LED FABRIC BRACELET

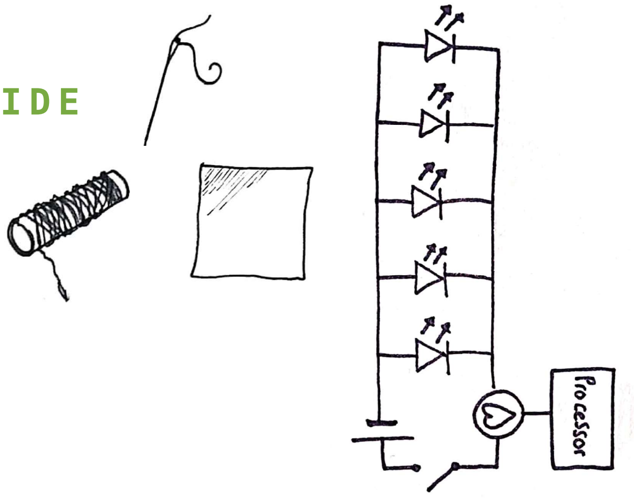
WITH A PULSE SENSOR

This tutotrial enhances the LED fabric bracelet from the previous tutorial, into a bracelet that responds to your heart rate! Using a pulse sensor and Arduino, you can create a bracelet that flashes at the same rate as your pulse!

What you need:

LED FABRIC BRACELET
ARDUINO UNO
COMPUTER WITH ARDUINO IDE
LED
CONDUCTIVE THREAD
NEEDLE AND THREAD
NORMAL FABRIC
CROCODILE CLIPS
PULSE SENSOR
VELCRO
CLEAR VINYL STICKER
GLUE GUN

