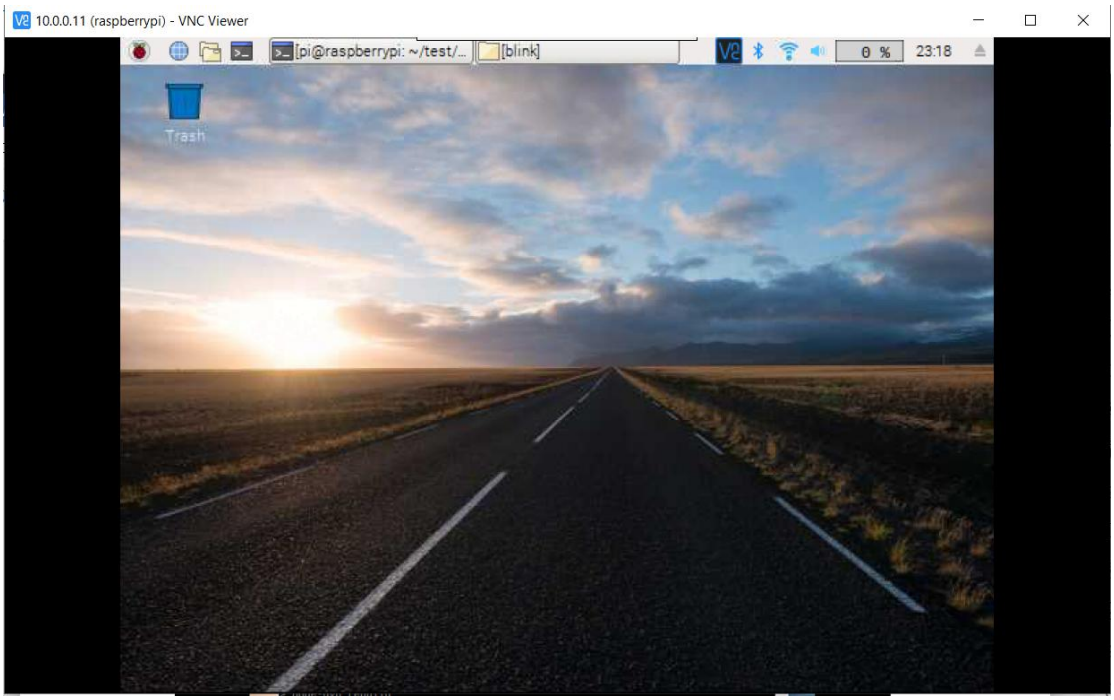
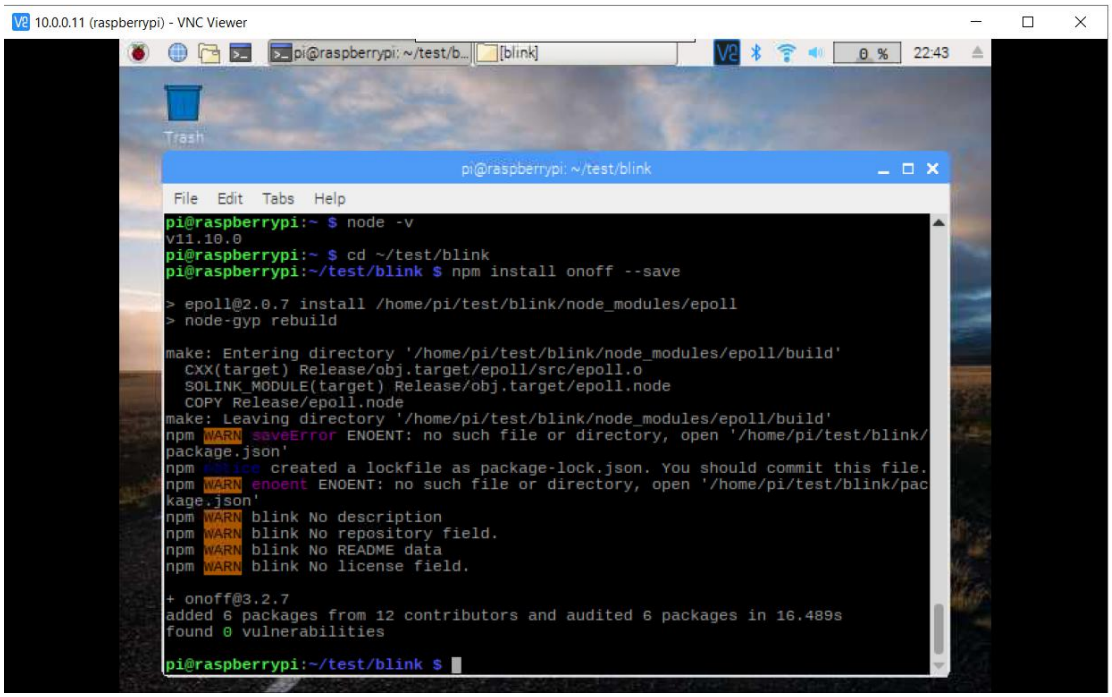
EE517 Internet of Things

