DIY Arduino Pulse Sensor

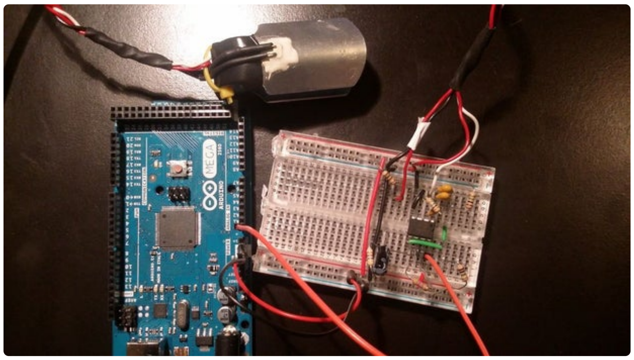
by ohoilett

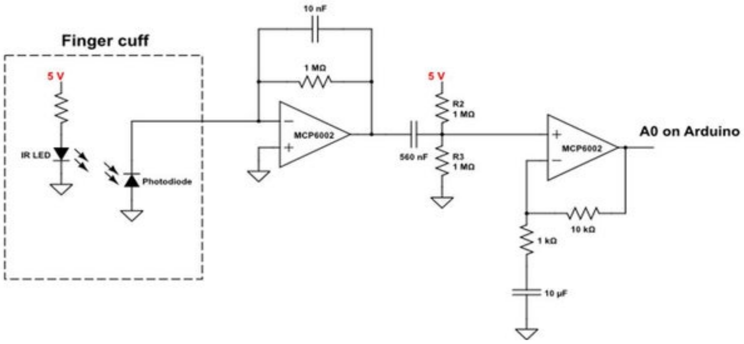
So I mentioned in an earlier Instructable that I am a teaching assistant (TA) for an introductory engineering course for biomedical engineering majors at Vanderbilt University. My main task as a TA is to re-write a few laboratory exercises that the students will be performed during the class. One of the laboratory exercises was to implement a simple circuit to measure the someone's heart rate using photoplethysmography.

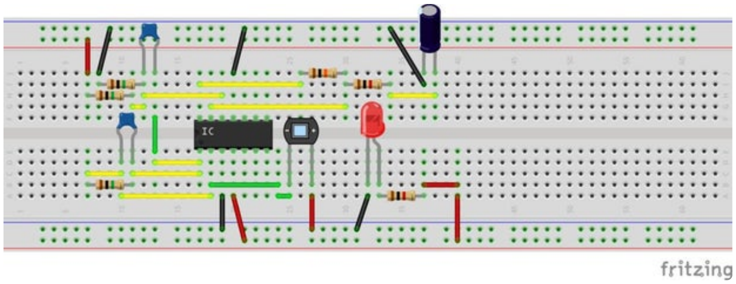
As blood is pumped through the body, the volume of blood in extremities such as your fingers increases and decreases with the pumping of the heart. The change in blood volume in the finger tips can be detected by shining a light through the finger and detecting the amount of light that passes through the finger using a photodiode. A photodiode is semiconductor that produces a current proportional to the amount of light that hits it. So when the blood volume in the fingers increases, less light is getting through the finger and hitting the photodiode.

The current produced by the photodiode is converted to a voltage by an amplifier and read by the Arduino. Here we go! I hope you enjoy reading and building!

*****NOTE: This is NOT a medical device.**







Step 1: Gather Your Tools

*****NOTE: This is NOT a medical device.**

You will need

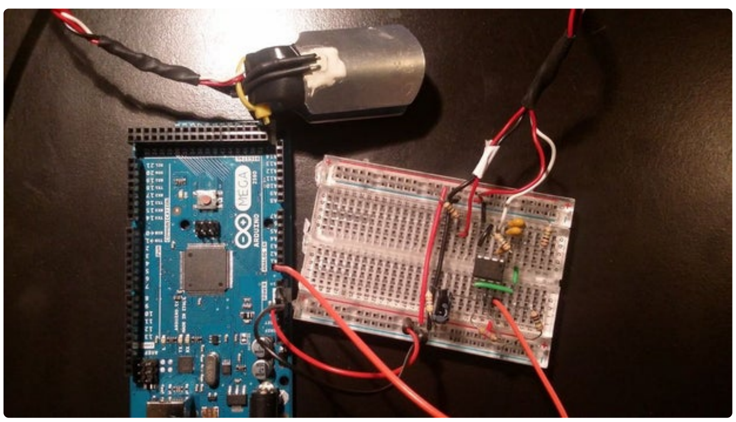
1 x Arduino (of course) (https://www.seeedstudio.com/Arduino-Mega2560-Rev3-p-695.html)
1 x General Purpose Op Amp
5 x Resistors
3 x Capacitors
1 x Photodiode
1 x IR LED

