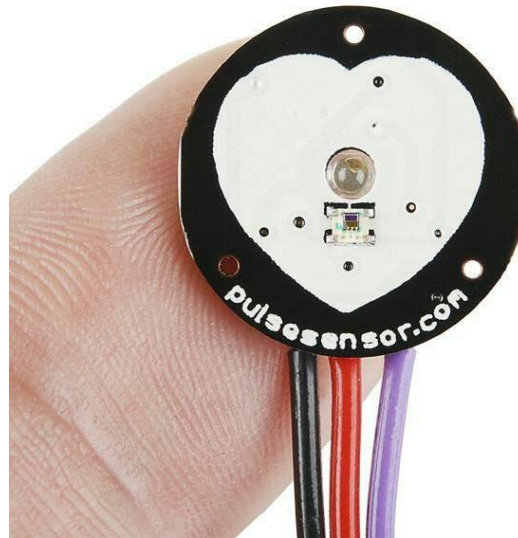


REF: B31-XD58C

XD-58C Pulsesensor Pulse Heart Rate Sensor



Description

The Pulse Sensor is a well-designed low-power plug-and-play heart-rate sensor for the Arduino. Anyone who wants to incorporate real-time heart-rate data into their work students, artists, athletes, makers, and game and mobile developers—can benefit from it.

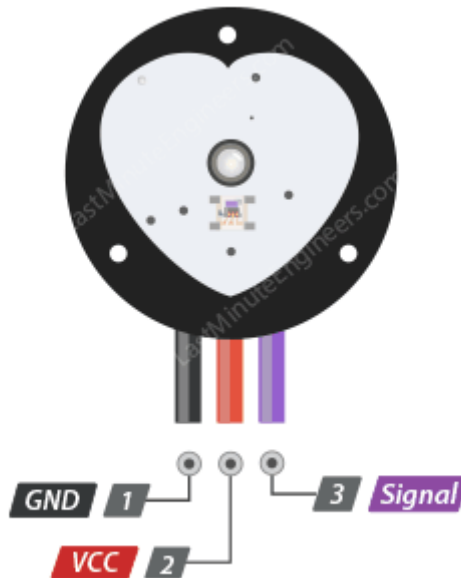
The best part is that this sensor plugs right into Arduino and easily clips onto a fingertip or earlobe. It is also super small (button-shaped) and has holes for sewing into fabric.

Specifications

Maximum Ratings	VCC	3.0 – 5.5V
	IMax (Maximum Current Draw)	< 4mA
	VOut (Output Voltage Range)	0.3V to Vcc
Wavelength	LED Output	565nm
	Sensor Input	525nm
Dimensions	L x W (PCB)	15.8mm (0.625")
	Lead Length	20cm (7.8")

Pinout

The sensor comes with a 24" flat ribbon cable with three male header connectors. The pinout is shown in the figure below.



Pin connection

Connecting the Pulse Sensor to an Arduino is a breeze. You only need to connect three wires: two for power and one for reading the sensor value.

The module can be supplied with either 3.3V or 5V. Positive voltage is connected to '+,' while ground is connected to '-'.' The third 'S' wire is the analog signal output from the sensor, which will be connected to the Arduino's A0 analog input.

The XD-58C Pulsesensor module brings out the following connections.

VIN - is the power pin. You can connect it to 3.3V or 5V output from your Arduino.

GND - is the ground.

S - wire is the analog signal output from the sensor, which will be connected to the Arduino's A0

ARDUINO PIN	XD-58C
3.3V	VIN
GND	GND
A0	S

Circuit diagram

Connecting the Pulse Sensor to an Arduino is a breeze. You only need to connect three wires: two for power and one for reading the sensor value.

The module can be supplied with either 3.3V or 5V. Positive voltage is connected to '+,' while ground is connected to '-.' The third 'S' wire is the analog signal output from the sensor, which will be connected to the Arduino's A0 analog input.

The following is the wiring diagram for the Pulse Sensor experiments:



Libraries

To run the following sketches, you must first install the 'PulseSensor Playground' library.

To install the library, navigate to Sketch > Include Library > Manage Libraries... Wait for the Library Manager to download the libraries index and update the list of installed libraries.

PulseSensor Playground by Joel Murphy,
Yury Gitman, Brad Needham

2.3.0 installed

Support at PulseSensor.com Code and Examples
for PulseSensor from PulseSensor.com

[More info](#)

2.3.0 ▼

REMOVE

Coding

To access the example sketches, navigate to File > Examples > PulseSensor Playground.

You will see a selection of example sketches. You can choose any of them to load the sketch

```
GettingStartedProject.ino
18 6) "Human-readable" code that is newbie friendly."
19
20 */
21
22
23 // Variables
24 int PulseSensorPurplePin = 0; // Pulse Sensor PURPLE WIRE connected to ANALOG PIN 0
25 int LED = LED_BUILTIN; // The on-board Arduino LED
26
27
28 int Signal; // holds the incoming raw data. Signal value can range from 0-1024
29 int Threshold = 580; // Determine which Signal to "count as a beat", and which to ignore.
30
31
32 // The Setup Function:
33 void setup() {
34   pinMode(LED,OUTPUT); // pin that will blink to your heartbeat!
35   Serial.begin(115200); // Set's up Serial Communication at certain speed.
36 }
37
38
39 // The Main Loop Function
40 void loop() {
41
42   Signal = analogRead(PulseSensorPurplePin); // Read the PulseSensor's value.
43   // Assign this value to the "Signal" variable.
44
45   Serial.println("Signal " + String(Signal)); // Send "reading " followed by the Signal value to Serial Plotter.
46
47
48   if(Signal > Threshold){ // If the signal is above "550", then "turn-on" Arduino's on-board LED.
49     digitalWrite(LED,HIGH);
50   } else {
51     digitalWrite(LED,LOW); // Else, the signal must be below "550", so "turn-off" this LED.
52   }
53
54
55   delay(20);
56 }
```

Result

In the Arduino IDE, choose Tools > Serial Plotter. You should see a wave similar to the image below, when you swipe your hand over the sensor.