Project: Controlling Leds flowing by node.js

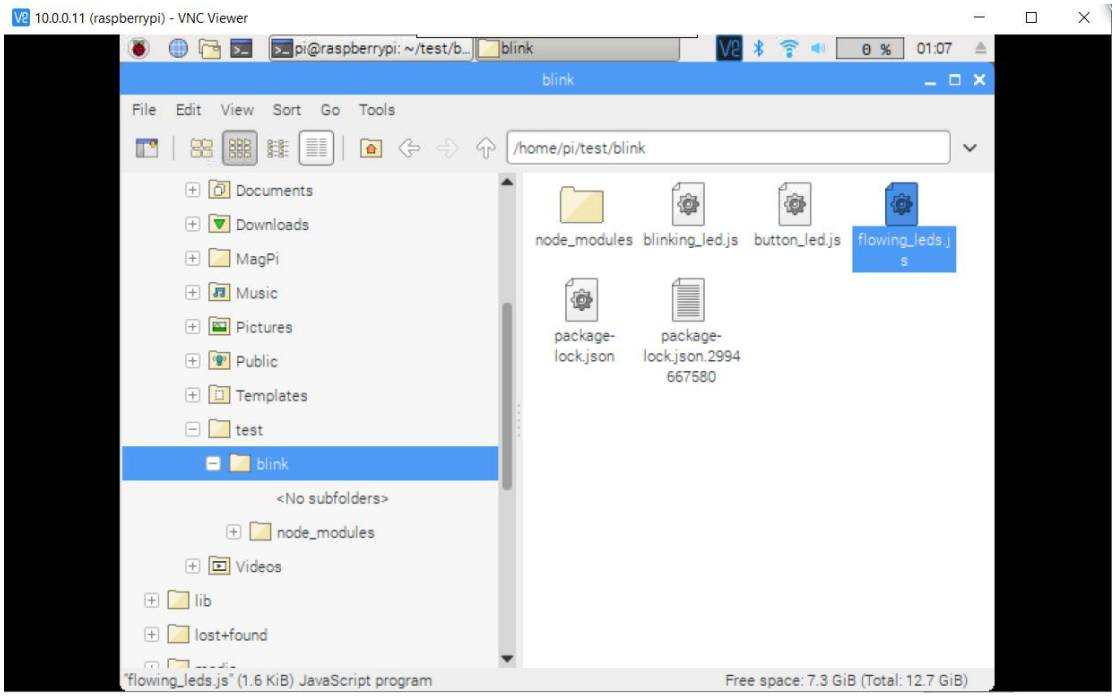
1. Access Raspberry Pi with your computer by using Putty or VNC. Here, I use VNC.



2. Install onoff packet under your direction



3. Create a flowing_leds.js file under your folder



flowing_leds.js

```
var Gpio = require('onoff').Gpio; //include onoff to interact with the GPIO
var LED04 = new Gpio(4, 'out'), //use declare variables for all the GPIO output pins
  LED17 = new Gpio(17, 'out'),
  LED27 = new Gpio(27, 'out'),
  LED22 = new Gpio(22, 'out'),
  LED18 = new Gpio(18, 'out'),
  LED23 = new Gpio(23, 'out'),
  LED24 = new Gpio(24, 'out'),
  LED25 = new Gpio(25, 'out');

//Put all the LED variables in an array
var leds = [LED04, LED17, LED27, LED22, LED18, LED23, LED24, LED25];
var indexCount = 0; //a counter
dir = "up"; //variable for flowing direction

var flowInterval = setInterval(flowingLeds, 100); //run the flowingLeds function
every 100ms

function flowingLeds() { //function for flowing Leds
  leds.forEach(function(currentValue) { //for each item in array
    currentValue.writeSync(0); //turn off LED
  });
  if (indexCount == 0) dir = "up"; //set flow direction to "up" if the count reaches
zero
  if (indexCount >= leds.length) dir = "down"; //set flow direction to "down" if
```

```
the count reaches 7
  if (dir == "down") indexCount--; //count downwards if direction is down
  leds[indexCount].writeSync(1); //turn on LED that where array index matches count
  if (dir == "up") indexCount++ //count upwards if direction is up
};

function unexportOnClose() { //function to run when exiting program
  clearInterval(flowInterval); //stop flow interwal
  leds.forEach(function(currentValue) { //for each LED
    currentValue.writeSync(0); //turn off LED
    currentValue.unexport(); //unexport GPIO
  });
};

process.on('SIGINT', unexportOnClose); //function to run when user closes using
ctrl+cc
```

4. Hardware connection

Pin 6 (Ground) connects to ground bus (the blue line on right side of the breadboard).

