Flow:

1. Pulse sensor: BPM xd-58c

Oxigenated blood blah blah...(Dharmin!). BPM sensor gives integer output in the range of 0 to 700. We connected it to microcontroller. We used Arduino nano.

Connection of Arduino nano and pulse sensor:

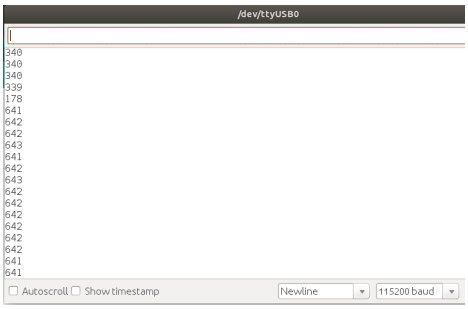
BPM ground --> ground
BPM VCC -----> 3.3 v
BPM Signal ----> Analog pin A0

2. Microcontroller: Arduino nano

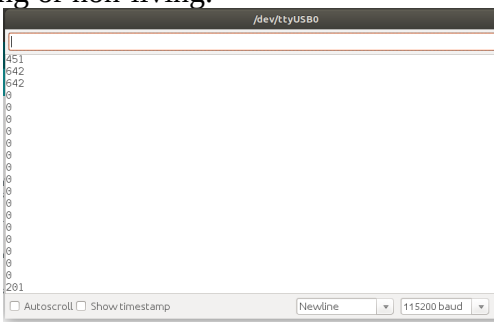
Micro controller reads Signal value from analog pin. The values are between 0 to 700. We analyzed the Signal variable.

Value of Signal variable:

1. ***Signal > 550*** Which is threshold value of BPM xd-58c. If Signal value goes beyond it, we show **disconnected.**

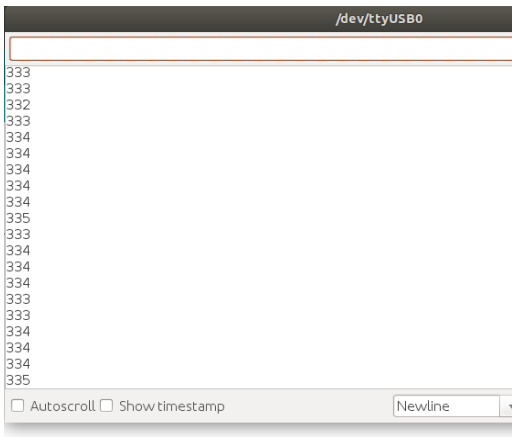


2. ***Signal < 10*** BPM sensor shows value below 10 when it is suddenly **connected** or get in touch to something living or non-living.

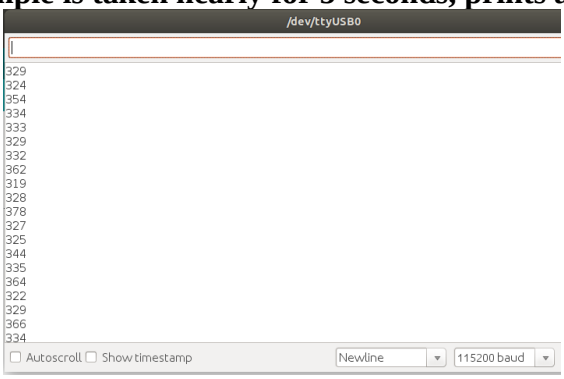


The above 2 observations dictate that the Sensor value is dependant upon intensity of incident ray and reflected way. Signal value might be subtraction of those two intensities.

1. ***Constant Signal value around 340:*** Which shows that there has been a **minimal noise** and difference between incident and reflection is constant. Which dictates that is **constant absorption of rays of bpm.** So, while bpm is connected to something but it **doesn't show any variation** in absorption. So, simply sensor is connected to a **non-living thing.**



2. ***Remarkable Variation in Signal value:*** Dictates **circulation of oxigenated blood** beneath the skin. A **pulse sample is taken nearly for 5 seconds, prints alive for a living.**



*There has been some error when suddenly connect to a living or a non living thing, but after 2 or 3 samples, the code shows the correct result.

3. Communication: Xbee s2c --- Xbee s2c pair.

Firstly, to establish communication between 2 xbees, we need to configure them. To configure, we need to set parameters of both Xbee devices accordingly. Guide of configuration of Xbee devices is mentioned after the flow.

At the Transmitter side, i.e. sensor side we have an End device Xbee. Connections of Xbee and arduino nano:

Xbee VCC (1) ---> 3.3 v (be careful, at 5 V xbee stops functioning PERMANENTLY)

Xbee Tx (2) -------> RX0

Xbee Rx (3) -------> TX1

Xbee ground (10) ---> ground

At the receiver side, we have two options, whether we want to see a continuous waveform of heart beats or just print dead or alive. For both the options, we need to have a Xbee at the receiver side.

In the second option, connect Xbee to your PC with Uno cable. (Use Explorer.) Open XCTU in your PC, Open Serial monitor and press CLOSE button. And output will be printed at the delay which we have set in the arduino code.

In the first option, we cannot see waveforms in xbee s2c. In online QnA forums, we found that a microcontoller is needed to program a xbee to print certain

waveforms. So, we connect another microcontroller at the receiver side. We can see a waveform on arduino ide serial plotter. Connections of Xbee s2c and microcontroller at receiver side:

Xbee VCC (1) ---> 3.3 v (be careful, at 5 V xbee stops functioning PERMANENTLY)

Xbee Tx (2) -------> RX0

Xbee Rx (3) -------> TX1

Xbee ground (10) ---> ground

Code for Transmitter side arduino and receiver side arduino are uploaded on GitHub. Below is the link:

<Link: >
<Link: >

Open Serial plotter to see the waveform.