

traffic light diy Kit Manual

No Prior Experience Required



DO YOU EVER WONDER HOW A TRAFFIC LIGHT WORKS, THE SOFTWARE BEHIND IT AND HOW WE CAN MODIFY THE SOFTWARE ?

The Moonshot Jr Traffic Light kit will help you understand traffic lights and the programming which helps it function. Let us get our hands on this kit and understand how it works. We will also understand the code and how it works to be able to modify it as we progress.

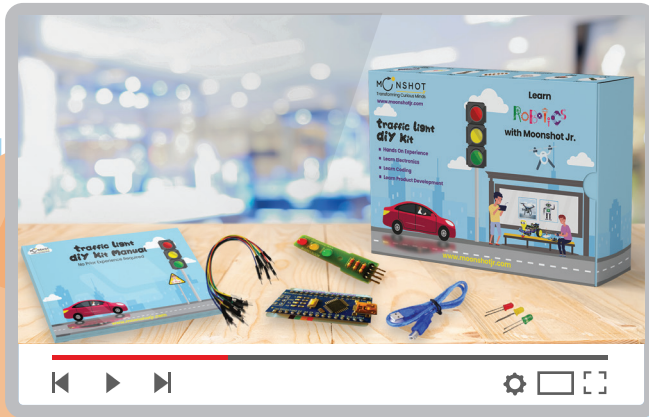


UNPACKING THE KIT

Visit our Tech Corner Blog at

<https://moonpreneur.com/tech-corner/moonshot-jr-traffic-light-kit/>

to view an online copy of this traffic light document



LIST OF ITEMS

Here is the list of items you will find as soon as you unpack the kit.



Controller Board - TL001



USB Cable - TL002



Mooncard - TL003



Traffic Light
Board x1 - TL004



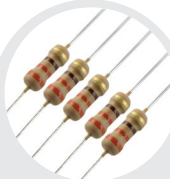
Connection
Wires - TL005



Bread Board - TL006



LEDs (3 Quantity) - TL007



Resistors 5x - TL008



Push Button - TL009



User Manual - TL010

LET'S START

Step 2

Insert the Controller Board (TL001) in the Bread Board (TL006). Ensure that the USB connector is at the edge of the Bread Board and the pins are aligning, as shown in the image below.

Step 4

Insert the Traffic Light Board (TL004) in the breadboard, as indicated.

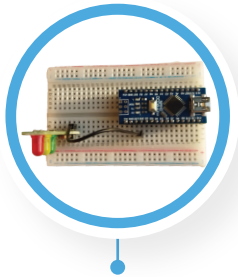
Step 1

Open the kit and locate the Controller board (TL001 - It is an Arduino Nano Compatible Board) . The Controller board has the pre-uploaded Software that will control the traffic light.

Step 3

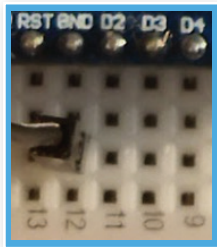
Find the Traffic Light Board (TL004) in the kit, it has 4 colored pins at one end. In the next step we will be connecting the 4 pins with the controller board using the provided wires.

Step 5



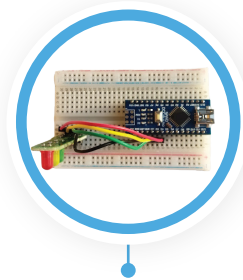
Use a black (or any other color) wire and insert one end to an empty hole in the breadboard in **front of the GND pin of the controller board.**

Insert the other end of the wire in the empty hole on the breadboard in **front of the traffic light board's black pin.**



Make sure the end of the wire is inserted in front of the GND pin of the controller board.

Step 6



Repeat the previous process for the remaining pins.

