Capstone Project

Automatic Health Monitoring Plat Form Using Deep Machine Learning Based On Zigbee Wireless Sensor

Prepared By-

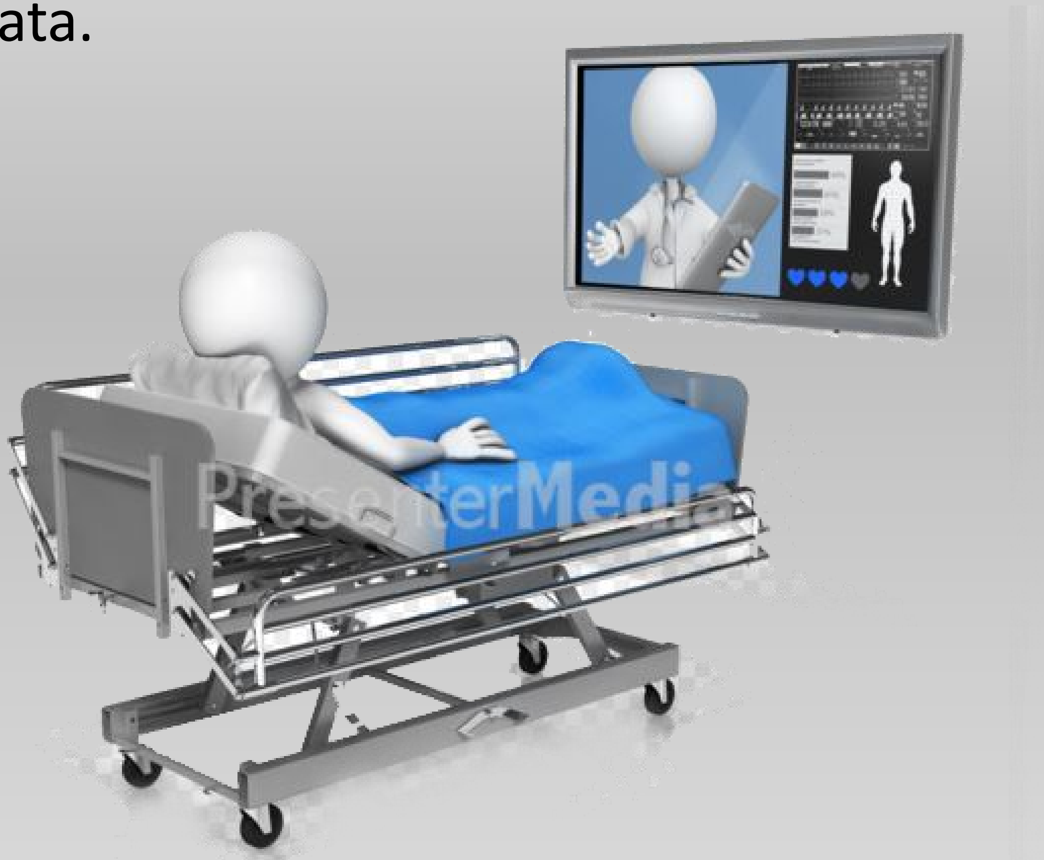
Nahom G Ghebremeskel

Guided By –

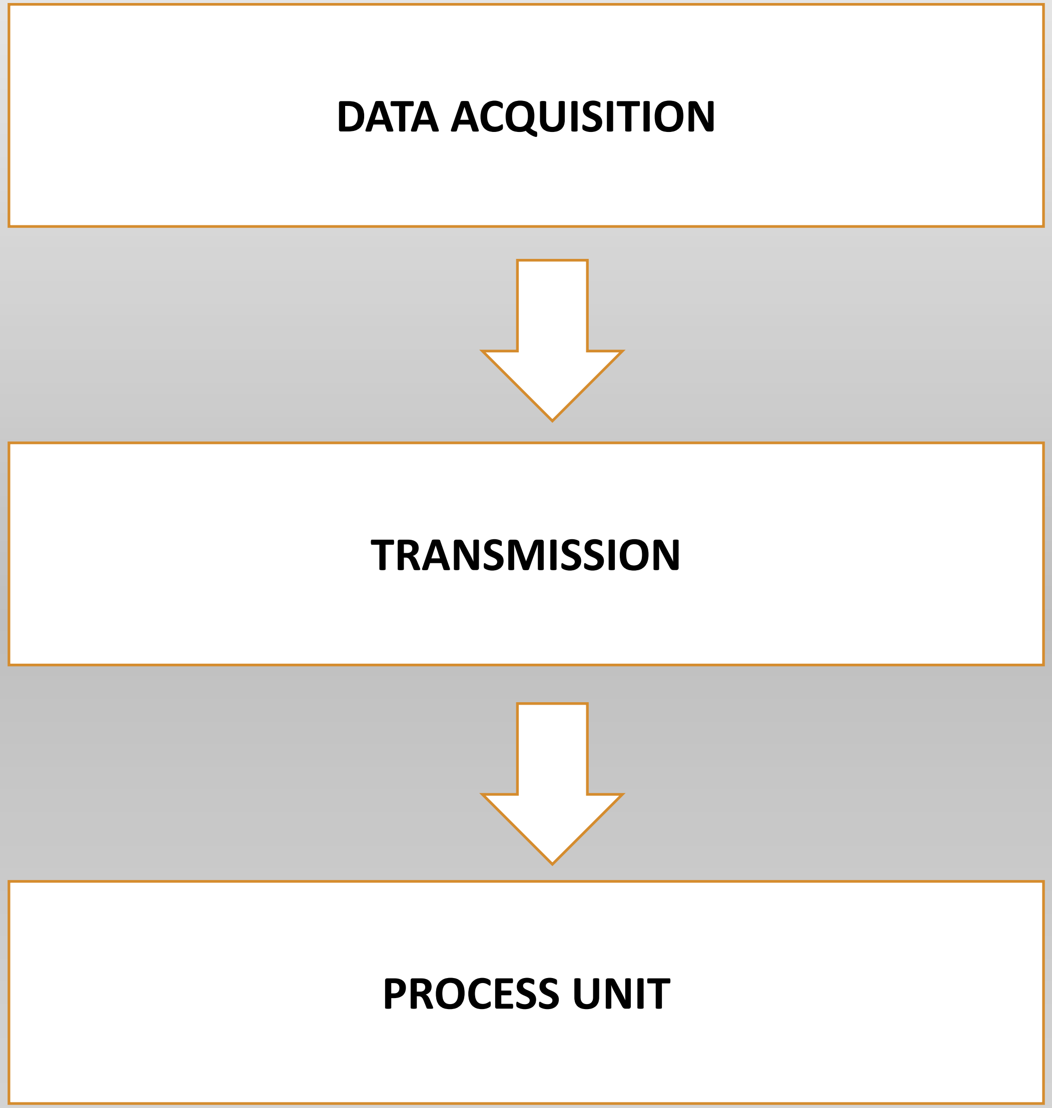
Dr. Vahid Emamian

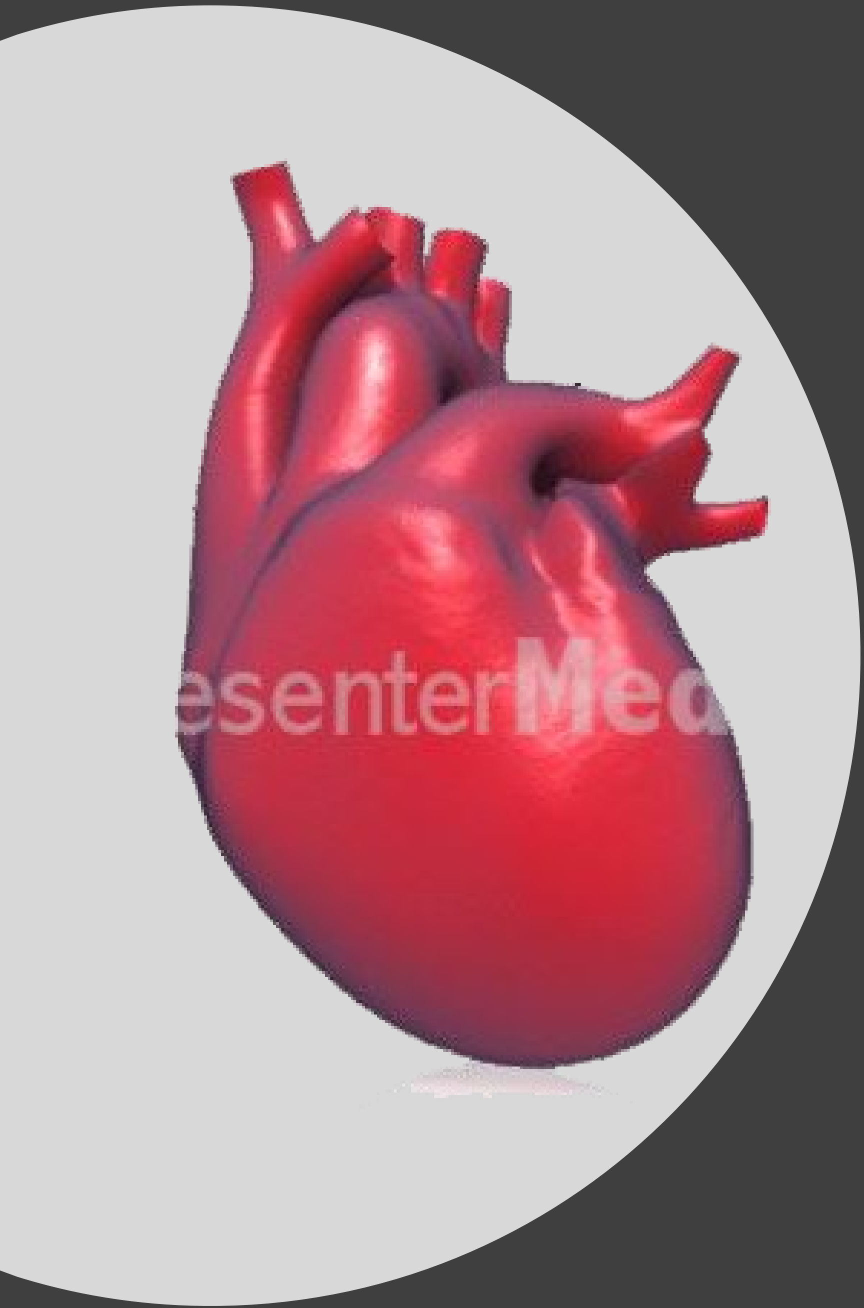
Project work

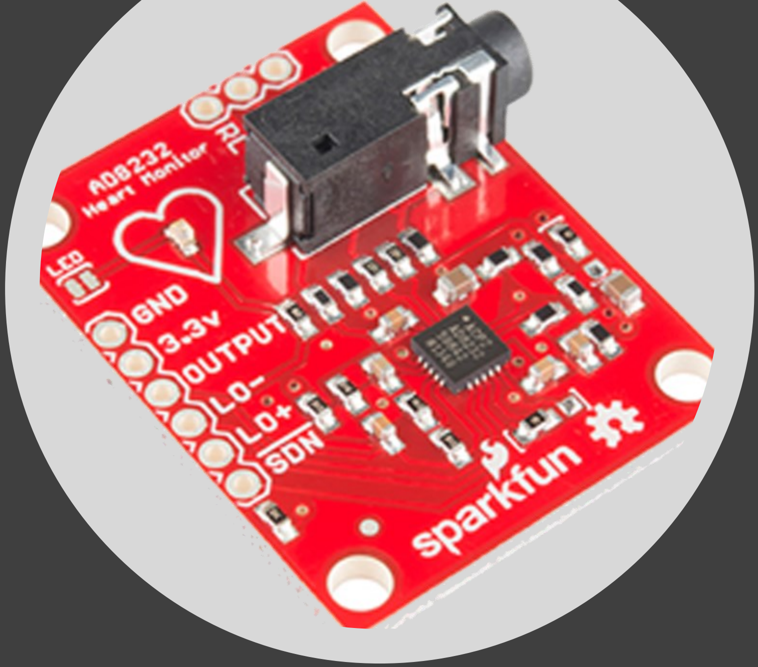
- develop a human health monitoring platform using deep machine learning and artificial intelligence (AI).
- Monitoring the health activity using wireless channel.
- Use an AI and deep machine learning platform to train the machine
- detect or predict heart disease from a collected data.



