BMEN 428/ECEN 489
CSCE 489/BMEN 689

Biopotential Sensing

Please **SIGN IN** at the front!

Turn in all **POST LABS** at the front!

Safety

If you have any questions at all, or feel uncomfortable performing the experiments please let the TA know!

Make sure that whenever you have electrodes connected to your body, your device is connected to a portable battery or personal computer (unplugged laptop), **NOT TO THE OUTLET**.

In other words, there should **NEVER** be an electrical path from your body to the wall outlet. If any of the TAs see you not complying with these guidelines, you will get an

AUTOMATIC *F* GRADE

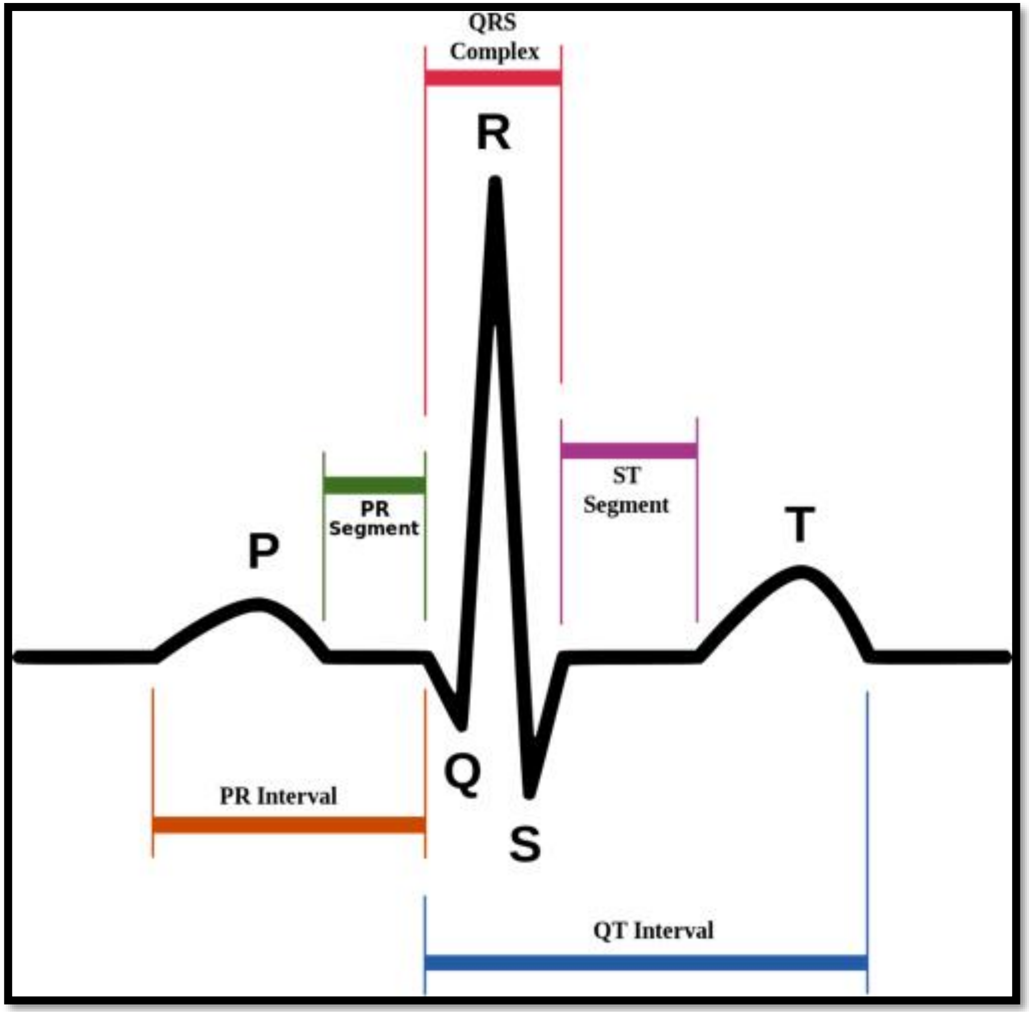
Previous Labs

This lab will make use of material and concepts previously learned.

It would be good to first make sure that you can get your code from previous labs working, in particular:

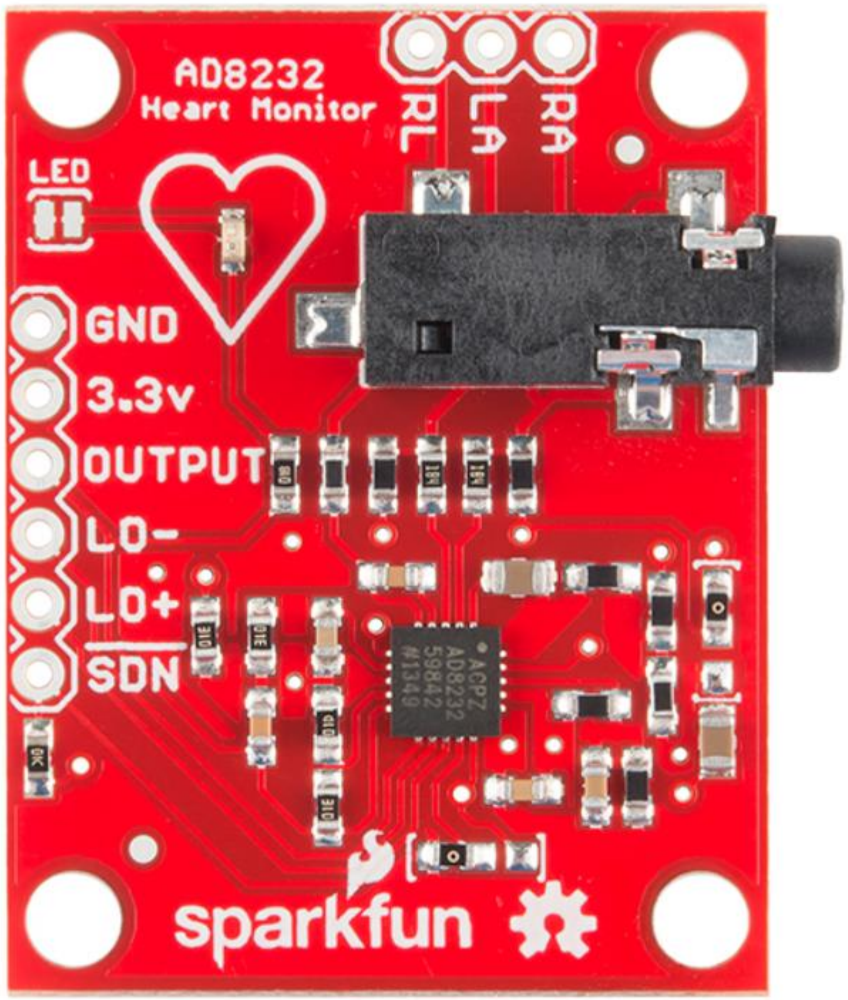
- **ADC, Wireless Communication**

ECG Signal


QRS
Complex
R
PR
Segment
ST
Segment
P
T
Q
S
PR Interval
QT Interval

SparkFun Single Lead Heart Rate Monitor

ECG Output Port



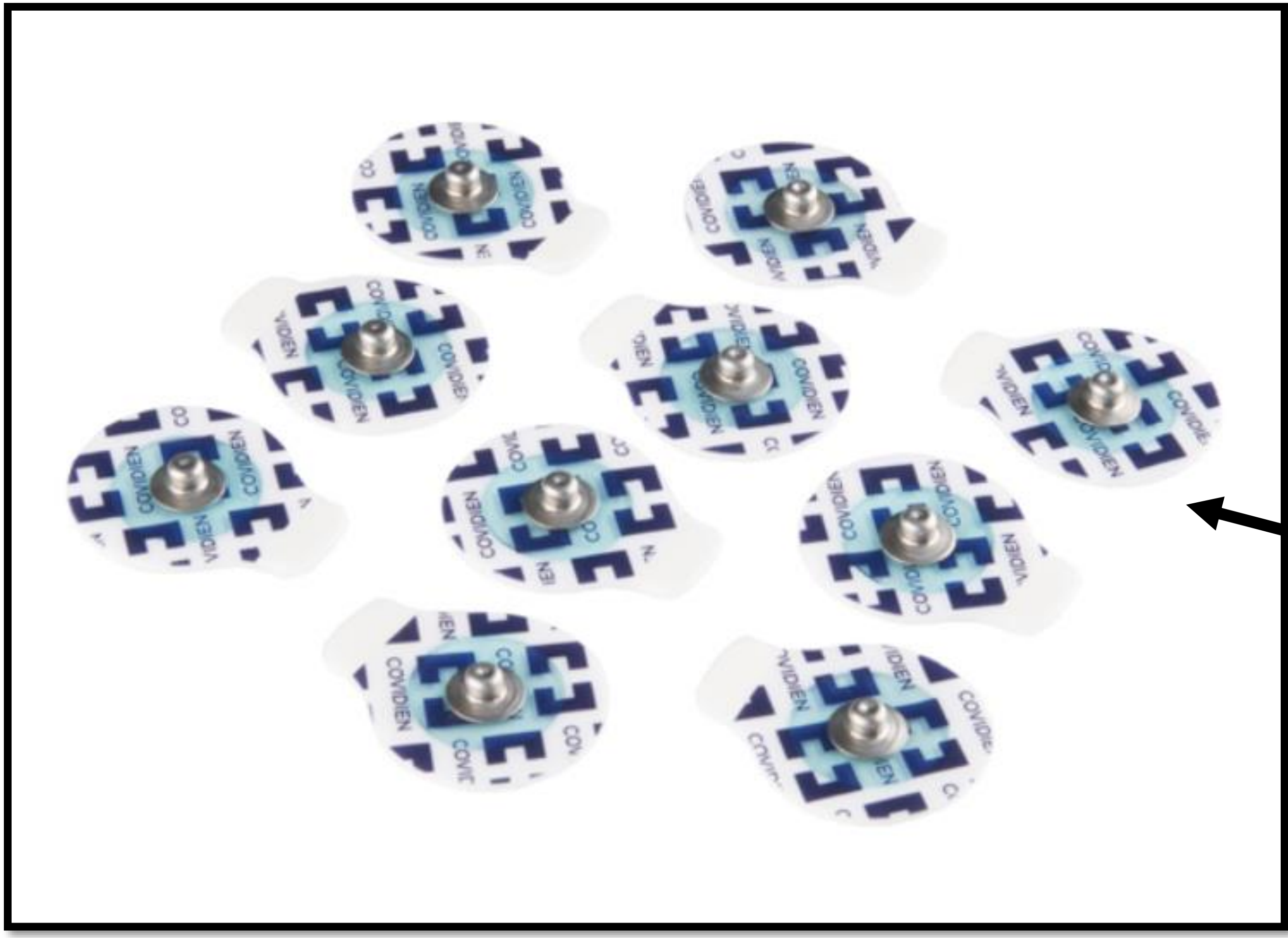
Electrode Pin Connection

AD8232

Sensor Cable - Electrode Pads (3 Connectors)

Make sure you match these to the "hook-up guide" when you place them in your body!

Biomedical Sensor Pads



These need to be attached to each electrode connector!

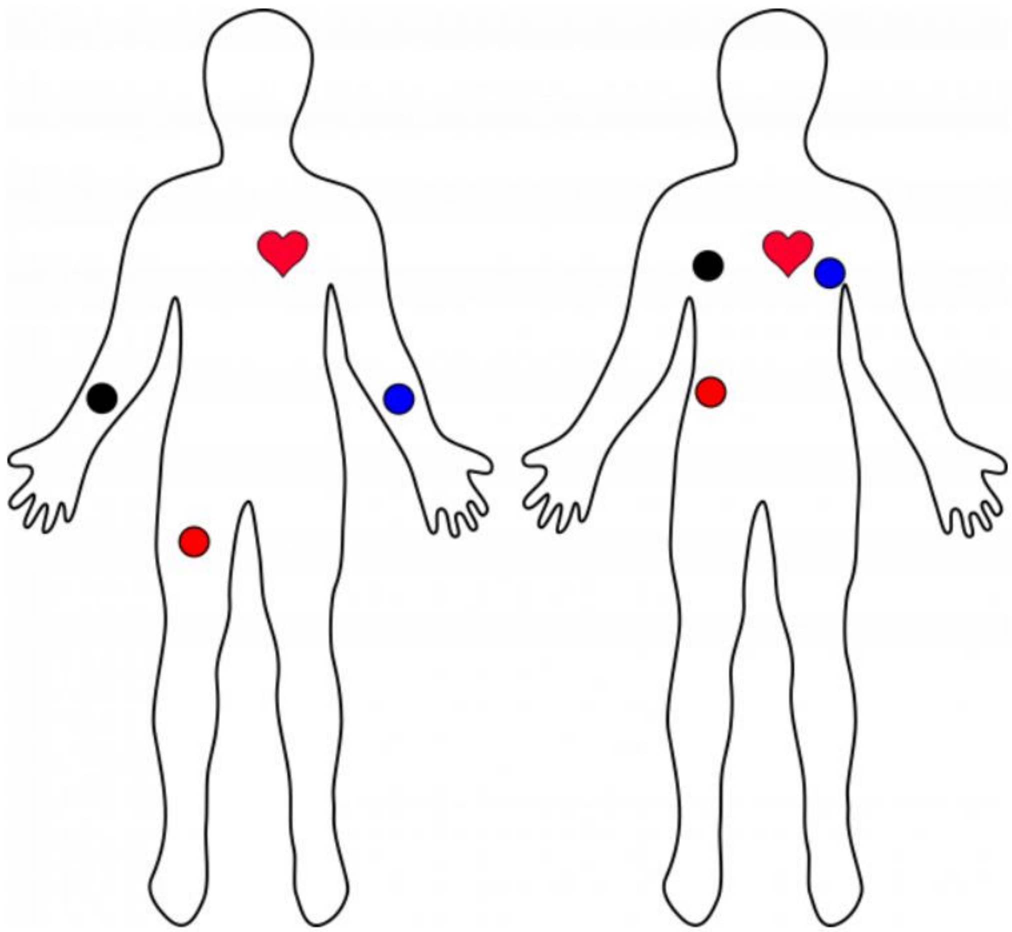
They should make “click” sound.

iMuto Portable Battery

When measuring ECG signals, your MSP430 should be connected to the battery, NOT the computer!

Typical Sensor Placement Locations

Pay attention to where the RIGHT and LEFT sides are! Your signal may be inverted!



Typical Sensor Placements

ECG Measurement Advice

Try to stay as calm as possible…

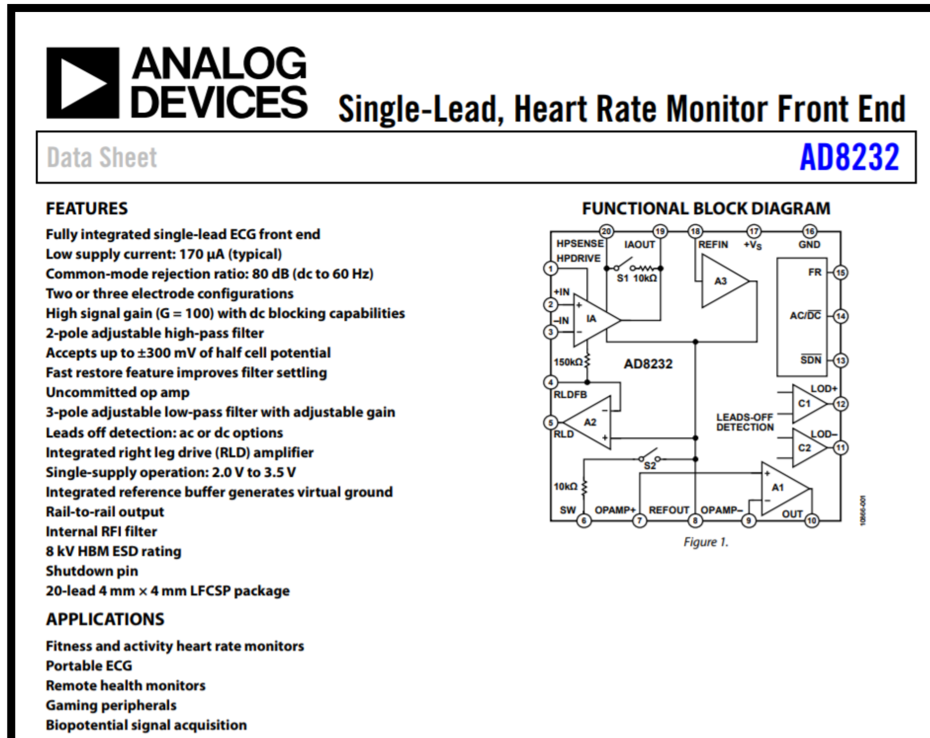
Do not move! The AD8232 is very sensible to motion artifacts, so any slight movements will distort your ECG signal.