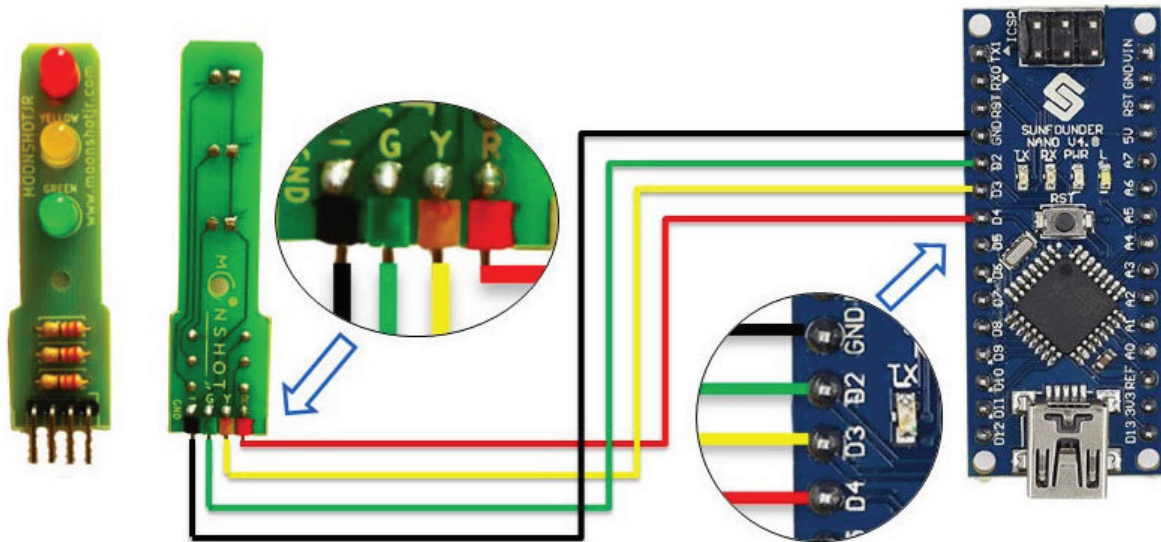
The Red wire connects to **pin D4** of the controller board.

The Yellow wire connects to **pin D3** of the controller board.

The Green wire connects to **pin D2** of the controller board.

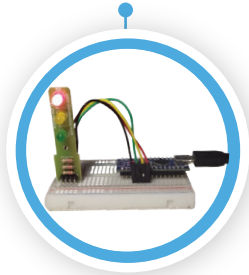
WIRE	TRAFFIC BOARD	CONTROLLER BOARD
RED	RED	D4
YELLOW	YELLOW	D3
GREEN	GREEN	D2
BLACK	BLACK	GND

This is the scheme of connection, note that the connection has to be made through the breadboard.



Step 7

Insert USB Cable into the connector of the controller board. Connect the other end of the cable in the USB Port of your computer or laptop.



Did you see the traffic lights glow in a particular order?

<https://moonpreneur.com/book-a-free-trial-m/>

Let us try to see how the software works. We will need to install the Arduino IDE first, it is a software that allows us to create, modify and upload software on the controller board. You will be able to learn the above steps and more things in great detail in our free trial class. You can sign up for the free trial class by clicking on the link given below.



Have you thought about what was inside the Controller board that made the traffic lights work?



Step 8a

Go to <https://www.arduino.cc/en/software> and download the latest version of Arduino IDE, depending on which system you use (Windows, Mac).

Downloads



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10



Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

Step 8b

After Installing the Arduino IDE, connect the Arduino board to your computer.

After starting the Arduino IDE. Click on tools to reveal this menu:

Refer to the above picture, and follow these steps

- Hover on Board and Select **Arduino Nano**
- Select **ATmega328P** (Old Bootloader) as the Processor.
- Select the correct Port (e.g. COM4).
- Make sure the Programmer is set to **AVRISP mkII**.

File Edit Sketch Tools Help

```
TRAFFIC_LIG
// C++ code
//Define Pins
const byte GRE
const byte YEL
const byte RED

//Initialize t
void setup()//
{
  pinMode(GRE
  pinMode(YELL
  pinMode(RED,
}
//Run code (wz
void loop()//T
```

Auto Format	Ctrl+T
Archive Sketch	
Fix Encoding & Reload	
Manage Libraries...	Ctrl+Shift+I
Serial Monitor	Ctrl+Shift+M
Serial Plotter	Ctrl+Shift+L

WiFi101 / WiFININA Firmware Updater

Board: "Arduino Nano"	>
Processor: "ATmega328P (Old Bootloader)"	>
Port: "COM4"	>
Get Board Info	
Programmer: "AVRISP mkII"	>
Burn Bootloader	