How it can be implemented?


DATA ACQUISITION
TRANSMISSION
PROCESS UNIT





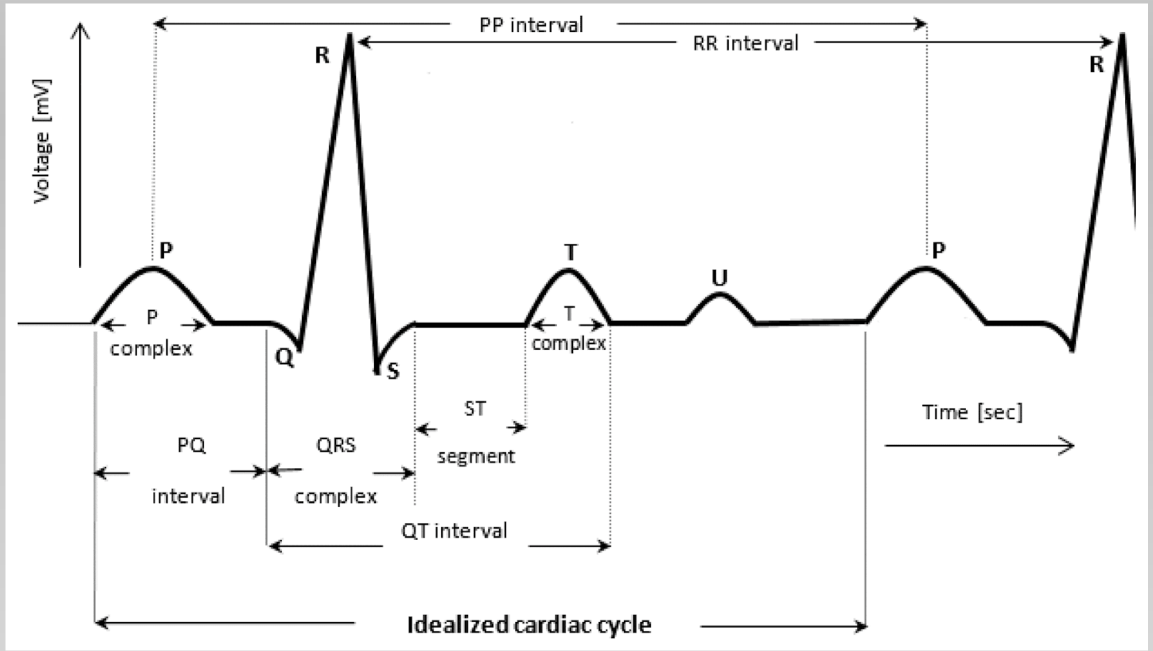
DATA ACQUISITION

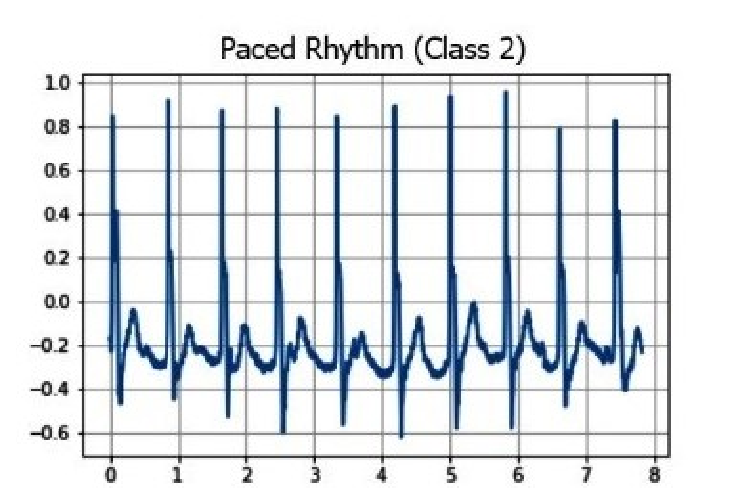
ELECTROCARDIOGRAM

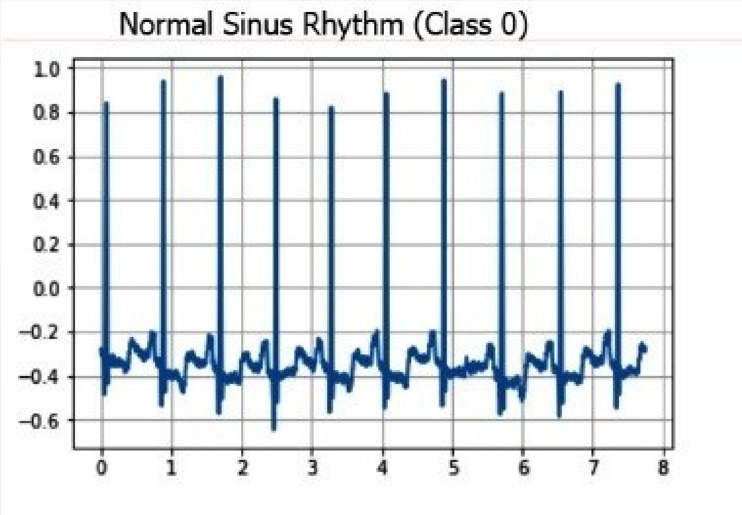
- Electrocardiogram (ECG) signals in the healthcare field are used to **identify cardiac problems** in a patient.
- ECG signal is the **only way** by which the **heart condition can be detected** very accurately.
- All features of ECG are must be in **specific limit** if it is not then it calls abnormal signal or its **Arrhythmia**.

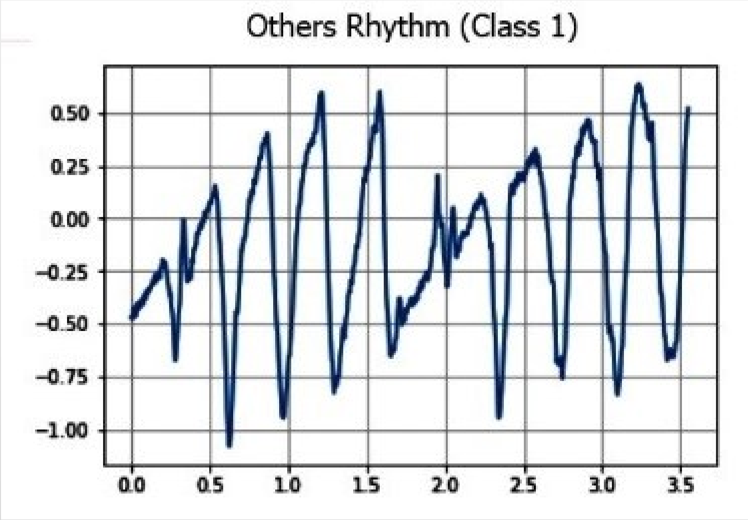
Abnormalities Characteristics features

1. Bradycardia Heart Rate < 60 BPM
2. Tachycardia Heart Rate > 100 BPM
3. First Degree AV Block QRS complex ≤ 0.1 Sec.
4. Second Degree AV Block 0.1 Seconds ≤ QRS complex ≤ 0.2 Seconds









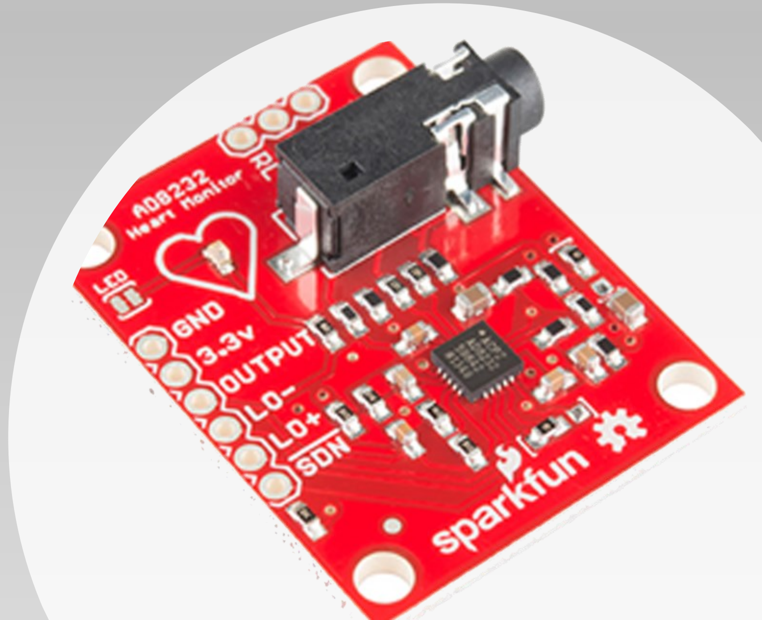
Types of Rhythm Used

AD 8232 SINGLE LEAD HEART RATE SENSOR

- It receives/reads the electrical activity of the heart
- It is intended to extract, amplify, and filter small biopotential signals
- It makes a noisy conditions, such as those created by motion or remote electrode placement.

AD 8232 sensor Negative Feedback

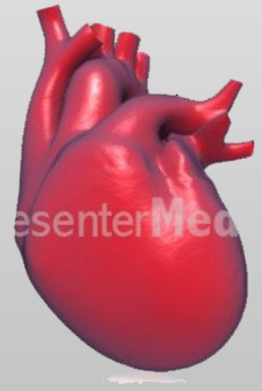
- It might last like 5 seconds to stabilize.
- The bandwidth is too big.
- makes noise because of the muscle signal and any movement .
- The noise goes all the way to 1k Hz, and a normal ECG doesn't go more than 250.
- Large SNR .
- Placement of the ECG leads give different reading.

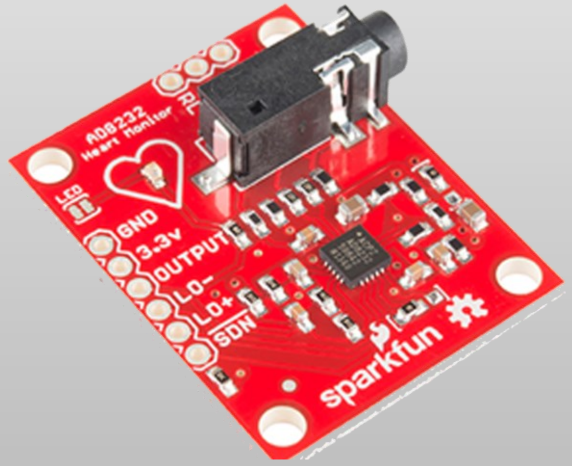


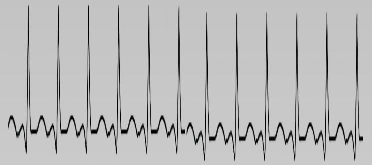
ARDUINO UNO

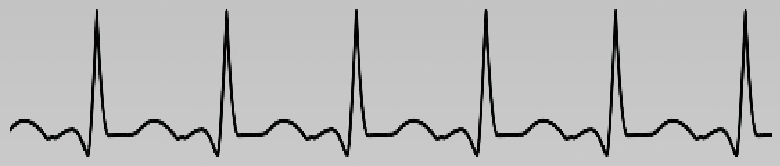
- It is a microcontroller board based on Atmega 328p.
- Arduino has
  - 14 digital input/output pins.
  - 6 analog inputs.
  - 16 MHz quart crystal.
  - USB connection, a power Jack and an ICSP heard.
- It contains 6 channel 10 bit analog to digital converter.
- The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board.