Maintain a constant breathing rate, as taking deep breaths can slightly change the conductivity around the heart and you'll get a different ECG waveform.

ADC Configuration

Read the datasheet of the AD8232 board to learn the output signal level in order to ensure you set up the range of the ADC on the MSP430 accordingly.

https://cdn.sparkfun.com/datasheets/Sensors/Biometric/AD8232.pdf


ANALOG DEVICES

Single-Lead, Heart Rate Monitor Front End

Data Sheet AD8232

FEATURES

- Fully integrated single-lead ECG front end
- Low supply current: 170 μA (typical)
- Common-mode rejection ratio: 80 dB (dc to 60 Hz)
- Two or three electrode configurations
- High signal gain (G = 100) with dc blocking capabilities
- 2-pole adjustable high-pass filter
- Accepts up to ±300 mV of half cell potential
- Fast restore feature improves filter settling
- Uncommitted op amp
- 3-pole adjustable low-pass filter with adjustable gain
- Leads off detection: ac or dc options
- Integrated right leg drive (RLD) amplifier
- Single-supply operation: 2.0 V to 3.5 V
- Integrated reference buffer generates virtual ground
- Rail-to-rail output
- Internal RFI filter
- 8 kV HBM ESD rating
- Shutdown pin
- 20-lead 4 mm × 4 mm LFCSP package

APPLICATIONS

- Fitness and activity heart rate monitors
- Portable ECG
- Remote health monitors
- Gaming peripherals
- Biopotential signal acquisition

FUNCTIONAL BLOCK DIAGRAM

Figure 1.

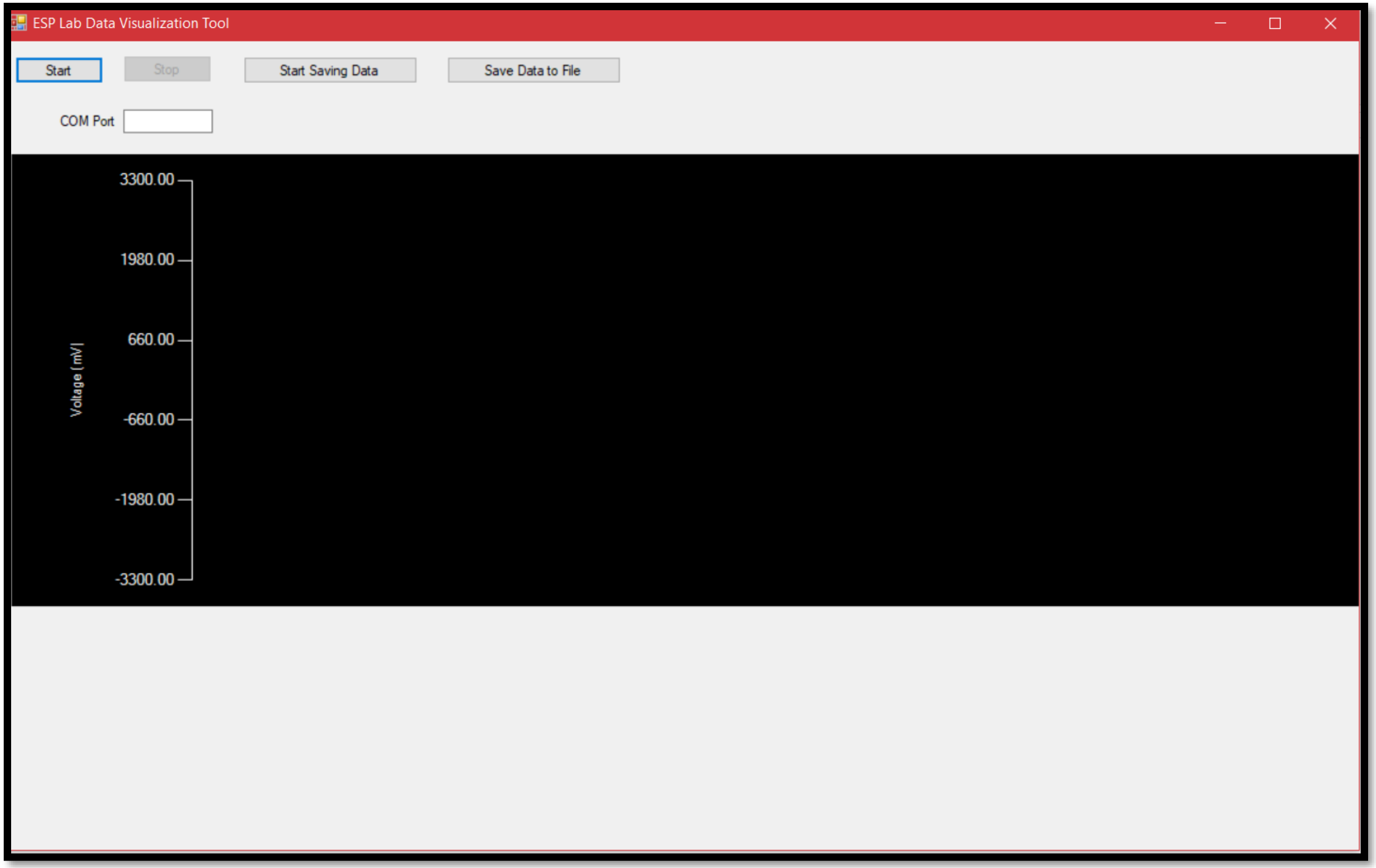
ADC Sampling Rate

Ensure that your sampling rate is high enough to detect the heart rate from the digitized ECG signal output from the ADC.

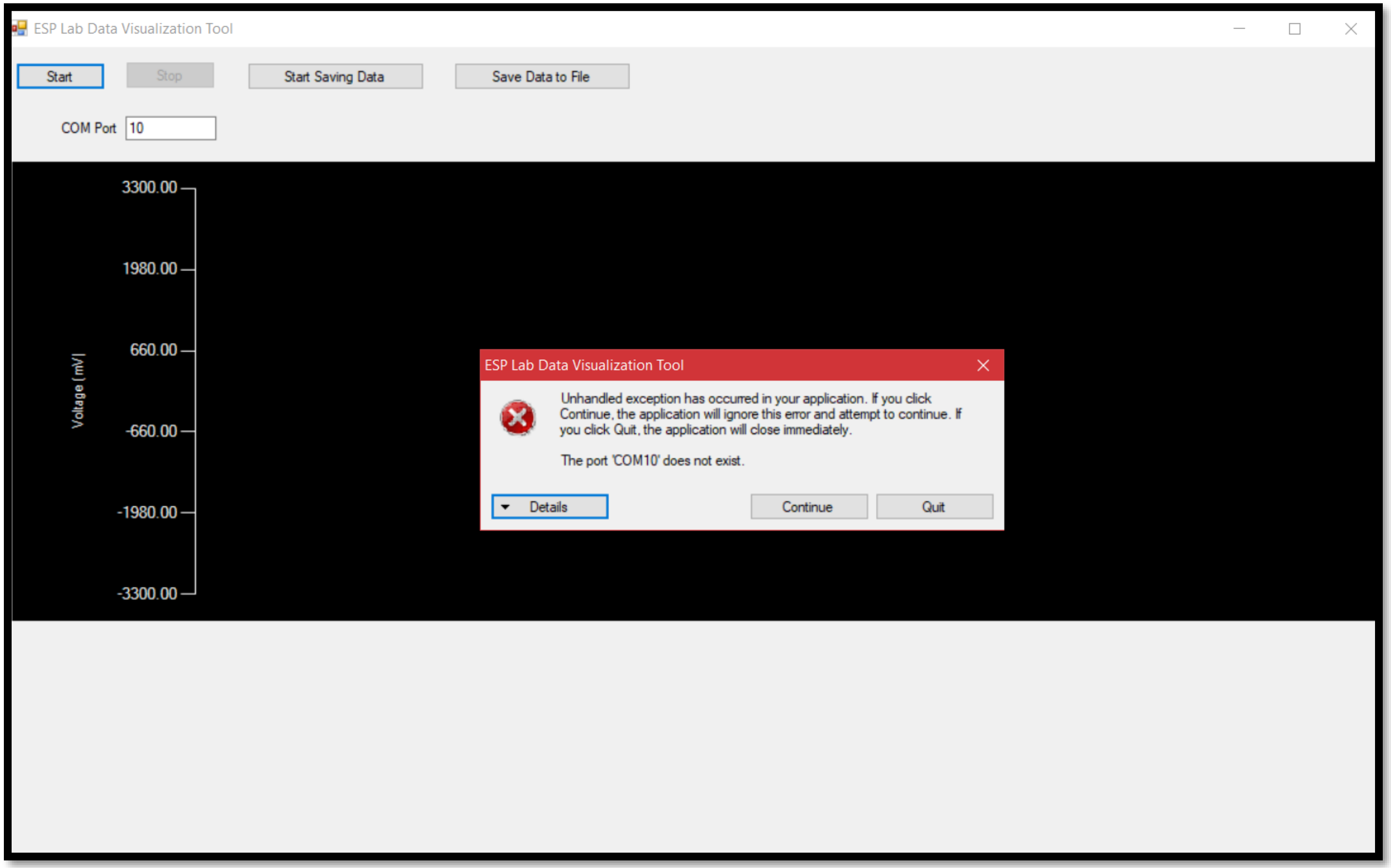
Ideally, you want your sampling rate to be as high as possible to capture all the frequency content of the ECG signal. However, there is a limiting factor here due to the requirement to transfer data in real-time to the PC.