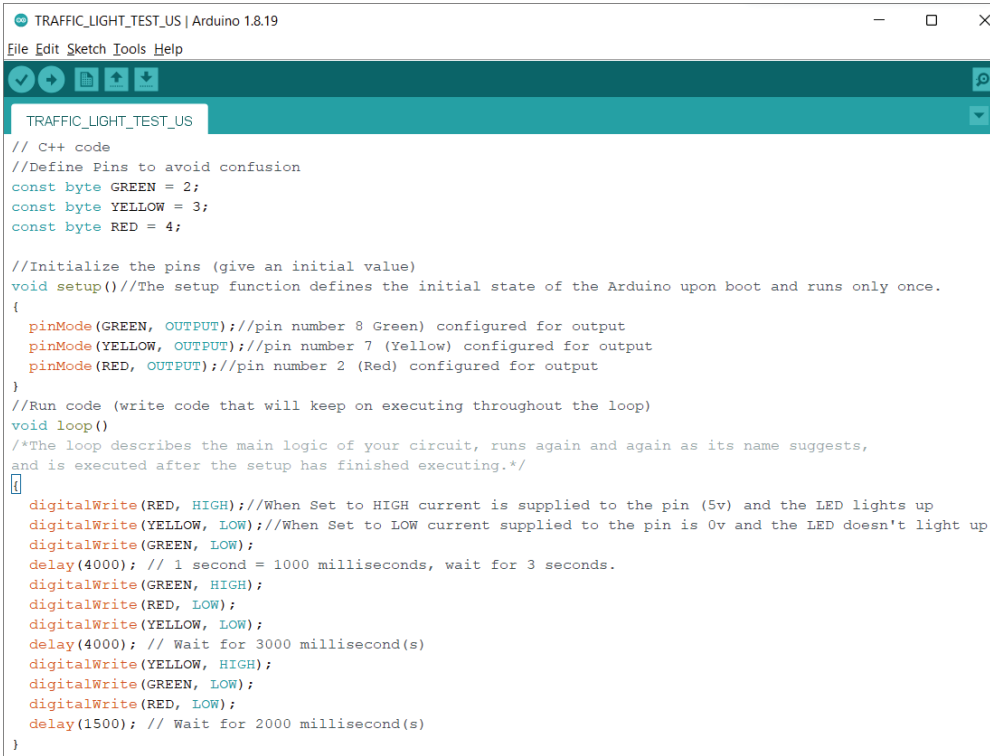
Step 8c

Visit <https://moonpreneur.com/tech-corner/moonshot-jr-traffic-light-kit/> to copy the working code for the traffic light. Alternatively, you can scan the QR code given below.



Step 8d

Paste the code in the editor.



```
TRAFFIC_LIGHT_TEST_US

// C++ code
//Define Pins to avoid confusion
const byte GREEN = 2;
const byte YELLOW = 3;
const byte RED = 4;

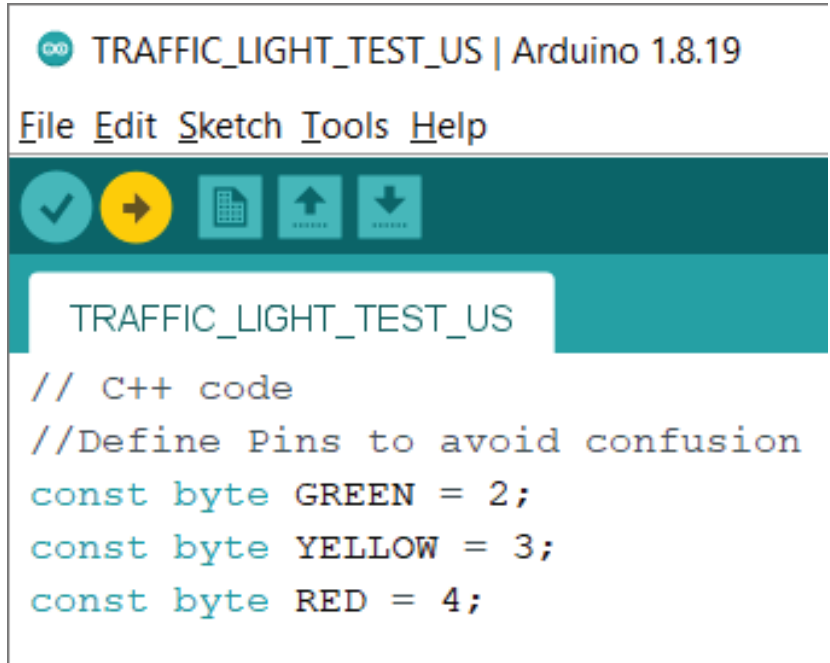
//Initialize the pins (give an initial value)
void setup()//The setup function defines the initial state of the Arduino upon boot and runs only once.
{
  pinMode(GREEN, OUTPUT);//pin number 8 Green) configured for output
  pinMode(YELLOW, OUTPUT);//pin number 7 (Yellow) configured for output
  pinMode(RED, OUTPUT);//pin number 2 (Red) configured for output
}
//Run code (write code that will keep on executing throughout the loop)
void loop()
/*The loop describes the main logic of your circuit, runs again and again as its name suggests,
and is executed after the setup has finished executing.*/
{
  digitalWrite(RED, HIGH);//When Set to HIGH current is supplied to the pin (5v) and the LED lights up
  digitalWrite(YELLOW, LOW);//When Set to LOW current supplied to the pin is 0v and the LED doesn't light up
  digitalWrite(GREEN, LOW);
  delay(4000); // 1 second = 1000 milliseconds, wait for 3 seconds.
  digitalWrite(GREEN, HIGH);
  digitalWrite(RED, LOW);
  digitalWrite(YELLOW, LOW);
  delay(4000); // Wait for 3000 millisecond(s)
  digitalWrite(YELLOW, HIGH);
  digitalWrite(GREEN, LOW);
  digitalWrite(RED, LOW);
  delay(1500); // Wait for 2000 millisecond(s)
}
```