Analog to digital conversion





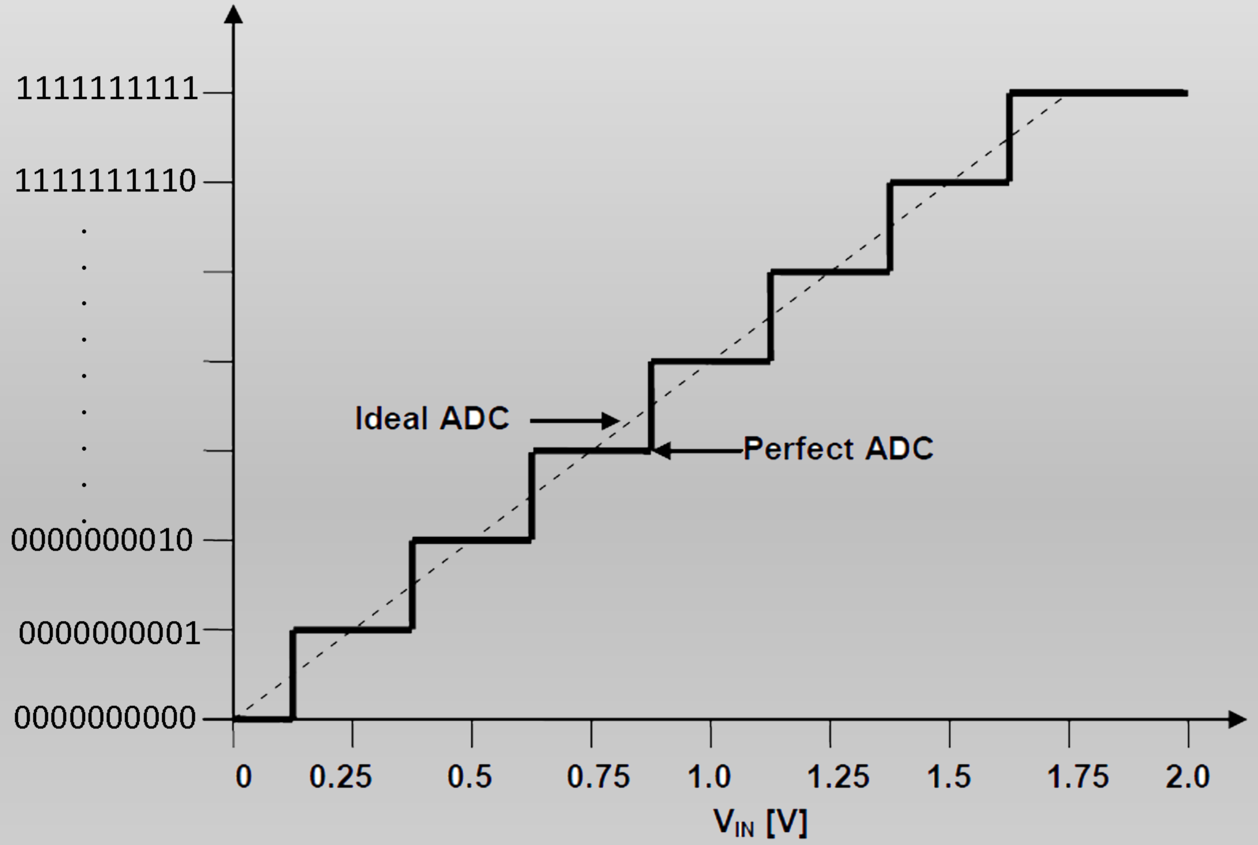




0	1	0	1	0	0	1	1
0	0	0	0	0	0	1	1

How Arduino ADC works

- It maps input voltages between 0 and 5 volts into integer values between 0 and 1023.
- It takes about 100 micro second to read an analog input.
- The maximum possible sampling rate is 9615 HZ



1	1	0	1	0	1	0	0	1	1

2^10 = 1024

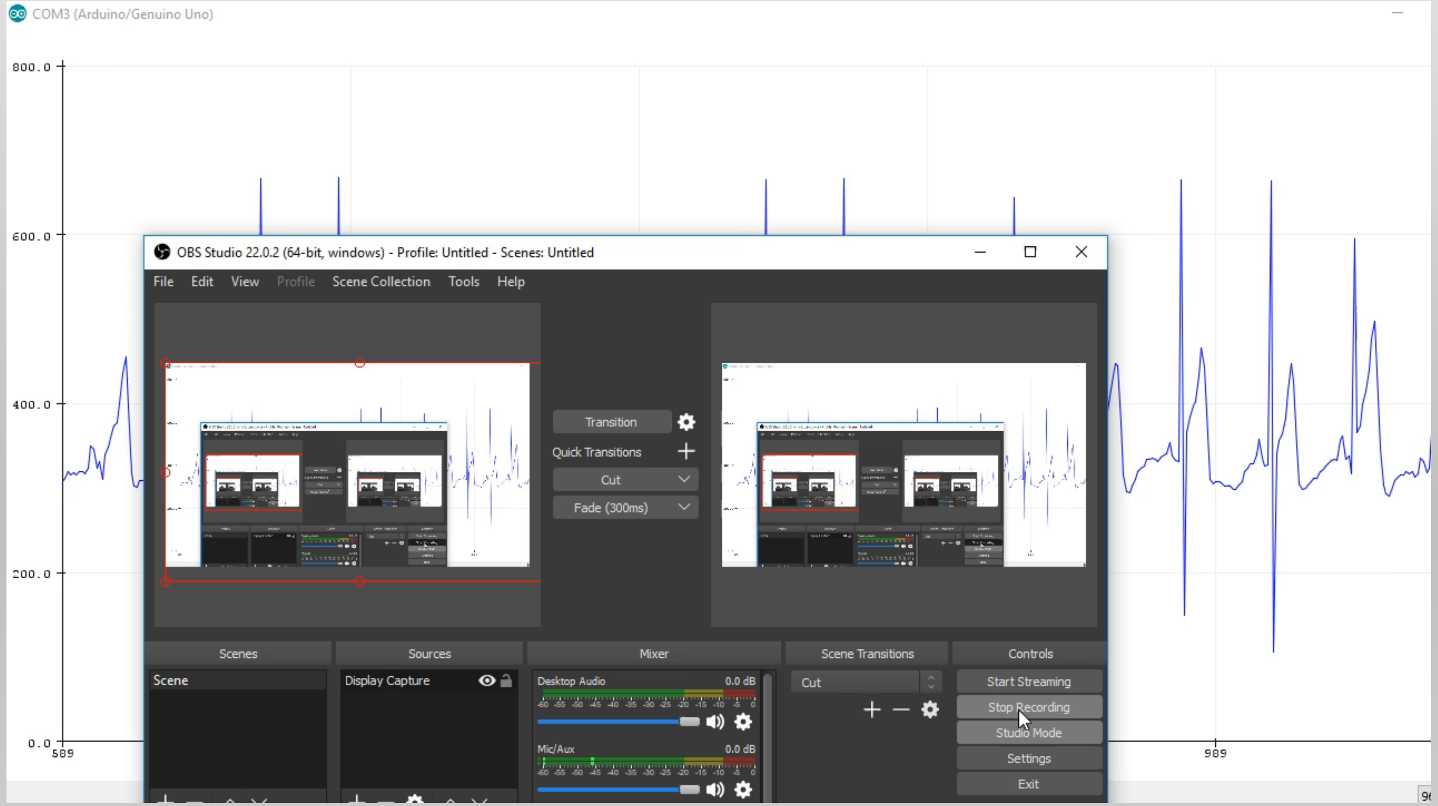
$$\frac{Resolution\ of\ the\ ADC}{System\ Voltage} = \frac{ADC\ Reading}{Analog\ Voltage\ Measured}$$

$$\frac{1023}{5.00V} = \frac{x}{2.12V}$$

$$\frac{1023}{5.00V} * 2.12V = x$$

$$x = 434$$

ECG READING AT THE ARDUINO IDE SERIAL PLOTER



TRANSMISSION

TRANSMITTING ECG DATA

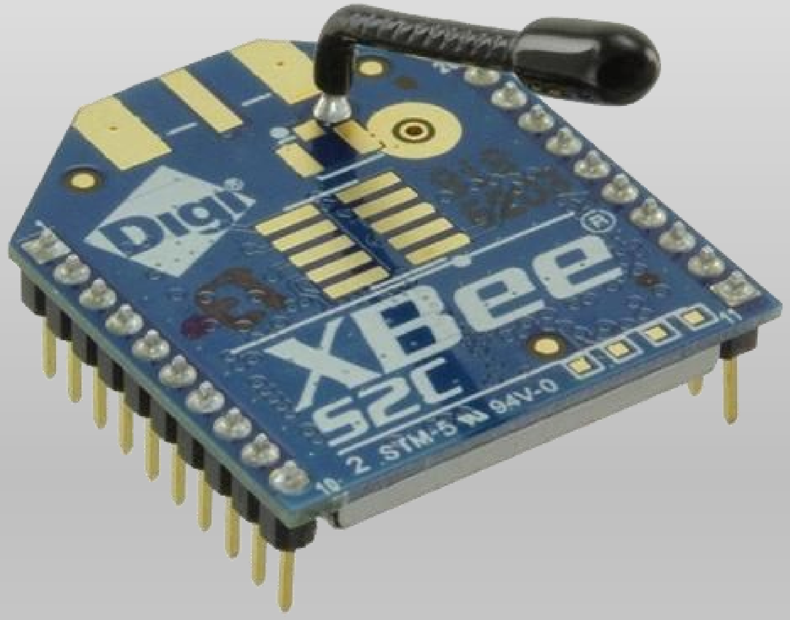
- Collected data transmit over a wireless channel to the platform
- Zigbee is high level communication protocols used to create personal area networks.
- Zigbee is an IEEE 802.15.4 based communication protocol.

ZIGBEE

- Based on IEEE 802.15.4 Standard
- Designed for sensor and control networks
- Used for applications that require:
  - Low Power Consumption
  - Low Data Rate
  - Long Range of distance

Xbee S2C

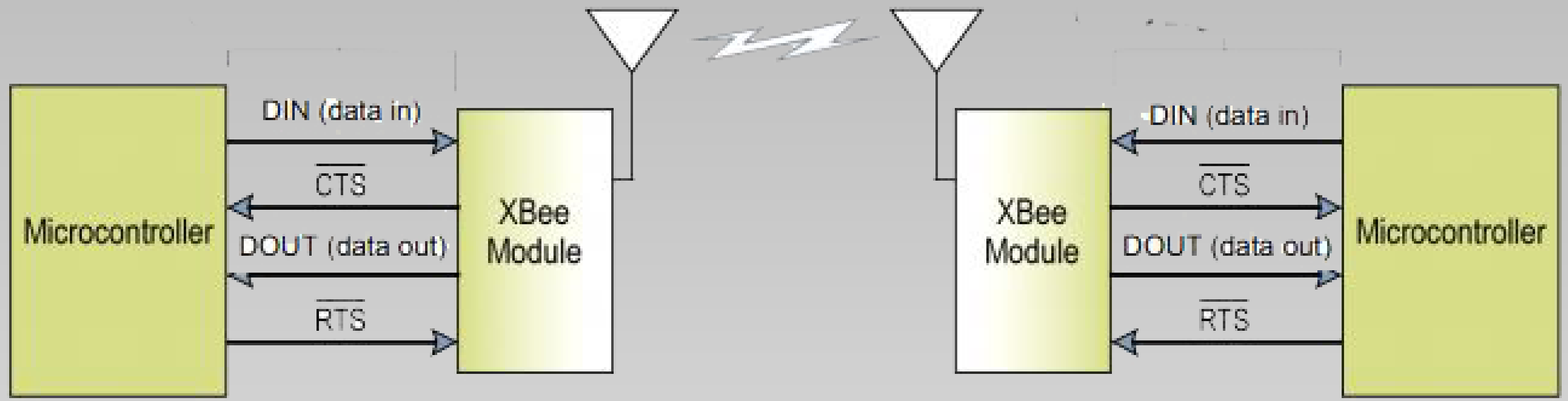
- RF Data Rate: 250kbps
- Throughput speed: 35kbps
- Frequency: ISM 2.4GHz
- Specs OK Temp: -40 to 85C
- Digital I/O pins: 11
- Analog input pins: 4
- Mesh routable Self Healing network
- Firmware: ZB ZigBee
- Operating Voltage: 2.1 – 3.6V
- Operating Current: 40mA@3.3V
- Indoor range: 40 Meters Line of sight range: 120 Meters Max
- Analog Pin Reading: 1.2V.