The UART data speed will remain fixed at 115200 bits per second, so you will need to ensure the sampling rate is slow enough that the UART has enough time to transfer the previous sample to the PC before the next one arrives.

PC GUI - SimpleSerial.exe



SimpleSerial.exe - “ERROR”



Reasons? No successful communication established, different baud rate settings, etc…

Sending Integer Types Through Bluetooth

IMPORTANT

MSP430 saves "int" variable types in 16 bit long registers, however, your ADC has a resolution of 10 bits, meaning that the sampled output of the ADC is 10 bits long (in straight binary).

You can successfully send "int" type variables through Bluetooth as long as their variable length is 8 bits long. This includes zero-padding, i.e. 0000 0010 instead of just 0010.

Because we are not sending "char" type variables as well, the receiving host will not need to change its conversion protocol.

Sending Integer Types Through Bluetooth

TYPE	Size	Representation	Minimum	Maximum
char,signed char	8	ASCII	-128	127
unsigned char,bool	8	ASCII	0	255
short, signed short	16	2's complement	-32768	32767
unsigned short	16	Binary	0	65535
int, signed int	16	2's complement	-32768	32767
unsigned int	16	Binary	0	65535
long, signed long	32	2's complement	-2147483648	2147483647
unsigned long	32	Binary	0	4294967295
long long,signed long long	64	2's complement	-9223372236854775808	9223372236854775807
unsigned long long	64	Binary	0	18446744073709551615
enum	16	2's complement	-32768	32767
float	32	IEEE 32-bit	1.175495E-38	3.402823E+38
double	32	IEEE 33-bit	1.175495E-38	3.402823E+38
long double	32	IEEE 34-bit	1.175495E-38	3.402823E+38
pointers,references, pointer to data members	16	Binary	0	0xFFFF
MSP430x large-data model pointers,references,pointer to data members	20	Binary	0	0xFFFFF
MSP430 function pointers	16	Binary	0	0xFFFF
MSP430X function pointers	20	Binary	0	0xFFFFF

SimpleSerial.exe - Settings

You will need to ensure that the outgoing data from the microcontroller is of the right format for the GUI to interpret correctly.

Make sure that the digitized data from the ADC is in the ‘straight binary’ format (see User’s Guide for details).

The UART can transmit only one byte at a time, whereas the data is at least 10-bit as you are using the ADC10. So you will need <<two successive>> UART transfers for each sample; ensure that for each sample you send the most significant byte first to the PC.

Communication Flow

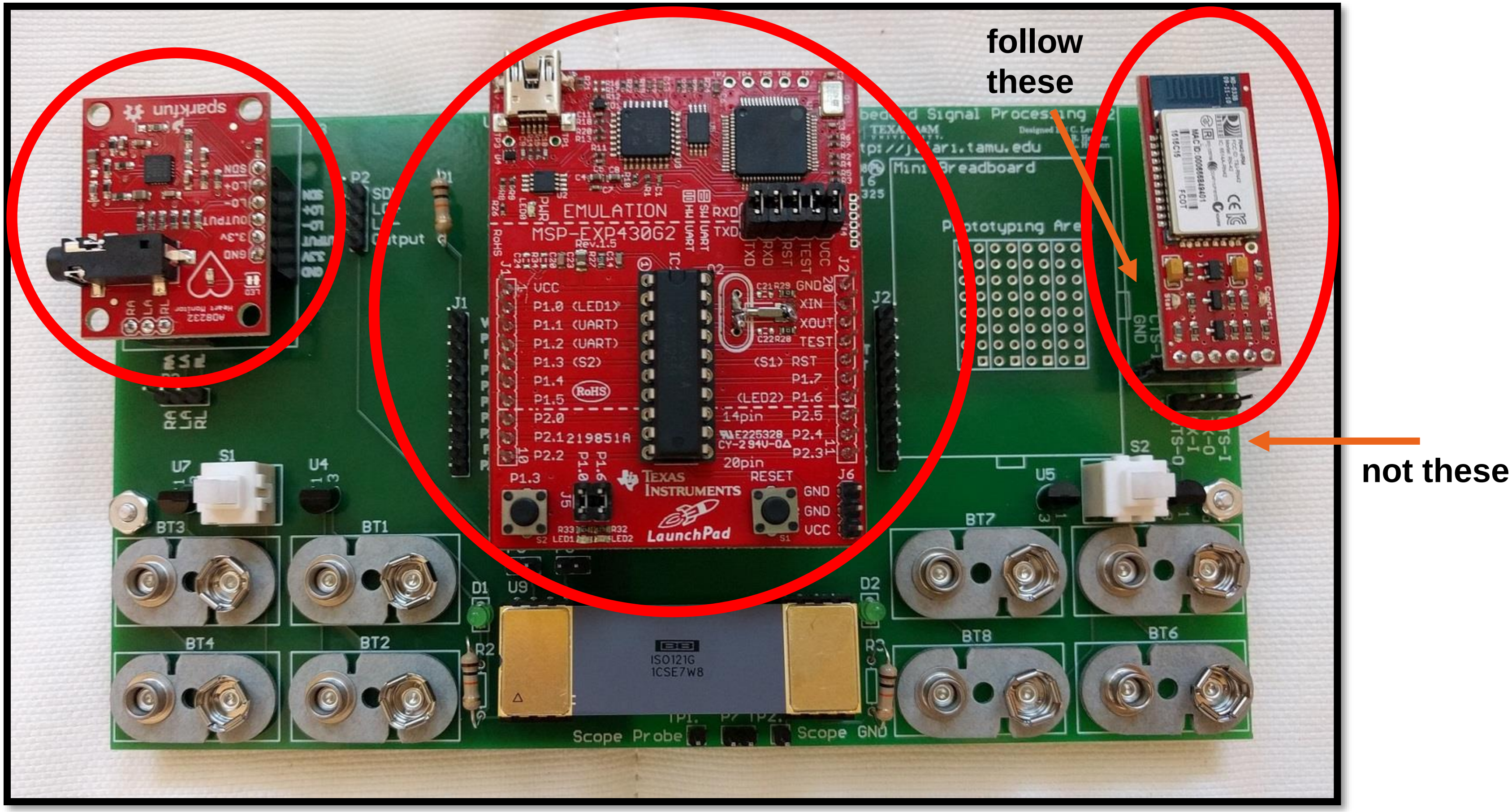
The overall flow will be initiated by the PC GUI. The GUI will send the letter ‘a’ to initiate the data transfer.

