



HEART & MIND TOWARDS EXCELLENCE

**ANONIMASI SINYAL EKG (ELEKTROKARDIOGRAM) UNTUK  
KEAMANAN TRANSMISI DATA PADA SEBUAH *NODE* SENSOR**



**TUGAS AKHIR**

**Program Studi**

**S1 Sistem Komputer**

**Oleh :**

**BRAMASTA AGNANDA SETIAWAN**

**14.41020.0033**

---

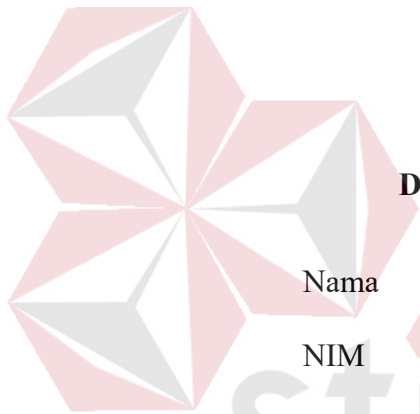
**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2018**

**ANONIMASI SINYAL EKG (ELEKTROKARDIOGRAM) UNTUK  
KEAMANAN TRANSMISI DATA PADA SEBUAH *NODE* SENSOR**

**TUGAS AKHIR**

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer



**Disusun Oleh :**

Nama : Bramasta Agnanda Setiawan

NIM : 14.41020.0033

Program : S1 (Strata Satu)

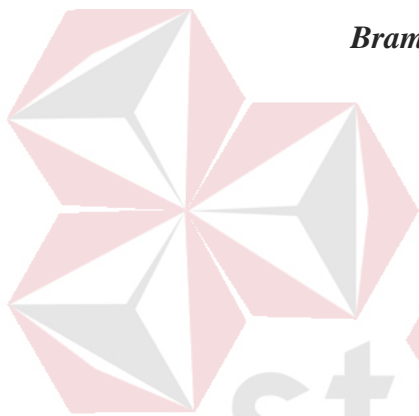
Jurusan : Sistem Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**

**2018**

***“Buanglah gengsi semata menjadi pribadi yang terus berkembang lebih baik.”***

***Bramasta Agnanda Setiawan***



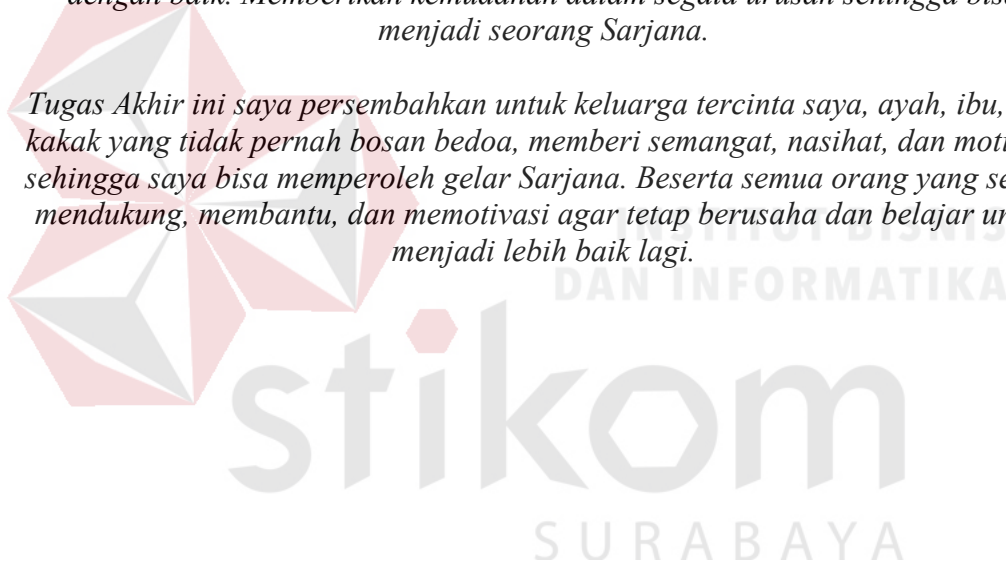
INSTITUT BISNIS  
DAN INFORMATIKA

stikom  
SURABAYA

*Alhamdulillah akhirnya penulis dapat menyelesaikan Tugas Akhir ini dengan baik.*

*Alhamdulillah rasa syukur yang tiada terhingga kepada Allah SWT, yang telah memberikan kemudahan dan kelancaran dalam menyelesaikan Tugas Akhir ini dengan baik. Memberikan kemudahan dalam segala urusan sehingga bisa menjadi seorang Sarjana.*

*Tugas Akhir ini saya persembahkan untuk keluarga tercinta saya, ayah, ibu, dan kakak yang tidak pernah bosan berdoa, memberi semangat, nasihat, dan motivasi sehingga saya bisa memperoleh gelar Sarjana. Beserta semua orang yang selalu mendukung, membantu, dan memotivasi agar tetap berusaha dan belajar untuk menjadi lebih baik lagi.*



## TUGAS AKHIR

### ANONIMASI SINYAL EKG (ELEKTROKARDIOGRAM) UNTUK KEAMANAN TRANSMISI DATA PADA SEBUAH NODE SENSOR

Dipersiapkan dan disusun oleh

**Bramasta Agnanda Setiawan**

**NIM : 14.41020.0033**

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji

Pada : Agustus 2018

#### Susunan Dewan Penguji

Pembimbing

I. Dr. Jusak

NIDN. 0708017101

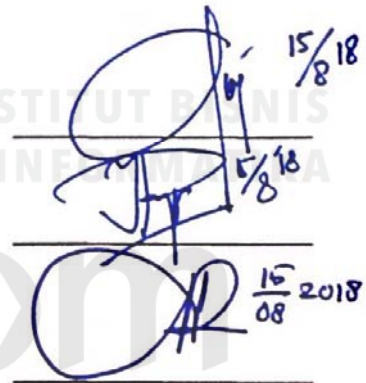
II. Ira Puspasari, S.Si., M.T.

NIDN. 0710078601

Penguji

I. Pauladie Susanto, S.Kom., M.T.

NIDN. 0729047501



15/8 2018  
15/8 2018  
15/8 2018

Tugas Akhir ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Sarjana



15/8

Dr. Jusak  
Dekan Fakultas Teknologi dan Informatika

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

## SURAT PERNYATAAN

### PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

Nama : Bramasta Agnanda Setiawan

NIM : 14.41020.0033

Program Studi : S1 Sistem Komputer

Fakultas : Fakultas Teknologi dan Informatika

Jenis Karya : Tugas Akhir

Judul Karya : ANONIMASI SINYAL EKG (ELEKTROKARDIOGRAM)

UNTUK KEAMANAN TRANSMISI DATA PADA SEBUAH  
NODE SENSOR

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Agustus 2018



Yang menyatakan

Bramasta Agnanda Setiawan

NIM : 14.41020.0033

## ABSTRAK

Sinyal EKG berisi informasi kesehatan penting seorang pasien yang harus dilindungi. Oleh karena itu, platform *e-Health* berbasis internet yang mengabaikan perlindungan informasi kesehatan merupakan ancaman bagi privasi pasien. Dalam sisi keamanan untuk melindungi transmisi sinyal EKG masih kurang

Dalam penelitian ini, penulis menggunakan metode algoritma yang didasarkan pada algoritma *Fast Fourier Transform* (FFT) yang cocok untuk sistem IMedT (*Internet of Medical Things*). Algoritma ini disebut sebagai algoritma Jusak-Seedahmed. Hal ini dikembangkan untuk membahas hal utama yang menjadi kendala di lingkungan IMedT, yaitu kebutuhan untuk mengamankan transmisi sinyal EKG. Dengan menggunakan metode ini, keamanan dari data sinyal EKG akan lebih baik.

Pada hasil penelitian, didapatkan bahwa hasil algoritma Jusak-Seedahmed dapat digunakan untuk melakukan anonimasi sinyal EKG dan enkripsi *key* dengan baik, dalam penelitian ini digunakan 3 macam sinyal EKG dengan frekuensi *sampling* 250 Hz, 500 Hz, 1000 Hz. Pengujian dengan menggunakan variasi panjang *key* yang berbeda algoritma Jusak-Seedahmed dapat melakukan anonimasi sinyal EKG. Untuk perubahan panjang *key* 128 dan *key* 256 dengan algoritma Jusak-Seedahmed tidak mempengaruhi waktu pemrosesan di MATLAB. Selain itu, didapatkan bahwa *key* dapat diunggah ke *email medical center* menggunakan protokol SMTP, sedangkan hasil anonimasi dapat diunggah ke *server* menggunakan protokol FTP.

*Keywords* : Sinyal EKG, Anonimasi, FFT.

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT karena berkat, rahmat, dan karuniaNya penulis dapat menyelesaikan penulisan Tugas Akhir ini dengan sebaik-baiknya. Penulis mengambil judul “Anonimasi Sinyal EKG (Elektrokardiogram) Untuk Keamanan Transmisi Data Pada Sebuah *Node* Sensor” ini sebagai salah satu syarat dalam menyelesaikan Tugas Akhir di Institut Bisnis dan Informatika Stikom Surabaya.

Pada kesempatan kali ini penulis juga ingin mengucapkan terima kasih kepada :

1. Allah SWT karena dengan rahmatnya dan hidayahnya penulis dapat menyelesaikan Laporan Tugas Akhir ini.
2. Orang Tua dan Saudara-saudara saya tercinta yang telah memberikan dorongan dan bantuan baik moral maupun materi sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini.
3. Bapak Dr. Jusak, selaku Dekan Fakultas Teknologi dan Informatika Institut Bisnis dan Informatika Stikom Surabaya dan dosen pembimbing pertama yang telah membantu serta memberikan dukungan kepada penulis sehingga penulis dapat melaksanakan Tugas Akhir ini dengan baik.
4. Ibu Ira Puspasari, S.Si., M.T., selaku dosen pembimbing kedua yang senantiasa memberikan dukungan kepada penulis sehingga penulis dapat melaksanakan Tugas Akhir ini dengan baik.
5. Bapak Pauladie Susanto, S.Kom., M.T., selaku pembahas yang telah membimbing penulis yang memberi masukan dalam menyusun buku Tugas



Akhir dan membimbing selama menempuh perkuliahan di Institut Bisnis dan Informatika Stikom Surabaya.

6. Seluruh dosen Pengajar Program Studi S1 Sistem Komputer yang telah mendidik, memberi motivasi kepada penulis selama masa kuliah di Institut Bisnis dan Informatika Stikom Surabaya.
7. Teman-teman angkatan 2014 maupun adik dan kakak angkatan Jurusan S1 Sistem Komputer yang mendukung dan membantu penulis selama masa dan penyusunan buku Tugas Akhir ini.
8. Teman-teman di rumah yang telah menghibur dan memberi semangat yang luar biasa kepada penulis.

Banyak hal dalam laporan Tugas Akhir ini yang masih perlu diperbaiki lagi. Oleh karena itu penulis mengharapkan saran dan kritik yang dapat membangun dari semua pihak agar dapat menyempurnakan penulisan ini kedepannya. Penulis juga memohon maaf yang besar jika terdapat kata- kata yang salah serta menyinggung perasaan pembaca. Akhir kata penulis ucapkan banyak terima kasih yang besar kepada para pembaca, semoga tulisan ini dapat bermanfaat bagi para pembaca.

Surabaya, Agustus 2018

Penulis

## DAFTAR ISI

	<b>Halaman</b>
HALAMAN JUDUL.....	i
HALAMAN SYARAT .....	ii
MOTTO .....	iii
HALAMAN PERSEMBAHAN .....	iv
HALAMAN PENGESAHAN.....	v
HALAMAN PERNYATAAN .....	vi
ABSTRAK .....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xvi
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan .....	4
1.5 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1 Jantung .....	6

2.1.1 Prinsip Kerja Jantung.....	7
2.1.2 Karakter Jantung Normal.....	7
2.1.3 Suara Jantung.....	7
2.1.4 Sifat Mekanik Jantung.....	8
2.2 Elektrokardiogram.....	9
2.2.1 Gelombang EKG Normal .....	9
2.3 Elektrokardiograf .....	12
2.4 Transformasi Fourier Diskrit.....	12
2.5 <i>Fast Fourier Transform</i> .....	13
2.6 Arduino .....	18
2.7 Arduino Uno.....	18
2.7.1 Daya.....	20
2.7.2 Memori .....	21
2.7.3 <i>Input dan Output</i> .....	21
2.7.4 <i>Software</i> Arduino IDE .....	22
2.8 Sensor Monitor Denyut Jantung AD8232.....	23
2.9 MATLAB.....	24
2.10 <i>File Transfer Protocol</i> .....	27
2.11 <i>Simple Mail Transfer Protocol</i> .....	27
2.12 <i>Server</i> .....	29
<b>BAB III METODE PENELITIAN .....</b>	<b>30</b>
3.1 Model Pengembangan.....	30
3.2 Metode Penelitian.....	30

3.3 Model Perancangan.....	32
3.4 Instrumen Penelitian.....	33
3.4.1 Alat Penelitian .....	33
3.4.2 Bahan Penelitian .....	33
3.5 Pengambilan Data Auskultasi Jantung.....	34
3.5.1 Peletakan Elektroda Jepit.....	34
3.5.2 Program Membaca Sensor EKG dan Mengirim ke PC .....	35
3.5.3 Pengambilan Sinyal Auskultasi Jantung.....	36
3.6 Perancangan Perangkat Lunak .....	37
3.6.1 Sinyal EKG .....	39
3.6.2 Pengondisian Sinyal.....	39
3.6.3 <i>Denoising</i> Sinyal.....	42
3.6.4 Transformasi FFT dan Partisi Sinyal EKG.....	45
3.6.5 Pemisahan $X_1[k]$ dari $X[k]$ .....	46
3.6.6 Modifikasi $X_2[k]$ .....	47
3.6.7 Pembuatan kunci keamanan $K$ .....	47
3.6.8 Pengunggahan pada <i>server</i> dan <i>email medical center</i> .....	47
BAB IV HASIL .....	48
4.1 Pengujian Pengambilan Data .....	48
4.1.1 Tujuan Pengujian Pengambilan Data.....	48
4.1.2 Kebutuhan Pengujian Pengambilan Data .....	48
4.1.3 Prosedur Pengujian Pengambilan Data.....	49
4.1.4 Hasil Pengujian Pengambilan Data .....	50

4.2	Pengujian <i>Denoising</i> .....	51
4.2.1	Tujuan Pengujian <i>Denoising</i> .....	51
4.2.2	Kebutuhan Pengujian <i>Denoising</i> .....	51
4.2.3	Prosedur Pengujian <i>Denoising</i> .....	52
4.2.4	Hasil Pengujian <i>Denoising</i> .....	53
4.3	Pengujian FFT .....	58
4.3.1	Tujuan Pengujian FFT .....	59
4.3.2	Kebutuhan Pengujian FFT .....	59
4.3.3	Prosedur Pengujian FFT .....	59
4.3.4	Hasil Pengujian FFT .....	60
4.4	Pengujian Anonimasi .....	66
4.4.1	Tujuan Pengujian Anonimasi .....	67
4.4.2	Kebutuhan Pengujian Anonimasi .....	67
4.4.3	Prosedur Pengujian Anonimasi.....	67
4.4.4	Hasil Pengujian Anonimasi .....	68
4.5	Pengujian Enkripsi .....	74
4.5.1	Tujuan Pengujian Enkripsi .....	75
4.5.2	Kebutuhan Pengujian Enkripsi .....	75
4.5.3	Prosedur Pengujian Enkripsi.....	75
4.5.4	Hasil Pengujian Enkripsi .....	76
4.6	Pengujian Perubahan Panjang <i>Key</i> .....	78
4.6.1	Tujuan Pengujian Perubahan Panjang <i>Key</i> .....	78
4.6.2	Kebutuhan Pengujian Perubahan Panjang <i>Key</i> .....	78

4.6.3	Prosedur Pengujian Perubahan Panjang <i>Key</i> .....	78
4.6.4	Hasil Pengujian Perubahan Panjang <i>Key</i> .....	79
4.7	Pengujian Unggah Data Pada <i>Server</i> .....	80
4.7.1	Tujuan Pengujian Unggah Data Pada <i>Server</i> .....	80
4.7.2	Kebutuhan Pengujian Unggah Data Pada <i>Server</i> .....	80
4.7.3	Prosedur Pengujian Unggah Data Pada <i>Server</i> .....	80
4.7.4	Hasil Pengujian Unggah Data Pada <i>Server</i> .....	81
4.8	Pengujian Unggah Data Pada <i>Email</i> .....	82
4.8.1	Tujuan Pengujian Unggah Data Pada <i>Email</i> .....	82
4.8.2	Kebutuhan Pengujian Unggah Data Pada <i>Email</i> .....	82
4.8.3	Prosedur Pengujian Unggah Data Pada <i>Email</i> .....	83
4.8.4	Hasil Pengujian Unggah Data Pada <i>Email</i> .....	84
BAB V	PENUTUP.....	85
5.1	Kesimpulan .....	85
5.2	Saran.....	86
DAFTAR	PUSTAKA .....	87
LAMPIRAN	.....	89
Lampiran 1.	<i>Listing</i> Program Arduino .....	89
Lampiran 2.	<i>Listing</i> Program MATLAB.....	89
BIODATA	PENULIS .....	97

## DAFTAR GAMBAR

Gambar 2.1 Anatomi Jantung Manusia.....	6
Gambar 2.2 Bunyi Jantung Normal .....	8
Gambar 2.3 Sinyal Jantung Normal .....	10
Gambar 2.4 QRS Kompleks dan Nilai Normal Amplitudo dan Durasi dari Gelombang .....	12
Gambar 2.5 Perbandingan Kompleksitas DFT dan FFT .....	15
Gambar 2.6 Algoritma Cooley-Tukey .....	15
Gambar 2.7 Algoritma Sande-Tukey .....	15
Gambar 2.8 Proses Kerja Radix-2.....	17
Gambar 2.9 Proses Kerja Radix-4.....	18
Gambar 2.10 Arduino Uno Sisi Depan (Kiri) dan Belakang (Kanan).....	19
Gambar 2.11 Tampilan <i>Software</i> Arduino IDE .....	23
Gambar 2.12 Sensor Monitor Denyut Jantung AD8232.....	24
Gambar 2.13 MATLAB versi R2014B.....	26
Gambar 3.1 Diagram Blok Sistem Keseluruhan.....	31
Gambar 3.2 Ilustrasi Pihak Dokter Memperoleh Sinyal EKG .....	31
Gambar 3.3 Diagram Blok Perancangan Sistem.....	32
Gambar 3.4 Letak Posisi Elektroda Jepit .....	34
Gambar 3.5 <i>Flowchart</i> Pengiriman Data Auskultasi Jantung.....	35
Gambar 3.6 Program Pengambilan Sinyal EKG Pada Arduino IDE.....	36
Gambar 3.7 Hasil Sinyal Auskultasi Jantung Selama 10 Detik.....	37
Gambar 3.8 Diagram Blok Perancangan Perangkat Lunak Anonimasi Sinyal EKG .....	38
Gambar 3.9 Data Proses Auskultasi Jantung .....	39
Gambar 3.10 <i>File</i> Txt Data Sinyal EKG.....	41
Gambar 3.11 Hasil Visualisasi Data EKG Setelah Dikonversi.....	41
Gambar 3.12 Program GUI Impor Sinyal EKG Pada MATLAB .....	42
Gambar 3.13 Proses <i>Denoising</i> Sinyal EKG.....	42
Gambar 3.14 Program GUI <i>Denoising</i> Sinyal EKG .....	45





## DAFTAR TABEL

Tabel 2.1 Susunan Data <i>Bit-Reversed</i> .....	16
Tabel 2.2 Spesifikasi Arduino Uno.....	19
Tabel 4.1 Data Sinyal EKG.....	51
Tabel 4.2 Pengujian Perubahan <i>Key</i> .....	79



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Jantung merupakan salah satu organ penting dalam tubuh manusia yang berperan dalam aliran sistem peredaran darah. Penyakit jantung menjadi salah satu penyebab kematian terbesar di dunia. Penyakit jantung dapat dideteksi lebih dini dengan menggunakan alat elektrokardiograf. Tes medis EKG berfungsi untuk mendeteksi kelainan pada jantung dengan mengukur aktivitas listrik yang dihasilkan oleh jantung.

