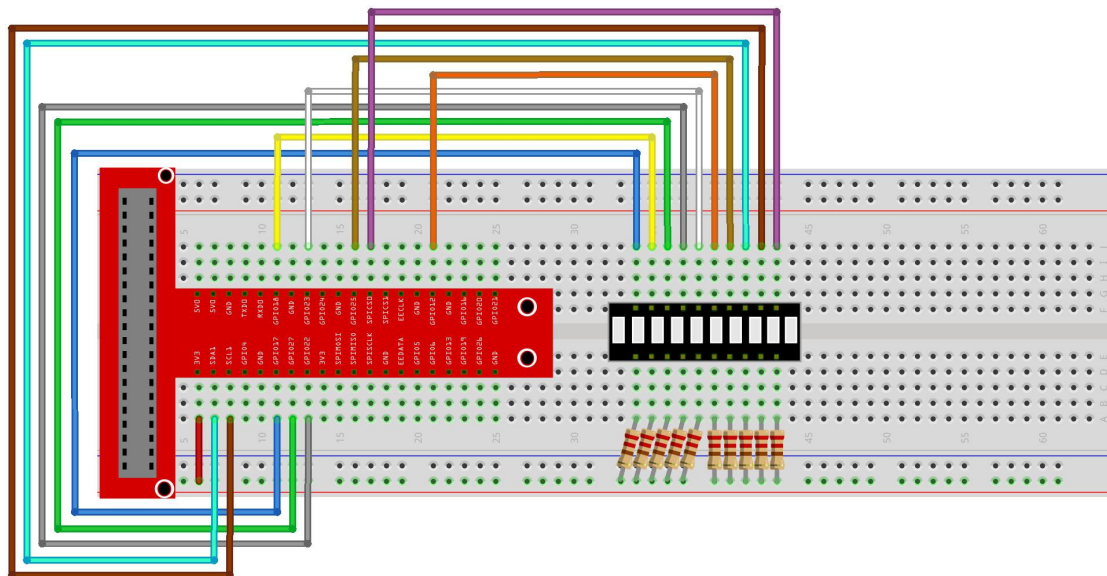LED bar graph is a component Integration consist of 10 LEDs. There are two rows of pins at its bottom.

Technical Parameters:

High quality, bright LED display

Pin Number: 20 pins

Segment Number: 10 segments

Wiring diagram：

C code：

```c
#include <wiringPi.h>
#include <stdio.h>
#define leds 10
int pins[leds] = {0,1,2,3,4,26,6,8,9,10};
void led_on(int n)//make led_n on
{
    digitalWrite(n, LOW);
}

void led_off(int n)//make led_n off
{
    digitalWrite(n, HIGH);
}
```

```c
int main(void)
{
    int i;
    printf("Program is starting ... \n");
    if(wiringPiSetup() == -1){ //when initialize wiring failed,print messageto
screen
        printf("setup wiringPi failed !");
        return 1;
    }
    for(i=0;i<leds;i++){        //make leds pins' mode is output
        pinMode(pins[i], OUTPUT);
    }
    while(1){
        for(i=0;i<leds;i++){    //make led on from left to right
            led_on(pins[i]);
            delay(100);
            led_off(pins[i]);
        }
        for(i=leds-1;i>-1;i--){    //make led on from right to left
            led_on(pins[i]);
            delay(100);
            led_off(pins[i]);
        }
    }
    return 0;
}
```

Python code：

```python
#!/usr/bin/env python3
import RPi.GPIO as GPIO
import time

ledPins = [11, 12, 13, 15, 16, 32, 22, 3, 5, 24]

def setup():
    print ('Program is starting...')
    GPIO.setmode(GPIO.BOARD)        # Numbers GPIOs by physical location
    for pin in ledPins:
        GPIO.setup(pin, GPIO.OUT)   # Set all ledPins' mode is output
```

```python
        GPIO.output(pin, GPIO.HIGH) # Set all ledPins to high(+3.3V) to off led

def loop():
    while True:
        for pin in ledPins:      #make led on from left to right
            GPIO.output(pin, GPIO.LOW)
            time.sleep(0.1)
            GPIO.output(pin, GPIO.HIGH)
        for pin in ledPins[10:0:-1]:        #make led on from right to left
            GPIO.output(pin, GPIO.LOW)
            time.sleep(0.1)
            GPIO.output(pin, GPIO.HIGH)

def destroy():
    for pin in ledPins:
        GPIO.output(pin, GPIO.HIGH)      # turn off all leds
    GPIO.cleanup()                        # Release resource

if __name__ == '__main__':      # Program start from here
    setup()
    try:
        loop()
    except KeyboardInterrupt:  # When 'Ctrl+C' is pressed, the child program
destroy() will be  executed.
        destroy()
```

Experimental results:

In the directory where the code file is located, execute the following

command

C：
gcc -Wall -o LightWater LightWater.c -lwiringPi
sudo ./LightWater

Python:
python LightWater.py

After the program is executed, you will see that LED Bar Graph starts with the flowing water way to be turned on from left to right, and then from right to left.