Pin 7 (GPIO4) connects with the anode of a LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin 11 (GPIO17) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin 13 (GPIO27) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin 15 (GPIO22) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

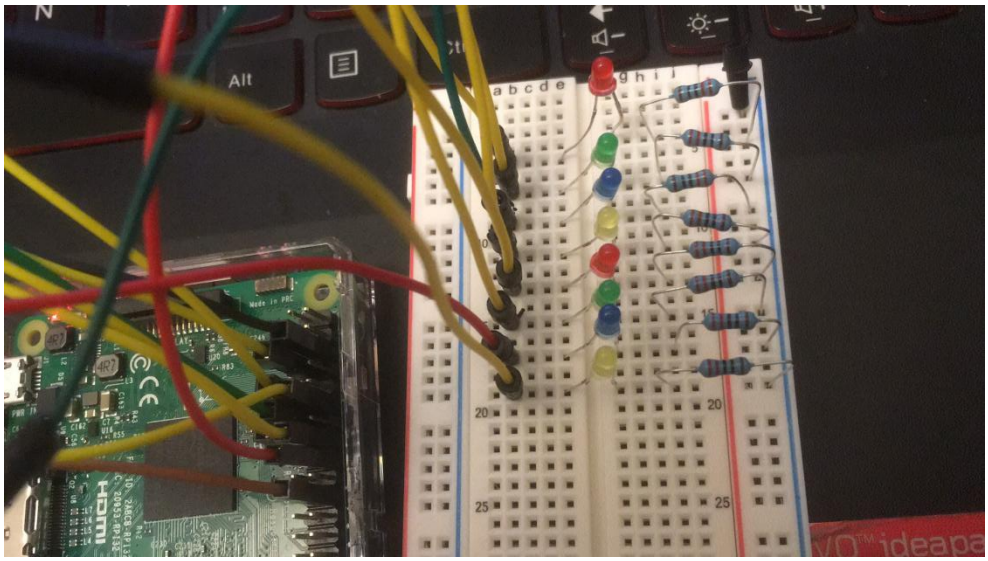
Pin 12 (GPIO18) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin 16 (GPIO23) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

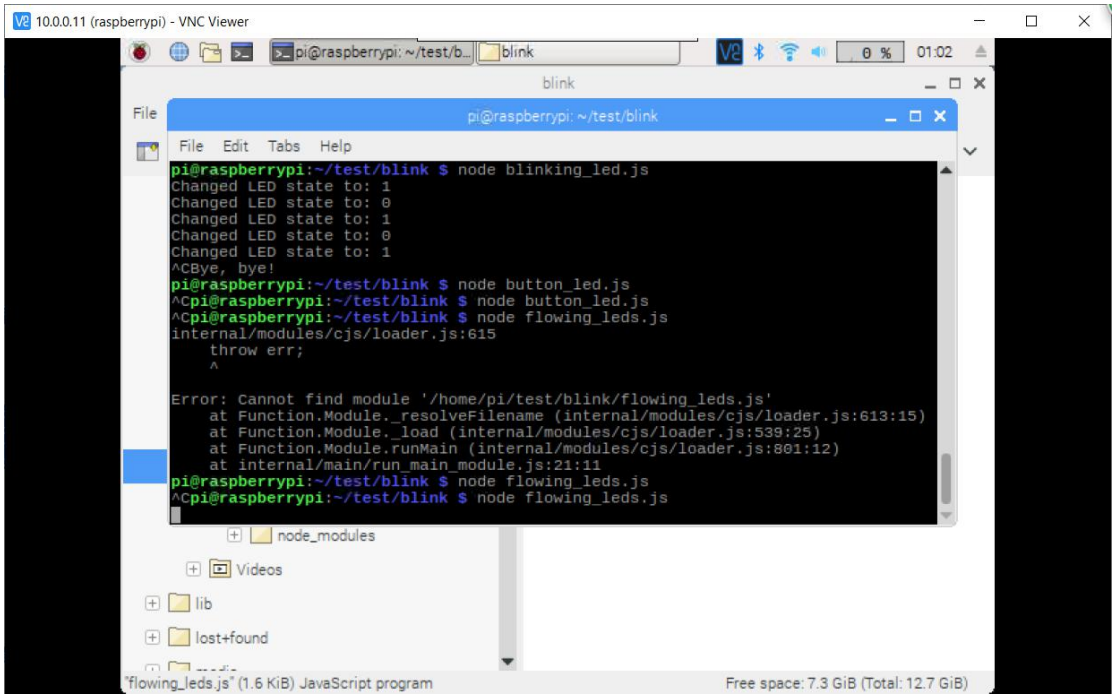
Pin 18 (GPIO24) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin 22 (GPIO25) connects with the anode of LED (which is the long leg), and the cathode of LED (which is the short leg) connects with a 220Ω resistor, then connects to ground bus.

