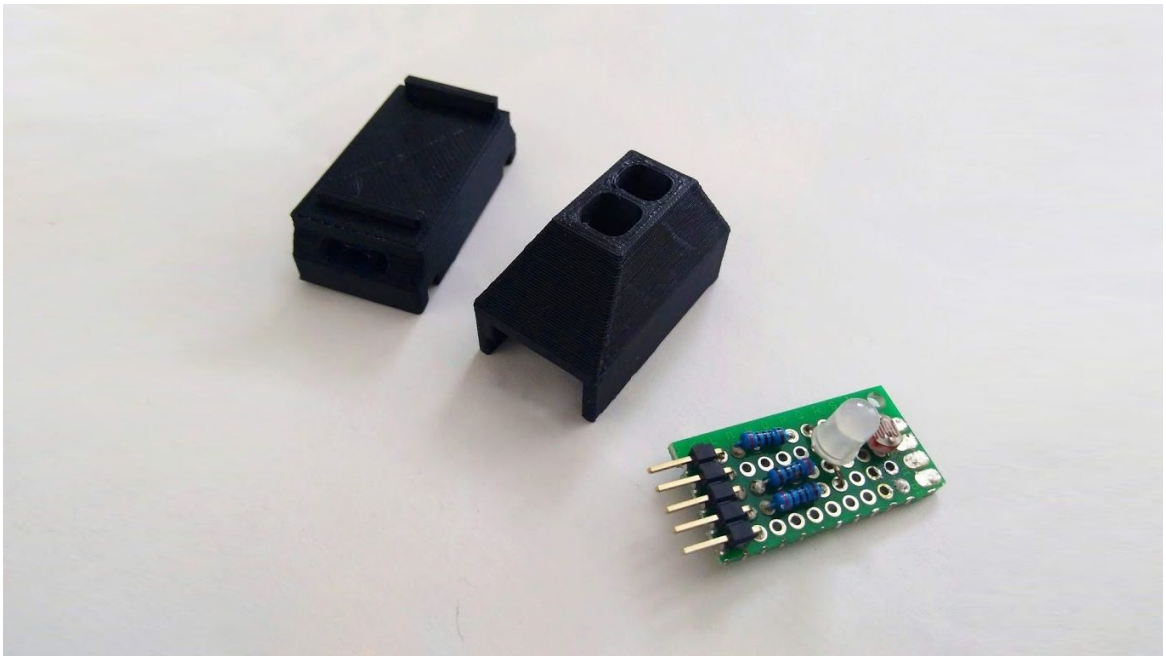
Adding a Color Sensor



A color sensor works by measuring the amount of red, green, and blue light reflected off of an object. This tutorial will use an RGB LED to cycle through red, green, and blue light, and a photoresistor to measure the light reflected in each case.

Parts Required:

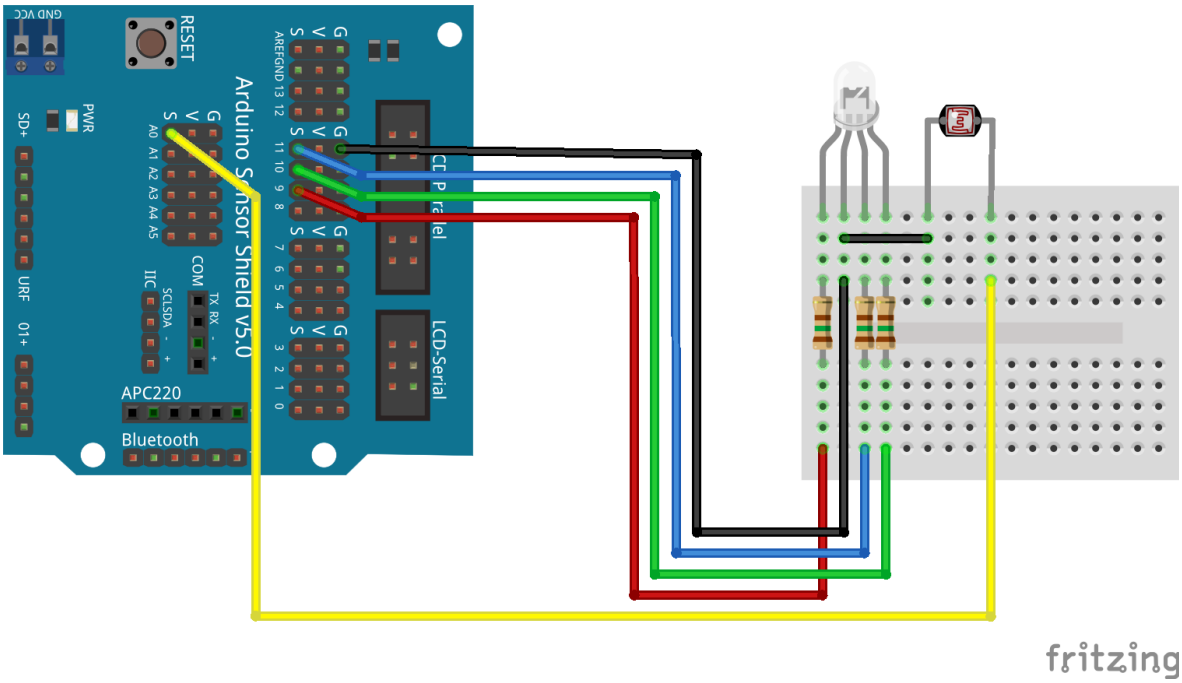
1 - Assembled Make-A-Pede
1 - RGB LED (common cathode)
1 - Photoresistor
3 - 150Ω Resistors
1 - M-M Jumper Wire
5 - M-F Jumper Wires