XBEE S2C SERIAL COMMUNICATION

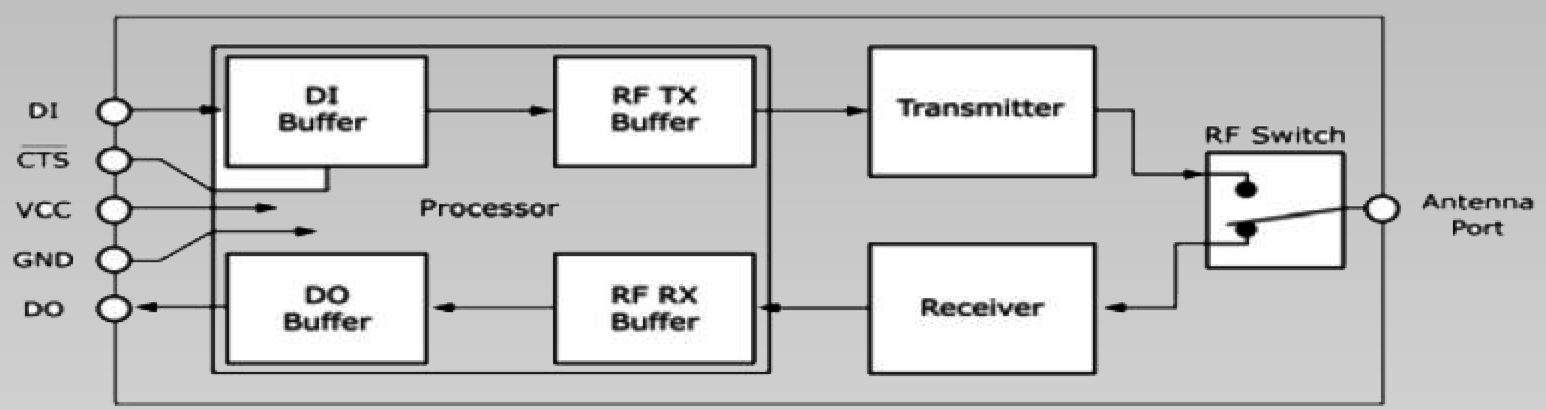
AT MODE

API MODE



Transparent operating mode (AT)

- Default xbee operate mode
- Data queued up for RF transmission
- Data is buffered in the DI buffer until
  - packetization Timeout
  - Maximum number of RF packet(100 byte)

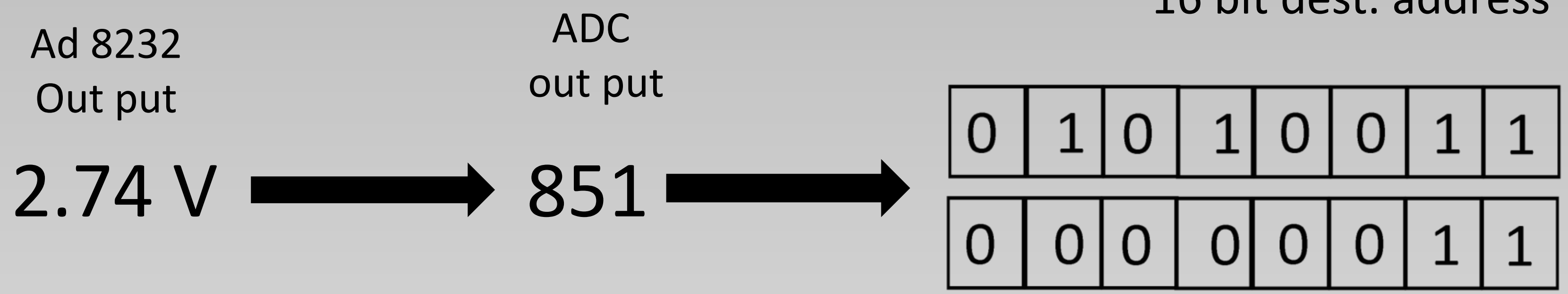


Application programming interface (API) mode.

- An alternative to the AT operating mode
- host application can interact with the networking capabilities of the module
- It lets the user decide how many byte wat to transmit.
- A host application can send data frames to the module that contain address and payload
- In API mode all data entering the module can immediately transmit.

Creating API Frame

- <XBEE.H>
- 8 data bit
- Convert 10 bits into two bytes
- MSB and LSB



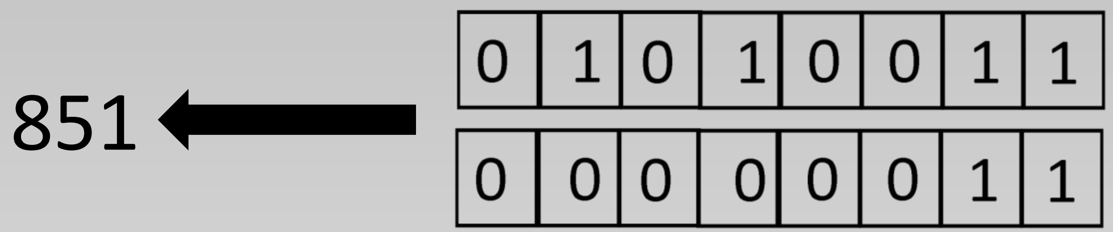
Start delimiter	7E
Length (18)	00 12
Frame Type	10
Frame ID	01
64 bit dest. address	00 13 A2 00 41 84 FB 3D
16 bit dest. address	FF FE
	53
	03
checksum	2D

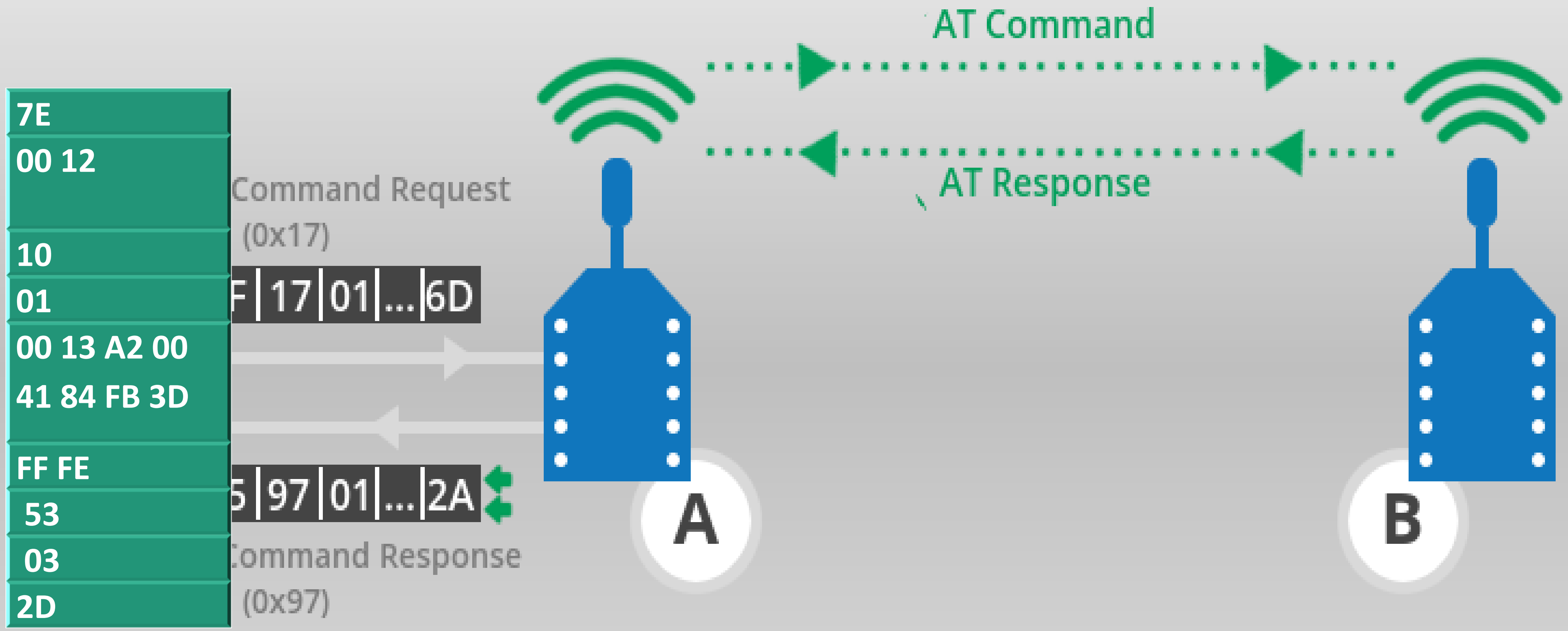
Extracting API frame at the receiver

- <XBEE.H>
- 8 data bit
- Convert two bytes into 10 bits
- Combine MSB and LSB

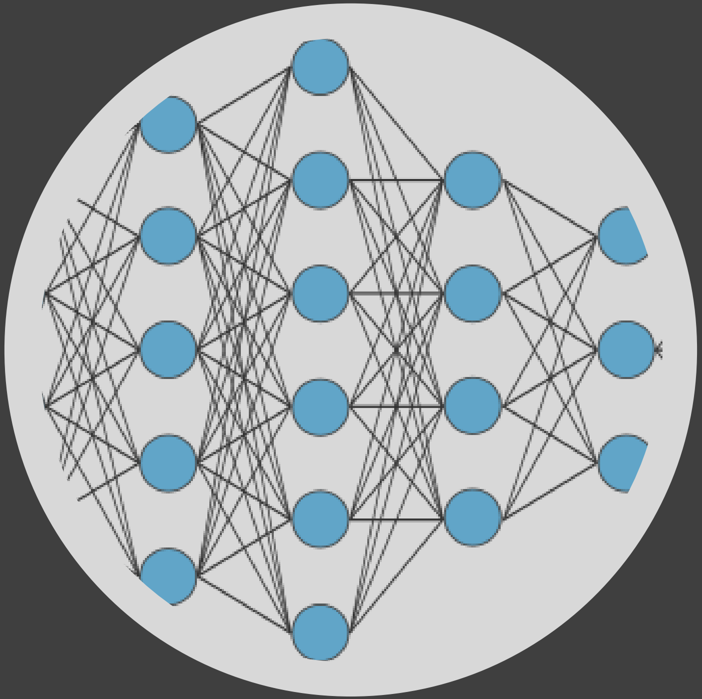
Start delimiter	7E
Length (18)	00 12
Frame Type	10
Frame ID	01
64 bit dest. address	00 13 A2 00 41 84 FB 3D
16 bit dest. address	FF FE
	53
	03
checksum	2D

Process unit





7E
00 12
10
01
00 13 A2 00
41 84 FB 3D
FF FE
53
03
2D
Command Request
(0x17)
F 17 01 ... 6D
5 97 01 ... 2A
Command Response
(0x97)
AT Command
AT Response
A
B