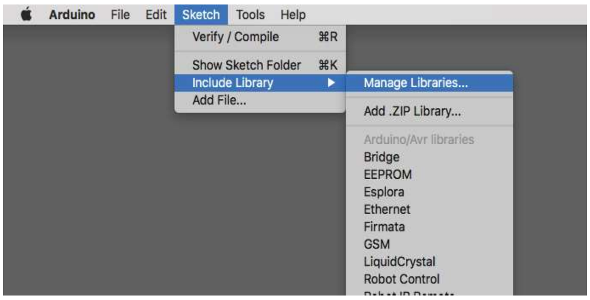
Circuit Diagram

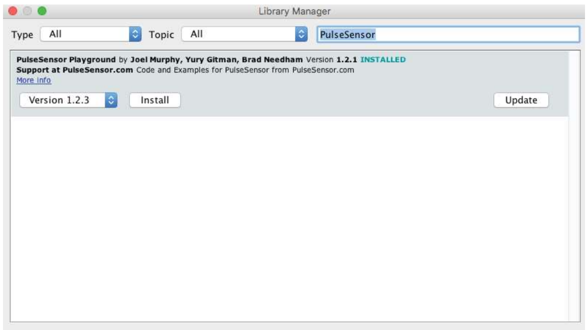


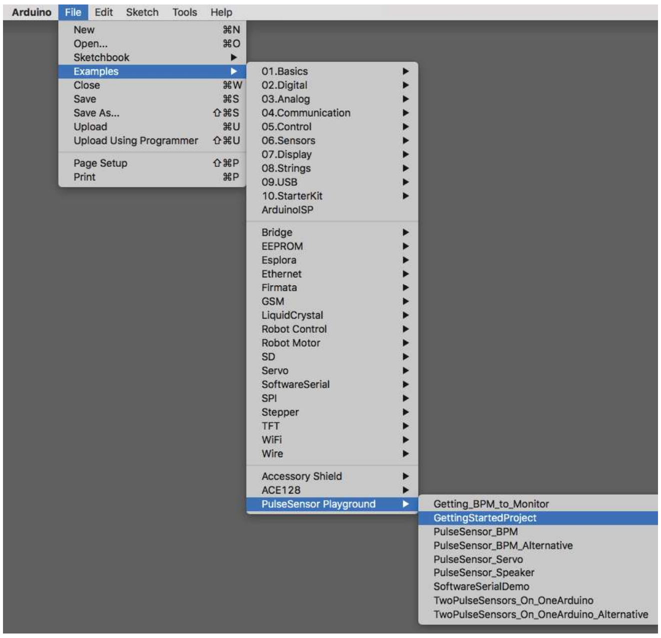
STEP 1

SETUP YOUR COMPUTER

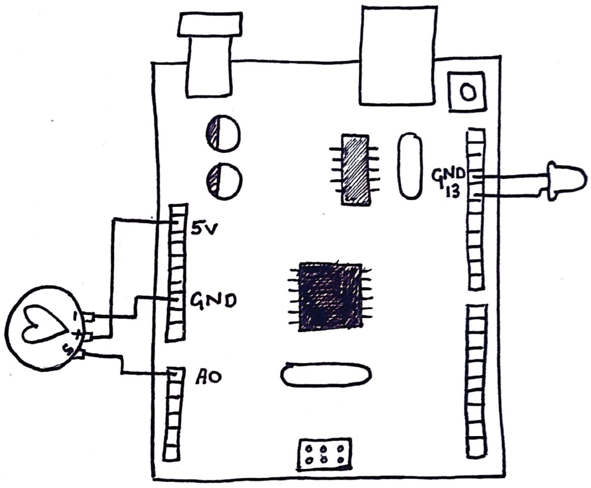
The first step is to prepare your computer and the code that you will run. Open up Arduino IDE and download the PulseSensor Playground library.
As shown in the screenshots: Sketch > Include Library > Manage libaries > (Search for PulseSensor Playground) > Install.
Once installed open up the GettingStartedProject code.
Files > Examples > PulseSensor Playground > GettingStarted Project.







STEP 2



BUILD A TEST CIRCUIT

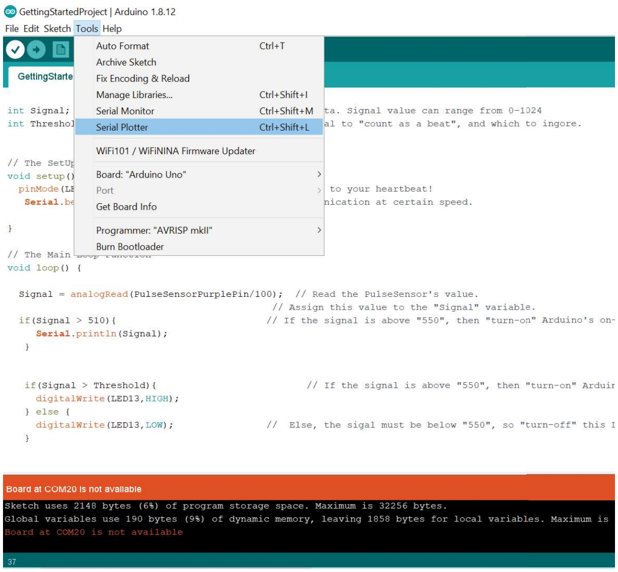
Now you need to test the code. To do this, the simplest way is to build a prototype circuit. With the Arduino, form the connections as shown in the illustration. This can be done by directly inserting the component into the ports or using jumper cables with a bread board.

LED negative leg, Pulse sensor negative = Ground

LED positive leg = 13

Pulse sensor positive = 5V

Pulse sensor signal (S) = A0



RUN THE CODE

Run the code by uploading it onto the Arduino board.

Make sure you have selected the correct board and COM port for your Arduino.

Tools > Board > Arduino Uno

Tools > Ports > COM...

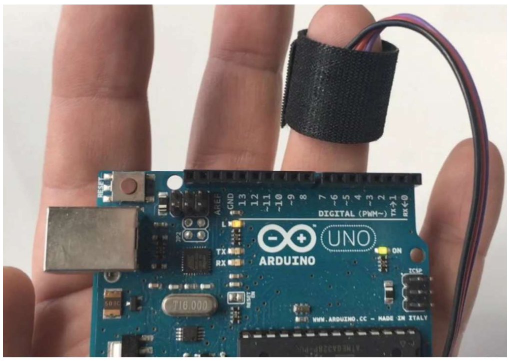
When you run the code, open the serial plotter to see the graph which is produced from the pulse sensor signal. Make sure to select 9600 baud

Tools > Serial plotter > 9600 baud

STEP 3

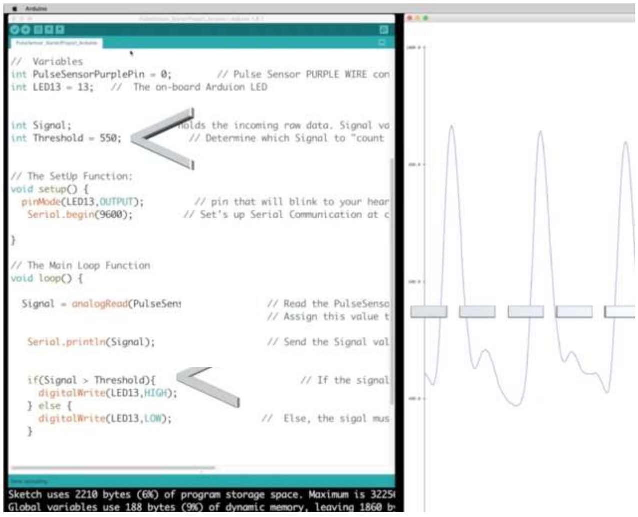
TEST AND MODIFY THE CODE

Set up the sensor as detailed in this document:



WWW.GENERATIONROBOTS.COM/MEDIA/DETECTEURDEPOULSAMPLIFIE/PULSESENSORAMPEDGETTINGSTARTEDGUIDE

Start with the sensor on your finger. Continue to watch the Serial plotter and once settled, move the sensor to find a good position on your wrist.
This will be where the sensor sits when you wear your bracelet.