Step 8e

Upload it to the arduino board by clicking on the upload button highlighted in the yellow color in the image below.

The code will be compiled and uploaded on the board. After the code has been uploaded on the arduino board the traffic light starts functioning.

Did you notice the green light stays on for a longer period of time?



UNDERSTANDING THE CODE AND MAKING CHANGES

1. All lines that begin with `//` are comments. Comments are skipped while when the code is executed, they enhance the understanding of the code. Following two lines are comments, for example.

```
// C++ code
//Define Pins to avoid confusion
```

2. In the block of code below, we define the pin numbers that we are going to use with a simple name to avoid confusion and make the code easy to understand.

```
const int GREEN = 2;
const int YELLOW = 3;
const int RED = 4;
```

3. In the block of code below, the setup is used to initialize or give initial values to the pins, which will decide how these pins will function throughout the execution of the code.

```
void setup()
{
  pinMode(GREEN, OUTPUT);
  pinMode(YELLOW, OUTPUT);
  pinMode(RED, OUTPUT);
}
```

4. Inside the loop resides the code that will keep on executing endlessly. The line of code given below turns on the Red LED while turning the other LED off, it waits for 4 seconds and then turns on only the Green LED while keeping other LEDs off. It again waits for 4 seconds and turns only the Yellow LED on, waits for 1.5 seconds and then starts over with the Red LED.

```
void loop()
{
  digitalWrite(RED, HIGH);//RED LED is ON
  digitalWrite(YELLOW, LOW);//YELLOW LED is OFF
  digitalWrite(GREEN, LOW);//GREEN LED
```

```
is OFF
delay(4000);// Wait for 4000ms or 4s
digitalWrite(GREEN, HIGH);
digitalWrite(RED, LOW);
digitalWrite(YELLOW, LOW);
delay(4000);
digitalWrite(YELLOW, HIGH);
digitalWrite(GREEN, LOW);
digitalWrite(RED, LOW);
delay(1500);
}
```

With the explanation given above, you can experiment with the code and make your own changes. If you wanted to change how fast traffic lights switch, you could change the value of delay.

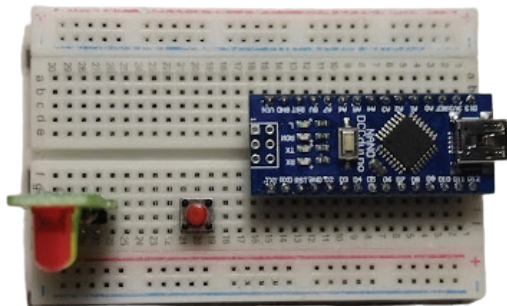
ADDITIONAL EXERCISE

Once you have understood how the above code works, add a button to stop the traffic lights on button press on the already implemented traffic light project.

Here is how you will achieve the above project

Step 1

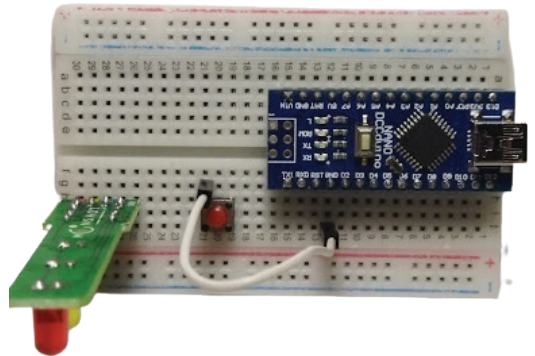
Disconnect and take your working traffic light project. Connect a button on the breadboard as shown.