Therefore you must set up a UART receive ISR that looks for the letter ‘a’ and only then starts sending out the converted ADC results to the PC.

The GUI automatically groups the incoming packets and converts the hex codes to the corresponding voltage level before plotting the data with some filtering.

Ideally, you should be able to see the regular ‘R-peaks’ corresponding to the heart beat once the hardware and software are set up correctly according to the lab procedure.

AD8232/Bluetooth/MSP430 Set-Up



Follow the labels on the “inner cavity” of the Bluetooth module!

Setting up the Bluetooth module (1)

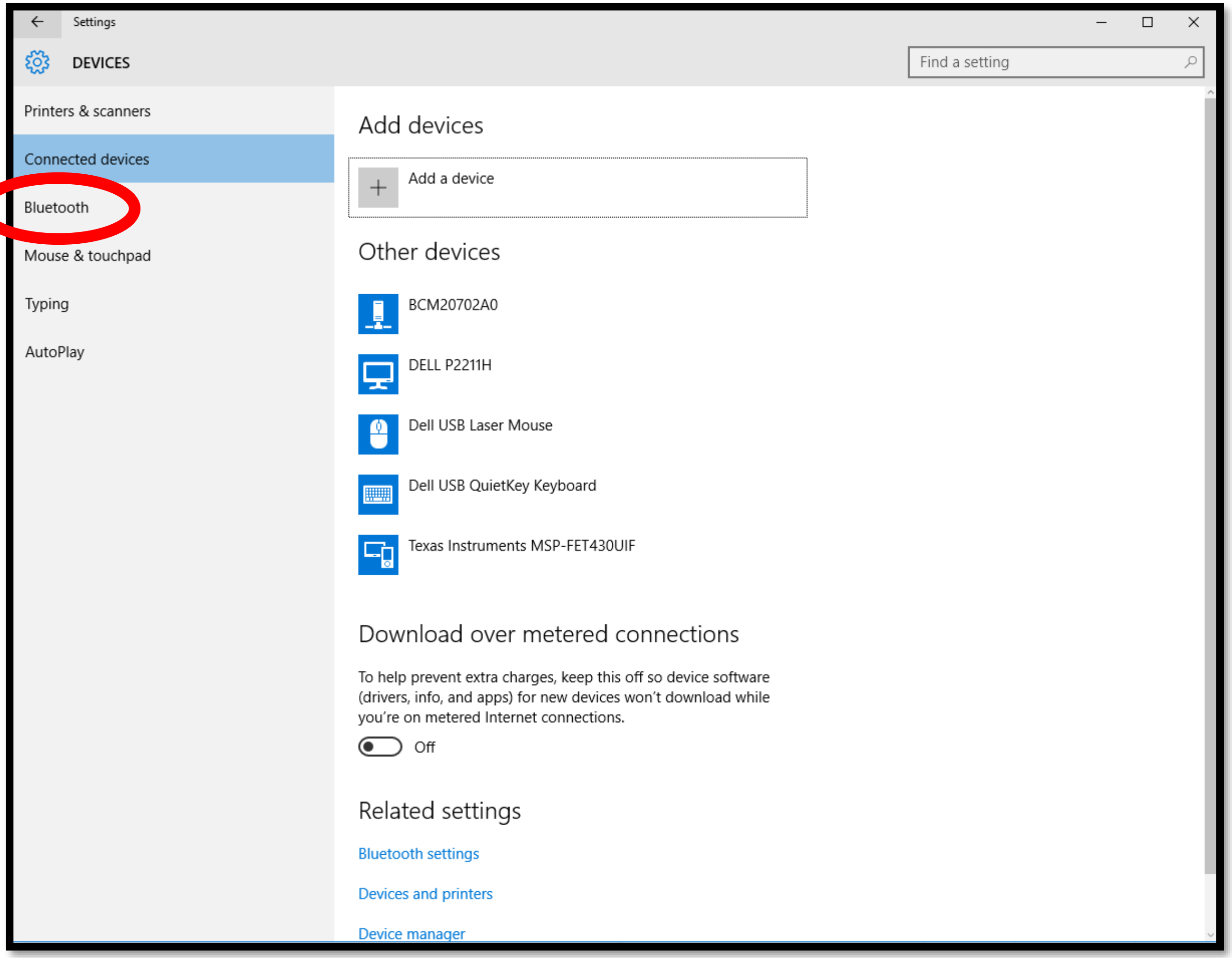
Connect the USB to your computer.

Setting up the Bluetooth module (2)

Click on the Windows icon and type:

"Add or remove devices"

Click on the Bluetooth tab



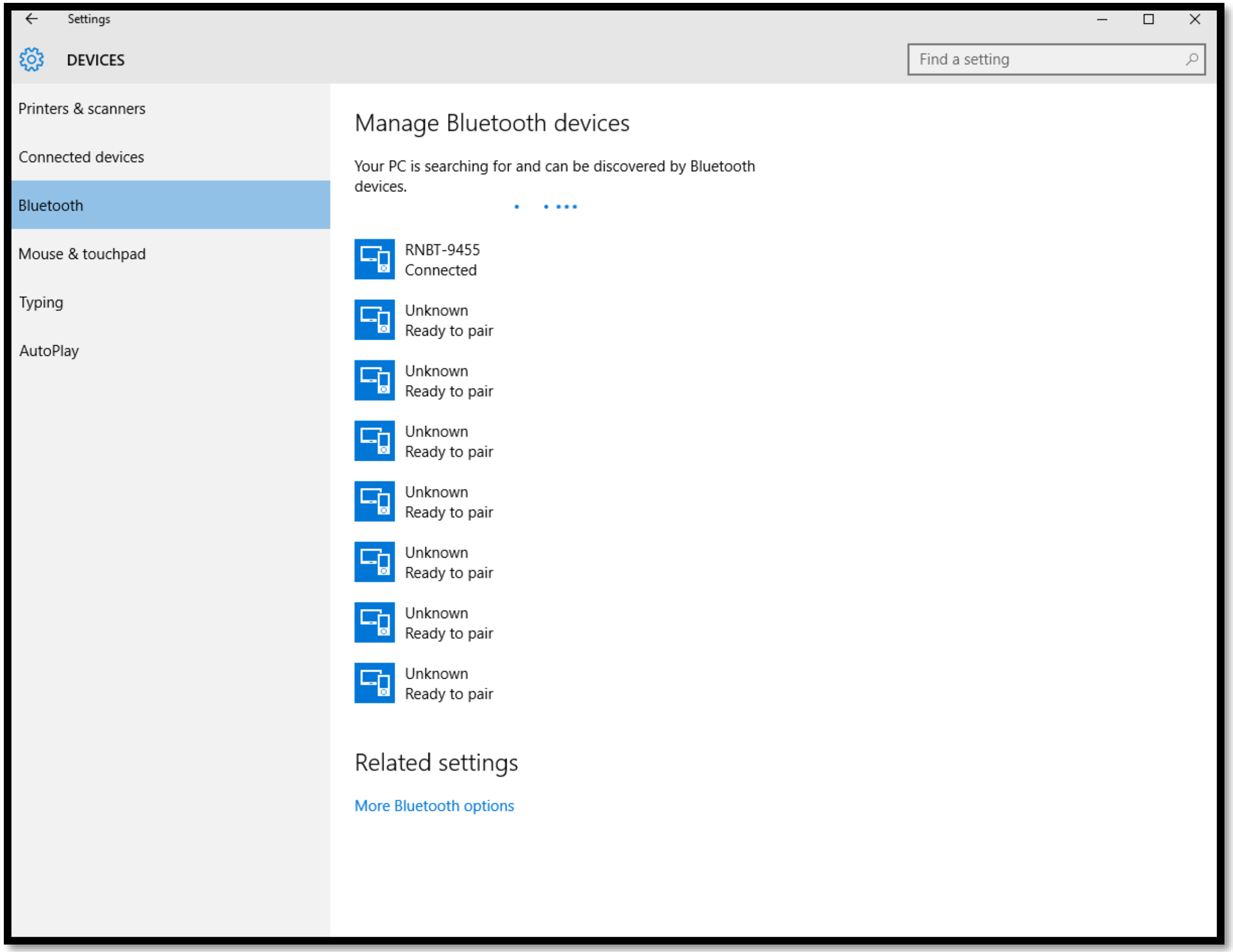
Setting up the Bluetooth module (3)

Look up your device!

