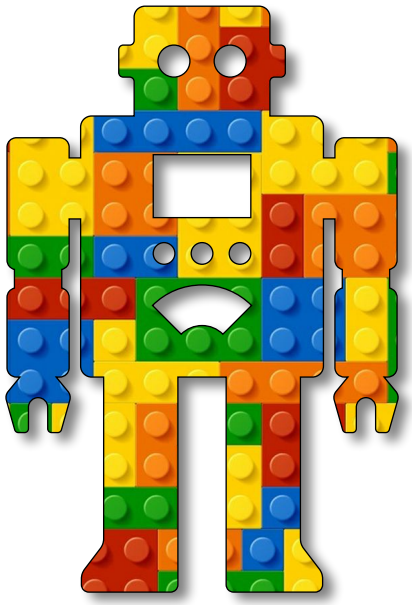
# Intelligent Machines Module A Unit #8 RGB LED



# Content

## Module A Unit #8 RGB LED

# Introduction to RGB LED

The board does have another built-in LED; it is an RGB LED, that means it has a red (LEDR) component, green (LEDG) component, and a blue (LEDB) component. So you can have a red, blue, or green LED, and you can combine them to create colours like yellow, cyan, magenta, or white. They are not PWM pins, so we cannot mix them together to create lots of colours as you can with p5.js.

# Sketch A8.1 built-in RGB LED

To start with, we will have red on for one second, green on for one second, blue on for one second, and all off for one second. This is slightly counterintuitive: for a particular colour LED to be on, it has to be set to LOW; conversely, to switch a colour LED off, it has to be set to HIGH. I know, confusing, isn't it?

Arduino sketch

```
void setup()
{
  pinMode(LEDR, OUTPUT);
  pinMode(LEDG, OUTPUT);
  pinMode(LEDB, OUTPUT);
}

void loop()
{
  // RED
  digitalWrite(LEDR, LOW);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, HIGH);
  delay(1000);

  // GREEN
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, LOW);
  digitalWrite(LEDB, HIGH);
  delay(1000);

  // BLUE
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, HIGH);
```

```
  digitalWrite(LEDB, LOW);
  delay(1000);

  // RGB OFF
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, HIGH);
  delay(1000);
}
```

## 🗒️ Notes

They should cycle through those RGB colours and then stop.

## 🛠️ Code Explanation

| | |
|---|---|
| `pinMode(LEDR, OUTPUT);` | LEDR is the red component of the LED, initialise it as an output |
| `digitalWrite(LEDR, LOW);` | Writes to the LEDR that it is to switch it on |
| `digitalWrite(LEDR, HIGH);` | Writes to the LEDR that it is to switch it off |
| `delay(1000);` | Waits for one second |

# Sketch A8.2 more colours

Adding in all the other colours for good measure.

Arduino sketch

```
void setup()
{
  pinMode(LEDR, OUTPUT);
  pinMode(LEDG, OUTPUT);
  pinMode(LEDB, OUTPUT);
}

void loop()
{
  // RED
  digitalWrite(LEDR, LOW);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, HIGH);
  delay(1000);

  // GREEN
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, LOW);
  digitalWrite(LEDB, HIGH);
  delay(1000);

  // BLUE
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, LOW);
  delay(1000);
```

```
  // YELLOW
  digitalWrite(LEDR, LOW);
  digitalWrite(LEDG, LOW);
  digitalWrite(LEDB, HIGH);
  delay(1000);

  // MAGENTA
  digitalWrite(LEDR, LOW);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, LOW);
  delay(1000);

  // CYAN
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, LOW);
  digitalWrite(LEDB, LOW);
  delay(1000);

// WHITE
  digitalWrite(LEDR, LOW);
  digitalWrite(LEDG, LOW);
  digitalWrite(LEDB, LOW);
  delay(1000);

  // RGB OFF
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, HIGH);
  delay(1000);
}
```

## 🗒️ Notes

Just for fun. If you could alter the value of each one, then you could create more colours, but they are either HIGH or LOW, unfortunately.

# Sketch A8.3 looping round the array

❗ New sketch.

Instead of hard coding everything, let's see if we can achieve the same thing but with just fewer lines of code. To achieve this, we will need to use `for()` loops and `arrays`. The array will contain the names of the LEDs. We are treating them as integers, even though they are clearly not numbers. We start with all the LED colours `off` (`HIGH`), then switch one LED colour `on` (`LOW`), one at a time.

Arduino sketch

```
int led[] = {LEDR, LEDG, LEDB};


void setup()
{
  for (int i = 0; i < 3; i++)
  {
    pinMode(led[i], OUTPUT);
  }
}


void loop()
{
  for (int i = 0; i < 3; i++)
  {
    digitalWrite(LEDR, HIGH);
    digitalWrite(LEDG, HIGH);
    digitalWrite(LEDB, HIGH);
    digitalWrite(led[i], LOW);
    delay(1000);
  }
}
```

## 🗒️ Notes

Cycles through the colours.

## 🌻 Challenges

1. How would you remove them all for one second?
2. What would happen if you didn't have them all off (HIGH) in the `for()` loop?

## 🛠️ Code Explanation

| | |
|---|---|
| `int led[] = {LEDR, LEDG, LEDB};` | Create an array called led. Put three elements into that array, the names of the LEDs. |
| `for (int i = 0; i < 3; i++)`<br>`{`<br>`  pinMode(led[i], OUTPUT);`<br>`}` | A for loop starts at 0 (i = 0), adds 1 each time (i++) and stops before it gets to 3 (i < 3). |
| `pinMode(led[i], OUTPUT);` | led[i] is the index reference in the array. index[0] is LEDR, index[1] is LEDG and index[2] is LEDB |

# Sketch A8.4 random selection

Another solution: Now it will select an LED at random.

❗ Remove the `for()` loop and the curly brackets.

Arduino sketch

```
int led[] = {LEDR, LEDG, LEDB};

void setup()
{
  for (int i = 0; i < 3; i++)
  {
    pinMode(i, OUTPUT);
  }
}

void loop()
{
  int i = random(0, 3);
  digitalWrite(LEDR, HIGH);
  digitalWrite(LEDG, HIGH);
  digitalWrite(LEDB, HIGH);
  digitalWrite(led[i], LOW);
  delay(1000);
}
```

## Notes

It will select the same one more than once. Notice that it is random between 0 and 3, that is, it is inclusive of 0 but up to 3 but not including 3, as we only have three LEDs.

## 🌻 Challenge

For a pleasing effect, make the delay much smaller, e.g. `100`.