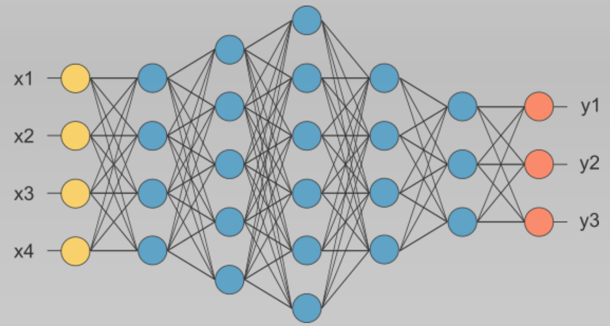


PROCESSING PLATFORM

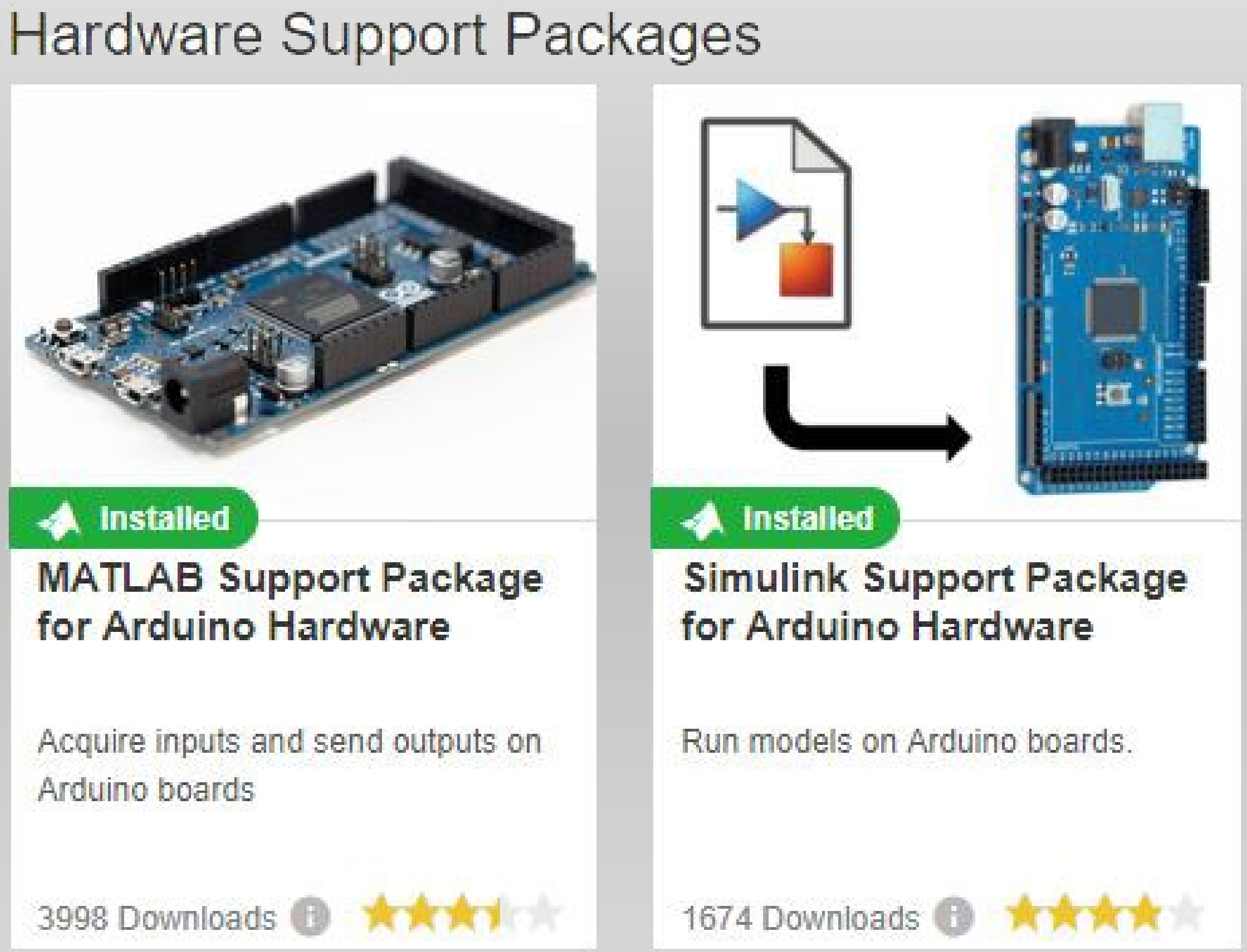
PROCESSING PLATFORM

1. Collecting the received data.
2. Data is processed using Deep neural network.

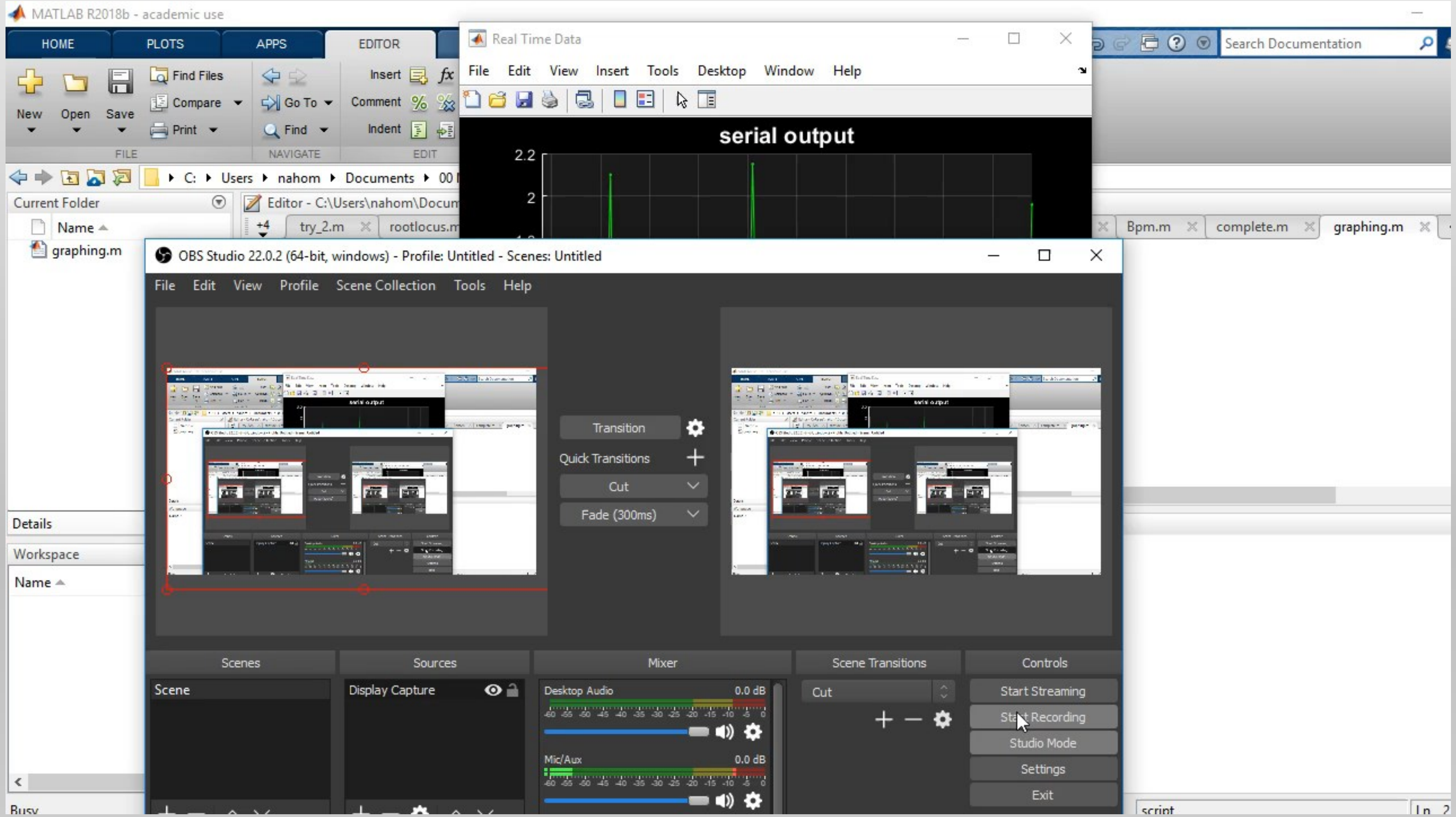
Received Data



CONNECT ARDUINO TO MATLAB

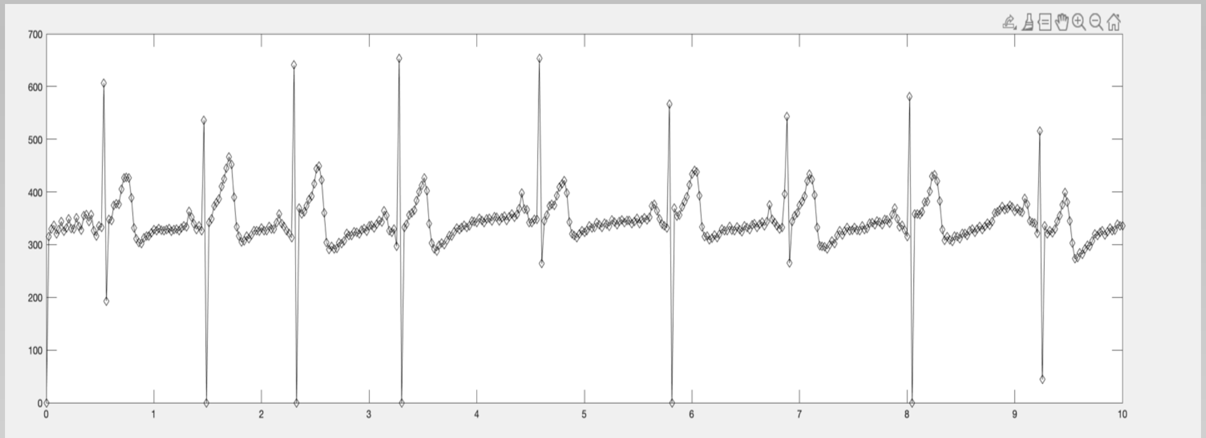


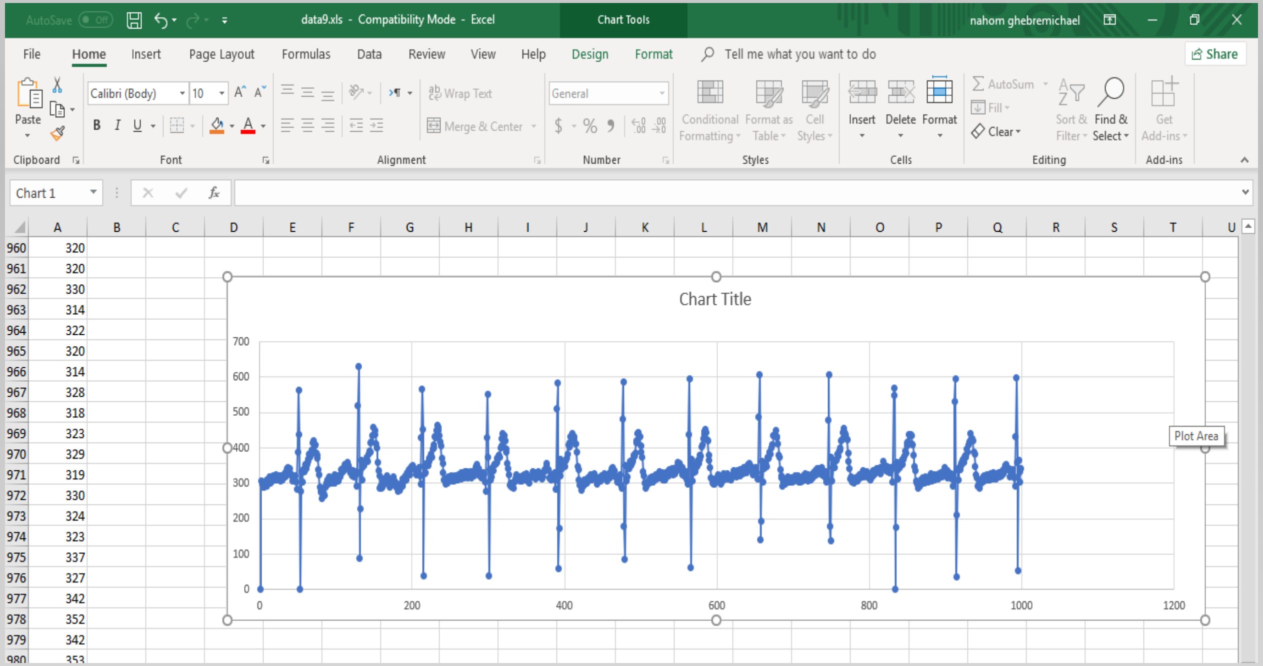
RECIEVED ECG SIGNAL REAL TIME MATLAB PLOTER



Collecting the received ECG data using MATLAB

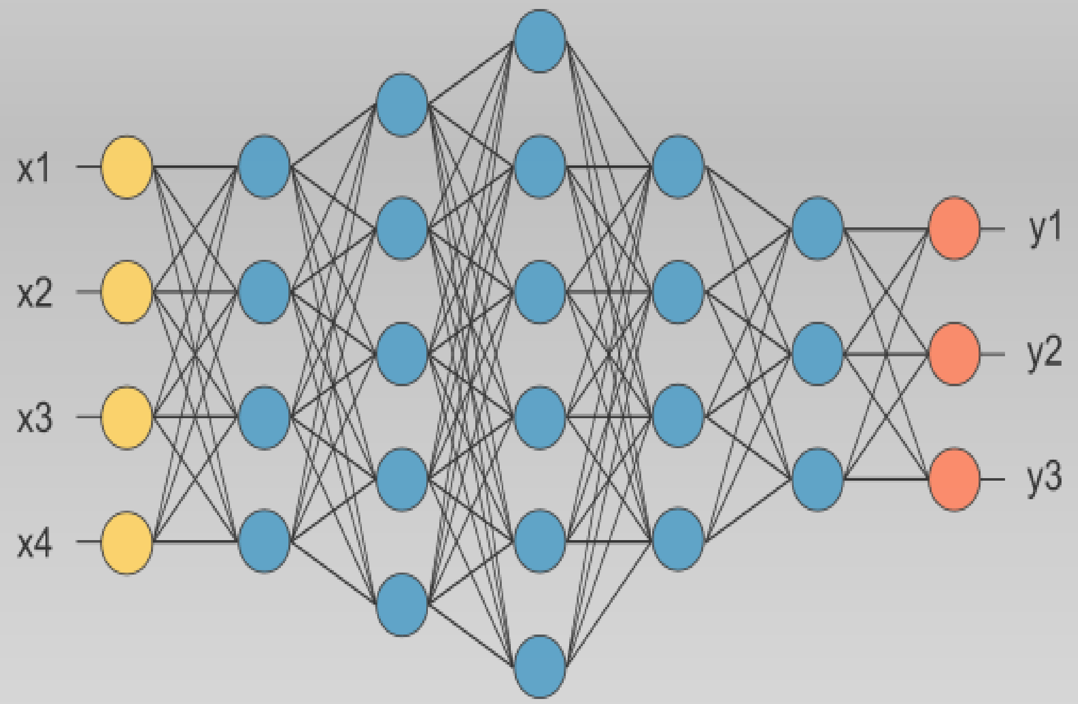
- We use MATLAB to read the received ECG data from the Arduino through serial communication.
- Use to store the reading data for an interval of time at a sampling frequency of 43 samples per second in the form .csv format.
- Plotting the received ECG signal in real time.



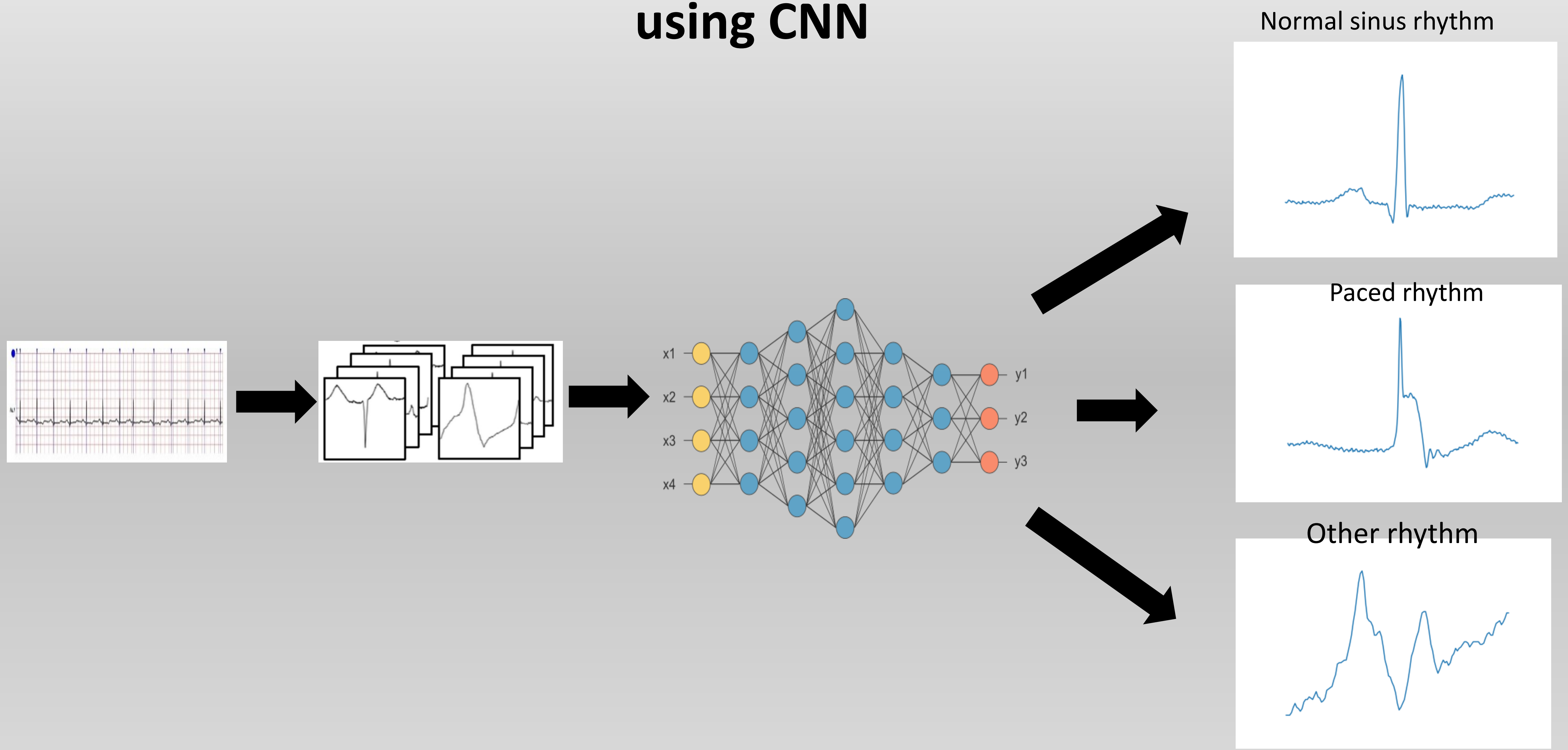