For the general purpose op amp, I used an MCP6002 which can be purchased from any major electronics distributor such as DigiKey.com, Mouser.com, or Newark.com. I purchased mine from Newark. I liked this op amp because it is single supply, meaning it only needs to be powered using ground (GND) and a positive voltage. It does not need a negative voltage supply like many other op amps. This enables it to be easily integrated with the Arduino which only runs on a positive supply. If you use a bipolar op amp, you have to make sure the op amp and Arduino are running on the same GND and positive supply. Do not connect the negative supply to the Arduino. You will fry something on it. I have done it before.

The resistor values that you will need are as follows:
3 x 1 MOhm resistors
1 x 10 kOhm resistor
1 x 1 kOhm resistor

The capacitor values that you will need are as follows:
1 x 10 nF
1 x 560 nF (I ended up using a 470 nF and 100 nF in parallel - which adds us to 570 nF - which is close enough)

For the photodiode and IR LED, the Bio-Instrumentation lab at Vanderbilt University made a few finger cuffs with integrated photodiode and IR LED, so we used those for the lab. You can purchase disposable finger cuffs online. You can make your own as well. You just need some sort of housing for the photodiode and IR LED to keep them really steady on someone's finger.



Step 2: Summary of Operation

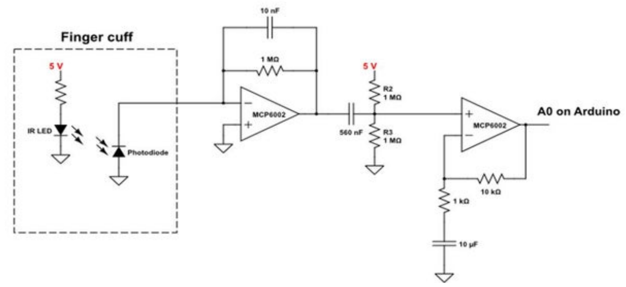
As the heart pumps blood through the body the volume of blood in extremities such as the fingers increases and decreases. We can detect the change in blood volume in the fingertips by shining a light through the finger and detecting the amount of light that passes through the finger using a photodiode.

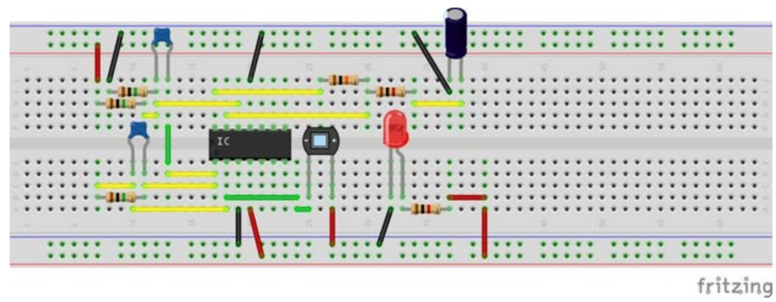
Step 3: Circuit Diagram

We will be creating the circuit depicted in the corresponding image. The circuit has three main parts:

1. Finger cuff
2. Transimpedance stage
3. High-pass filter
4. AC gain-stage

I will explain these various parts in the next sections.