Sinyal EKG berisi informasi kesehatan penting seorang pasien. Oleh karena itu, sinyal EKG bersifat unik untuk setiap individu dalam jangka waktu yang panjang. Sinyal EKG juga dapat bertindak sebagai identitas biometrik untuk membedakan informasi spesifik yang dimiliki orang tertentu. Fitur ini membawa konsekuensi langsung pada transmisi sinyal EKG dari *node* sensor ke penyedia layanan kesehatan melalui jaringan publik yang membuatnya rentan terhadap serangan dari luar. Oleh karena itu, platform *e-Health* berbasis internet yang mengabaikan perlindungan informasi kesehatan merupakan ancaman bagi privasi pasien. Sayangnya, belum ada *platform e-Health* yang menerapkan perlindungan transmisi sinyal EKG.

Pada tahun 2016, telah dilakukan penelitian oleh Wahyu Rokhman Sugiarto yang berjudul “Rancang Bangun Alat Elektrokardiograf Untuk Visualisasi, Perekaman, dan Penyimpanan Sinyal Jantung”, penelitian ini melakukan

pembuatan alat yang berfungsi merekam, menyimpan, dan memvisualisasikan sinyal EKG untuk selanjutnya dianalisis. Namun untuk proses analisis tidak hanya melihat secara visual saja, oleh karena itu terdapat penelitian tentang analisis frekuensi.

Pada tahun 2016, telah dilakukan penelitian oleh Pangky Ari Wibowo yang berjudul “Analisis *Time-Frequency* Pada Sinyal ECG (*Electrocardiogram*) dengan Menggunakan *Continuous Wavelet Transform*”, penelitian ini melakukan transformasi sinyal EKG ke *domain* frekuensi dan menganalisis sinyal EKG dengan menggunakan metode *continous* wavelet pada jantung normal.

Dengan acuan pada penelitian di atas, proses transmisi data tidak memiliki keamanan. Karena itu, dalam penelitian ini sebelum proses transmisi data, akan dilakukan proses anonimasi untuk memberikan tambahan fitur keamanan bagi data EKG. Hal ini dikarenakan data EKG merupakan data personal yang harus dilindungi. Jika terdapat data EKG yang tersimpan di *server* dicuri oleh *hacker*, data EKG itu sudah berubah bentuk dan hanya bisa dikembalikan bila ada kunci. Metode anonimasi yang berbasis algoritma *Fast Fourier Transform* (FFT) diusulkan untuk menguraikan sinyal EKG, dengan menggantikan koefisien dari nodus frekuensi rendah dengan angka nol dan merekonstruksi sinyal EKG frekuensi tinggi untuk proses anonimasi. Metode ini sebagai peningkatan kinerja dari metode sebelumnya, yaitu metode anonimasi berdasarkan algoritma wavelet. Algoritma ini dilengkapi dengan fungsi reversibel atau operasi untuk menyembunyikan fitur *fiducial* dan *non-fiducial* dari sinyal EKG yang normal maupun tidak normal. Di sisi penerima, hanya personil resmi yang memiliki kunci rahasia dan mengetahui

fungsi reversibel yang dapat merekonstruksi sinyal EKG asli dari sinyal EKG yang telah dianonimkan.

Dalam penelitian ini, penulis menggunakan metode algoritma anonimasi didasarkan pada algoritma *Fast Fourier Transform* (FFT) yang cocok untuk sistem IMedT (*Internet of Medical Things*). Algoritma ini selanjutnya disebut sebagai algoritma Jusak-Seedahmed. Hal ini dikembangkan untuk membahas hal utama yang menjadi kendala di lingkungan IMedT, yaitu kebutuhan untuk mengamankan transmisi sinyal EKG dan merupakan suatu metode yang efisien. Dengan menggunakan metode ini, keamanan dari data sinyal EKG akan lebih baik.

## 1.2 Rumusan Masalah

Adapun rumusan masalah dalam pengerjaan tugas akhir ini adalah Bagaimana menerapkan metode anonimasi data sinyal EKG dengan menggunakan algoritma Jusak-Seedahmed berdasarkan algoritma FFT untuk memberikan fitur keamanan dalam proses transmisi data bagi sinyal EKG.

## 1.3 Batasan Masalah

Dalam perancangan dan pembuatan alat ini, terdapat beberapa batasan masalah, antara lain :

1. Data uji yang digunakan adalah sinyal EKG untuk jantung normal dengan *sampling* waktu 4 ms selama 10 detik, 2 ms selama 11 detik, 1 ms selama 13 detik.
2. Alat sistem tertanam yang digunakan adalah Arduino Uno dengan perangkat lunak pengolah Arduino IDE dan MATLAB 2014B.

## 1.4 Tujuan

Tujuan dari penelitian yang dilakukan adalah menerapkan metode anonimasi data sinyal EKG dengan menggunakan algoritma Jusak-Seedahmed berdasarkan algoritma FFT untuk memberikan fitur keamanan dalam proses transmisi data bagi sinyal EKG.

## 1.5 Sistematika Penulisan

Adapun Pembahasan Tugas Akhir ini secara garis besar tersusun dari 5 (lima) bab, yaitu diuraikan sebagai berikut :

### 1. BAB I PENDAHULUAN

Pada bab ini dibahas mengenai latar belakang masalah, batasan masalah, tujuan penulisan, dan sistematika penulisan.

### 2. BAB II LANDASAN TEORI

Pada bab ini dibahas mengenai teori penunjang dari permasalahan, yaitu mengenai jantung, Elektrokardiogram, Elektrokardiograf, Transformasi Fourier Diskrit, *Fast Fourier Transform*, Arduino, Arduino Uno, MATLAB, *File Transfer Protocol*, *Simple Mail Transfer Protocol*, dan *server*.

### 3. BAB III METODE PENELITIAN DAN PERANCANGAN SISTEM

Pada bab ini dibahas mengenai model pengembangan, metode penelitian, model perancangan, instrumen penelitian, pengambilan data auskultasi jantung, perancangan perangkat lunak.

### 4. BAB IV HASIL DAN PEMBAHASAN

Pada bab ini dibahas mengenai pengujian pengambilan data, pengujian *Fast Fourier Transform*, pengujian anonimasi sinyal, pengujian enkripsi, pengujian

perubahan panjang *key*, pengujian unggah data pada *server*, dan pengujian unggah data pada *email*.

## 5. BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan yang didapat dari hasil penelitian berdasarkan rumusan masalah serta saran untuk perkembangan penelitian selanjutnya.

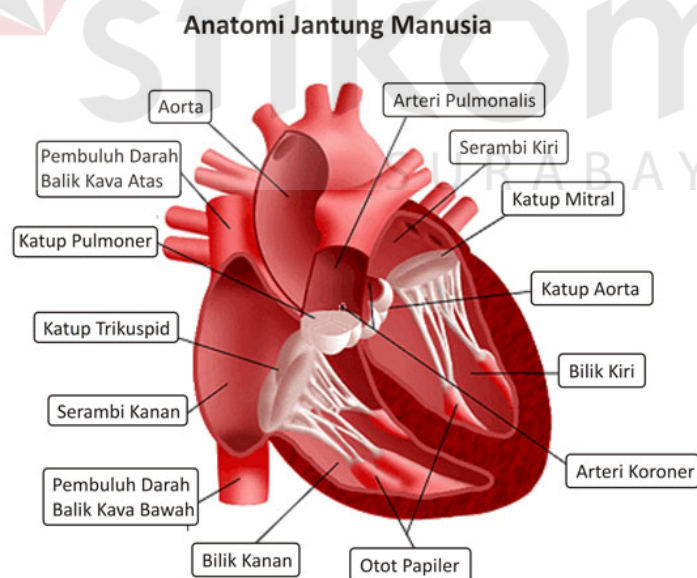


## BAB II

### LANDASAN TEORI

#### 2.1 Jantung

Jantung merupakan salah satu organ penting dalam tubuh manusia. Jantung terdiri atas bilik dan serambi. Jantung berfungsi sebagai pompa untuk menekan darah agar dapat disalurkan keseluruh tubuh dan kembali ke jantung. Otot-otot jantung berfungsi untuk memompa darah dari satu ruangan menuju ruangan lain. Saat terjadi proses pemompaan, katup jantung akan terbuka sehingga darah dapat mengalir menuju ruangan yang dituju dan akan langsung menutup untuk mencegah aliran balik darah (Wibowo, 2016). Anatomi jantung manusia ditunjukkan pada Gambar 2.1.



Gambar 2.1 Anatomi Jantung Manusia

### 2.1.1 Prinsip Kerja Jantung

Keadaan jantung menguncup menandakan jantung sedang memompa darah (kontraksi), sedangkan saat mengembang menandakan jantung sedang tidak memompa (relaksasi). Cara kerja jantung sebagai berikut :

- a. Jika kedua serambi mengembang, maka darah dari pembuluh balik akan masuk ke serambi.
- b. Jika kedua serambi menguncup dan bilik mengembang, maka darah dari serambi masuk ke bilik.
- c. Jika kedua bilik menguncup, maka darah keluar dari bilik menuju ke pembuluh aorta.

### 2.1.2 Karakter Jantung Normal

Untuk mengetahui normal tindaknya jantung maka ada beberapa karakteristik yang dapat dilihat seperti :

1. Koefisien maksimum rata-rata sebesar 6753,489 (Wibowo, 2016)
2. Energi maksimum rata-rata sebesar 58322836,81 (Wibowo, 2016)
3. Jarak normal untuk interval R-R pada sinyal EKG adalah 0,6 s sampai 1 s (Wibowo, 2016)

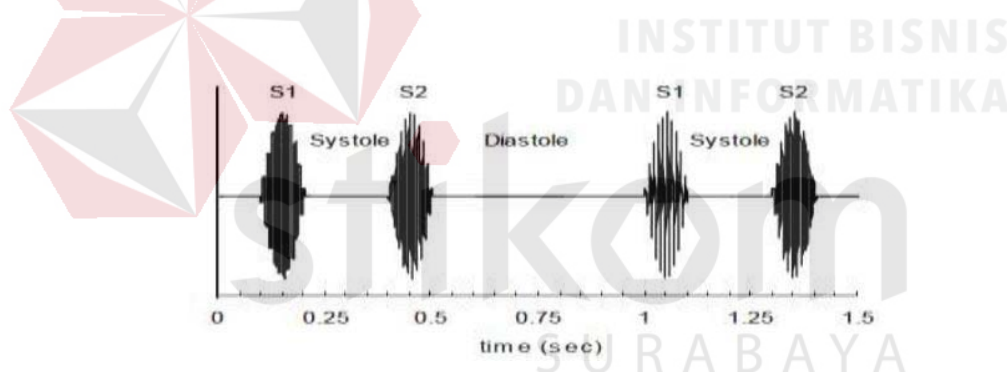
### 2.1.3 Suara Jantung

Detak jantung menghasilkan dua suara yang berbeda yang dapat didengarkan pada stetoskop, yang sering dinyatakan dengan lub-dub. Suara lub disebabkan oleh penutupan katup triscupid dan mitral (atrioventrikular) yang memungkinkan aliran darah dari atrium (serambi jantung) ke ventricle (bilik jantung) dan mencegah aliran



balik dan dapat disebut dengan suara jantung pertama (S1) yang terjadi pada awal systole (periode jantung berkontraksi). Suara dub disebut suara jantung kedua (S2) dan disebabkan oleh penutupan katup semilunar (*aortic* dan *pulmonary*) yang membebaskan darah ke sistem sirkulasi paru-paru dan sistemik. Katup ini tertutup pada akhir systole dan sebelum katup atrioventrikular membuka kembali. Suara jantung ketiga (S3) sesuai dengan berhentinya pengisian atrioventrikular, sedangkan suara jantung keempat (S4) memiliki korelasi dengan kontraksi atrial. Suara S4 ini memiliki amplitudo yang sangat rendah dan komponen frekuensi rendah (Rizal, 2007).

Dalam kondisi normal, pada dasarnya terdapat dua macam bunyi jantung, yaitu S1 dan S2 seperti ditunjukkan Gambar 2.2.



Gambar 2.2 Bunyi Jantung Normal

#### 2.1.4 Sifat Mekanik Jantung

Sebagai Siklus jantung adalah peristiwa yang terjadi pada jantung mulai dari awal suatu denyut jantung sampai dengan mulainya denyut jantung berikutnya yang termasuk di dalamnya periode kontraksi dan relaksasi. Setiap siklus jantung terdiri dari peristiwa listrik-potensial aksi, dan mekanik-kontraksi didalam sistem

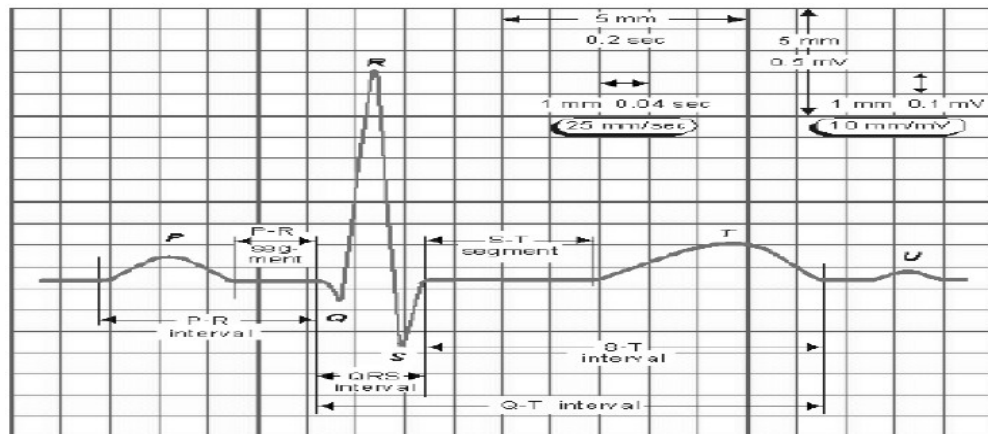
kardiovaskuler. Tekanan yang ditimbulkan oleh kontraksi jantung diubah menjadi aliran yang bertujuan untuk menyediakan kebutuhan oksigen dan nutrisi bagi seluruh jaringan tubuh siklus jantung terdiri dari satu periode relaksasi yaitu diastol, dimana terjadi pengisian jantung dengan darah, kemudian diikuti oleh periode kontraksi yang disebut sistol. Dalam setiap siklus, terjadi perubahan tekanan pada atria, ventrikel maupun aorta serta terjadi perubahan volume ventrikel. Semua peristiwa mekanik ini sesuai dengan aktifitas listrik yang dapat dicatat dengan EKG. Selain itu, peristiwa mekanik akibat kontraksi jantung akan menimbulkan suara jantung akibat menutupnya katup jantung.

## **2.2 Elektrokardiogram**

Elektrokardiogram adalah sinyal fisiologi yang diperoleh dari aktivitas kelistrikan jantung. Sinyal ini direkam melalui perangkat elektrokardiograf, sebuah perangkat keras yang berfungsi mencatat aktifitas listrik pada jantung. Prinsip kerja alat ini yaitu dengan mengukur perbedaan potensial listrik pada tubuh manusia. Perbedaan muatan inilah yang mengakibatkan tubuh manusia mampu menghasilkan sinyal listrik yang dapat mengatur syaraf, otot, dan organ tubuh lainnya.