TRAINING OUR PLATFORM USING DEEP LEARNING NEURAL NETWORK

- Physio Bank datasets will use to distinguish normal and abnormal ECG signals
- Convolutional Neural Network (CNN) to detect various arrhythmias in arbitrary length ECG dataset.
- Our platform will be trained using deep neural network with Physio Bank dataset.
- To implement CNN I may use MATLAB or Python (TensorFlow)

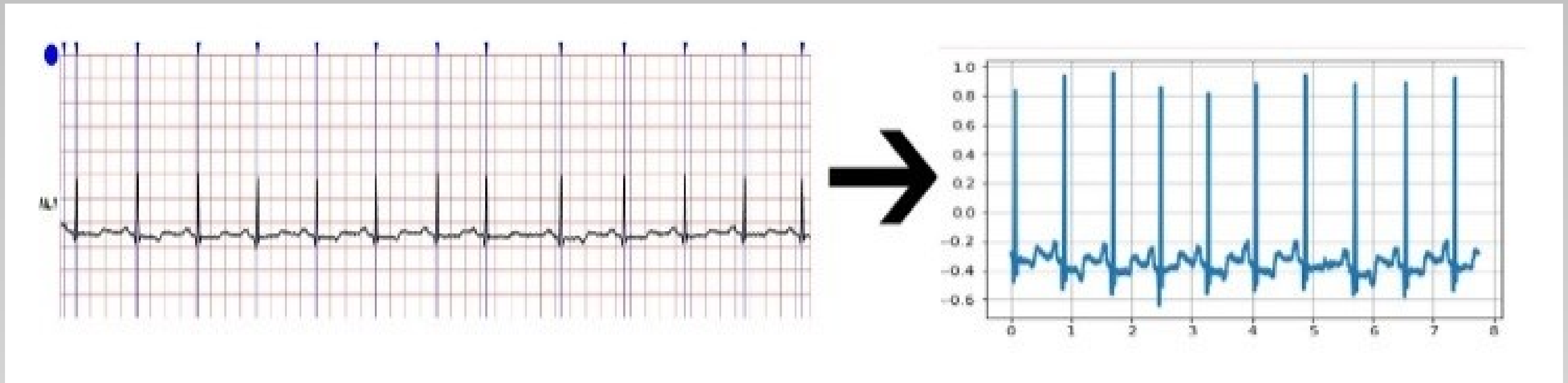


Detection of Arrythmia and Normal Sinus by using CNN



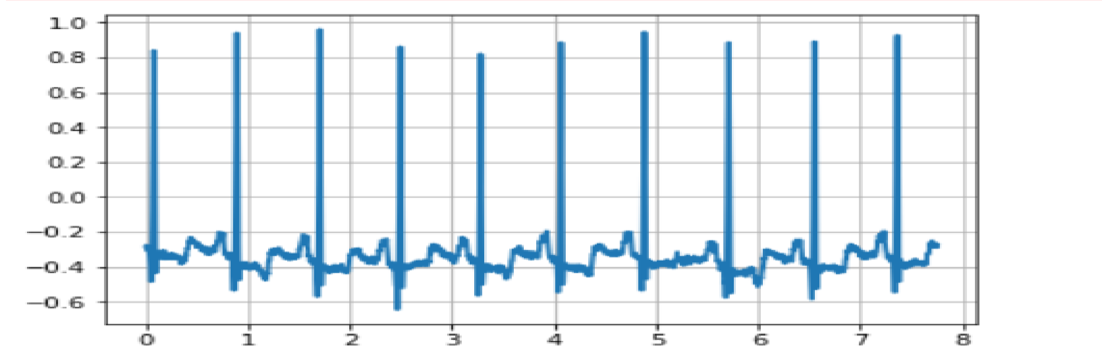
Data Pre-Process

- The MIT-BIH database contains approximately 110,000 ECG beats with 15 different types of arrhythmia including normal.
- We used 7177 Normal beat ECG Images (class 0), 8917 Paced rhythm ECG images (class 2) and 472 Other rhythm ECG images (class 1).
- Convert the ECG signal into image using python.