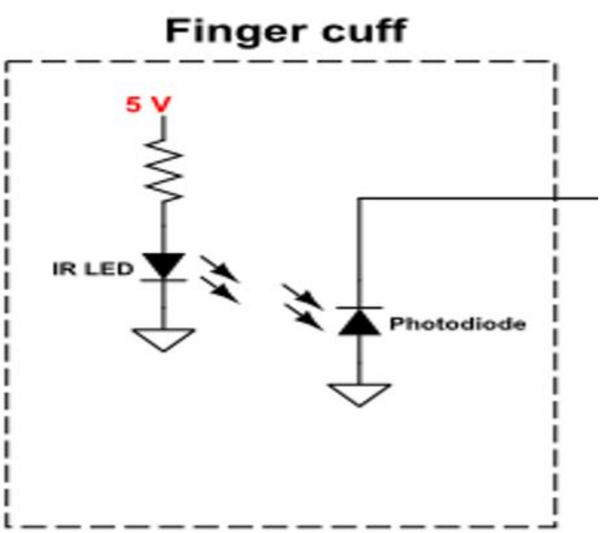




Step 4: Finger Cuff

As I stated before, the finger cuff has a photodiode (I don't know the part number) and IR LED integrated into it. Both components have wires that connect them to the Arduino.

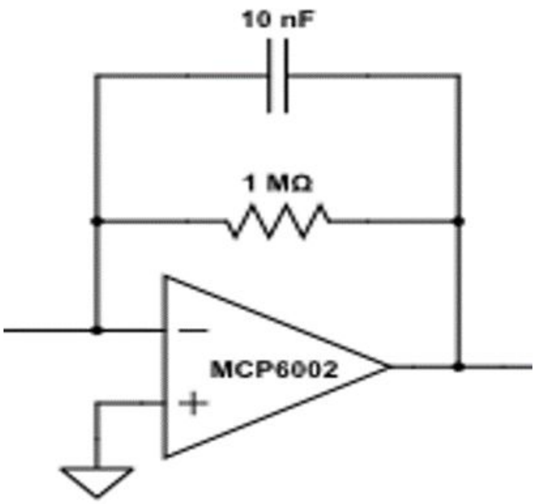
Please see Notes 1 and 2 in "Step 10: Troubleshooting."



Step 5: Transimpedance Amplifier

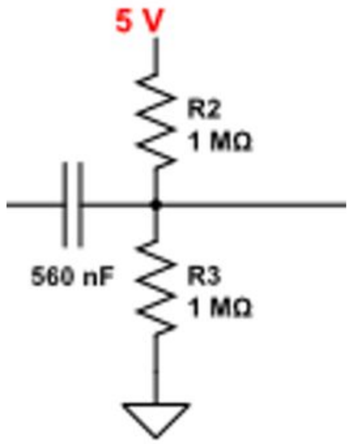
The next portion of our circuit is what is known as a transimpedance amplifier or current-to-voltage converter. A transimpedance amplifier is an amplifier configuration that allows us to convert a current to a voltage. In our case, the current produced by the photodiode flows through the resistor in the op amp's feedback loop. We will see a voltage at the output of the op amp with accordance to Ohm's Law, V = I*R. The current produced by a photodiode can be pretty small (micro-amps), so this is why we are using such a large resistor (1 MOhm) as the gain-setting resistor. A high gain-resistor such as this is typical for transimpedance amplifiers. The capacitor in the feedback loop helps reduce high-frequency noise.

Please see Note 2 in "Step 10: Troubleshooting."



Step 6: High Pass Filter

Our next stage is a high-pass filter. A high-pass filter allow us to remove low-frequency signals. In our case, we are removing signals that are slower than a normal pulse signal. Specifically, we are removing the DC bias from our signal. When the IR LED shines light through the finger, most of the light is absorbed by the tissue. Our circuit will in turn produce a voltage that corresponds to the absorbance of light by the tissue, not the pulsatility of the artery (which is what we really want). For this reason, we use a high pass filter to get only the absorbance due to the pulsatility of the blood in the arteries (our pulse).