### **2.2.1 Gelombang EKG Normal**

Dalam sinyal EKG memiliki beberapa ciri pada setiap gelombang yang penting untuk digunakan dalam memberi diagnosa pada jantung. Sinyal EKG yang normal dapat dilihat pada Gambar 2.3.



Gambar 2.3 Sinyal Jantung Normal

**a. Gelombang P**

Gelombang awal yang merupakan hasil depolarisasi di kedua atrium disebut gelombang P. Lebar gelombang P yang normal kurang dari 0,12 s dan amplitudonya tidak lebih dari 0,3 mV. Gelombang P yang normal selalu defleksi positif (cembung ke atas) di semua sandapan dan selalu defleksi negatif (cekung ke bawah) di sandapan aVR (Wibowo, Jusak, & Puspasari, 2016)

**b. Gelombang Q**

Gelombang Q merupakan gelombang defleksi negatif setelah gelombang P. Lebar dari gelombang Q tidak lebih dari 0,04 s dan dalamnya kurang dari 1/3 tinggi gelombang R.

**c. Gelombang R**

Gelombang R merupakan gelombang defleksi positif setelah gelombang P atau setelah gelombang Q. Gelombang ini umumnya selalu positif di semua sandapan, kecuali aVR dan hal ini masih normal.

**d. Gelombang S**

Gelombang ini merupakan gelombang defleksi negatif setelah gelombang R atau gelombang Q. Gelombang S berangsur-angsur menghilang pada sandapan V1-V6. Gelombang ini sering terlihat lebih dalam di sandapan V1 dan aVR. Hal ini masih normal.

**e. Gelombang T**

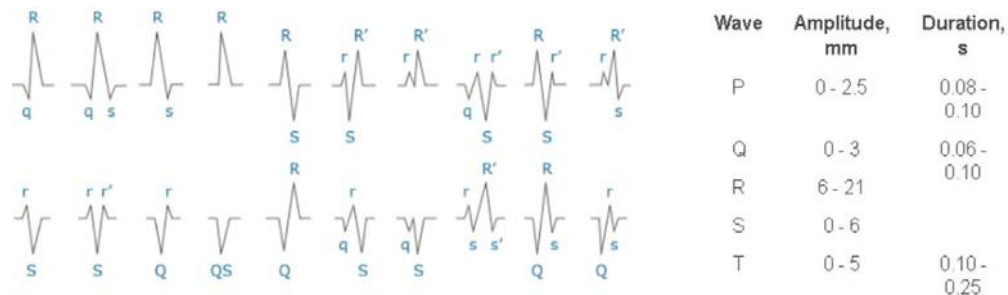
Gelombang T adalah hasil repolarisasi di kedua ventrikel. Normalnya positif dan *inverted* (terbalik) di aVR. Pada sandapan V1 bisa memiliki gelombang T yang positif atau negatif.

**f. Interval P-R**

Interval P-R adalah selisih waktu awal kompleks QRS dengan waktu awal gelombang P. Interval ini menggambarkan waktu yang diperlukan depolarisasi atrium sampai awal depolarisasi ventrikel atau waktu yang diperlukan impuls listrik dari nodus SA menuju serabut Purkinje dan normalnya 0,12 - 0,20 s.

**g. Kompleks QRS**

Gelombang kedua setelah gelombang P biasa disebut kompleks QRS. Gelombang QRS merupakan hasil depolarisasi yang terjadi di kedua ventrikel yang dapat direkam oleh mesin EKG. Normalnya lebar kompleks QRS adalah 0,06-0,12 s dengan amplitudo yang bervariasi tergantung pada sandapan. Tidak setiap kompleks QRS memuat gelombang Q, gelombang R, dan gelombang S. Nilai normal dari QRS Kompleks dapat dilihat pada Gambar 2.4.



Gambar 2.4 QRS Kompleks dan Nilai Normal Amplitudo dan Durasi dari Gelombang

### 2.3 Elektrokardiograf

Elektrokardiograf adalah alat bantu yang berfungsi mendiagnosa gangguan pada jantung. Alat ini merekam informasi kondisi jantung yang diambil dengan memasang elektroda pada badan. Penggunaan alat ini dipelopori oleh Einthoven pada tahun 1903 dengan menggunakan Galvanometer. Senar Galvanometer ini merupakan suatu instrumen yang sangat peka untuk mencatat perbedaan kecil dari tahanan milivolt pada jantung.

### 2.4 Transformasi Fourier Diskrit

Transformasi Fourier Diskrit atau disebut dengan *Discrete Fourier Transform* (DFT) adalah model transformasi Fourier yang dikenakan pada fungsi diskrit, dan hasilnya juga diskrit. Misalkan  $f(x)$  periodik. Diasumsikan  $N$  merupakan bagian – bagian yang diukur dari  $f(x)$  yang diambil pada interval  $0 \leq x \leq 2\pi$  dengan jarak yang teratur dari titik-titik

$$x_k = \frac{2\pi k}{N}, k = 0, 1, \dots, N-1. \quad (1)$$

DFT didefinisikan dengan:

$$\hat{f}_j = \sum_{k=0}^{N-1} f_k e^{ijx_k}, f_k = f(x_k), j = 0, \dots, N-1 \quad (2)$$

Persamaan (2) inilah yang akan menghasilkan spektrum frekuensi dari sinyal.

Misal diberikan

$$e_{jk} = e^{-ijx_k} = e^{-\frac{2\pi jk}{N}} = w^{jk}, w = w_N = e^{-\frac{2\pi i}{N}} \quad (3)$$

dimana  $j, k \in \{0, 1, \dots, N-1\}$ . DFT seperti pada Persamaan (3) dinamakan dengan DFT 1 dimensi, DFT semacam ini banyak digunakan dalam pengolahan sinyal digital.

## 2.5 Fast Fourier Transform

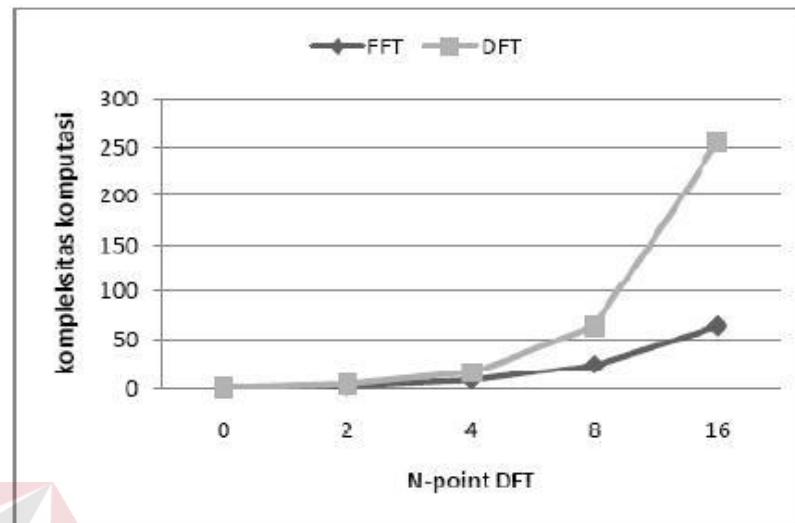
*Fast Fourier Transform* (FFT) adalah transformasi *fourier* yang dikembangkan dari algoritma *Discrete Fourier Transform* (DFT). Dengan metode FFT, laju komputasi dari perhitungan transformasi *fourier* dapat ditingkatkan. Komputasi DFT adalah komputasi yang memerlukan waktu untuk proses *looping* dan memerlukan banyak memori. Dengan menerapkan algoritma FFT, perhitungan DFT dapat dipersingkat, dalam hal ini proses *looping* dapat direduksi. Dilihat dari metode yang digunakan, FFT dibagi menjadi dua yaitu DIT (*Decimation in Time*) dan metode DIF (*Decimation in Frequency*), namun keduanya memiliki fungsi yang sama yaitu untuk mentransformasi sinyal menjadi frekuensi dasarnya. Decimation adalah proses pembagian sinyal menjadi beberapa bagian yang lebih kecil yang bertujuan untuk memperoleh waktu proses yang lebih cepat. Jika *input*

sinyal pada *time domain* dari *N-points* adalah  $x(n)$ , langkah awal yang dilakukan adalah dengan memisahkan menjadi 2 bagian yang sama (*N/2 points*) (Dianputra, 2014).

FFT (*Fast Fourier Transform*) merupakan salah satu metode untuk transformasi sinyal suara dalam *domain* waktu menjadi sinyal dalam *domain* frekuensi, artinya proses perekaman suara disimpan dalam bentuk digital berupa gelombang *spectrum* suara yang berbasis frekuensi sehingga lebih mudah dalam menganalisa *spectrum* frekuensi suara yang telah direkam (Sipasulta, 2014).

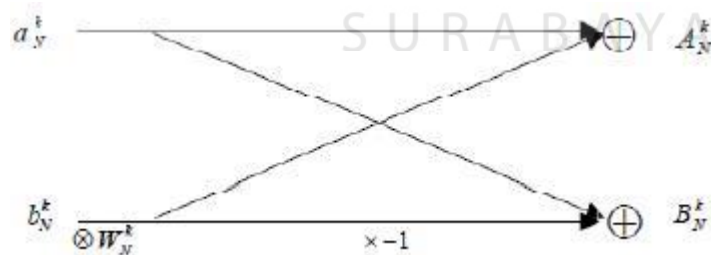
FFT dipergunakan untuk mengurangi kompleksitas transformasi yang dilakukan dengan DFT. Sebagai perbandingan, bila kita menggunakan DFT, maka kompleksitas transformasi kita adalah sebesar  $O(N^2)$ , sementara dengan menggunakan FFT, selain waktu transformasi yang lebih cepat, kompleksitas transformasi pun menurun, menjadi  $O(N \log(N))$ . Untuk jumlah *sample* yang sedikit mungkin perbedaan kompleksitas tidak begitu terasa, namun lain ceritanya bila kita mengambil jumlah *sample* yang sedikit lebih banyak. Misalnya kita hanya mengambil 2 *sample*, dengan menggunakan DFT, tingkat kompleksitas transformasi kita adalah 4, sementara dengan menggunakan FFT kompleksitasnya sebesar 0,602. Perbedaan yang semakin mencolok tampak bila kita mengambil jumlah *sample* yang lebih banyak lagi, misalnya kita ingin meninjau 64 titik *sample*, maka kompleksitas dengan menggunakan DFT adalah sebesar 4096, sementara dengan menggunakan FFT kompleksitasnya menjadi 115,6. Perbedaan yang sangat mencolok melihat perbandingan yang mencapai hampir 40 kali lipatnya. Kompleksitas transformasi ini terutama menjadi vital saat

diimplementasikan pada perangkat riil. Perbandingan kompleksitas DFT dan FFT dapat dilihat pada Gambar 2.5

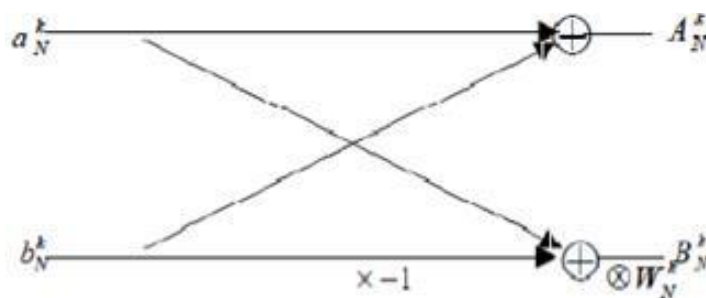


Gambar 2.5 Perbandingan Kompleksitas DFT dan FFT  
(Sumber : <https://deeto88.wordpress.com>)

Secara umum, terdapat dua buah pendekatan yang dijalankan dalam algoritma FFT. DIT merupakan usulan pendekatan yang dikemukakan oleh Cooley-Tukey, sementara DFT merupakan usulan pendekatan algoritma yang dikemukakan oleh Sande-Tukey. Perbedaan utama antara dua pendekatan ini dapat dilihat pada Gambar 2.6 dan Gambar 2.7.



Gambar 2.6 Algoritma Cooley-Tukey



Gambar 2.7 Algoritma Sande-Tukey



Dalam implementasi perangkat, pendekatan DIF lebih disenangi karena data input dapat langsung dimasukkan sesuai dengan posisi aslinya, *natural*, sementara dalam pendekatan DIT data input harus terlebih dahulu diubah secara *bit-reversed*. Pengertian susunan data masukan *natural* atau *bit-reversed* secara lebih mudah dapat dilihat pada Tabel 2.1.

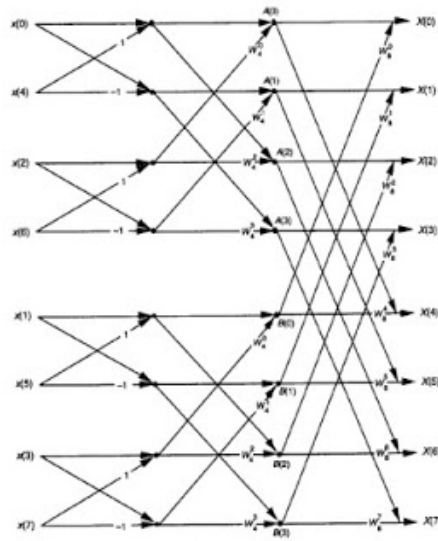
Tabel 2.1 Susunan Data *Bit-Reversed*

Input ke –	Deretan bit pada urutan normal	Deretan <i>bit-reversed</i>
1	000 (0)	000 (0)
2	001 (1)	100 (4)
3	010 (2)	010 (2)
4	011 (3)	110 (6)
5	100 (4)	001 (1)
6	101 (5)	101 (5)
7	110 (6)	011 (3)
8	111 (7)	111 (7)

Dalam transformasi sinyal waktu diskrit dengan menggunakan FFT dikenal istilah *butterfly*. *Butterfly* adalah elemen terkecil dari suatu operasi FFT. Masing-masing jenis pendekatan algoritma FFT dapat dikelompokkan lagi menjadi beberapa jenis. Beberapa jenis yang sering digunakan dalam pendekatan FFT adalah Radix-2, Radix-4 serta Radix-8.

### Radix-2

Radix-2 berarti, sejumlah *N-sample* yang akan ditransformasikan dibagi menjadi 2 kelompok dalam tiap kali rekursi yang terjadi. Pada Gambar 2.8 berikut ini dapat dilihat bagaimana proses kerja Radix-2

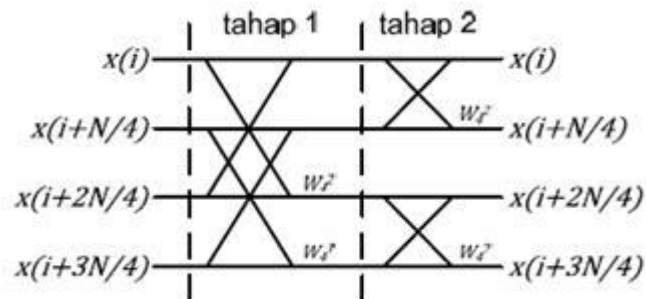


Gambar 2.8 Proses Kerja Radix-2

Dari contoh di atas kita dapat melihat, untuk mentransformasikan 8 titik *sample* dengan menggunakan Radix-2, maka dibutuhkan 3 tahap proses, angka ini diperoleh dari  $\log_2 8 = 3$ . Oleh karena tiap tahapannya data masukan dikomputasi menjadi dua bagian, maka jumlah data masukan Radix-2 harus sebesar

#### Radix-4

Radix-4 membagi data masukan dalam 4 kelompok dalam tiap kali rekursi. Pada dasarnya Radix-4 adalah Radix-2 untuk 4 titik *sample*. Oleh karena itu, dapat dikatakan bahwa Radix-4 adalah salah satu generalisasi dari Radix-2. Oleh karena jumlah titik *sample* harus sebesar , maka ada beberapa kasus yang dapat ditangani dengan menggunakan algoritma Radix-2 namun tidak dapat ditangani dengan menggunakan Radix-4. Pada Gambar 2.9 akan ditunjukkan visualisasi dari algoritma Radix-4



Gambar 2.9 Proses Kerja Radix-4

## 2.6 Arduino

Arduino adalah *prototype platform* elektronik *open source* yang terdiri mikrokontroler, bahasa pemrograman, dan IDE. Arduino adalah alat untuk membuat aplikasi interaktif, yang dirancang untuk mempermudah proyek bagi pemula, tapi masih fleksibel bagi para ahli untuk mengembangkan proyek – proyek yang kompleks.

## 2.7 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 pin *input* dari *output* digital dimana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP *header*, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board* Arduino Uno ke *Personal Computer* dengan menggunakan kabel USB atau listrik dengan AC yang-ke *adaptor*-DC atau baterai untuk menjalankannya.

Uno berbeda dengan semua *board* sebelumnya dalam hal koneksi USB-to-*serial* yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter

*USB-to-serial* berbeda dengan *board* sebelumnya yang menggunakan *chip* FTDI driver *USB-to-serial*.

Nama “Uno” berarti satu dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian *board* USB Arduino, dan sebagai model referensi untuk *platform* Arduino. Pada Gambar 2.10 ditunjukkan perangkat Arduino Uno dari sisi depan dan sisi belakang.



Gambar 2.10 Arduino Uno Sisi Depan (Kiri) dan Belakang (Kanan)

Secara umum Arduino terdiri dari dua bagian, yaitu:

1. *Hardware*: papan *input/output* (I/O)
2. *Software*: *software* arduino meliputi IDE untuk menulis program, *driver* untuk koneksi dengan *Personal Computer*, contoh program dan *library* untuk pengembangan program. (Djuandi, 2011). Berikut adalah spesifikasi dari Arduino Uno pada Tabel 2.2.