Your device name should be:

RNBT – XXXX

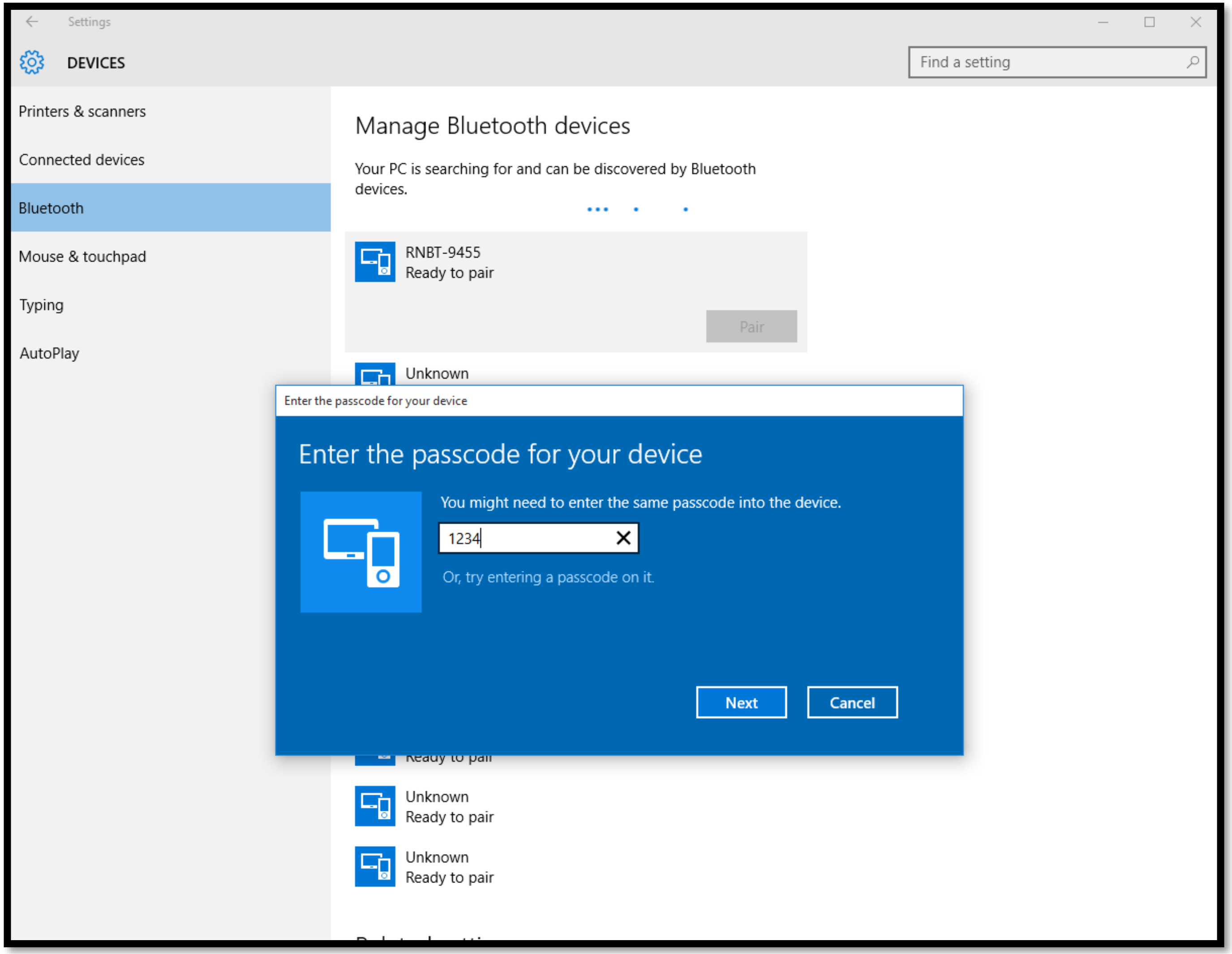
Where the XXXX correspond to the last 4 digits of the MAC ID in your Bluetooth module.



Setting up the Bluetooth module (4)

Password:

"1234"

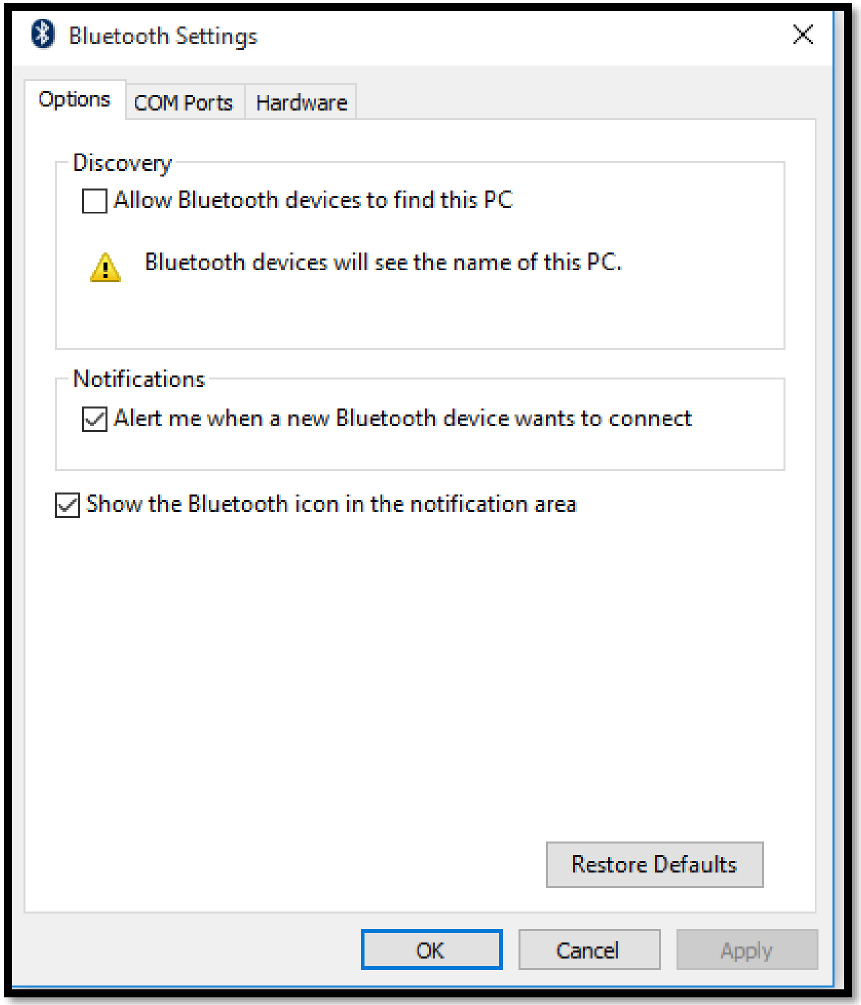


Setting up the Bluetooth module (5)

Once you've paired your device, click below on the

"More Bluetooth Settings"

And the window on the right should pop-up.