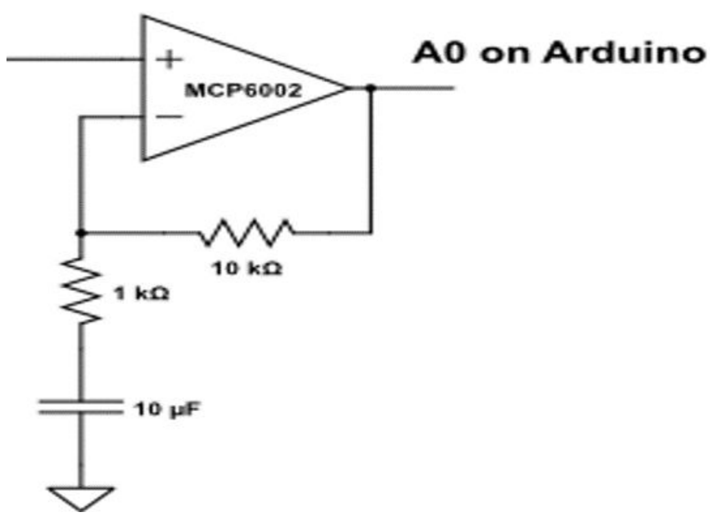
Step 7: AC Gain Stage

The final portion of our circuit is an AC gain stage using a non-inverting op amp. A non-inverting op amp amplifies and input voltage in accordance to the equation 1+R2/R1 where R2 is the resistor in the feedback loop and R1 is the resistor that is connected to ground. Notice the capacitor in the our non-inverting op amp. A capacitor blocks DC signals and only allows AC signals to pass. This means that only the AC signal which corresponds to our pulse gets amplified, not the Vref voltage that we applied in the previous stage.

This was a neat little circuits trick that was pointed out to me by MattDougan. Check his page out if you get a chance.

The signal at the output of our AC gain stage is read by the analog-to-digital converter on our Arduino.

Please see Note 4 in "Step 10: Troubleshooting."



Step 8: Software Interface

For the class that I am TA-ing for, I actually wrote a LabVIEW VI for this circuit. LabVIEW is a pretty powerful engineering program that allows for automation of experiments, data visualization, etc. It is extremely useful, but unfortunately proprietary so the average person does not have access to it. For this reason, I have slightly modified the Arduino graph ing example in Processing so that it will plot the pulse waveform. This Instructable is still a work in progress and there are a lot of things I would like to change. I would like to create a custom user interface for the viewing the pulse waveform, as well as the heart beat, but this should be pretty good for now. Included are the Arduino, Processing, and LabVIEW codes.

Enjoy!

If you have any questions, please feel free to ask.