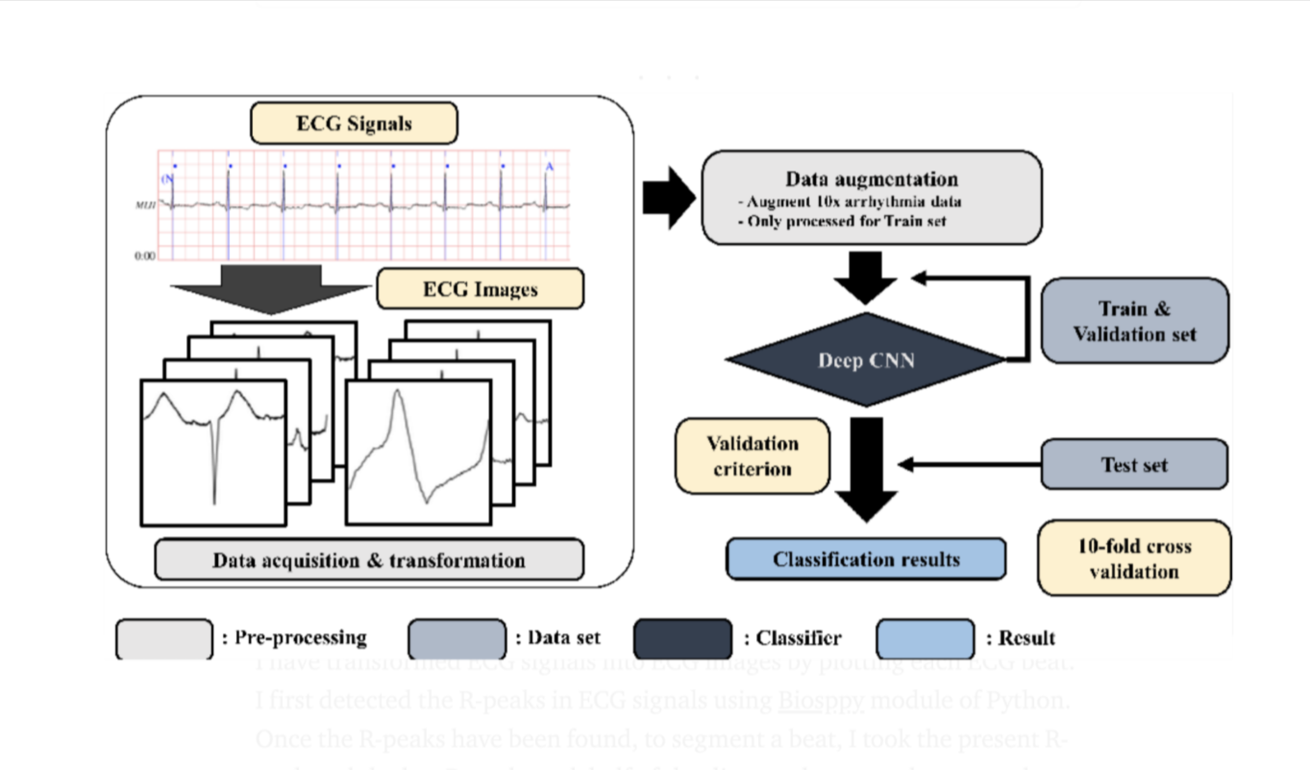


ECG signal into ECG Beat segments images

- detected the R-peaks in ECG signals using Biosppy module of Python.
- took the present R-peak and the last R-peak,
- took half of the distance between the two and included those signals in the present beat.
- converted these segmented signals into images using Matplotlib and OpenCV.

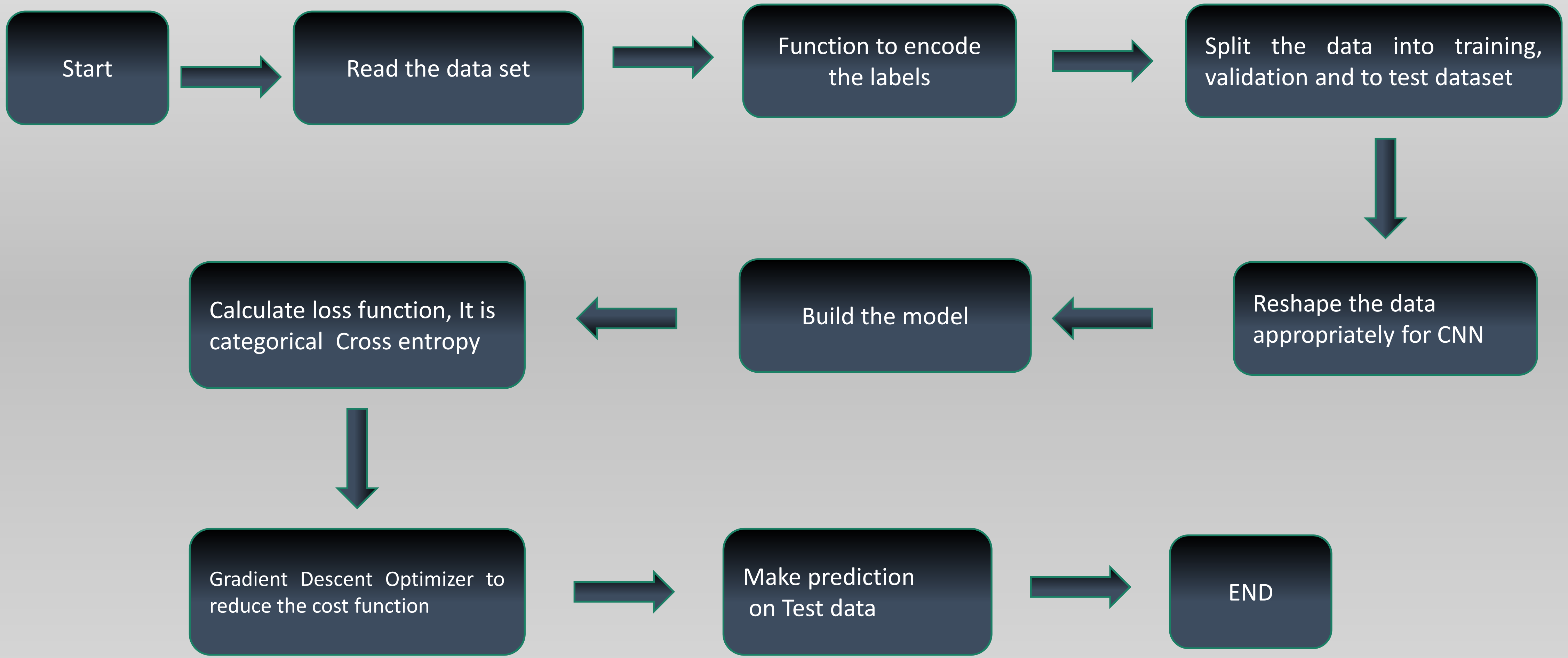


ECG Image data set



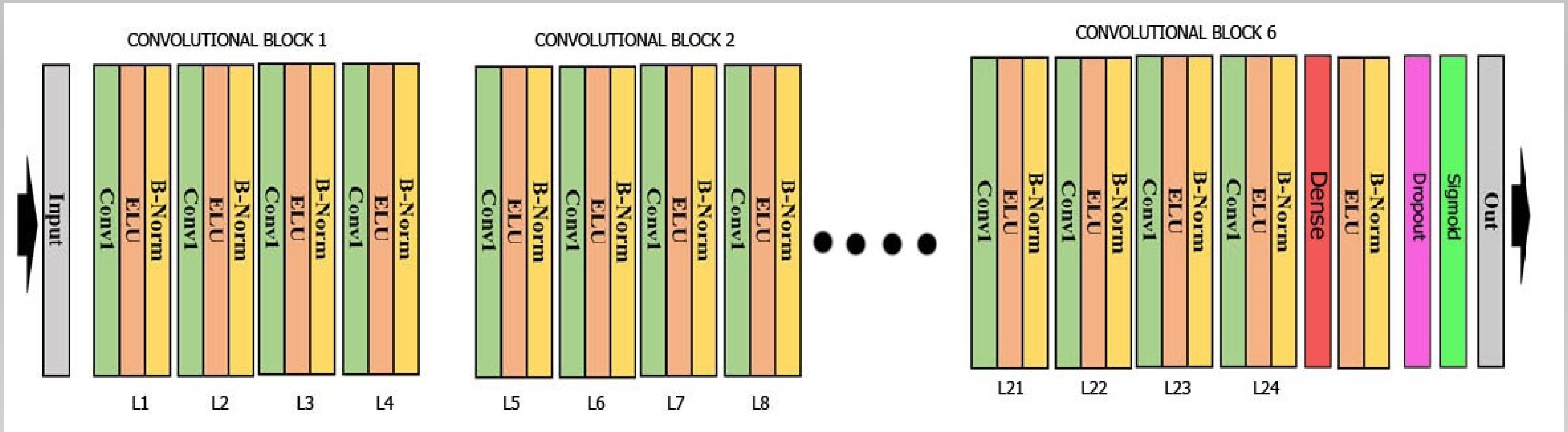
- We used 7177 Normal beat ECG Images (class 0), 472 Other rhythm ECG images (class 1) and 8917 Paced rhythm ECG images (class 2) .
- Processed data Augmentation for training set.
- After data augmentation and K fold cross validation, the proposed CNN algorithms used 953360 ECG beat images for training and 238340 ECG beat images for validation.