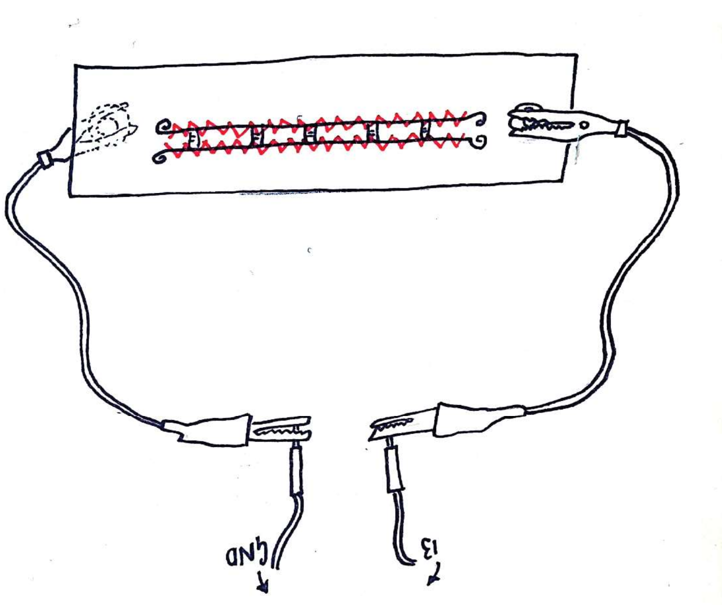
When moving the sensor there will be a lot of noise which will effect the results. Once it has settled again, you can modify the code to make it more accurate.

On the serial plotter, you should be able to see a spike for every heart beat. You can set a more accurate threshold value in the code as indicated by the arrow. Define the threshold value by reading off the serial plotter at which value is surpassed with every heart beat.

STEP 4

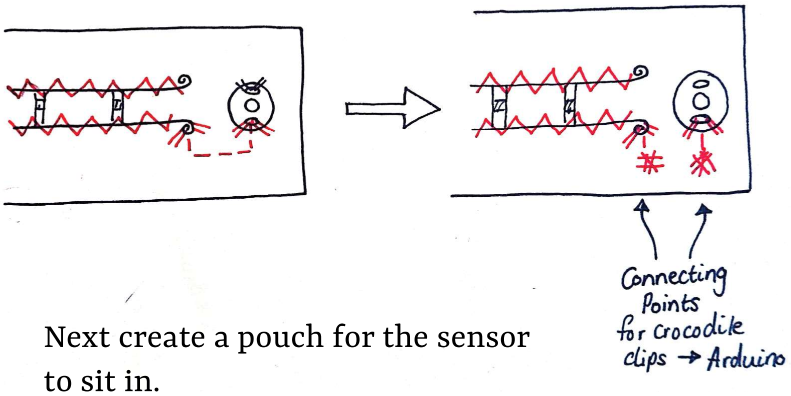
TEST WITH THE BRACELET

Now that the code is working well and the LED on the Arduino is flashing at the same rate as your pulse, you need to integrate the sensor with the LED bracelet.
First of all, test it will work using crocodile clips as shown in the illustration. Here you are completing the circuit between the two poppers with the Arduino circuit. Here, the bracelet is a replacement for the LED in the test circuit. This means that the crocodile clip connected to the positive popper needs to connect to pin 13 and the other popper to the ground using jumper cables.