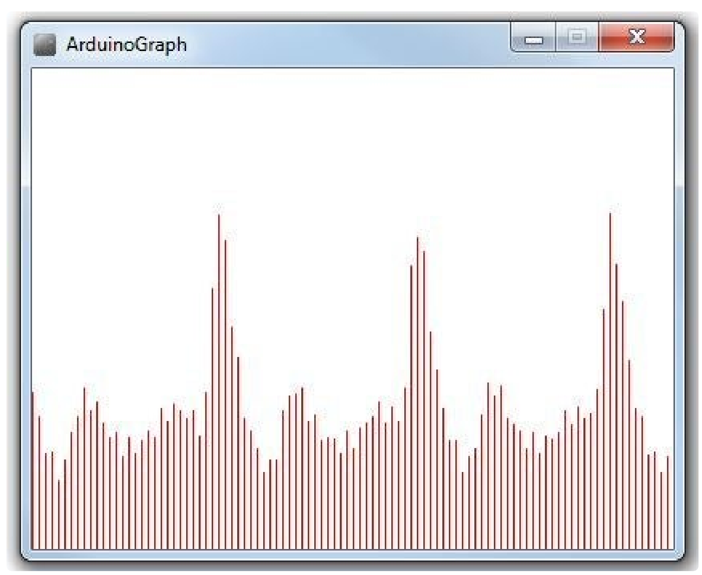
Project GitHub repository

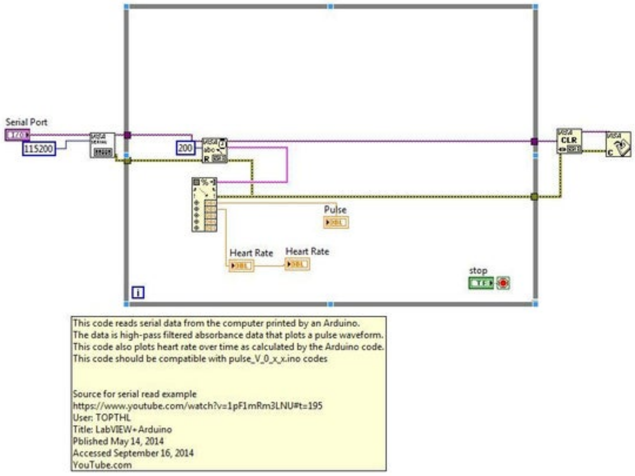
https://www.instructabl... Download

https://www.instructabl... Download

https://www.instructabl... Download

Step 9: A Few Screenshots

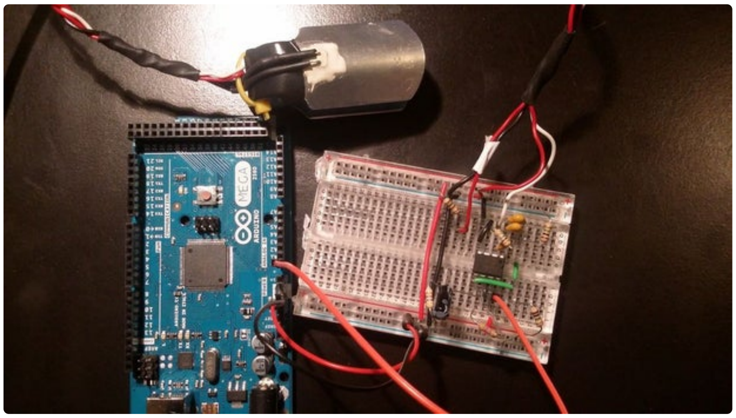




Step 10: Troubleshooting

After speaking with a few people who had problems getting the sensor to work, I have put together a list of common pitfalls and how to remedy them.

1. The positioning of your finger between the LED and photodiode is very critical to obtaining a stable signal. For the class that I developed this circuit for, we had the LED and photodiode fixed in a finger splint (as seen in the intro picture), which ensured stability.
2. You may need to decrease the brightness of the LED. With your set up, check the output of the transimpedance amplifier (Step 5). It should be within your power rails (lower than 5V and greater than 0V if powering with +5V and GND)
3. You may need to add more gain to your pulse sensor circuit. I would suggest this gain stage https://www.circuitlab.com/circuit/jp7tu98zj5x6/pu...
4. The high pass filter created by the 1k resistor and 10uF capacitor (Step 7) is not very well tuned for the pulse frequency. The cut-off frequency for the 1k resistor and 10uF capacitor is about 16Hz which is well above the pulse frequency of 1.2Hz. Unfortunately, I simply overlooked this when first making the circuit. That being said, the circuit still worked really well. I created several of them for the class I was TA-ing for and the students were able to get really good pulse data (Step 9). I think having the LED and photodiode enclosed in a finger splint really improved signal integrity which offsetted the effects of the miscalculated cut-off frequency. As a result, I think adding the extra gain stage as suggested in point 3 (above) should help correct for the miscalculated cutoff frequency. You could also modify the RC values for the high pass filter to be 1uF and 330k to create a high-pass filter cutoff of 0.5Hz, which will help preserve our pulse signal. Accordingly, you should change the value of the 10k resistor to be **at least** 330k. Any value higher than 330k will give you a pulse signal gain that is equal to the ratio of the two resistors.
5. Also the LabVIEW GUI does have a few bugs in it when it comes to reading in serial data, so it's a bit finicky. If you know how to properly set up the serial connection settings in LabVIEW, please share. Otherwise, the Processing GUI should work, it just looks boring. All else fails, the Serial Plotter option in Arduino would be the easiest. The Serial Plotter didn't exist when this project was first built.



Hi man nice work there, i have to build a pulse oximeter for the last exercise for my degree! I used a lightsensor OPT101 with an ifra-red led, not a photodiode can this run?

The OPT101 actually contains a photodiode and a transimpedance amplifier, so you're good.

Can I directly replace MPC 6002 with LM358. I have a few of them with me.

Hi, I'm having trouble asking you to provide transmitter and receiver sensors

hello,

I do not want to use ArdiNo, because my teacher told you to use the breadboard, maybe guide me

Hello, where should I add the extra gain stage?

Hello. I've built the circuit but can't get it to work.

How to adjust the LED signal and photodiode?

