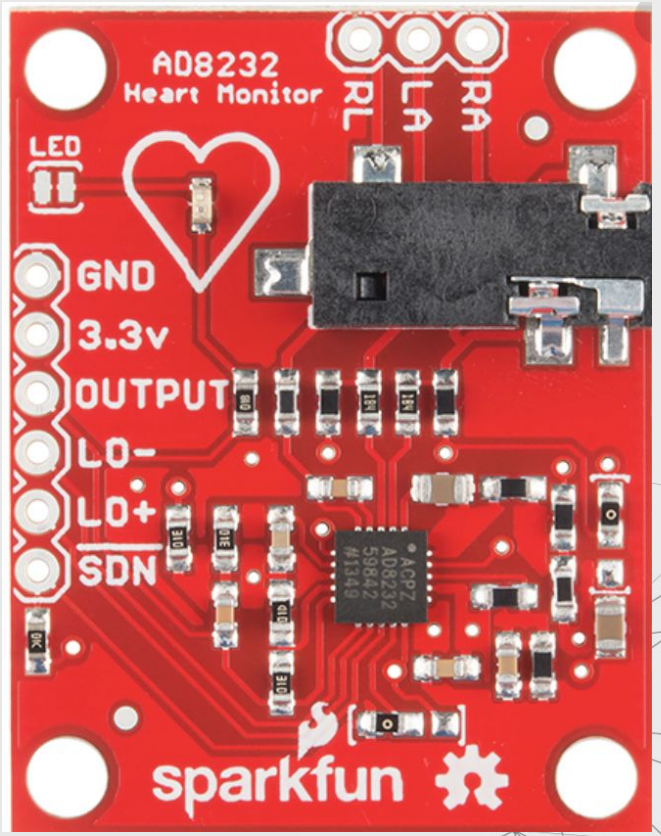
Phase 1: Heart Monitor

Aya Farag 900160580

TABLE OF CONTENTS

About the Project

For my project I’ve chosen to implement a simple heart monitor. The set requirements is that I have to develop it using the STM32 Module and the ECG Sensor, and using both I should be able to collect ECG signal, and report it to the PC over USB link, and the data received should be graphed , I have to report the heart beat rate, set a sampling rate and collect one minute worth of data. The ECG data must be graphed and I should provide UI elements, to select COM port, and baud rate, and UI elements to set a sampling rate.

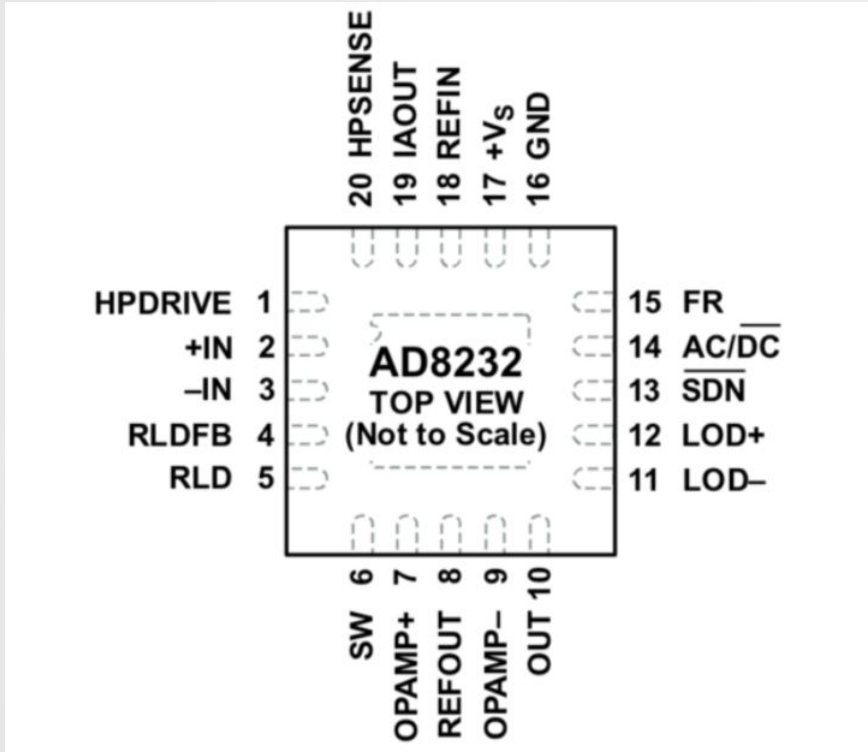


Hardware/Pins

The ECG sensor outputs an analog signal, so I will use an ADC 3202/On Chip ADC, to convert the signal coming out of the ECG sensor before it is sent to the Microcontroller.