Tabel 2.2 Spesifikasi Arduino Uno

Mikrokontroler	ATmega328
Tegangan Operasi	5V
Tegangan <i>Input</i> (disarankan)	7-12 V

Tegangan <i>Input</i> (batas)	6-20V
I / O Pin Digital	14 (dimana 6 sebagai <i>output</i> PWM)
Pin <i>Input</i> Analog	6
Arus DC per I / O Pin	40 mA
Arus DC untuk 3.3V Pin	50 mA
Memori Flash	32 KB ( ATmega328 ) yang 0,5 KB digunakan oleh bootloader
SRAM	2 KB ( ATmega328 )
EEPROM	1 KB ( ATmega328 )
Kecepatan jam	16 MHz

Program Arduino diprogram dengan menggunakan Arduino *Software* (IDE) *Integrated Development Environment* yang umum bagi semua papan dan berjalan secara *online* dan *offline* (Arduino.cc, 2017).

### 2.7.1 Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal (otomatis). Daya eksternal (*non-USB*) daya dapat berasal baik dari AC-ke adaptor-DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan plug jack pusat-positif ukuran 2.1mm konektor POWER. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin pin *header* dari konektor POWER.

Kisaran kebutuhan daya yang disarankan untuk *board* Uno adalah 7 sampai dengan 12 volt, jika diberi daya kurang dari 7 volt kemungkinan pin 5 volt Uno dapat beroperasi tetapi tidak stabil kemudian jika diberi daya lebih dari 12V,

regulator tegangan bisa panas dan dapat merusak *board* Uno. Pin listrik adalah sebagai berikut:

- VIN : Tegangan masukan kepada *board* Arduino ketika itu menggunakan sumber daya eksternal (sebagai pengganti dari 5 volt koneksi USB atau sumber daya lainnya).
- 5V : Catu daya digunakan untuk daya mikrokontroler dan komponen lainnya.
- 3v3 : Sebuah pasokan 3,3 volt dihasilkan oleh regulator *on-board*.
- GND. *Ground* pin.

### 2.7.2 Memori

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk *bootimporer*), 2 KB dari SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan EEPROM *library*).

### 2.7.3 Input dan Output

Masing-masing dari 14 pin digital di Uno dapat digunakan sebagai *input* atau *output*, dengan menggunakan fungsi *pinMode* (), *digitalWrite* (), dan *digitalRead* (), beroperasi dengan daya 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull-up* resistor (secara *default* terputus) dari 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus:

- *Serial*: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data *serial*. Pin ini dihubungkan ke pin yang berkaitan dengan *chip Serial* ATmega8U2 USB-to-TTL.

- Interupsi Eksternal: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu interrupt pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.
- PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite* ().
- SPI: 10 (SS), 11 (Mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan SPI library.
- LED: 13. Ada built-in LED terhubung ke pin digital 13. Ketika pin bernilai nilai HIGH, LED on, ketika pin bernilai LOW, LED off.

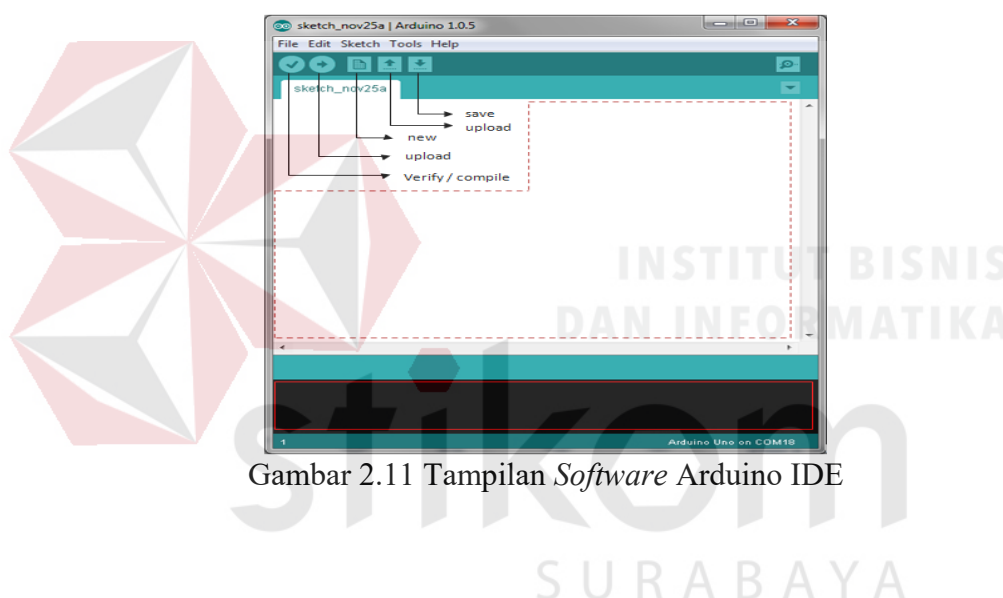
Uno memiliki 6 masukan analog, berlabel A0 sampai dengan A5, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda). Selain itu, beberapa pin memiliki fungsi khusus:

- I2C: A4 (SDA) dan A5 (SCL). Dukungan I2C (TWI) komunikasi menggunakan perpustakaan Wire.
- Aref: Tegangan referensi (0 sampai 5V saja) untuk *input* analog. Digunakan dengan fungsi *analogReference* ().
- *Reset*: Baris ini LOW untuk *reset* mikrokontroler.

#### 2.7.4 Software Arduino IDE

Arduino IDE adalah *software* yang ditulis menggunakan java dan berdasarkan pengolahan seperti, avr-gcc, dan perangkat lunak *open source* lainnya (Djuandi, 2011). *Software* Arduino IDE ditunjukkan pada Gambar 2.11. Arduino IDE terdiri dari:

1. *Editor* program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.
2. *Verify / Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *processing*, yang dipahami oleh mikrokontroler adalah kode biner.
3. *Uploader*, sebuah modul yang memuat kode biner dari *Personal Computer* ke dalam memori mikrokontroler di dalam papan Arduino.



Gambar 2.11 Tampilan *Software* Arduino IDE

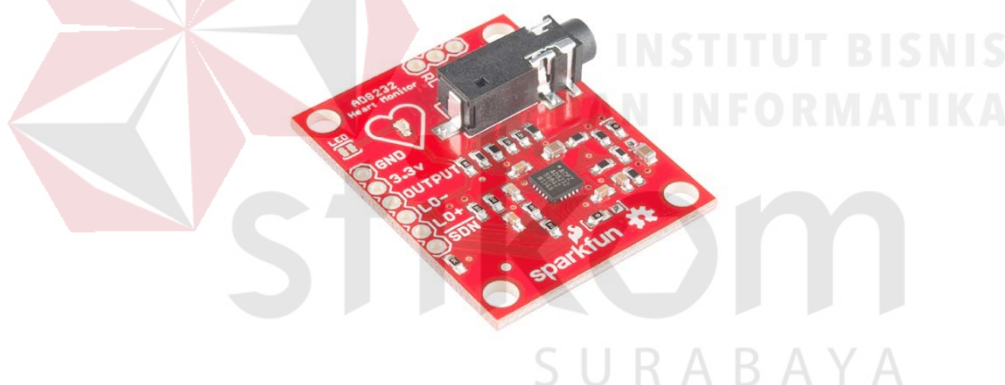
## 2.8 Sensor Monitor Denyut Jantung AD8232

Sensor AD8232 *SparkFun Single Lead Heart Rate Monitor* adalah *board* yang digunakan untuk mengukur aktivitas kelistrikan jantung. Aktivitas listrik ini dapat dipetakan sebagai EKG atau Elektrokardiogram dan *output* sebagai pembacaan analog. Sinyal EKG bisa sangat bising, Sensor AD8232 *Single Lead Heart Rate Monitor* bertindak sebagai op-amp untuk membantu mendapatkan sinyal yang jelas dari PR dan QT Interval dengan mudah.



AD8232 adalah blok pengkondisi sinyal yang terintegrasi untuk sinyal EKG dan aplikasi pengukuran biopotensial lainnya. Hal ini dirancang untuk mengekstrak, memperkuat, dan menyaring sinyal biopotensial kecil dalam kondisi bising, seperti yang dibuat oleh gerakan atau penempatan elektroda terpicil.

Sensor Monitor Denyut Jantung AD8232 memiliki sembilan koneksi dari IC, kabel, atau konektor lainnya. SDN, LO+, LO-, *OUTPUT*, 3.3V, GND pin penting untuk mengoperasikan monitor ini dengan Arduino atau mikrokontroler lainnya. Selain itu, di *board* ini terdapat pin RA (Lengan Kanan), LA (Lengan Kiri), dan RL (Kaki Kanan) menggunakan elektroda jepit. Sensor Monitor Denyut Jantung AD8232 dapat dilihat pada Gambar 2.12.



Gambar 2.12 Sensor Monitor Denyut Jantung AD8232

## 2.9 MATLAB

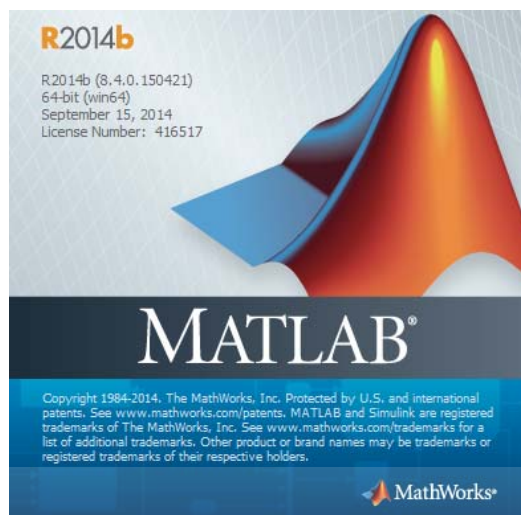
MATLAB (*Matrix Laboratory*) adalah suatu program untuk analisis dan komputasi numerik dan merupakan suatu bahasa pemrograman matematika lanjutan yang dibentuk dengan dasar pemikiran menggunakan sifat dan bentuk matriks. Pada awalnya, program ini merupakan *interface* untuk koleksi rutin-rutin numerik dari proyek LINPACK dan EISPACK, dan dikembangkan menggunakan bahasa FORTRAN namun sekarang merupakan produk komersial dari perusahaan

Mathworks Inc. yang dalam perkembangan selanjutnya dikembangkan menggunakan bahasa C++ dan *assembler* (utamanya untuk fungsi-fungsi dasar MATLAB).

MATLAB telah 8 berkembang menjadi sebuah *environment* pemrograman yang canggih yang berisi fungsi-fungsi *built-in* untuk melakukan tugas pengolahan sinyal, aljabar linier, dan kalkulasi matematis lainnya. MATLAB juga berisi *toolbox* yang berisi fungsi-fungsi tambahan untuk aplikasi khusus. MATLAB bersifat *extensible*, dalam arti bahwa seorang pengguna dapat menulis fungsi baru untuk ditambahkan pada *library* ketika fungsi-fungsi *built-in* yang tersedia tidak dapat melakukan tugas tertentu. Kemampuan pemrograman yang dibutuhkan tidak terlalu sulit bila memiliki pengalaman dalam pemrograman bahasa lain seperti C++, PASCAL, atau FORTRAN.

MATLAB merupakan merk *software* yang dikembangkan oleh Mathworks.Inc. merupakan *software* yang paling efisien untuk perhitungan *numeric* berbasis matriks. Dengan demikian jika di dalam perhitungan kita dapat menformulasikan masalah ke dalam format matriks maka MATLAB merupakan *software* terbaik untuk penyelesaian numeriknya. MATLAB yang merupakan bahasa pemrograman tingkat tinggi berbasis pada matriks sering digunakan untuk teknik komputasi numerik, untuk menyelesaikan masalah-masalah yang melibatkan operasi matematika elemen, matrik, optimasi, aproksimasi dan lain-lain. Sehingga Matlab banyak digunakan pada : (1) Matematika dan Komputansi, (2) Pengembangan dan Algoritma, (3) Pemrograman modeling, simulasi, dan

pembuatan prototype, (4) Analisa Data , eksplorasi dan visualisasi, (5) Analisis numerik (Cahyono, 2013). *Software* MATLAB ditunjukkan pada Gambar 2.13.



Gambar 2.13 MATLAB versi R2014B

Pada intinya MATLAB merupakan sekumpulan fungsi-fungsi yang dapat dipanggil dan dieksekusi. Fungsi-fungsi tersebut dibagi-bagi berdasarkan kegunaannya yang dikelompokkan didalam *toolbox-toolbox* yang ada pada MATLAB. *Toolbox* merupakan kumpulan koleksi dari fungsi-fungsi MATLAB (*M-files*) yang memperluas lingkungan MATLAB untuk memecahkan masalah-masalah tertentu. *Toolbox-toolbox* yang tersedia pada MATLAB antara lain:

1. *Signal Processing Toolbox*
2. *Control Systems Toolbox*
3. *Neural Networks Toolbox*
4. *Fuzzy Logic Toolbox*
5. *Wavelets Toolbox*
6. *Simulation Toolbox*
7. *Image Processing Toolbox*

### 2.10 *File Transfer Protocol*

*File Transfer Protocol* (FTP) adalah protokol jaringan standar yang digunakan untuk mentransfer *file Personal Computer* dari satu *host* ke *host* lain melalui jaringan berbasis TCP, seperti Internet. FTP dibangun di atas arsitektur *server* klien dan menggunakan kontrol terpisah dan koneksi data antara klien dan *server*. Pengguna APP dapat mengotentikasi dirinya dengan menggunakan protokol masuk yang jelas, biasanya dalam bentuk nama pengguna dan kata sandi, namun dapat terhubung secara anonim jika *server* dikonfigurasi untuk mengizinkannya. Untuk transmisi aman yang melindungi *username* dan *password*, dan mengenkripsi isinya, FTP sering diamankan dengan SSL / TLS (FTPS) (Ranie, Leena, Preeti Narula dan Neeti Panchal, 2014).

FTP (*FileTransfer Protocol*) umumnya berfungsi sebagai media tukar menukar *file* atau data dalam suatu *network* yang menggunakan TCP koneksi. FTP yang digunakan menggunakan berbasis *Open Source* guna menunjang tingkat stabilitas tinggi dan tidak mudah terinfeksi virus dan *malware*. FTP merupakan metode protokol pilihan yang paling tepat dalam penyimpanan *file/data* secara cepat dalam proses unggah dan *downimpor* dari *Personal Computer server* ke *client* tanpa menggunakan *flashdisk* untuk mengambil data dari *Personal Computer server* (Arman, Molavi, 2017).

### 2.11 *Simple Mail Transfer Protocol*

*Simple Mail Transfer Protocol* adalah standar protokol yang digunakan untuk pengiriman *electronic mail (email)* di internet. SMTP adalah protokol yang cukup sederhana, berbasis teks dimana protokol ini menyebutkan satu atau lebih penerima

*email* untuk kemudian diverifikasi. Jika penerima *email valid*, maka *email* akan segera dikirim. SMTP menggunakan *port 25* dan dapat dihubungi melalui program telnet. Agar dapat menggunakan SMTP *server* lewat nama *domain*, maka *record DNS (Domain Name Server)* pada bagian MX (*Mail Exchange*) digunakan.

*Sendmail* adalah *Mail Transfer Agent* pertama yang mengimplementasikan *port 25*. Kemudian, pada tahun 2001, ada sedikitnya 50 program *Mail Transfer Agent* yang mengimplementasikan SMTP baik sebagai *client* maupun sebagai *server*. Contoh *Mail Transfer Agent* yang populer adalah: Exim (ditulis oleh Philip Hazel), IBM Postfix, Qmail (ditulis oleh D.J Bernstein), dan Microsoft *Exchange Server*.

Karena protokol SMTP berawal dari protokol yang benar-benar berbasis teks ASCII, maka SMTP tidak bekerja terlalu baik dalam mengirimkan *file-file binary*. Standar untuk mengencode *file-file* biner agar dapat dikirimkan lewat SMTP dikembangkan dan memerlukan standar-standar seperti MIME (*Multipurposes Internet Mail Extensions*). Saat ini, hampir semua SMTP *server* mendukung 8BITMIME, yang dapat mengirimkan *file* biner semudah mengirimkan *file* teks.

SMTP hanya protokol yang melakukan “*push*”, artinya dia hanya bisa mengambil *email* dari *client* tetapi tidak bisa melakukan “*pull*”, yaitu melayani pengambilan *email* di *server* oleh *client*. Pengambilan pesan atau *email* tersebut dilakukan dengan menggunakan protokol tersendiri yaitu protokol POP3 (*Post Office Protokol*) atau IMAP (*Internet Message Access Protokol*).

### 2.12 *Server*

*Server* adalah sebuah tempat yang dipenuhi dengan berbagai macam informasi dimana *server* memiliki tugas utama untuk memberikan sebuah *service* atau layanan bagi para *client* yang terhubung dengannya.



## BAB III

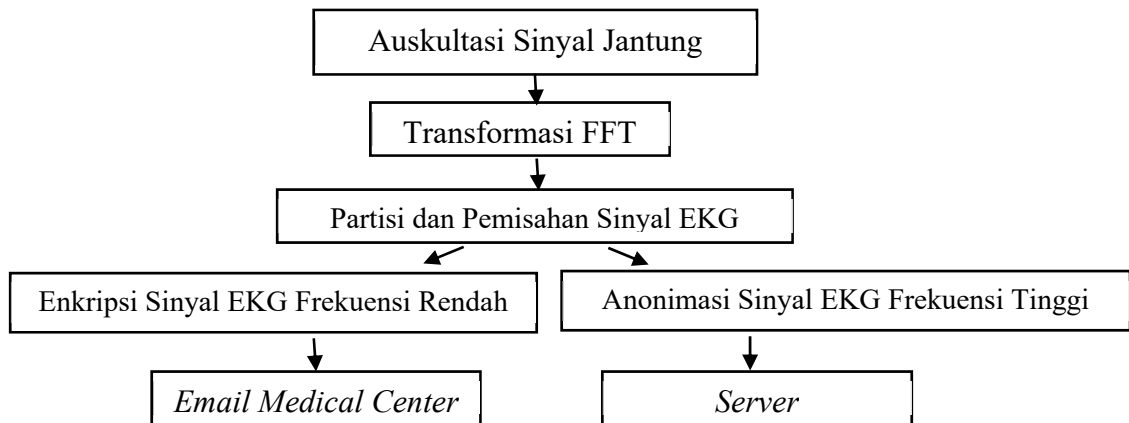
### METODE PENELITIAN DAN PERANCANGAN SISTEM

#### 3.1 Model Pengembangan

Tujuan dari penelitian yang dilakukan adalah menerapkan metode anonimasi data sinyal EKG dengan menggunakan metode anonimasi berdasarkan algoritma FFT untuk memberikan fitur keamanan dalam proses transmisi data bagi sinyal EKG, Metode ini merupakan sebuah pengembangan dari penelitian-penelitian sebelumnya yang masih belum menerapkan fitur keamanan sinyal EKG.