The whole scheme is tightly fixed

Using TL072

3 x 1 MOhm resistors

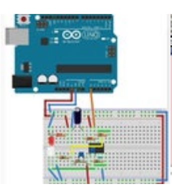
1 x 10 kOhm resistor

1 x 1 kOhm resistor

1 x 10 nF

1 x 560 nF (470 nF and 100 nF)

Ir led and Photodiode






Sorry for the stupid question. Tell me, here all non-polar capacitors ? Or 10 uf polar?

No problem. The 10uF capacitor is polar, all others are non-polar.

thank for you

Yeah dude no problem.

hello. help me how many watt Resistors and

Capacitors?

The standard 1/4 Watt resistors will be fine.

Capacitors aren't generally described in wattage capability. The 10V capacitors should be fine. Peak current shouldn't be a problem considering the pulse sensor is pretty low current device.

Thank you..for your work

Thank you for reading!

I'm having trouble downloading the pulse displays located on step 8, what are some alternatives if I want to display the pulse from the arduino?

If I am going to use the NTE928M (LM358) do I need to worry about negative supply? Or will I implement it just like I would If I were use the MCP6002?

Thank you!

There is a device called seelablet which works on ubuntu. Is it possible to connect the output from opamp directly to that device.

why is it showing values even if i dont have finger between ir and reciever?

Because the LED is still hitting the photodiode which produces a current.

Hello ,i am using tcrt5000 instead of the photodiode and a IR led .Does this change the arduino program that you have made and if yes how ?

I replied to your direct message. Check your inbox.

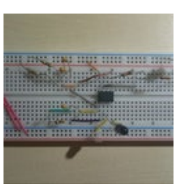
But for everyone else's benefit, it should be okay. I would suggest a dual supply op amp like the LM741 or the LM358 instead of the MCP6002. If you are familiar enough with op amps, you will have to appropriately bias the transimpedance amplifier to prevent "negative" voltages. If that doesn't make any sense, go with the LM741 or LM358.

This the circuit that i use.its exactly as your circuit but the only difference is that i use tcrt5000.I have buit this circuit more than 15 times and use your arduino code, but for some reason it doesnt work and i have tried both mcp6002 and lm358.Every time i open the serial monitor too see the pulse value ,it shows first the absorbance which is around 1,30 in my case and then after the "," it shows the value of the pulses(bpm)to be 190 or less and it doesnt change when i put my finger on the sensor .i am not sure id the problem is the code or the circuit but i cannot fix it. can help me with that????? maybe suggest me an other programm for measuring bmp(i care only about the serial monitor not the processing.Thanks

Hey I did revise my circuit a bit, but it seems like it will not work like this.

Im using the MCP6002 now with the same resistors and capacitors like you.

It still shows me values like 190 in the graph. Can you look over it another time? That would be great

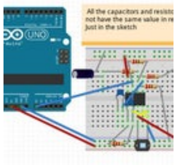


Hi. Im a german student who wants to build your Pulse Sensor but there
are some difficulties that I'm not able to solve. I hope you can help me
with it.

My Pulse Sensor does not work and I really dont know why. In the Plotter it only shows 0 as a pulse or 190 all the time.

Could you look over it and maybe tell me if I do something wrong?

It's my first time working with Fritzing so I'm sorry for the mess.



Hi,
I'm happy to help. A few questions.
1.) Which op amp are you using?
2.) What are the values of your resistors and capacitors?
3.) How familiar are you with circuit diagrams

Many of these issue are a bit hard to describe. I would be happy to do a Google Hangout with you and go over the circuit if you would like.

A few things I noticed.
1.) You need to power the TL072 op amp with a positive and negative voltage. Not 5V and GND. Maybe +5V and -5V.
2.) The (+) lead of the photodiode should be grounded. ---- Currently, you have it placed at the non-inverting pin of the op amp.
3.) The (-) lead of the photodiode should be placed at the inverting node of the op amp. ---- Currently, you have it placed at the inverting node, the negative supply of the op amp, and the GND of the Arduino.
4.) The non-inverting pin of the op amp should not be connected to the 5V.
5.) Your second capacitor is not connected to the output of the first op amp.
6.) The rest of these issues are a bit difficult to describe so I would just say look over the second part of your circuit. The components are not connected to where they should be.

Hi thanks for the reply:).