Step 2

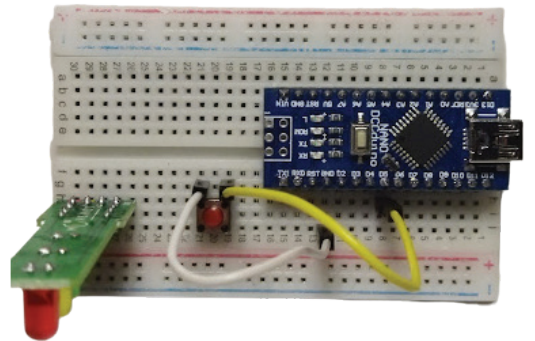
Take a wire connect one end of it in front of the GND pin of the controller board

Insert the other end of the wire in the empty hole on the breadboard in front of the button's left pin.

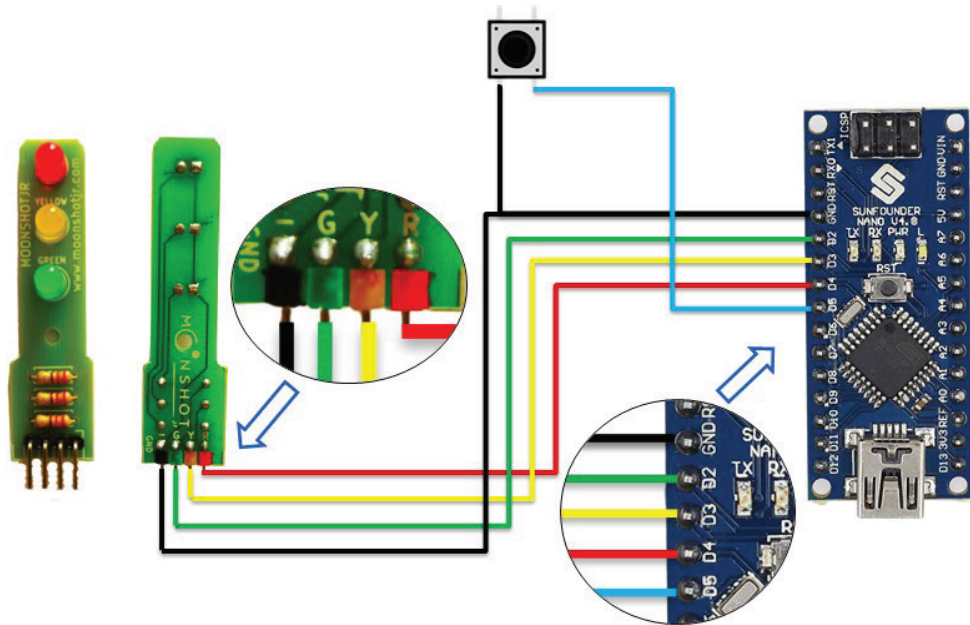


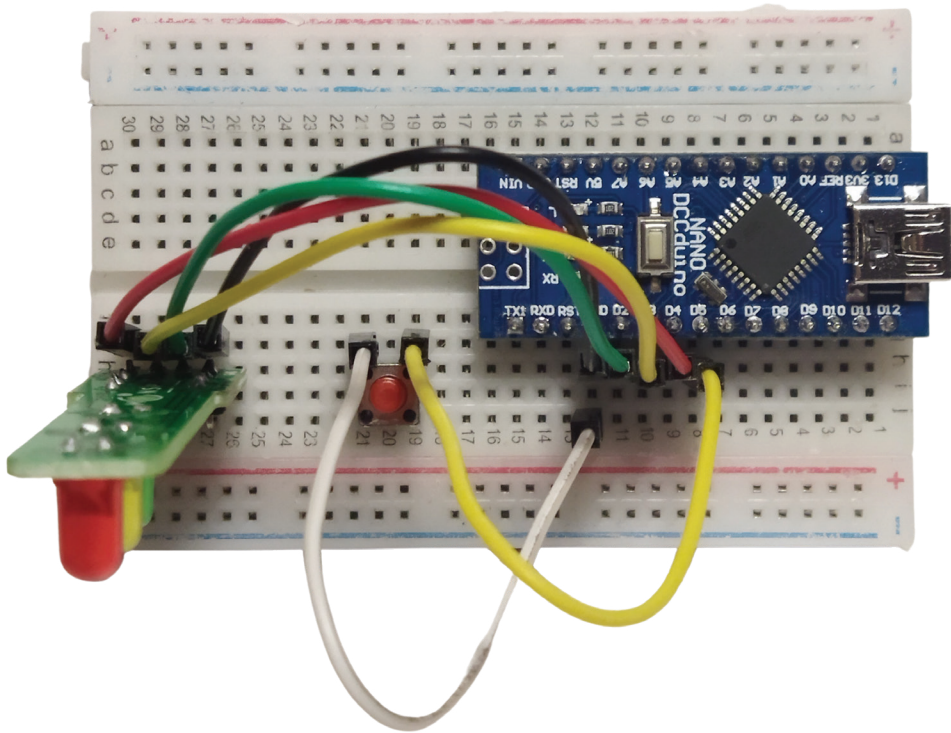
Step 3

Take another wire and connect one end in front of pin D5 of the controller board, connect the other end in front of the button's right pin.



This is the scheme of connection, note that the connection has to be made through the breadboard.





Step 4

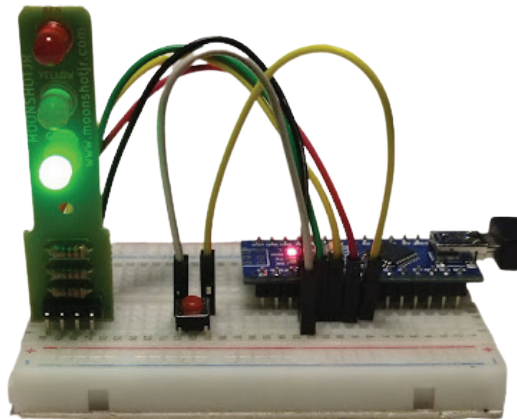
Visit