OR

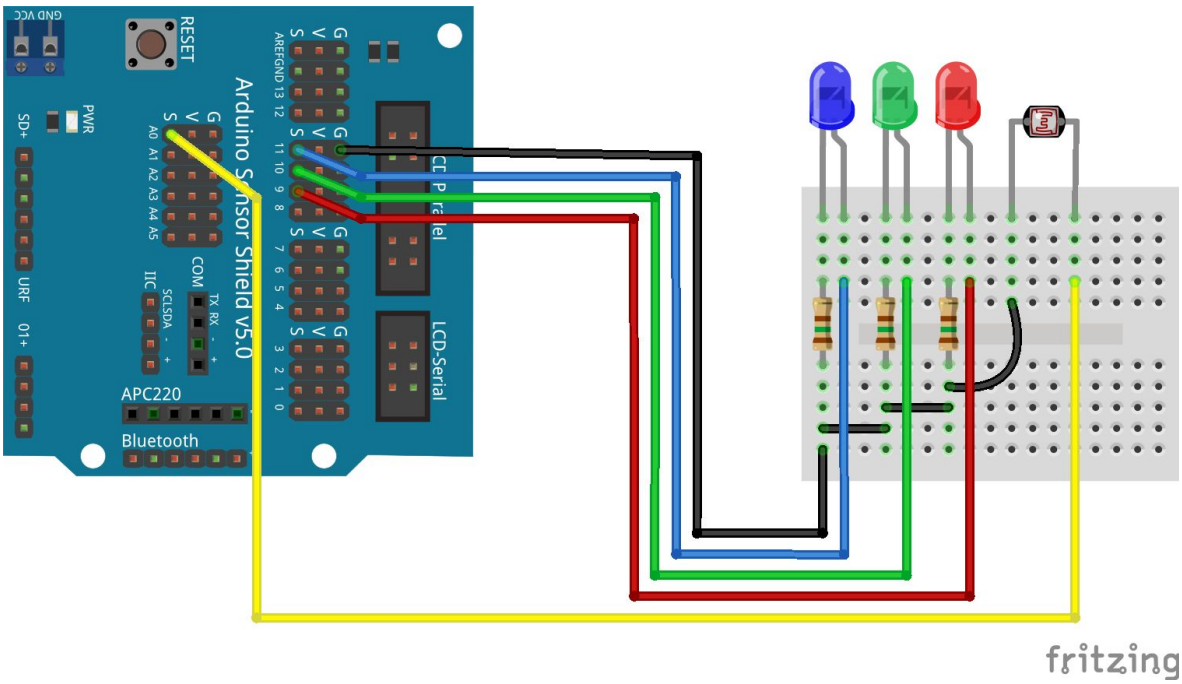
1 - Assembled Make-A-Pede
1 - Red LED
1 - Green LED
1 - Blue LED
1 - Photoresistor
3 - 150Ω Resistors
3 - M-M Jumper Wires
5 - M-F Jumper Wires