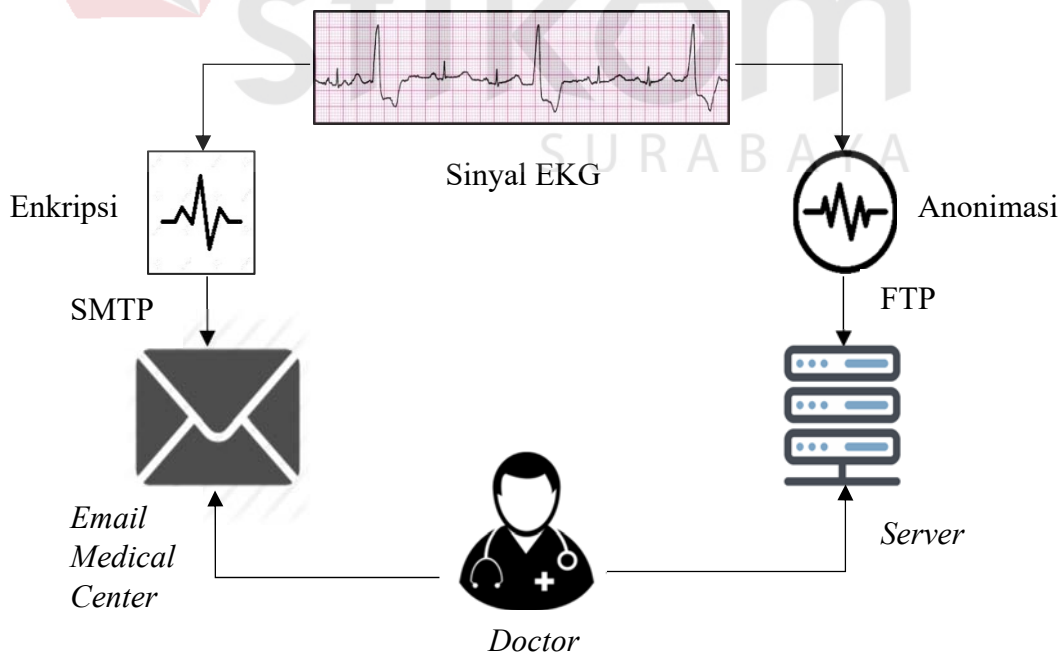
#### 3.2 Metode Penelitian

Metode Penelitian yang digunakan dalam pengerjaan penelitian meliputi studi kepustakaan dan penelitian laboratorium. Studi kepustakaan dilakukan untuk mencari referensi atau informasi dari berbagai buku, jurnal, dan artikel yang terkait dengan penelitian antara lain: Arduino, MATLAB, algoritma FFT, pemecahan rumus dalam program, pengambilan data secara *offline*. Berdasarkan referensi atau informasi studi kepustakaan yang diperoleh maka dilakukan penelitian laboratorium dengan perancangan sistem pada Gambar 3.1. Data sinyal jantung EKG diambil dari perekaman sinyal jantung EKG melalui Arduino. Untuk membangun sistem keamanan pada transmisi data terdiri dari beberapa proses utama, yaitu transformasi sinyal EKG, partisi komponen batas *domain* frekuensi dan anonimasi sinyal EKG frekuensi tinggi yang melibatkan beberapa subproses untuk dikirim ke *server*, serta enkripsi sinyal EKG frekuensi rendah untuk dikirim ke *email medical center*.



Gambar 3.1 Diagram Blok Sistem Keseluruhan

Berdasarkan Gambar 3.1, Hasil dari sinyal EKG yang sudah dianonimasi dan kunci yang sudah dienkripsi diunggah ke tempat tujuan yang berbeda. Hal ini dilakukan untuk meningkatkan tingkat keamanan data dari data sinyal EKG. Jika terdapat data EKG yang tersimpan di *server* dicuri oleh *hacker*, data EKG itu sudah berubah bentuk dan hanya bisa dikembalikan bila ada kunci yang tersimpan di *email medical center*. Pada Gambar 3.2 merupakan ilustrasi pihak dokter untuk memperoleh dan menggabungkan sinyal EKG.

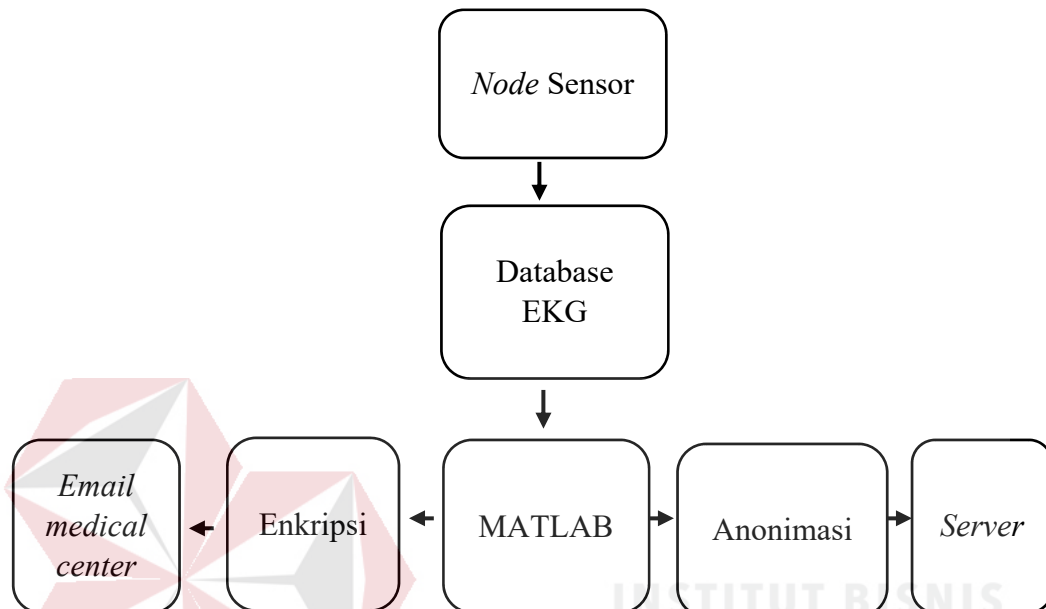


Gambar 3.2 Ilustrasi Pihak Dokter Memperoleh Sinyal EKG



### 3.3 Model Perancangan

Pada perancangan ini penulis menggambarkan perancangan sistem seperti gambar di bawah ini:



Gambar 3.3 Diagram Blok Perancangan Sistem

Berikut adalah penjelasan dari Gambar 3.3:

a. *Node Sensor*

Pada bagian ini dilakukan proses pengambilan data melalui transmisi sinyal auskultasi jantung dengan *node sensor*. *Node sensor* terdiri atas Mikrokontroler Arduino, Sensor *Heart Rate Monitor* (Modul EKG), dan elektroda jepit.

b. Database EKG

Sinyal EKG yang akan diolah sudah diperoleh dari *file* yang direkam melalui Arduino. Dalam *file* data EKG ini terdapat dua kolom data. Untuk data yang pertama adalah data *time* (ms) dan yang kedua adalah data amplitudo (V).

c. MATLAB

Aplikasi MATLAB difungsikan untuk pengolahan data EKG dari *database*. Proses pengolahan data meliputi: Transformasi FFT, Partisi Sinyal, Anonimasi sinyal EKG yang akan dikirim ke *server*, dan enkripsi sinyal yang akan dikirim ke *email medical center*.

### 3.4 Instrumen Penelitian

Instrumen penelitian yang akan digunakan dalam penelitian ini terdiri dari beberapa alat dan bahan pendukung untuk pembuatan dan pengujian program. Instrumen yang digunakan sebagai berikut.

#### 3.4.1 Alat Penelitian

Alat yang digunakan untuk menyelesaikan penelitian ini adalah *Personal Computer (PC) / Laptop* dan *Node Sensor*. *Personal Computer (PC) / Laptop* digunakan untuk membuat simulasi 2 dimensi dengan menggunakan program MATLAB. *Node Sensor* digunakan untuk transmisi aukultasi jantung.

#### 3.4.2 Bahan Penelitian

Bahan yang akan diteliti oleh penulis dalam penelitian kali ini meliputi beberapa aspek sebagai berikut :

1. Pengolahan data menggunakan *software* MATLAB.
2. *Database* sampel terdiri dari 6 sinyal EKG normal.
3. *Server* dan *email medical center*.

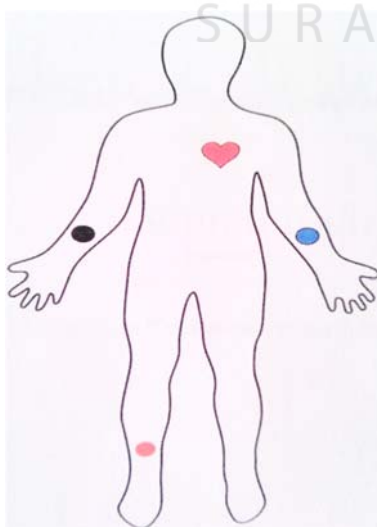
### 3.5 Pengambilan Data Auskultasi Jantung

Pada transmisi sinyal auskultasi ini, dibutuhkan algoritma pengiriman data dan analisa dari hasil pengiriman itu sendiri agar dapat diketahui seberapa baik sistem yang telah dibangun.

#### 3.5.1 Peletakan Elektroda Jepit

Dalam transmisi sinyal auskultasi, komponen terpenting adalah data yang diambil dari auskultasi jantung. Maka tahapan yang pertama dilakukan adalah mengambil data auskultasi dengan meletakkan sensor pada pergelangan kedua tangan dan pergelangan kaki kiri.

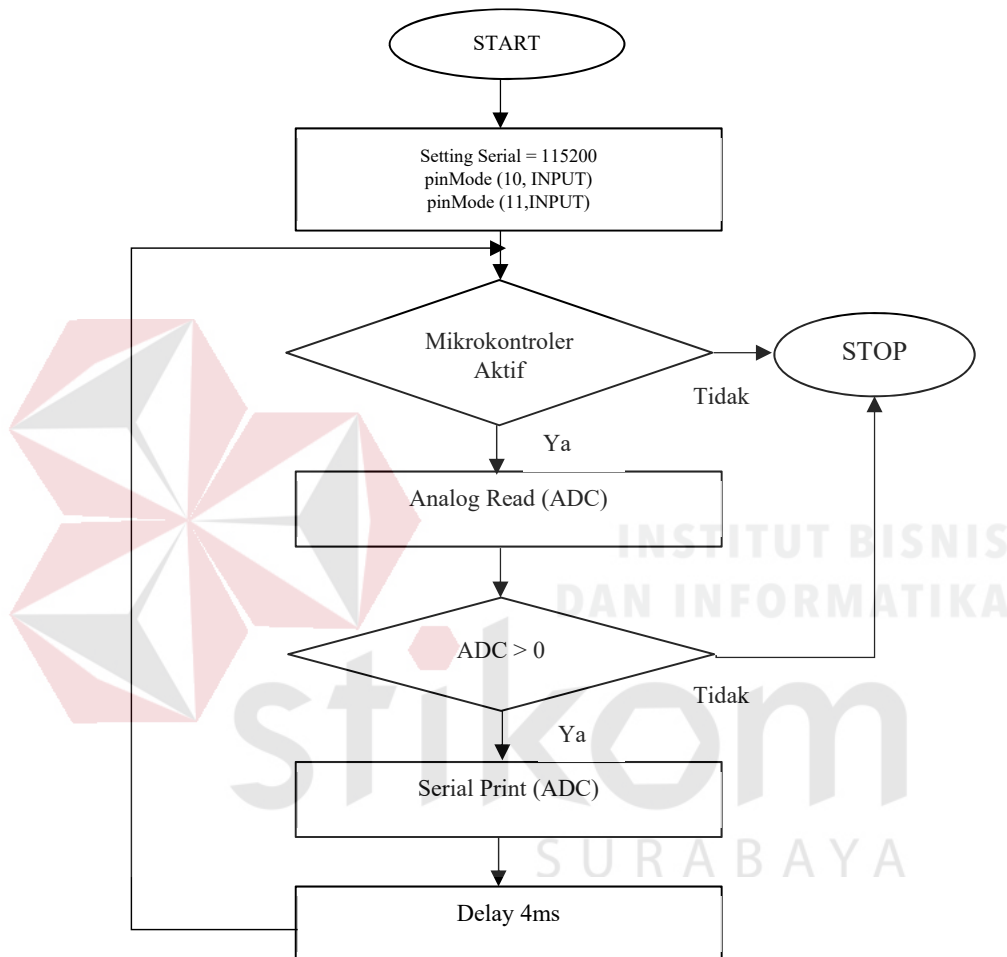
Posisi jantung manusia adalah pada tulang iga manusia ke 6 disebelah kiri dada manusia, atau 5 cm diatas uluh hati di sebelah kiri. Peletakkan sensor sangat berpengaruh, karena apabila sensor tidak diletakkan pada bagian yang tepat maka data yang akan diterima berupa *noise*. Posisi peletakkan sensor dapat dilihat pada Gambar 3.4.



Gambar 3.4 Letak Posisi Elektroda Jepit

### 3.5.2 Program Membaca Sensor EKG dan Mengirim ke PC

*Flowchart* untuk mengambil sinyal EKG terhadap tubuh pasien berdasarkan pembacaan sensor EKG dan dikirim ke PC untuk ditampilkan dan disimpan data sinyal EKG terdapat pada Gambar 3.5.



Gambar 3.5 *Flowchart* Pengiriman Data Auskultasi Jantung

Pada dasarnya konsep dari sensor jantung adalah menerima setiap suara, kemudian melakukan pengambilan tegangan yang keluar dari tubuh melalui sensor EKG dan data sensor EKG tersebut diolah pada mikrokontroler menjadi data digital melalui pin ADC.

### 3.5.3 Pengambilan Sinyal Auskultasi Jantung

Proses pengambilan data dilakukan saat semua alat terpasang dan proses pengiriman data berlangsung. Seperti yang dijelaskan pada Gambar 3.6, dapat melihat apakah sensor sudah berada di posisi yang tepat. Berikut adalah program pengambilan data EKG pada Arduino IDE seperti pada Gambar 3.6 di bawah ini.

```

unsigned long time;
unsigned long waktu;
void setup() {
  // initialize the serial communication
  Serial.begin(115200);
  pinMode (10, INPUT); // Setup for leads off detection LO +
  pinMode (11, INPUT); // Setup for leads off detection LO -
}

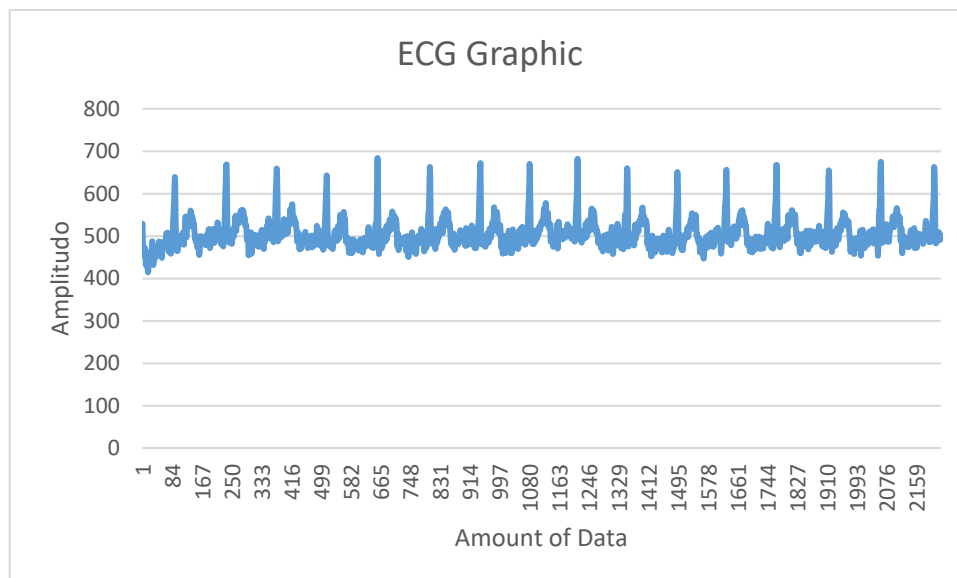
void loop() {
  time = millis();
  Serial.print(time);
  Serial.print(" ,");
  if(time <= 10000)
  {
    bacadata();
  }
  else
  {
    while(1){}
  }
}

void bacadata()
{
  Serial.print(' ');
  Serial.println(analogRead(A0));
  delay(4);
}