(1) I'm using the TL072CP op amp

(2) Values are the same as yours but splitting the 560 nF to 470 nF and 100 nF

(3)Hard question actually. I would assume it's okayish, but my knowledge about circuit diagrams are not the best.

To the solutions. I actually didn't know that I need to power the TL072CP op amp with a positive and negative voltage.

(5) That's actually just a dumb mistake because of my mess I did there so in reality that one is not there.

(6) I will.

So my problem now is how to put up a negative voltage. Since I thought the TL072CP is the same as the MCP6002, but the MCP6002 probably doesnt need negative voltage, does it?

And this question may be a bit stupid, but what does non inverting/inverting pin actually mean? Is it the meaning of IN- and IN+. To be honest I thought that this would mean negative input and positive input..

And to google Hangout. I'd like to write with you private, but maybe tomorrow, because it's already really late in Germany. And my english talking skill are not the polished so I dont prefer talk.

I would also suggest checking out this circuit. https://www.instructables.com/id/Heart-Sensor-With-AutoAdjusted-Threshold-and-Heart/ It has a better gain response.

1) That's fine
2) Good
3) Hopefully the Fritzing diagrams make things easier

5) Cool
6) Cool

The non-inverting pin of an op amp is the input labeled with a "IN+"
Similarly the inverting pin of an op amp is the input labeled with a "IN-"

Hi, I do have some difficulties in finding a circuit to produce a negative voltage for the Arduino. So will it be the better choice to buy a op amp like the MCP6002?

Yes. I recommend the MCP6002 because it is a single supply op amp and can be easily powered form an Arduino. The LM358 should work as well.

can i use the op amp TL072CP?

Yeah. That'll be fine.

I'm getting a lot of noise, what would you recommend doing to get rid of it?

you can try building a bandpass filter from 0.8-3.0Hz, this would effectively block out all frequencies outside of the heart beat rage.

Can you take a picture?

Thank you very much... :-) just did'n understand why same answer 5 times? :-P

Oh. Haha, sorry about that. My browser was acting up I guess.

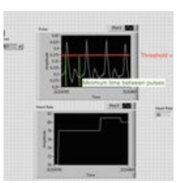
Can i use LM324 or TL074 instead of MCP6002 as it's not available here... If i can then is there any change required in the circuitry. One more thing i would like to know that can i use SEN-00114 emitters and detectors or TCRT5000 optical sensor. If not what should be the alternative

All those should work fine. Just realize the TL074 needs a negative power supply.

Also, I think the TCRT5000 has an emitter and a detector.

hii, I was not able to understand the code. There are some terms like threholding,refractor period and absorbance. You commented that we need to first find value of thersholding, how can we find it

Maybe this will also help



Thresholding - the threshold is the minimum signal amplitude (how high the signal is) for us to count a heart beat. If the signal is below a certain level, we will not count a heart beat. If it goes above that level, we will count a heart beat. You have to look at the amplitude (y-axis) of the pulse waveform in the LabVIEW or Processing sketch to determine what should be the threshold.

Absorbance - absorbance is the relationship between the intensity of the light emitted by the LED and the intensity of the light detected by the photodiode. The code plots these values to show the pulse waveform.

Refractory period - minimum time between heart rates. Heart rate will be at most 4 Hz (and 4 Hz is really, really fast for a human heart rate), so at least every 250 ms. The code takes account of that. If the code detects a heart beat at time x, it will wait 250 ms before it tries to detect another heart beat.

can we use ua741cp for same resistor values??

Yes, but certain other parts of the circuit may need to change since the UA741 needs +5 V and -5 V to operate. But it should be okay if you power it correctly.

can we use ua741cp for same resistor values??

Which IDE you have used to run above given code. LABVIEW is IDE or what . can i use another IDE. reply me on jadhaoakash99@gmail.com

Upload the Arduino code using the Arduino IDE. View the data using Processing.org or LabVIEW.

Which IDE you have used to run above given code. LABVIEW is IDE or what . can i use another IDE. reply me on jadhaoakash99@gmail.com

SIR,COULD YOU PLEASE MAIL ME THE CODE FOR HEART RATE SENSOR,I AM USING TCRT5000 SENSOR.MY EMAIL ID IS kingmakerboss786@gmail.com