Architecture for CNN Model

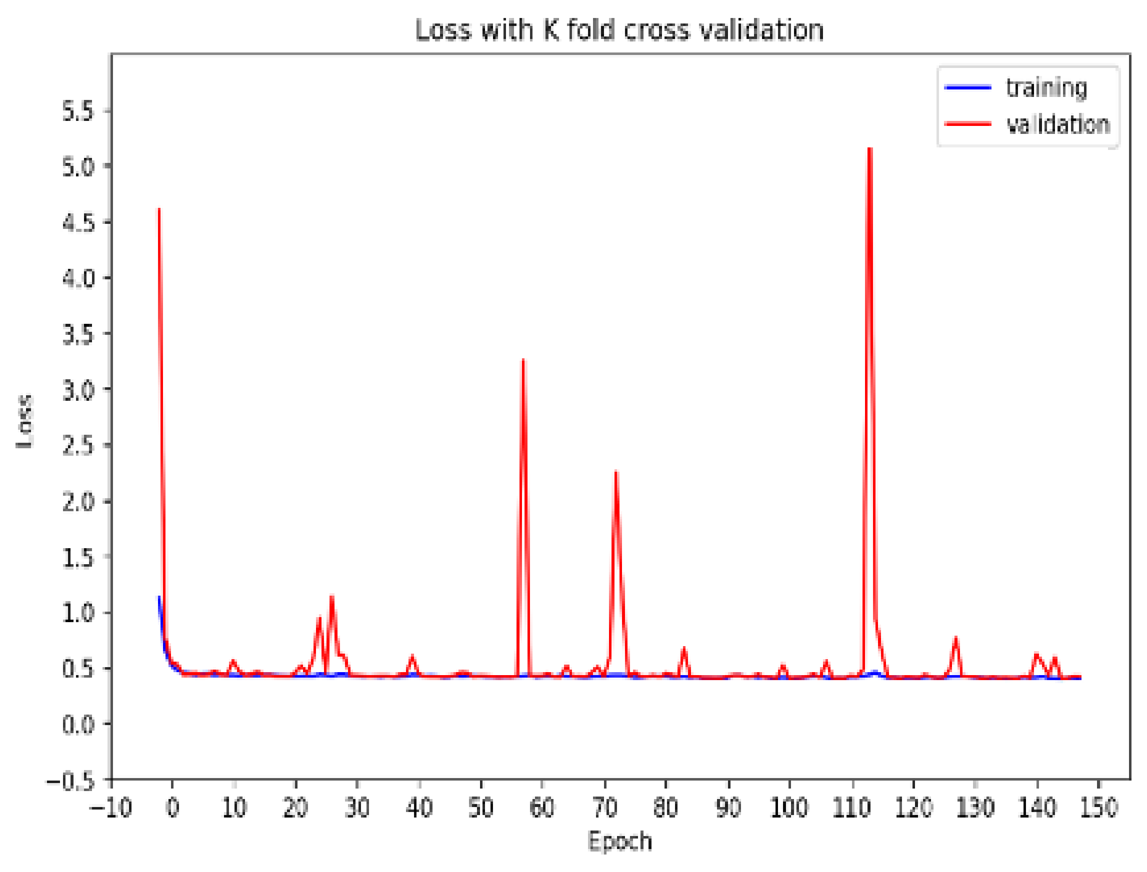


Convolutional Neural Network Layers

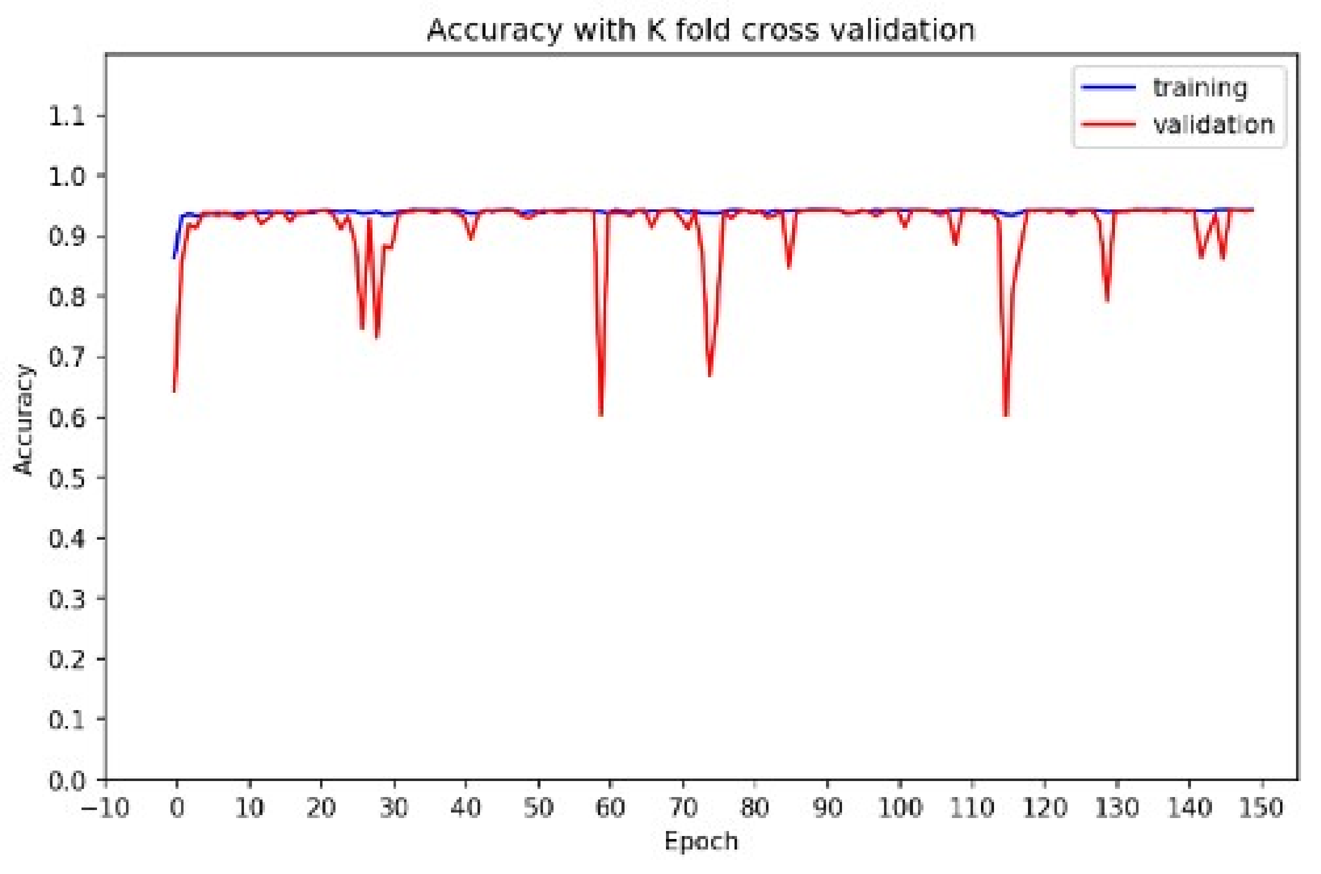
- We used an ECG image with 200 X 200 x 1 input dimension of the network.
- We used 24 hidden layers.
- The ReLU function was used to activate each hidden layer .
- batch normalization was used to normalize the input layer by adjusting and scaling the activations.



RESULTS

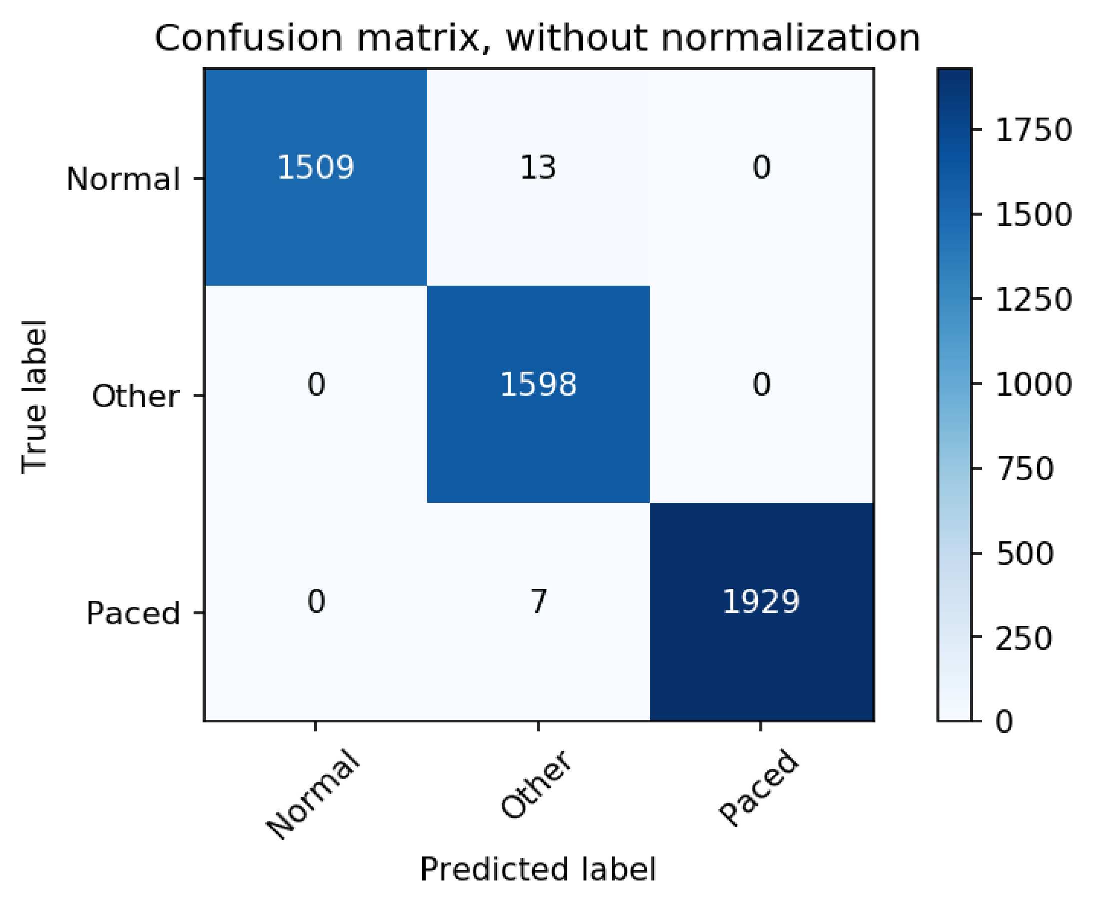


Loss with K fold cross validation
training
validation
Loss
Epoch

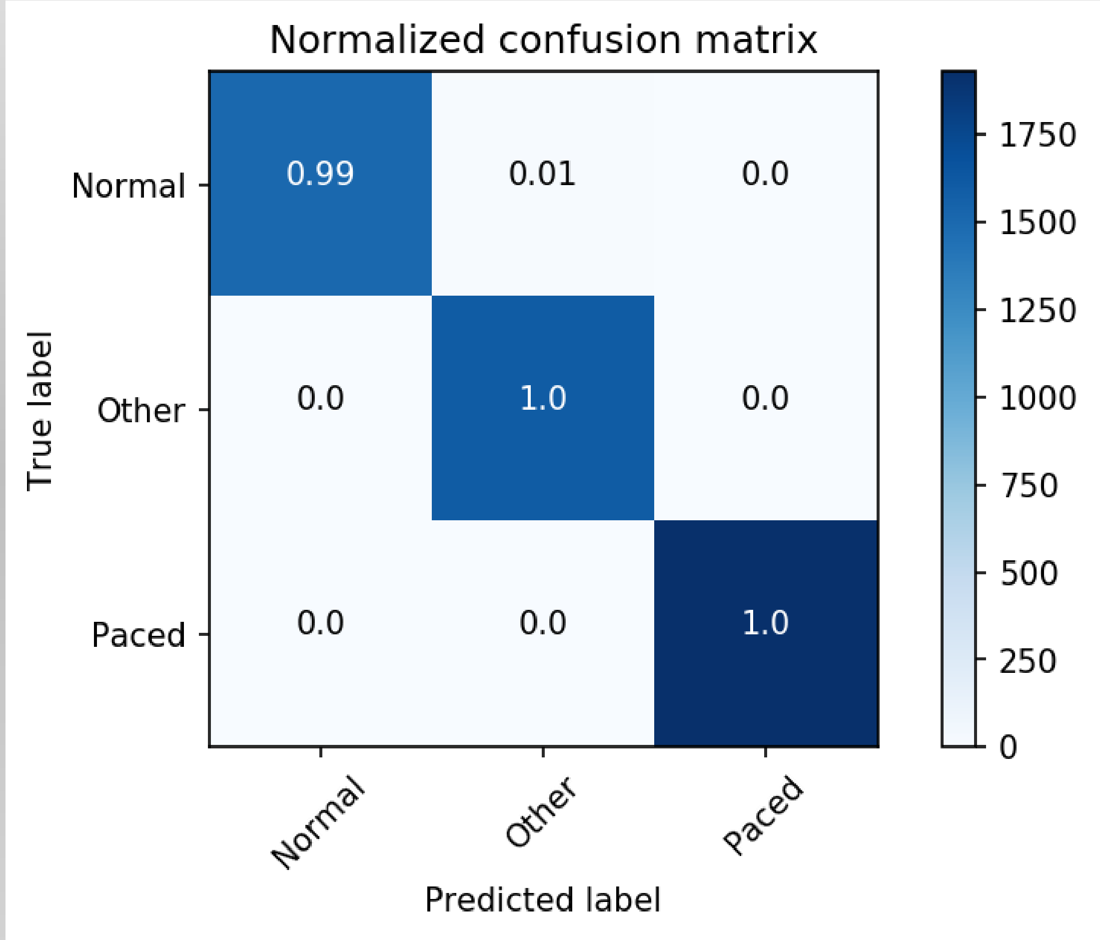


Accuracy with K fold cross validation
training
validation
Accuracy
Epoch

RESULTS



Confusion matrix, without normalization
Normal
Other
Paced
True label
1509
13
0
0
1598
0
0
7
1929
Normal
Other
Paced
Predicted label
1750
1500
1250
1000
750
500
250
0



Normalized confusion matrix
Normal
Other
Paced
True label
0.99
0.01
0.0
0.0
1.0
0.0
0.0
0.0
1.0
Normal
Other
Paced
Predicted label
1750
1500
1250
1000
750
500
250
0


Thank you