```

Gambar 3.6 Program Pengambilan Sinyal EKG Pada Arduino IDE

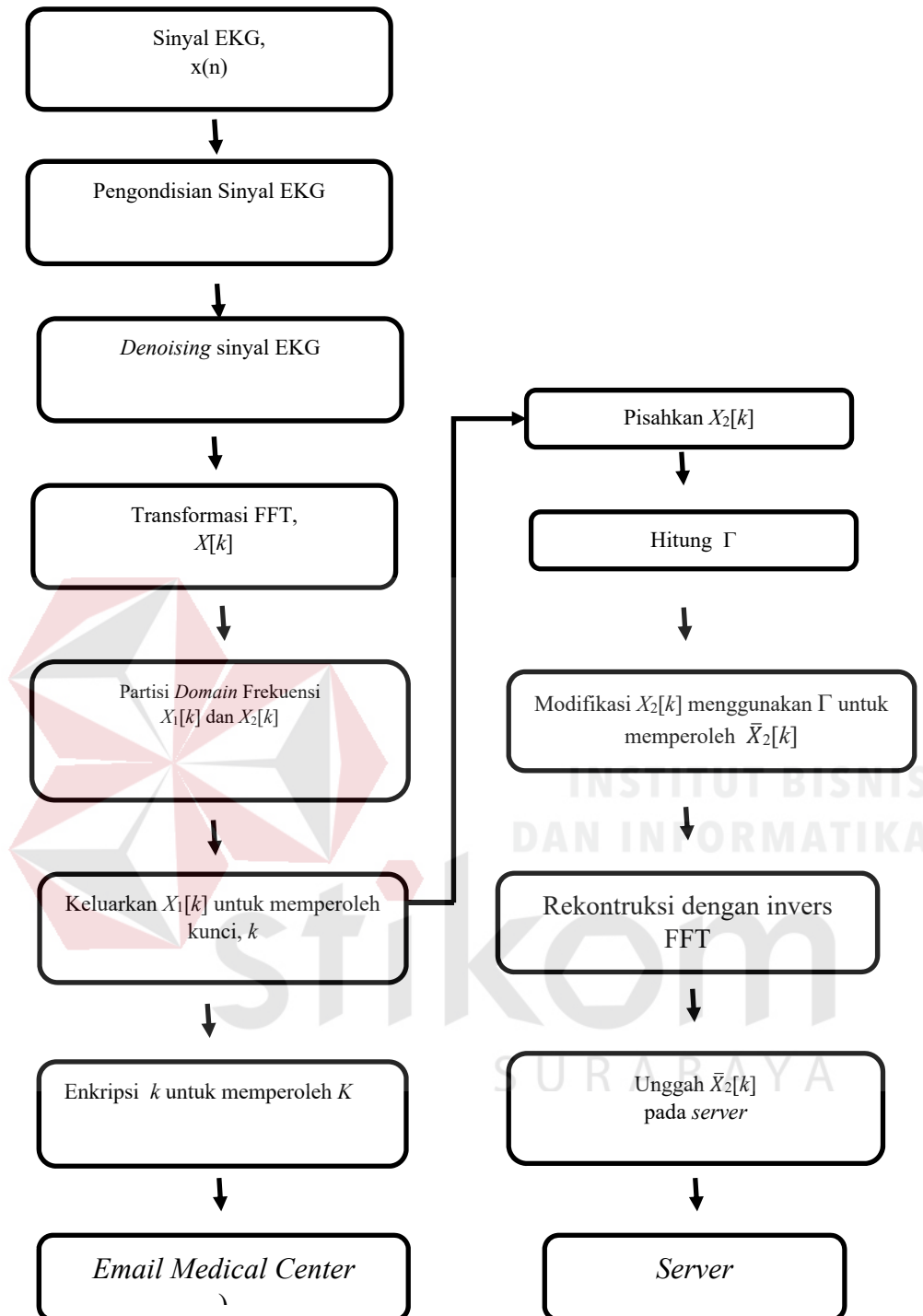
Berikut contoh pengambilan sinyal jantung dilakukan selama 10 detik untuk mendapatkan hasil sinyal EKG yang telah diperoleh dan telah divisualisasikan dalam bentuk grafik pada Gambar 3.7.



Gambar 3.7 Hasil Sinyal Auskultasi Jantung Selama 10 Detik

### 3.6 Perancangan Perangkat Lunak

Pada penelitian ini membutuhkan perancangan perangkat lunak (*software*) untuk menjalankan sistem untuk proses anonimasi dengan benar. Beberapa algoritma yang digunakan adalah mengambil data dari *database*, *denoising* sinyal, transformasi FFT, partisi *domain*, modifikasi sinyal, rekonstruksi sinyal, enkripsi *key*, pengunggahan pada *server* dan *medical center*. Pada Gambar 3.8 menggambarkan diagram blok rancangan perancangan lunak dalam proses anonimasi sinyal EKG secara keseluruhan.




Gambar 3.8 Diagram Blok Perancangan Perangkat Lunak Anonimasi Sinyal EKG

Berikut adalah penjelasan dari gambar 3.8.

### 3.6.1 Sinyal EKG

Data sinyal EKG diperoleh dari pengambilan auskultasi jantung melalui Arduino Uno. Berikut di bawah ini data yang diperoleh dalam proses auskultasi jantung dapat dilihat pada Gambar 3.9. Dalam *file* data EKG ini terdapat dua data. Untuk data pertama adalah data *time(ms)* dan pada data kedua adalah data amplitudo dalam bentuk keluaran ADC.



9805	, 261
9809	, 268
9815	, 264
9819	, 255
9823	, 256
9828	, 266
9832	, 267
9837	, 258
9841	, 254
9845	, 263
9850	, 269
9854	, 262
9860	, 254
9864	, 257
9868	, 267
9873	, 266

Gambar 3.9 Data Proses Auskultasi Jantung

Data akan diolah di Microsoft Excel agar nilai amplitudo dari sinyal EKG dalam satuan ADC diubah menjadi nilai dalam satuan *volt* dan disimpan dalam *file* txt.

### 3.6.2 Pengondisian Sinyal

Pengondisian sinyal adalah tahap dimana sinyal yang telah diambil akan disimpan dalam format *file* txt. Data sinyal EKG akan dikonversi dari nilai ADC ke satuan tegangan dan disimpan dalam *file* txt untuk diimpor ke MATLAB. Pada Gambar 3.8 menunjukkan tampilan *amplitudo* dalam satuan ADC. Maka salin data



yang sudah diperoleh ke dalam *software* Microsoft Excel untuk dirubah ke dalam satuan tegangan. Cara merubah data menjadi satuan tegangan dengan perhitungan rumus :

$$x = (\text{data} * 5 / 1023) - 2,5$$

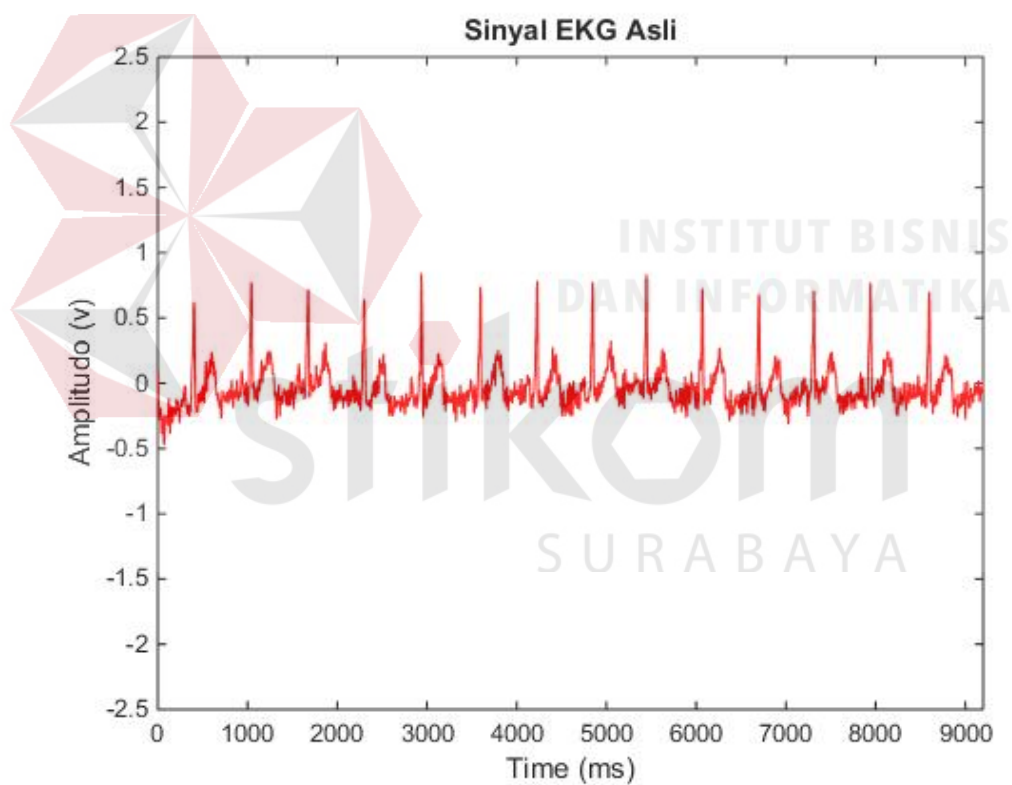
Berikut adalah penjelasan dari rumus merubah data menjadi satuan tegangan:

- a. Perkalian dengan 5 : Data yang diambil dari tegangan referensi antara 0 V sampai 5 V.
- b. Pembagian dengan 1023 : Sinyal auskultasi jantung telah dikonversi menjadi data ADC dengan resolusi 10 bit.
- c. Pengurangan -2,5 : Untuk memberikan tampilan hasil visualisasi sinyal EKG dengan nilai amplitudo antara -2,5 V sampai 2,5 V.

Setelah data dikonversi, maka *file* akan disimpan menjadi *format* txt agar selanjutnya data bisa dibaca dan diolah oleh *software* MATLAB. Pada saat pengambilan data jantung selain posisi jantung yang tepat, hasil dari sinyal auskultasi jantung juga terpengaruh oleh gerakan yang dilakukan oleh subjek uji. Hal tersebut akan membuat banyak *noise* dan sinyal jantung tidak akan terbentuk dengan baik. Pada Gambar 3.9 adalah data sinyal EKG yang sudah diolah menjadi *file* txt. Data EKG terbagi menjadi tiga kolom data. Pada kolom pertama adalah data *time* (ms), pada kolom kedua adalah tanda baca koma (,) yang berfungsi sebagai pemisah data, pada kolom ketiga adalah data amplitudo (V). Untuk impor *file* txt pada MATLAB menggunakan fungsi *load*. Contoh data sinyal EKG dalam format *file* txt seperti pada Gambar 3.10 dan Pada Gambar 3.11 adalah hasil visualisasi data sinyal EKG yang sudah dikonversi ke satuan *volt*.

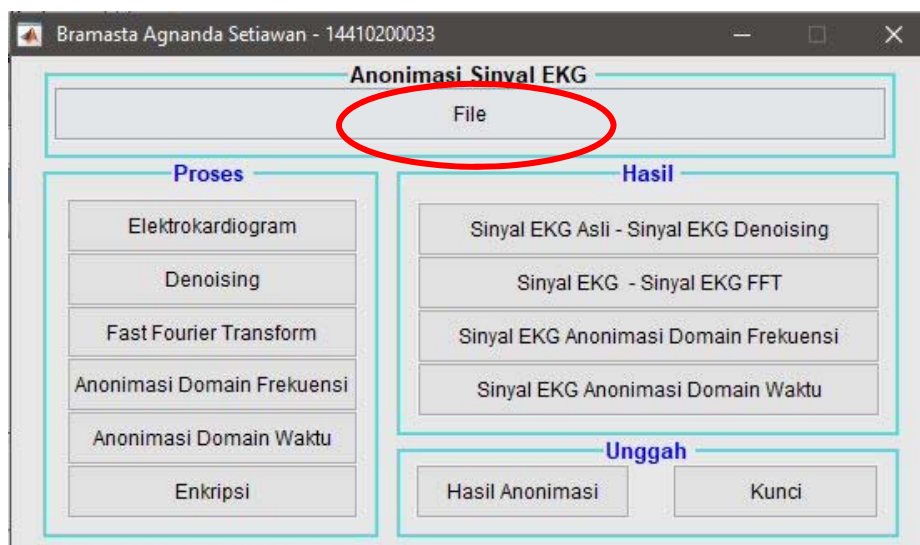
8	,	-0.119745846
12	,	-0.227272727
17	,	-0.285923754
21	,	-0.227272727
25	,	-0.227272727
30	,	-0.188172043
34	,	-0.207722385
38	,	-0.261485826
43	,	-0.38856305
48	,	-0.383675464
52	,	-0.281036168
56	,	-0.25659824
61	,	-0.349462366
65	,	-0.442326491

Gambar 3.10 File Txt Data Sinyal EKG



Gambar 3.11 Hasil Visualisasi Data EKG Setelah Dikonversi

Pada penelitian ini, proses impor dapat dilakukan dengan memilih *button* lingkaran merah pada program GUI seperti pada Gambar 3.12.



Gambar 3.12 Program GUI Impor Sinyal EKG Pada MATLAB

### 3.6.3 Denoising Sinyal

*Denoising* digunakan untuk menghapus data sinyal yang tidak diperlukan. Pada penelitian ini dilakukan untuk mencari *threshold* dengan karakteristik sinyal *input* dengan penerapan *thresholding rules*. Terdapat 2 metode *thresholding* yaitu *global thresholding* dan *level dependent thresholding*. Pencarian nilai *threshold* dengan metode *global thresholding* menggunakan karakteristik dari panjang data sinyal *input*. Sedangkan dengan metode *level dependent thresholding* yaitu mencari nilai *threshold* berdasarkan level resolusi/level dekomposisi. Pada penelitian ini *threshold* yang digunakan adalah metode *level dependent thresholding*. Berikut ini cuplikan program proses *denoising* dapat dilihat pada Gambar 3.13.

```
den = wden(amp, 'sqtwolog', 's', 'sln', 4, 'db5');
```

Gambar 3.13 Proses *Denoising* Sinyal EKG

`wden` adalah fungsi standar pada MATLAB untuk proses *denoising* dengan menggunakan metode *level dependent thresholding*. Pada penelitian ini menggunakan parameter sebagai berikut:

- a. `amp` adalah data sinyal EKG asli.
- b. `sqtwlog` digunakan untuk *threshold* universal.
- c. `s` adalah *soft thresholding*.
- d. `sln` digunakan untuk *rescalling* menggunakan estimasi tingkat *noise* berdasarkan tingkat awal.
- e. `level` untuk tingkatan level yang digunakan. Pada penelitian ini tingkatan level yang digunakan adalah level 4.
- f. `wname` adalah *mother* wavelet yang digunakan. Pada penelitian ini menggunakan dB5 karena adanya kemiripan bentuk dari sinyal EKG dengan mother wavelet dB5.

Model yang mendasari untuk *noise signal* pada dasarnya adalah dari bentuk berikut:

$$s(n) = f(n) + \sigma e(n)$$

Di mana waktu  $n$  memiliki jarak yang sama. Dalam model yang paling sederhana, anggaplah bahwa  $e(n)$  adalah *Gaussian white noise*  $N(0,1)$  dan tingkat *noise*  $\sigma$  seharusnya sama dengan 1. Tujuan *denoising* adalah untuk menekan kebisingan bagian dari sinyal  $s$  dan untuk memulihkan  $f$ . Proses *denoising* berlangsung dalam tiga langkah:

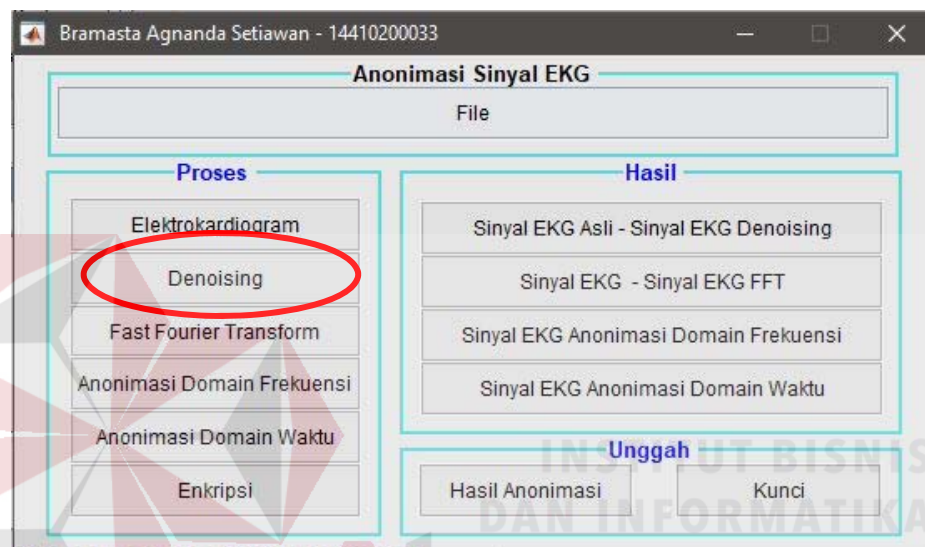
1. Penguraian. Pilih sebuah wavelet, dan pilih sebuah level  $N$ . Hitung dekomposisi wavelet dari sinyal  $s$  pada level  $N$ .
2. Detail koefisien *thresholding*. Untuk setiap level dari 1 hingga  $N$ , pilih ambang dan terapkan *thresholding* halus ke koefisien detail.

3. Rekonstruksi. Hitung rekonstruksi wavelet berdasarkan pada koefisien pendekatan asli tingkat  $N$  dan koefisien detail yang dimodifikasi dari level 1 hingga  $N$ .

Dalam prakteknya, model dasar tidak dapat digunakan secara langsung. Bagian ini memeriksa opsi yang tersedia, untuk menangani penyimpangan model. Parameter scal harus ditentukan terlebih dahulu. Ini sesuai dengan metode *rescaling*.