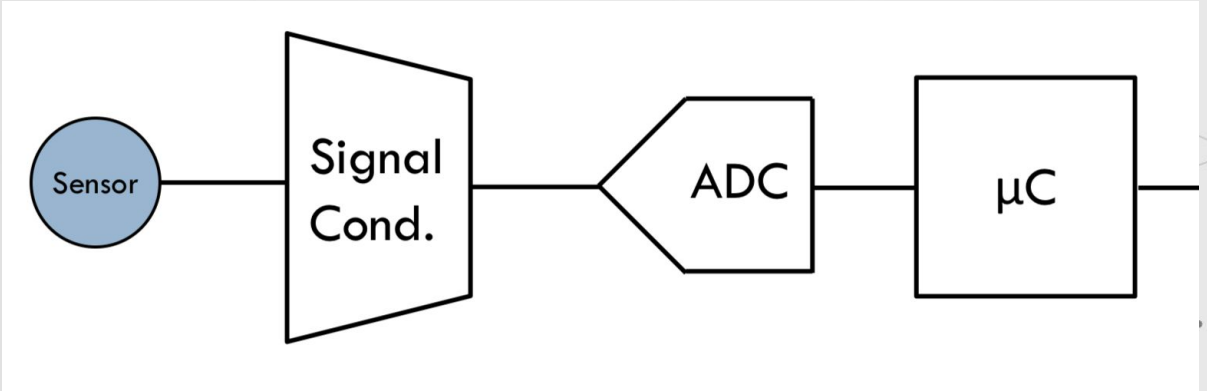
The ECG sensor has 15 pins. I will only be using 3-pins, GND, 3.3v, output. LO-, LO+, and SDN. LO-, and LO+ are leads -off comparators, LO- is always low, and LOD+ will be high. To detect leadoff, the ECG monitors the impedance between each differential-sensing electrode and the lead- off electrode. The impendence measurement provides an input for measuring the respiration rate.

The pint OUT outputs the fully conditioned heart. Rate signal which will be connected to an ADC. The also wont be using the SDN pin, because its useful only for low-power applications. So the LO-, LO+ and OUT pins will be inputs to the PC.



ADC

Since the Heart Monitor outputs an analog signal ,the signal had to first be converted into a digital signal in order to be processed, therefore I used one of the 2 ADC’s located on the stm32 microcontroller , and it's connected to pin A0. Data is transmitted through this pin and then I read the values at a rate of 2 Beats Per Second, therefore it's currently running on 2HZ.



Obtaining Data for One Minute

I currently obtain data for one minute only using polling,and using SysTick. So what I do is I have a variable called myTicks and I increment it in the systick handler, and in the main loop I check if the counter reached 1 minute and if it does I stop obtaining data from the ADC.

Next Step: Do it through interrupts rather than polling.

Calculating BPM

I currently obtain data for one minute only using polling,and using SysTick. So what I do is I have a variable called counted and I increment it if the value coming is not 0. So it checks for high values.

Code Walkthrough

```
while (1)
{
  if(myTick<60000)
  {
    HAL_ADC_Start(&hadc1); //Start ADC
    HAL_ADC_PollForConversion(&hadc1,100); // wait for conversion to complete
    adc_value = HAL_ADC_GetValue(&hadc1); //get the value
    HAL_ADC_Stop(&hadc1); // stop adc
    HAL_Delay(500);
    write= ((float)adc_value *3.3) / (1.0*(pow(2,12)-1.0));
      if(adc_value!=0)
        counter++;
    sprintf(out,"%f",write);
    out[6] = ' ';
    HAL_UART_Transmit(&huart1,(uint8_t *) out,sizeof(out),10);
    HAL_UART_Transmit(&huart1,&r[0],1,10);
    if(myTick >= 60000)
    {
      sprintf(out2,"%d",counter);
      HAL_UART_Transmit(&huart1,(uint8_t *) e,sizeof(e),10);
      HAL_UART_Transmit(&huart1,(uint8_t *) out2,sizeof(out2),10);
    }
  }
```

PySerial- Com Ports and Baud Rate

```
import tkinter as tk
import time
COM = input("Enter Com port: ")
br = input("Enter BaudRate: ")
ser = serial.Serial(COM ,baudrate = br,timeout = 1)
ser.flushInput()
ser.flushOutput()


while True:
    line = ser.read(ser.in_waiting)
    if line == b'':
        time.sleep(1)  # be sure we don't try too often to request data
        continue
    else:
        line = line.decode('ascii')
        print(line)
```

To Do	Set Due Date
Get enough Data for a minute to get Beats per minute, and view it + Taking COM Rate and Baud Rate from user	May 9th,2020
View Heart Beat Signal-Graphed	May 15th,2020
Set sampling Rate / Report/ Presentation	May 16th , 2020
Documentation on GitHub	May 18th, 2020
Finish Project/Finalized	May 20th,2020

DEMO

Github Link: https://github.com/ayashaker98/Embbeded-Project-REP

Thank you