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I²C)		DC Power 5v	04
05	GPIO03 (SCL1 , I²C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)		(I²C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

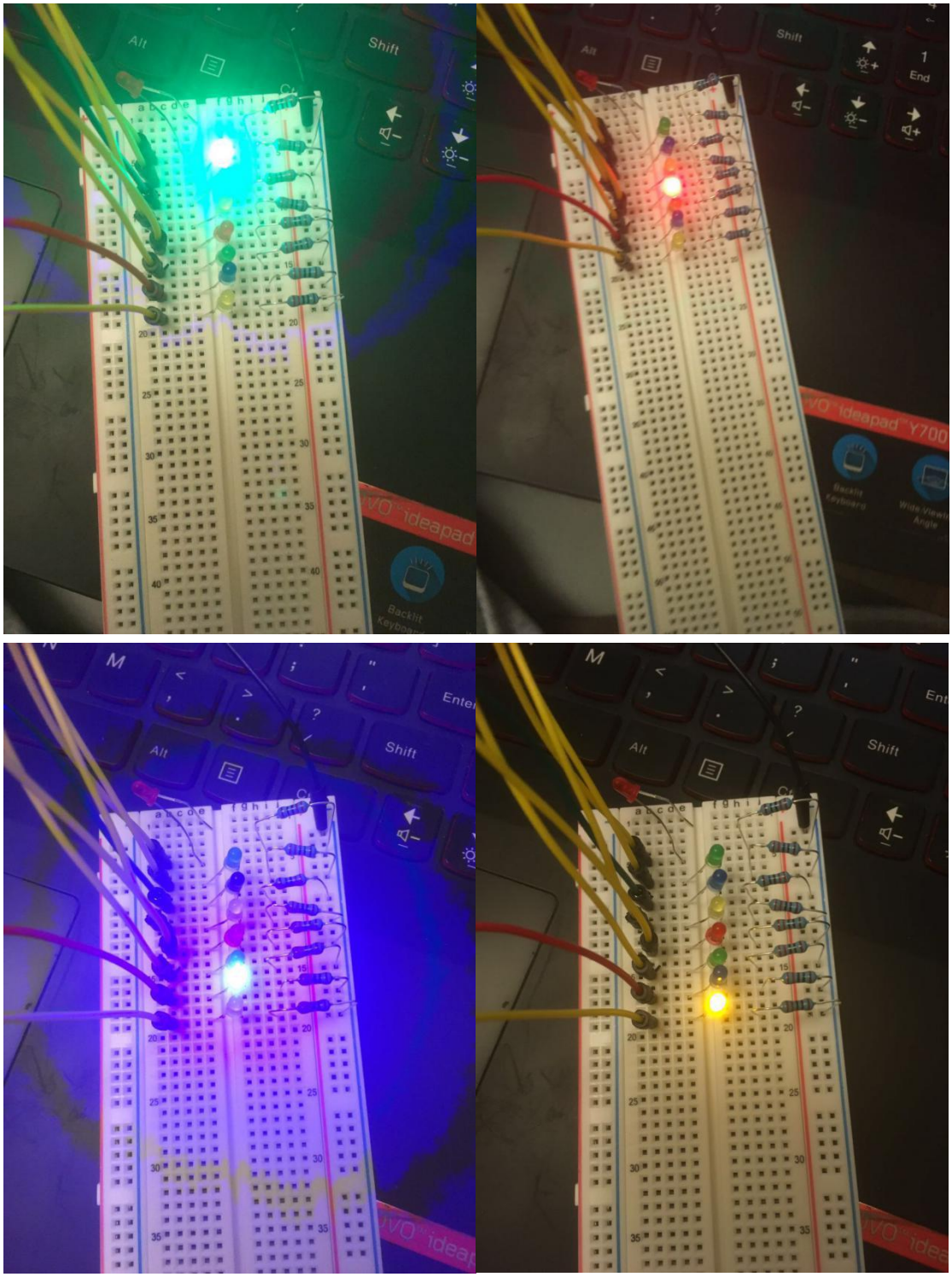


5. Run node flowing_leds.js



6. Simulation:

Leds will be flowing in sequence



[Demo:https://drive.google.com/file/d/14jngfdH1-SZNVBDsu-gX8vGLiuwC7aZJ/view?usp=sharing](https://drive.google.com/file/d/14jngfdH1-SZNVBDsu-gX8vGLiuwC7aZJ/view?usp=sharing)