Wiring

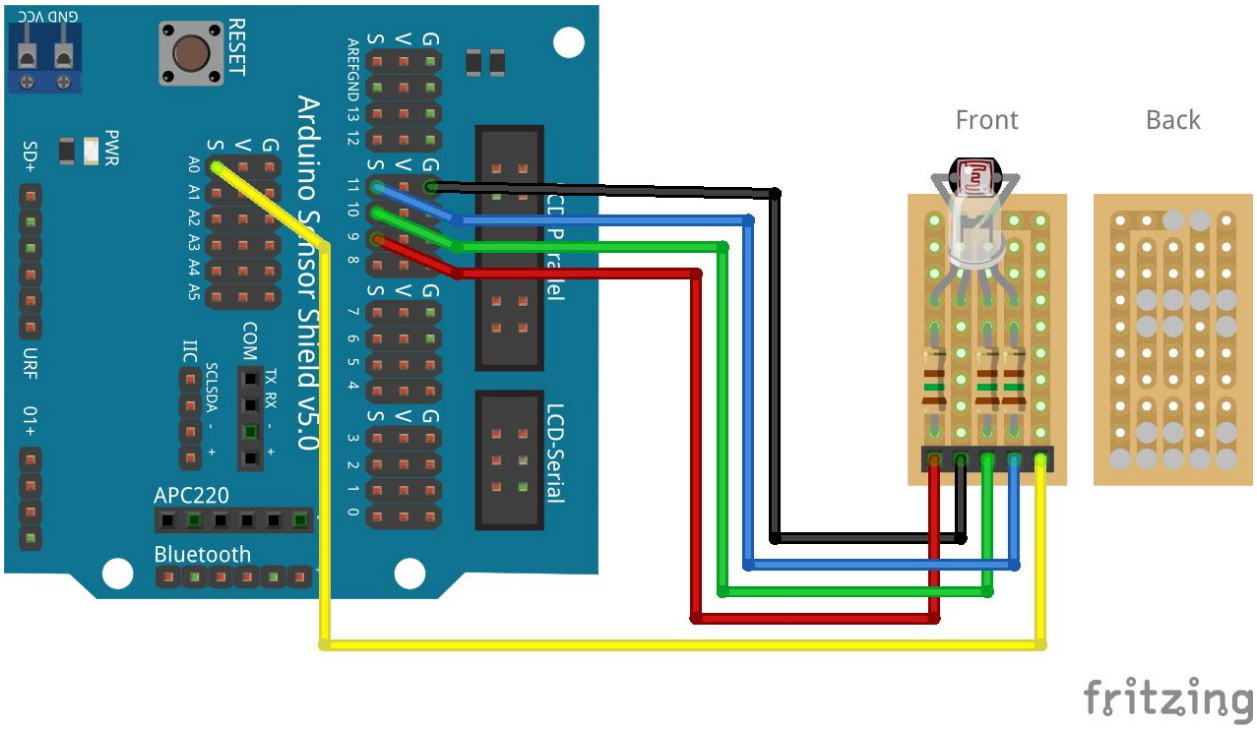
If you are using an RGB LED, connect it and the photoresistor to your Make-A-Pede as shown below:



If you are using three LEDs, connect them and the photoresistor to your Make-A-Pede as shown below:



After you test the circuit, you may wish to transfer it to a protoboard so you can move it to a different location on you robot, such as facing forward or at the ground. An example layout is shown below:



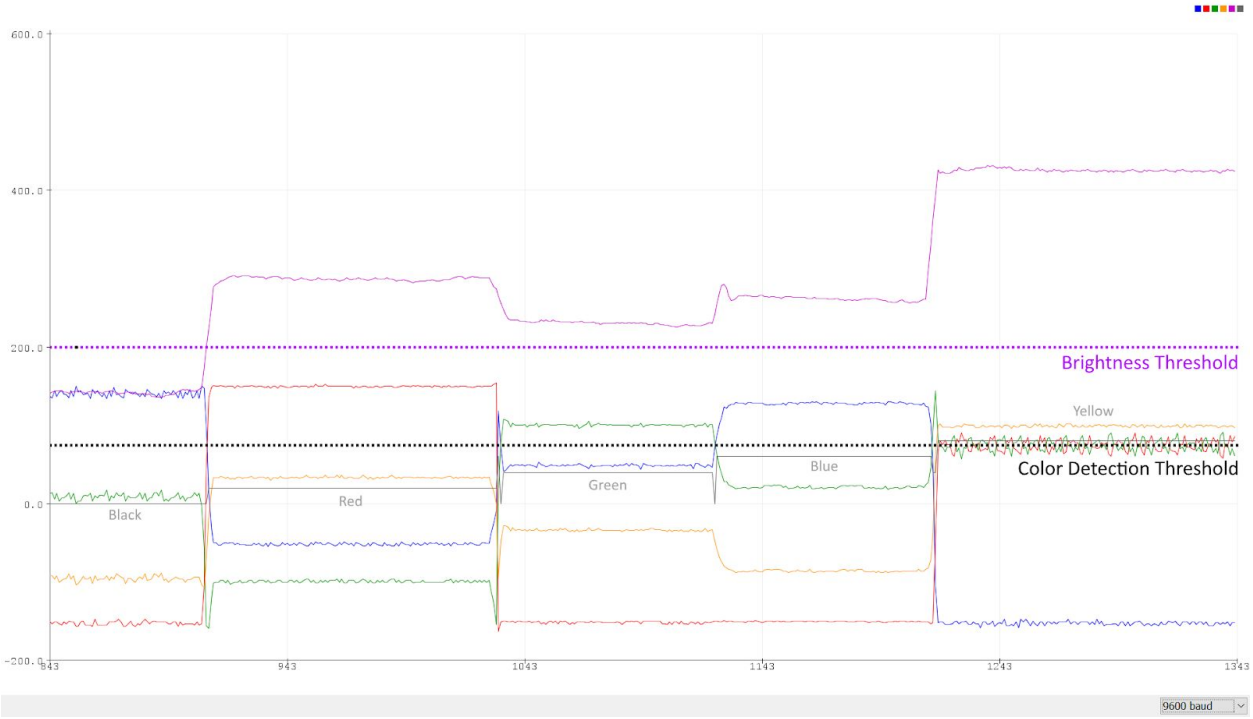
Light Shield

The light sensor needs to be separated by a divider from the LED(s) to prevent it from seeing the light coming directly off of them. This can be built using cardboard and tape, or 3D printed. Depending on the placement of your sensor, you may need additional shielding to block overhead light.

Testing

Plug your Arduino into your computer. Open the ColorDemo.ino program by opening the Arduino IDE and going to File → Examples → Make-A-Pede → ColorDemo. Place a white card under the sensor, upload the program onto your Arduino, and then open the serial plotter from the tools menu.

The LED will first calibrate itself using the card. It will then begin to rapidly cycle through red, green, and blue while reading the photoresistor to get the reflected light value for each color. Try placing different colored cards/objects under the sensor while observing the results on the serial plotter. Below is an example of what the plotter will look like:



The red, green, blue, and yellow lines show the measured value for their respective colors. The purple line shows the ambient brightness and the grey line shows the selected color.

The color detection threshold (used to decide what color the sensor is seeing) can be adjusted at the top of the program using the variable detectionThreshold. The brightness threshold (used to determine when the sensor is seeing black) is automatically adjusted based on the ambient lighting.

For an example of the color sensor in use, see the ColorDemoIR.ino program (found in File → Examples → Make-A-Pede → ColorDemoIR). This program uses the same calibration procedure as ColorDemo, but will perform different actions based on the detected color instead of using the serial plotter. ColorDemoIR also requires an IR distance sensor.

Programming

The color sensor is controlled using these commands:

`setupRGB(redPin, greenPin, bluePin);`

setupRGB is used to set which pins will be used to control the LED(s). Default values are 9, 10, and 11.

`setRGBColor(color, brightness);`

setRGBColor is used to set the LED to a specific color. Valid color inputs are:

0 or LED_OFF - Off
1 or LED_RED - Red
2 or LED_YELLOW - Yellow
3 or LED_GREEN - Green
4 or LED_CYAN - Cyan
5 or LED_BLUE - Blue
6 or LED_MAGENTA - Magenta
7 or LED_WHITE - White

The default color value is 0. Valid brightness inputs are 0-255 and can only be used if the LED is connected to PWM pins.

`analogRead(sensorPin);`

analogRead is used to read the brightness value of the photoresistor. Returned values are on a scale of 0 to 1023.

`pinMode(sensorPin, INPUT_PULLUP);`

This command is required to enable the internal pullup resistor for the photoresistor.