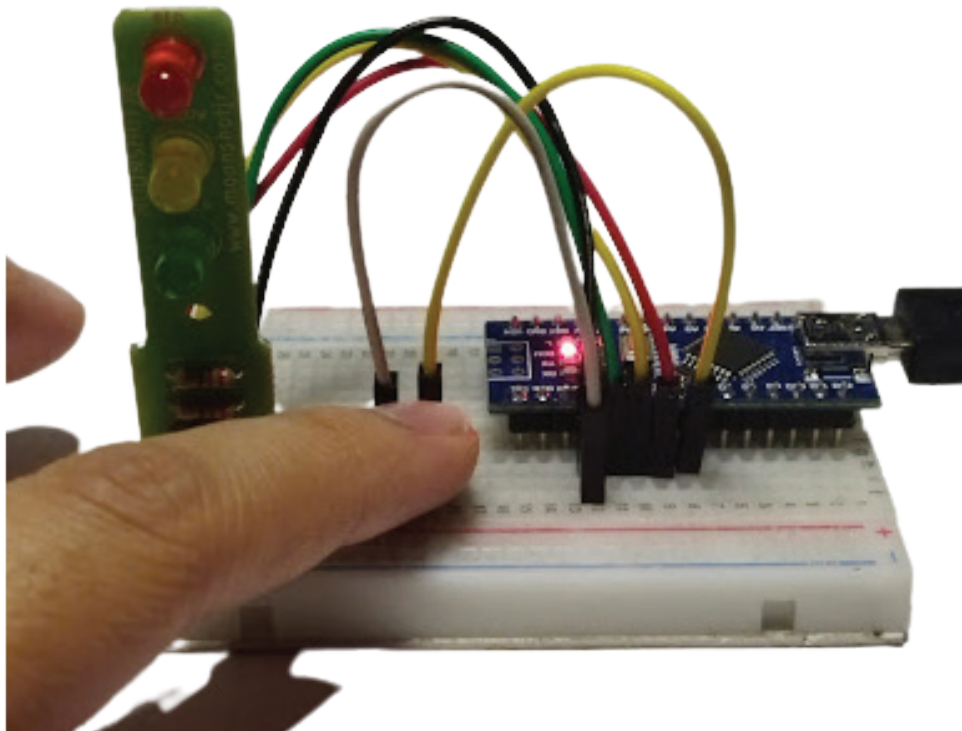
<https://moonpreneur.com/tech-corner/moonshot-jr-traffic-light-kit/>

to copy the working code.

Step 5

Open the Arduino IDE, connect the controller board, paste the code and upload it on the board. Notice that, when you press the button the Traffic Light stops, when you release the button the traffic light starts working again.





NEXT STEP

Do you want to learn more about automation and how you can make a smart Traffic Light and more...

Stop here and register for a free trial class to know how you could actually do that.

Visit Free trial class at

<https://moonpreneur.com/book-a-free-trial-m/>