- Pilihan scal = 'one' sesuai dengan model dasar.
- Secara umum, tingkat *noise* yang sudah diperoleh dapat diabaikan. Koefisien detail  $CD_1$  (skala terbaik) pada dasarnya adalah koefisien *noise* dengan standar deviasi sama dengan  $\sigma$ . Rata-rata deviasi absolut dari koefisien adalah diperoleh dari  $\sigma$ . Estimasi yang digunakan sangat penting karena jika koefisien level 1 berisi rincian  $f$ .
- Pilihan scal = 'sln' berfungsi untuk *rescaling threshold* menggunakan estimasi *level noise* berdasarkan pada koefisien *level* pertama.
- Nilai *threshold* harus di *rescale* bergantung dari estimasi *level noise*. strategi yang sama digunakan dengan memperkirakan tingkat  $\sigma_{lev}$  berdasarkan level. Estimasi ini diimplementasikan dalam *file* *wnoisest*, yang mengatur struktur dekomposisi wavelet dari sinyal asli  $s$  secara langsung.
- Pilihan scal = 'mln' untuk mengatur *rescaling threshold* menggunakan estimasi *level-dependent noise level*.

Pada proses *denoising* ini *input* yang dibutuhkan adalah sinyal EKG. *Denoising* ini dilakukan untuk menghapus data sinyal yang tidak diperlukan saat terekam pada proses perekaman sinyal jantung. Pada penelitian ini, proses *denoising* dapat dilakukan dengan memilih *button* lingkaran merah pada program GUI seperti pada Gambar 3.14.



Gambar 3.14 Program GUI *Denoising* Sinyal EKG

### 3.6.4 Transformasi FFT dan Partisi Sinyal EKG

Proses transformasi FFT diikuti dengan proses partisi *domain* frekuensi. Proses partisi *domain* frekuensi ini adalah fase yang paling penting untuk melakukan anonimasi sinyal EKG, dimana pada fase ini sinyal EKG terbagi menjadi dua bagian  $X1[k]$  and  $X2[k]$ . Bagian pertama  $X1[k]$  merupakan hasil transformasi sinyal dengan frekuensi rendah, sedangkan bagian kedua  $X2[k]$  merupakan hasil transformasi sinyal dengan frekuensi tinggi. Sehingga dapat didefinisikan sebagai berikut.

$$X[k] \equiv \left\{ \underbrace{X_1[0 \dots P]}_{\text{low freq. component}}, \underbrace{X_2[(P+1) \dots Q]}_{\text{high freq. component}} \right\}$$

Dimana  $P$  sebagai nilai *secret key* dan  $Q = N - 1$ . Nilai  $Q$  dapat diasumsikan sebagai:

1. Nilai  $Q$  untuk  $X[k]$  harus dipilih dengan hati-hati untuk memastikan bahwa sampel sinyal EKG berisi frekuensi tinggi lebih dari 250 Hz. Asumsi ini akan menjamin bahwa semua komponen penting dari sinyal EKG yang meliputi QRS kompleks, gelombang P, dan gelombang T yang telah diekstrak masih tetap utuh.
2. Hubungan dari parameter  $P$  dan  $Q$  adalah  $0 \equiv Q \pmod{P}$ . Dimana  $\text{mod}()$  adalah operasi modulus yang akan selalu bernilai positif.

### 3.6.5 Pemisahan $X_1[k]$ dari $X[k]$

Pada fase ini, dilakukan pemisahan komponen  $X_1[k]$  dari  $X[k]$  untuk memperoleh *key* yang belum terenkripsi  $\kappa$ . *Key* ini berisi frekuensi yang paling rendah dari sampel sinyal EKG. *Key* dapat didefinisikan sebagai  $\kappa = \{X_1[k]: k = 0, \dots, P\}$ , dimana  $P$  adalah panjang kunci yang diinginkan. Dengan proses pemisahan  $X_1[k]$  dari  $X[k]$ , komponen  $X_2[k]$  memegang informasi penting dari sinyal EKG.

### 3.6.6 Modifikasi $X_2[k]$

Modifikasi komponen  $X_2[k]$  menggunakan fungsi reversibel tertentu. Dalam penelitian ini, penulis membagi komponen  $X_2[k]$  dengan  $\Gamma$  agar perhitungan dari algoritma tidak terlalu kompleks. Komponen  $X_2[k]$  yang termodifikasi dapat ditulis sebagai berikut:

$$\overline{X}_2[k] = \left\{ \frac{x_2[k]}{\Gamma}; k = P + 1, \dots, Q \right\},$$

Dimana  $\Gamma$  didefinisikan sebagai  $\Gamma = \kappa + \text{offset}$ , dengan  $\text{offset} = |\min(\kappa)| + \eta$ . Parameter  $\eta$  adalah nilai konstan untuk mencegah pembagian dengan nol, sementara  $|\cdot|$  adalah operasi bilangan absolut. Setelah dimodifikasi, sinyal hasil anonimasi direkonstruksi menggunakan algoritma invers FFT.

### 3.6.7 Pembuatan kunci keamanan $K$

Proses pemberian keamanan dibuat melalui *software* MATLAB. Pembuatan kunci keamanan dapat diselesaikan melalui proses dan enkripsi pada komponen  $\kappa$  dengan  $\Gamma$  sesuai dengan persamaan  $K = E(\kappa, \Gamma)$ , dimana operator  $E$  merupakan operasi enkripsi atau pembuatan sandi. Dalam proses enkripsi digunakan rumus Enkripsi =  $key * 7 + 11$ .

### 3.6.8 Pengunggahan pada *server* dan *email medical center*

Mengunggah sinyal EKG yang telah dianonimasi untuk diamankan pada *server* sebagai repositori data kesehatan dan Sinyal EKG yang telah terenkripsi dikirim ke *email medical center*. Protokol yang digunakan dalam proses pengunggahan data pada *server* adalah FTP, sedangkan Protokol yang digunakan dalam proses pengunggahan data pada *email medical center* adalah SMTP.



## BAB IV

### HASIL PENGUJIAN DAN PEMBAHASAN

Dalam bab ini penulis akan menguraikan dan menjelaskan hasil dari penelitian tugas akhir. Pengujian yang dilakukan meliputi pengujian perangkat lunak (*software*) dan kinerja keseluruhan sistem yang telah dibuat.

#### 4.1 Pengujian Pengambilan Data

Pengujian pengambilan data dilakukan dengan menggunakan Arduino yang telah diprogram dan dieksekusi melalui Arduino IDE. Pengujian dilakukan kepada 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, dan 4 ms.

##### 4.1.1 Tujuan Pengujian Pengambilan Data

Pengujian ini dilakukan untuk memperoleh data sinyal EKG. Selanjutnya data tersebut diolah menjadi *file* txt. Sehingga data dalam bentuk *file* txt dapat dibaca oleh *software* MATLAB untuk dianonimasi.

##### 4.1.2 Kebutuhan Pengujian Pengambilan Data

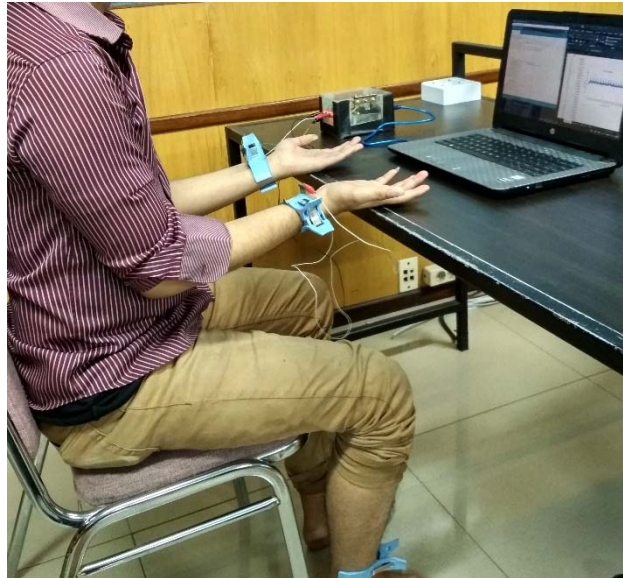
Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. Arduino UNO
- b. Kabel USB
- c. Sensor *Heart Rate Monitor* dan elektroda jepit
- d. *Personal Computer/Laptop*

- e. *Software* Arduino IDE
- f. *Software* Microsoft Excel

#### 4.1.3 Prosedur Pengujian Pengambilan Data

- a. Penentuan lokasi untuk pengambilan data harus dengan suasana lokasi yang kondusif, karena bisa mengakibatkan *noise* pada saat pengambilan data.
- b. Hubungkan Arduino dengan sensor *heart rate* dan elektroda jepit.
- c. Hubungkan Arduino dan *Personal Computer* dengan menggunakan kabel USB.
- d. Letakkan masing-masing sensor pada pergelangan tangan kanan, tangan kiri, dan kaki kanan subjek uji agar mendapatkan sinyal jantung yang tepat. Posisi peletakkan sensor dapat dilihat pada Gambar 4.1.
- e. Buka *software* Arduino IDE. Isi perintah untuk melakukan pengambilan data dalam waktu 10 detik, 11 detik, dan 14 detik pada subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, dan 4 ms dalam program bahasa C.
- f. Apabila telah selesai untuk mengisi perintah, pilih “*verfiy*” untuk cek apabila terdapat perintah yang salah dalam bahasa C dan pilih “unggah” untuk memasukkan perintah tersebut ke dalam Arduino Uno.
- g. Setelah proses unggah sudah berhasil, pilih *icon serial monitor*. Maka *window serial monitor* akan menampilkan hasil pengambilan data yang diperoleh.
- h. Salin data yang ada di *serial monitor* ke dalam *software* Microsoft Excel untuk diolah dan disimpan menjadi *file txt*.



Gambar 4.1 Peletakan Elektroda Jepit Pada Subjek Uji

#### 4.1.4 Hasil Pengujian Pengambilan Data

Hasil pengujian unggah program ke Arduino dapat dilihat pada Gambar 4.2 yang menunjukkan Arduino berhasil diisi dengan program yang telah ditulis dalam *software* Arduino IDE.

```

ambil_data | Arduino 1.8.3
File Edit Sketch Tools Help
Upload
ambil_data
void setup() {
  // initialize the serial communication
  Serial.begin(115200);
  pinMode (10, INPUT); // Setup for leads off detection LO +
  pinMode (11, INPUT); // Setup for leads off detection LO -
}
void baca data()
{
  Serial.print(' ');
  Serial.println(analogRead(A0));
  delay(4);
}
void loop()
{
  time = millis();
  Serial.print(time);
  Serial.print(",");
  if(time <= 10000)
  {
    baca data();
  }
  else
  {
    while(1){}
  }
}
Done compiling.
Sketch uses 2184 bytes (6%) of program storage space. Maximum is 32768.
Global variables use 194 bytes (9%) of dynamic memory, leaving 1856.
27 Arduino/Genuino Uno on COM6

```

Gambar 4.2 Unggah Program Berhasil

Program yang di-unggah ke Arduino merupakan program untuk mengirimkan data menggunakan *serial*. Proses pengiriman terjadi apabila Arduino masih dihubungkan dengan USB *Personal Computer*, maka penerimaan data yang dikirim dapat dilihat di *menu serial monitor* pada *software* Arduino IDE seperti yang dijelaskan pada sub Bab 3.6.2. Data sinyal EKG yang diperoleh dalam pengujian ini ditunjukkan pada Tabel 4.1.

Tabel 4.1 Data Sinyal EKG

ID	Data Uji	Waktu <i>Sampling</i>
bs10089601	1	4 ms
ss25029501	2	4 ms
bs10089602	3	2 ms
ss25029502	4	2 ms
bs10089603	5	1 ms
ss25029503	6	1 ms

## 4.2 Pengujian *Denoising*

Pengujian *denoising* dilakukan menggunakan *software* MATLAB. Pengujian ini dilakukan pada sinyal EKG yang telah berhasil di impor di MATLAB.

### 4.2.1 Tujuan Pengujian *Denoising*

Pengujian ini dilakukan untuk mengurangi *noise* pada sinyal jantung EKG.

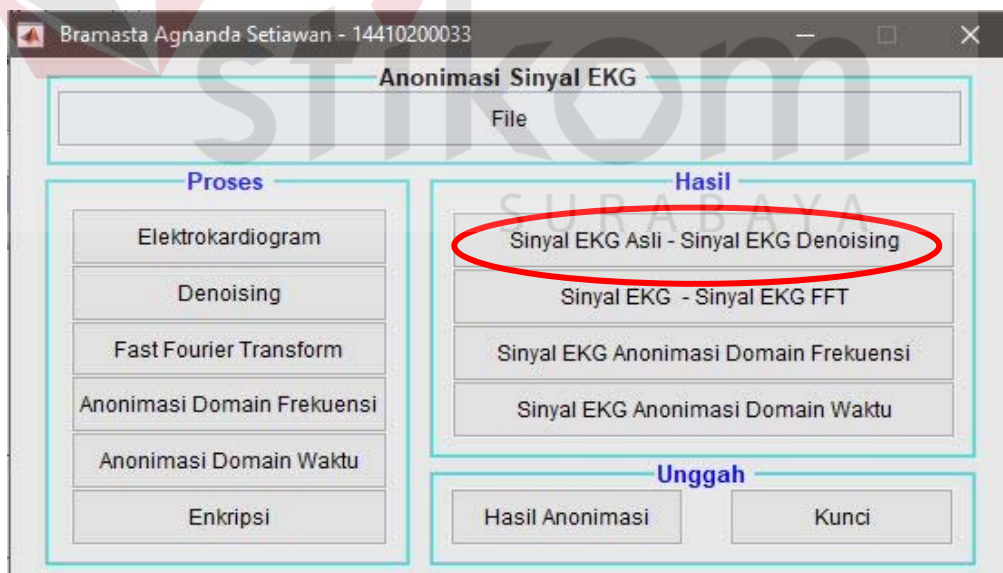
### 4.2.2 Kebutuhan Pengujian *Denoising*

Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software MATLAB*
- c. Data sinyal EKG berekstensi txt

#### 4.2.3 Prosedur Pengujian *Denoising*

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.
- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 3.6.3.
- d. Amati perubahan dari sinyal EKG sebelum dan setelah proses *denoising*. Hasil pengujian *denoising* dapat dilihat dengan memilih *button* lingkaran merah di program GUI seperti pada Gambar 4.3.



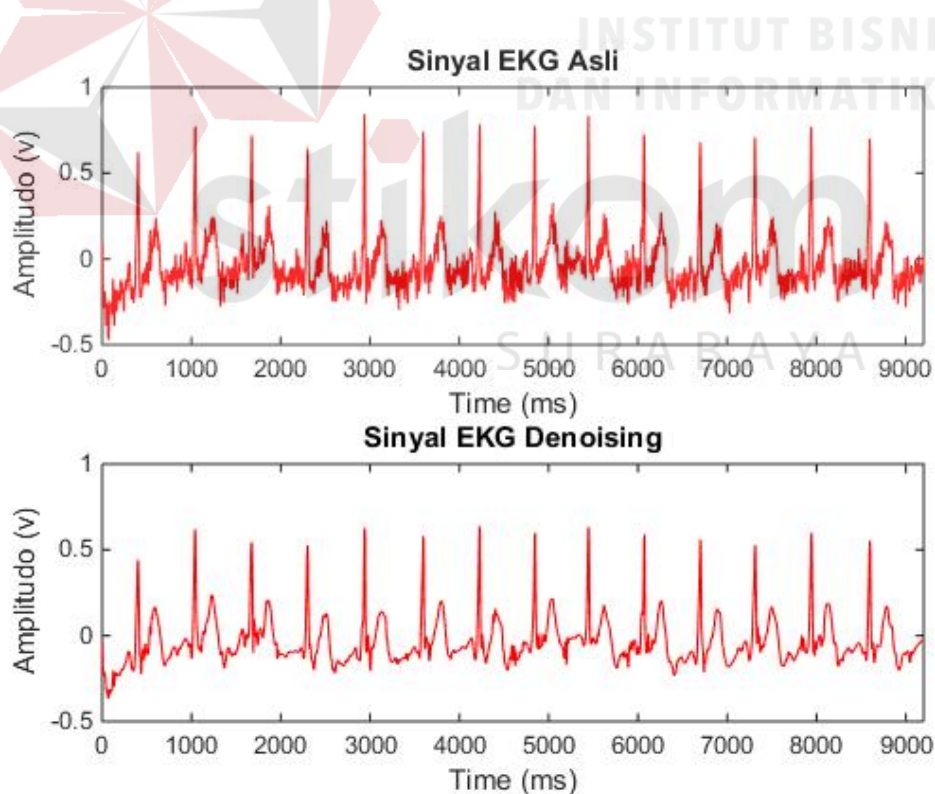
Gambar 4 3 Program GUI Pengujian *Denoising*

#### 4.2.4 Hasil Pengujian *Denoising*

Hasil pengujian dari prosedur yang dilakukan, terdapat 6 sampel data uji. Data sinyal EKG dari *file* txt yang berhasil terimpor dalam *software* MATLAB, selanjutnya dilakukan proses *denoising* untuk mengurangi *noise* yang ada pada sinyal EKG dengan menghapus sinyal -sinyal yang menyerupai *noise*. Berikut di bawah ini adalah hasil akhir pengujian proses *denoising* pada 6 sampel data uji:

##### a. Pengujian Data Uji 1

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 1 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data uji 1 yang dapat dilihat pada Gambar 4.5.