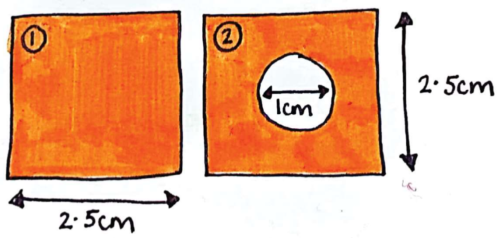


STEP 5 MODIFY THE BRACELET

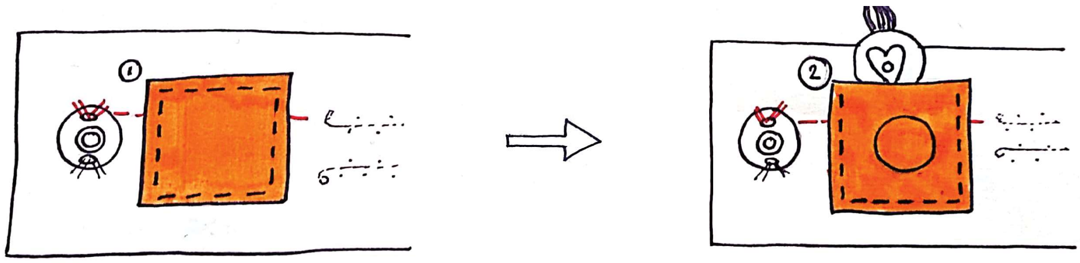
Now you know that it works, you can modify the bracelet. To do this, undo the conductive thread stitching between the thin copper wire and the popper.
Restitch two connecting points with conductive thread.



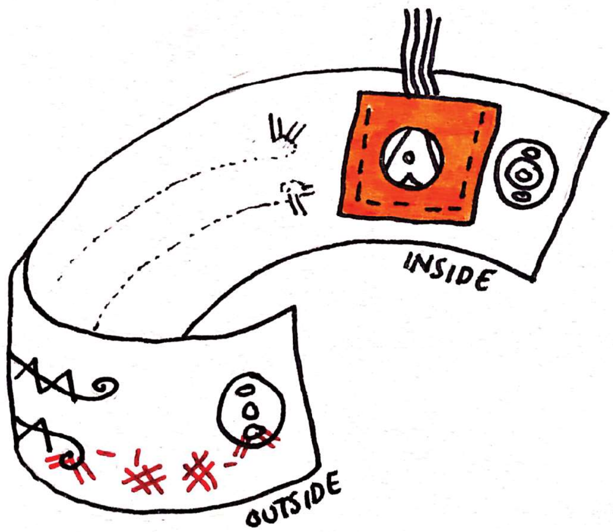
Next create a pouch for the sensor to sit in.



Cut out two squares of fabric (both around 2.5cm by 2.5cm). Cut a small circle in the centre of one of the pieces. This should be big enough for the sensor and light on the pulse sensor to be visible but not big enough for the sensor to fall through.



Sew down the full square piece first - this is to prevent short circuiting.
Then sew down the second piece with the circle cut out. Sew along only three sides as shown in the illustration. The pulse sensor should then be able to slide in when connected to the Arduino.

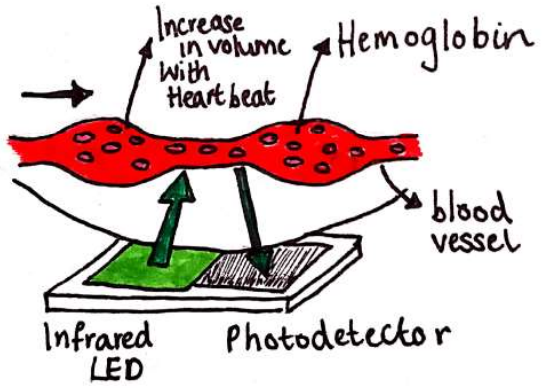


How it Works

In our blood there is oxygenated hemoglobin, which are types of cells that are carrying oxygen around our body. These can absorb light. This means that if there is more oxygenated hemoglobin in our blood more light is absorbed.

Your heart beat forces oxygenated blood around your body, so that the oxygen can be delivered to your organs. The blood is pushed round your body by blood vessels increasing and decreasing in volume.

With every heart beat, the vessels increase in volume, increasing the amount of oxygenated hemoglobin and increasing the amount of light that is absorbed.



The pulse sensor uses a green Infrared LED (the light source) and a photodiode sensor (a light detector). The infared LED shines infared light and the reflected light is then detected by the photodiode sensor.

The photodiode sensor generates a small voltage and current when light is shone onto it (like a miniture solar panel). The size of the voltage is proportional to the amount of light detected.

Therefore when the volume of the blood vessel increases, with every heart beat there is more oxygenated hemoglobin at that point. This means that more light is absorbed and so less will be reflected. The photodiode will sense less light being reflected and a lower voltage will be given out. When the volume of the blood vessel reduces after the heart beat, the amount of light absorbed is also reduced. More light is then reflected and picked up by the photodiode, giving a larger voltage.

This small change in voltage with every heartbeat can then be amplified through the circuit. In our circuit, the signal from the sensor is continuously being sent to the Arduino, and the amplified result is plotted on the serial plotter. The graph that is shown shows regular peaks and troughs with the repetitive variation in the amount of light being absorbed when the heart beats and the blood vessel volume changes.