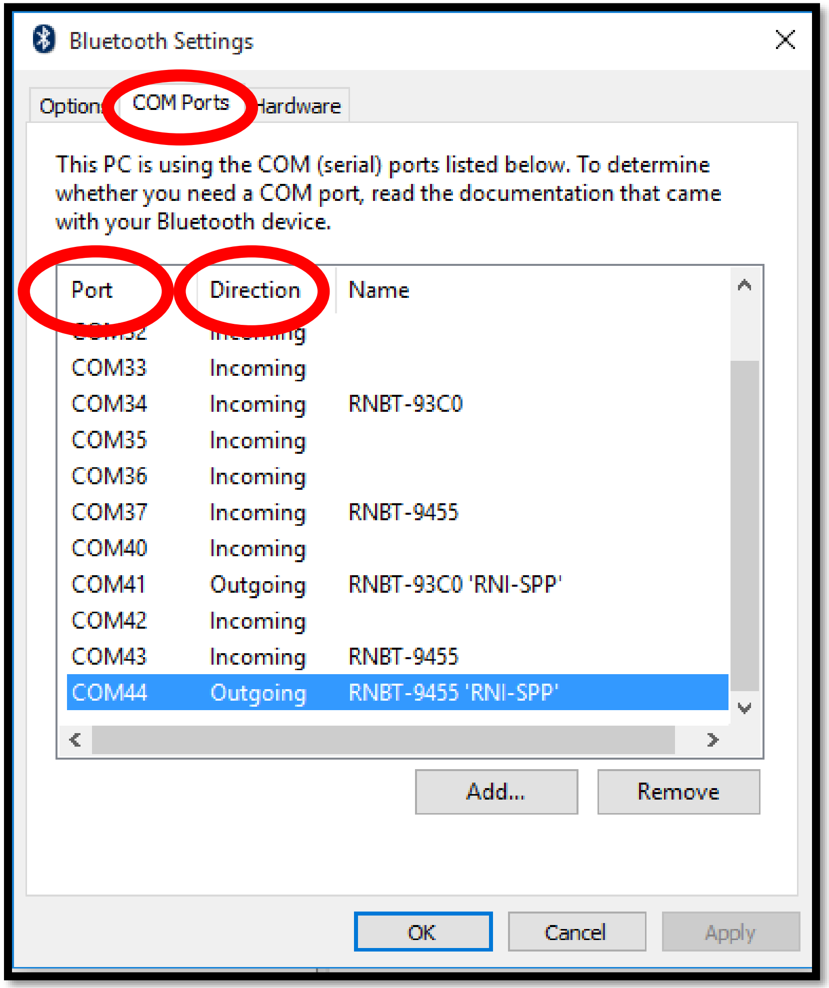


Setting up the Bluetooth module (6)

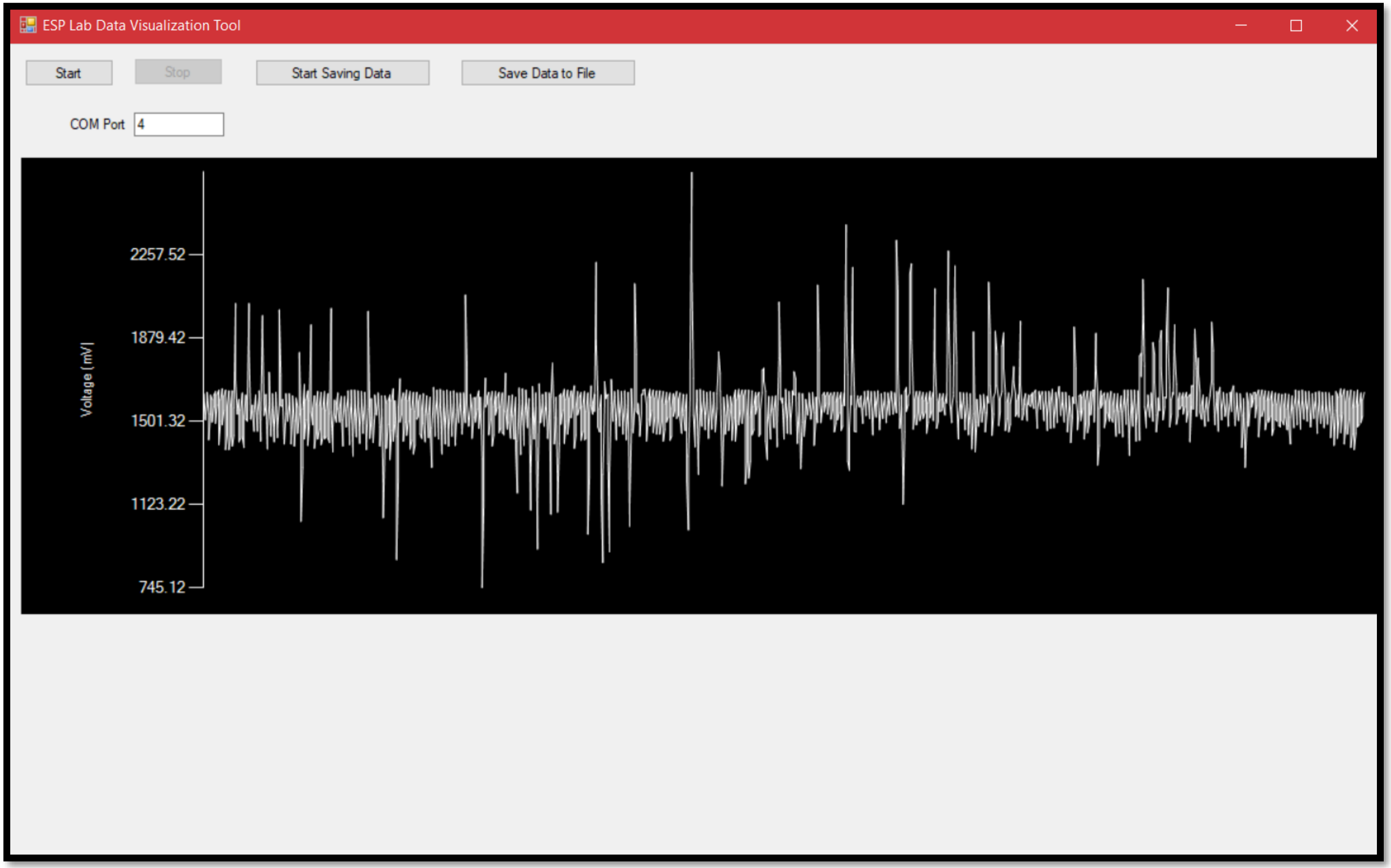
Search for the COM port of your device and write it down.

Look for the “Outgoing” direction.

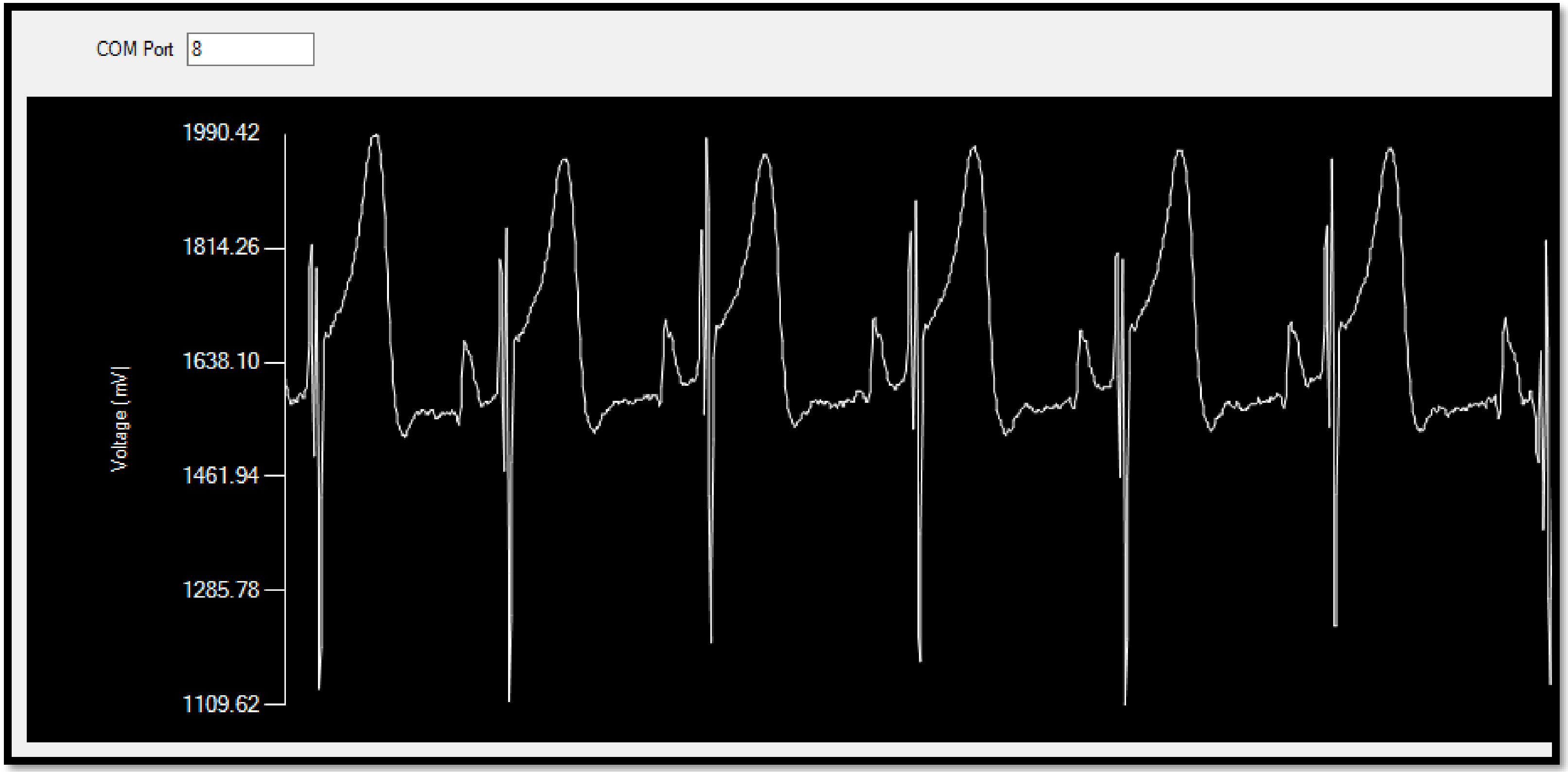
The name of the device may have a “RNI-SPP” embedded to it after the device name.



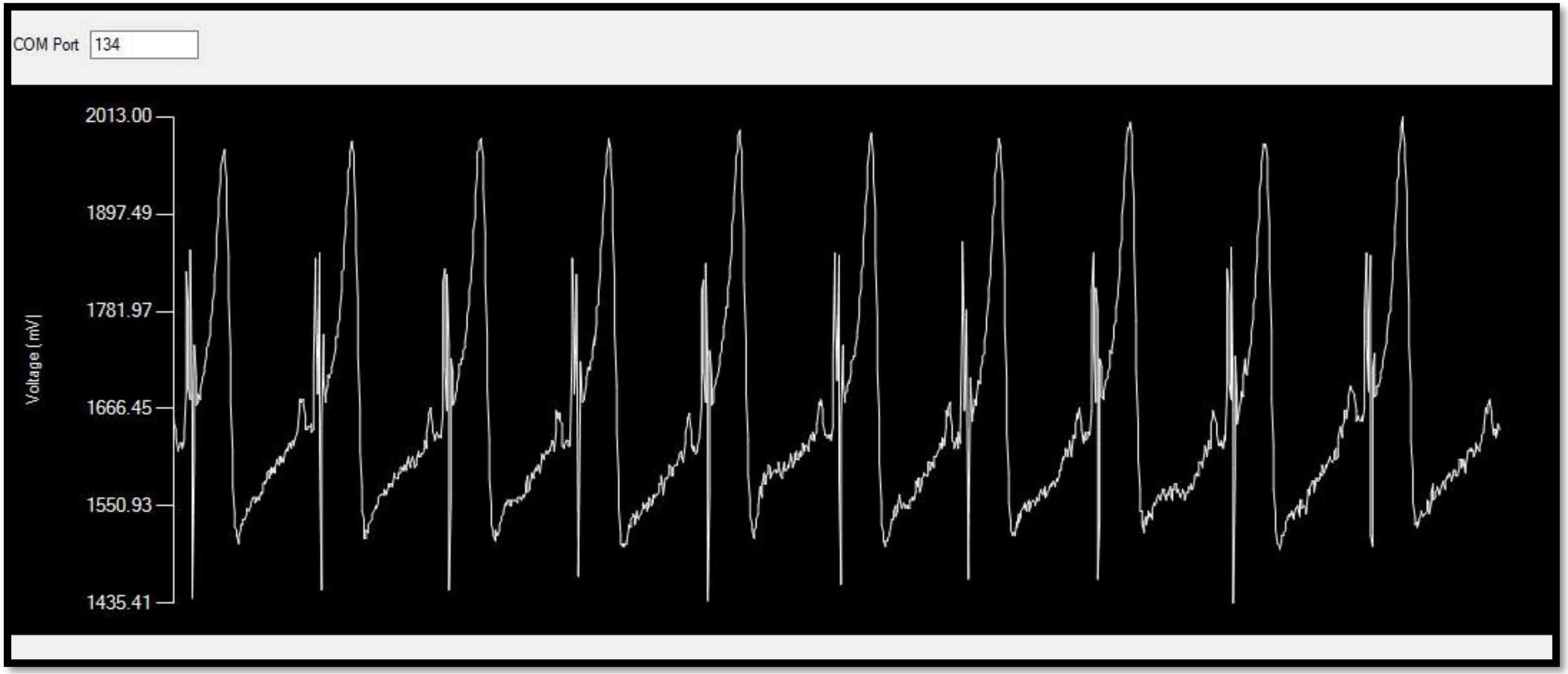
Sensor Noise



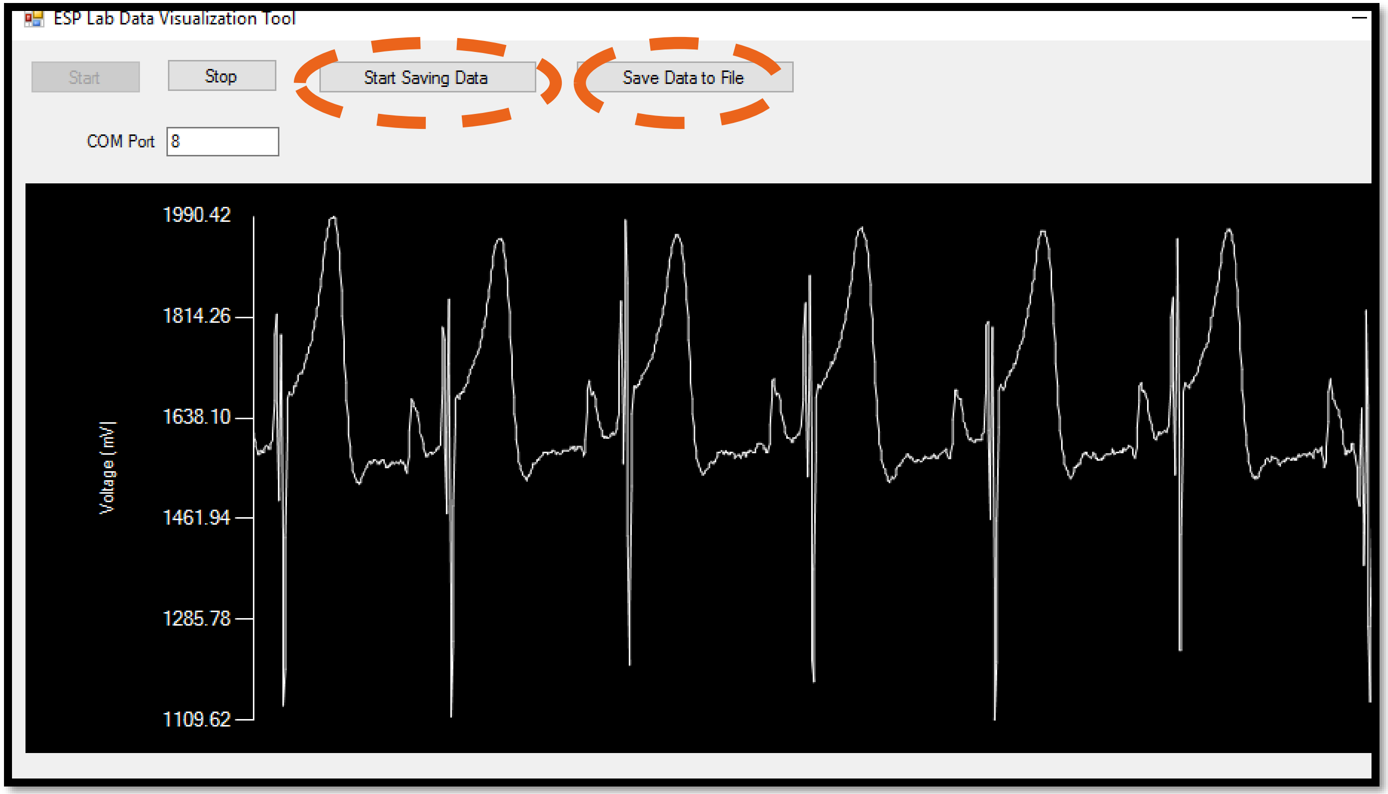
Actual ECG Measurement



Actual ECG Measurement



Take some measurements!



You'll probably use this data for the project! Take advantage of it now!

Expected Results