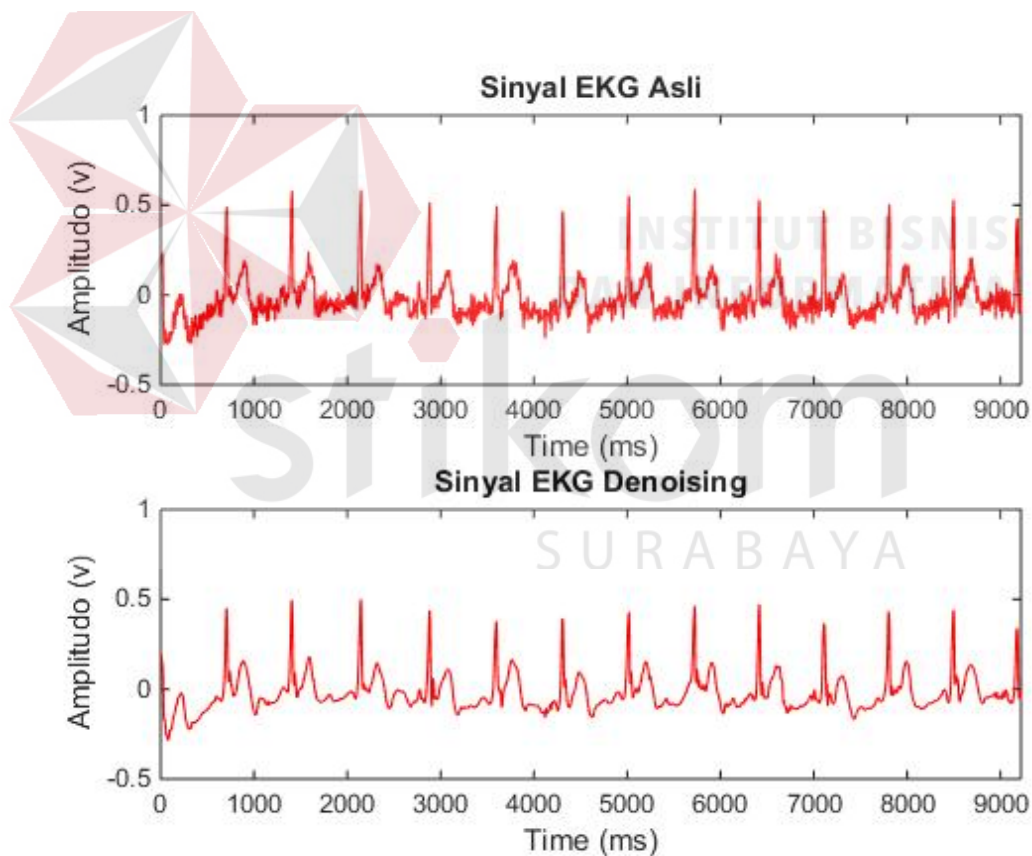


Gambar 4.4 Tampilan Hasil *Denoising* Data Uji 1

Berdasarkan Gambar 4.4, sinyal - sinyal menyerupai *noise* sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

#### b. Pengujian Data Uji 2

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 2 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data uji 2 yang dapat dilihat pada Gambar 4.5.

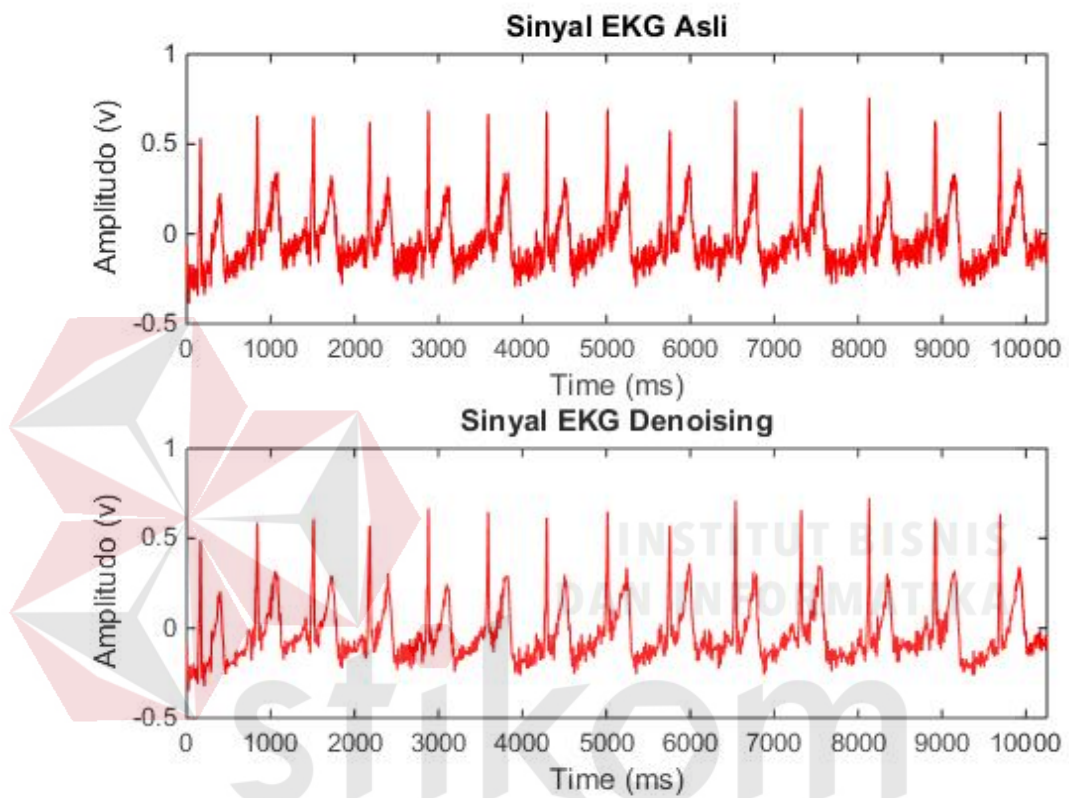


Gambar 4.5 Tampilan Hasil Denoising Data Uji 2

Berdasarkan Gambar 4.5, sinyal - sinyal menyerupai noise sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

### c. Pengujian Data Uji 3

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 3 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data data uji 3 yang dapat dilihat pada Gambar 4.6.



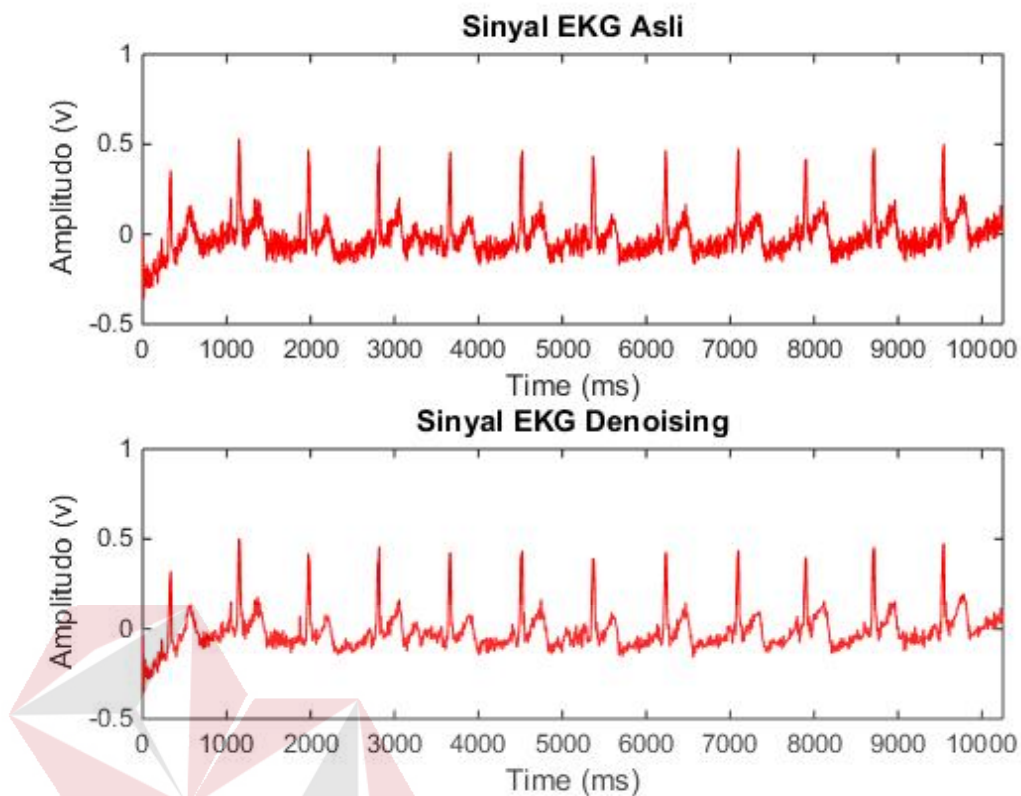
Gambar 4.6 Tampilan Hasil *Denoising* Data Uji 3

Berdasarkan Gambar 4.6, sinyal - sinyal menyerupai *noise* sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

### d. Pengujian Data Uji 4

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 4 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data uji 4 yang dapat dilihat pada Gambar 4.7.



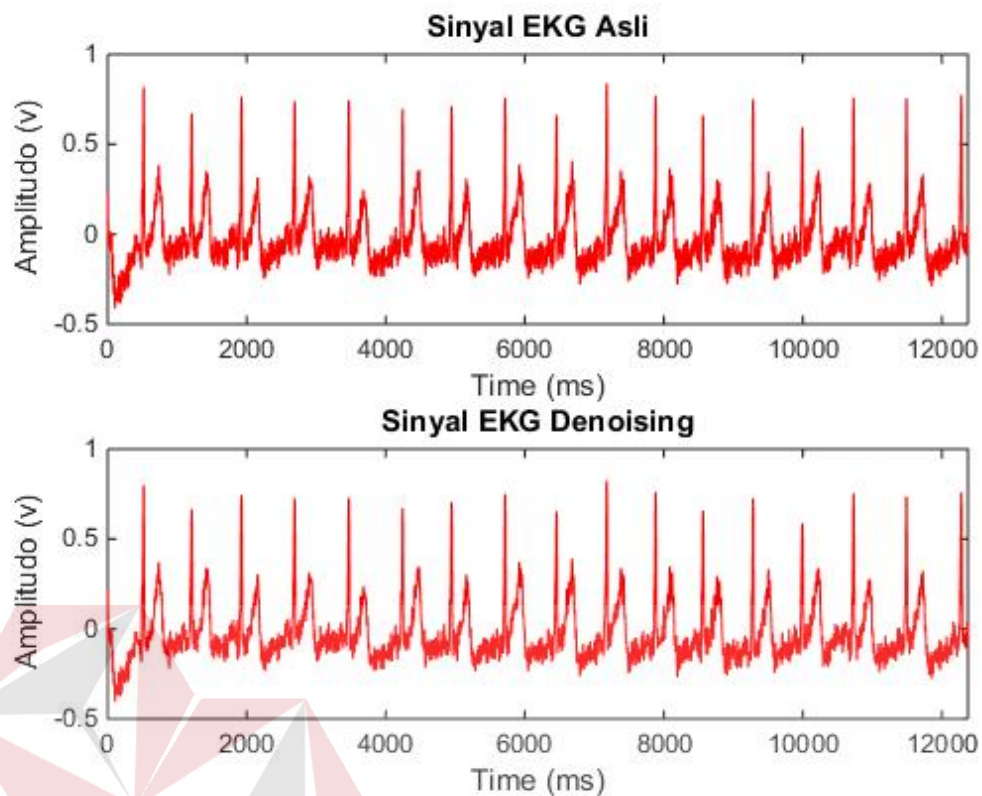


Gambar 4.7 Tampilan Hasil *Denoising* Data Uji 4

Berdasarkan Gambar 4.7, sinyal - sinyal menyerupai *noise* sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

#### e. Pengujian Data Uji 5

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 5 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data uji 5 yang dapat dilihat pada Gambar 4.8.

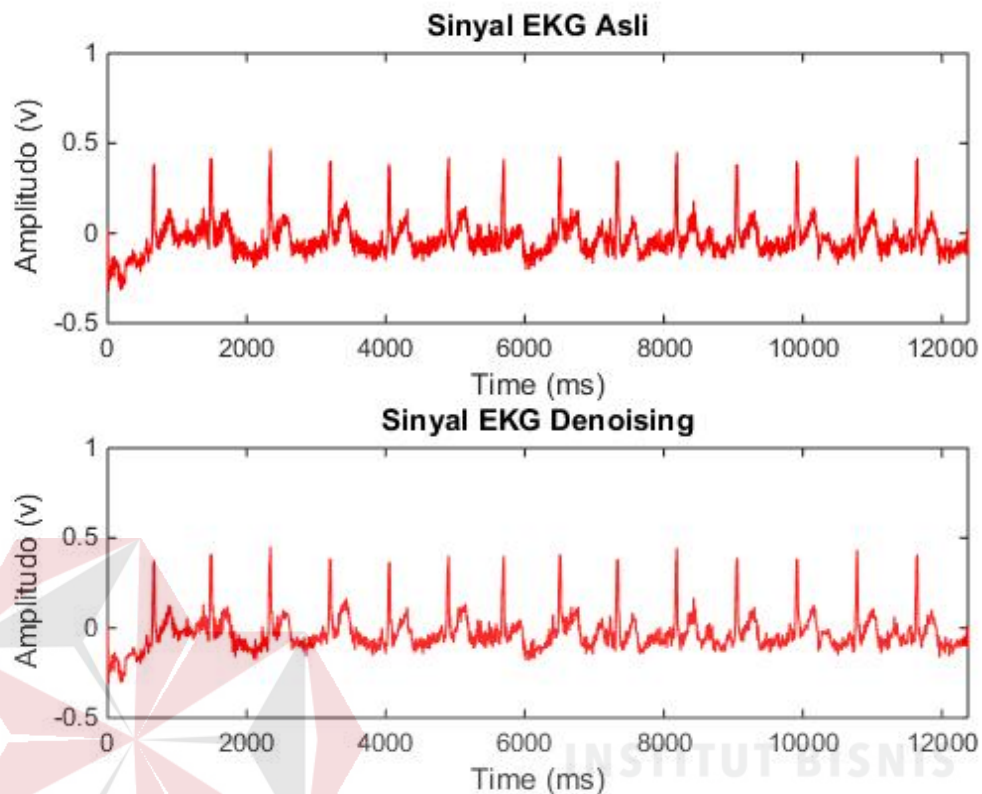


Gambar 4.8 Tampilan Hasil *Denoising* Data Uji 5

Berdasarkan Gambar 4.8, sinyal - sinyal menyerupai *noise* sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

#### f. Pengujian Data Uji 6

Hasil dari pengujian proses *denoising* sinyal jantung EKG pada data uji 6 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil *denoising* data uji 6 yang dapat dilihat pada Gambar 4.9.



Gambar 4.9 Tampilan Hasil *Denoising* Data Uji 6

Berdasarkan Gambar 4.9, sinyal - sinyal menyerupai *noise* sudah hilang pada sinyal EKG yang sudah mengalami proses *denoising*. Hal ini menunjukkan proses *denoising* sinyal jantung EKG telah berhasil.

### 4.3 Pengujian FFT

Pengujian *Fast Fourier Transform* dilakukan menggunakan *software* MATLAB untuk mantransformasi data sinyal jantung EKG dari *domain* waktu ke *domain* frekuensi. Seperti yang dijelaskan dalam sub Bab 3.7, akan dilakukan proses *denoising* terhadap sinyal EKG sebelum sinyal tersebut ditransformasikan ke *domain* frekuensi menggunakan metode FFT.

#### 4.3.1 Tujuan Pengujian FFT

Pengujian ini dilakukan untuk mengubah sinyal EKG dari *domain* waktu ke *domain* frekuensi. Pengubahan ini bertujuan agar sinyal lebih mudah untuk dimodifikasi (anonimasi).

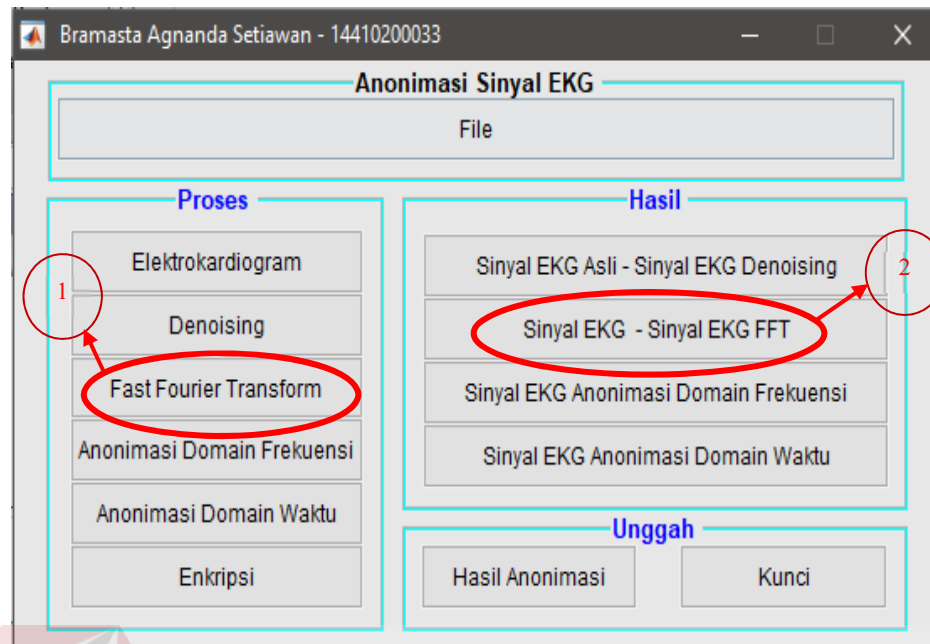
#### 4.3.2 Kebutuhan Pengujian FFT

Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software* MATLAB
- c. Data sinyal EKG berekstensi txt

#### 4.3.3 Prosedur Pengujian FFT

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.
- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 3.6.3.
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms. Pengujian FFT dapat dilakukan dengan memilih *button* dengan lingkaran merah nomor 1 di program GUI seperti pada Gambar 4.10.
- e. Amati data yang sudah berubah bentuk dari bentuk sinyal *domain* waktu menjadi *domain* frekuensi. Hasil Pengujian FFT dapat dilihat dengan memilih *button* lingkaran merah nomor 2 di program GUI seperti pada Gambar 4.10.



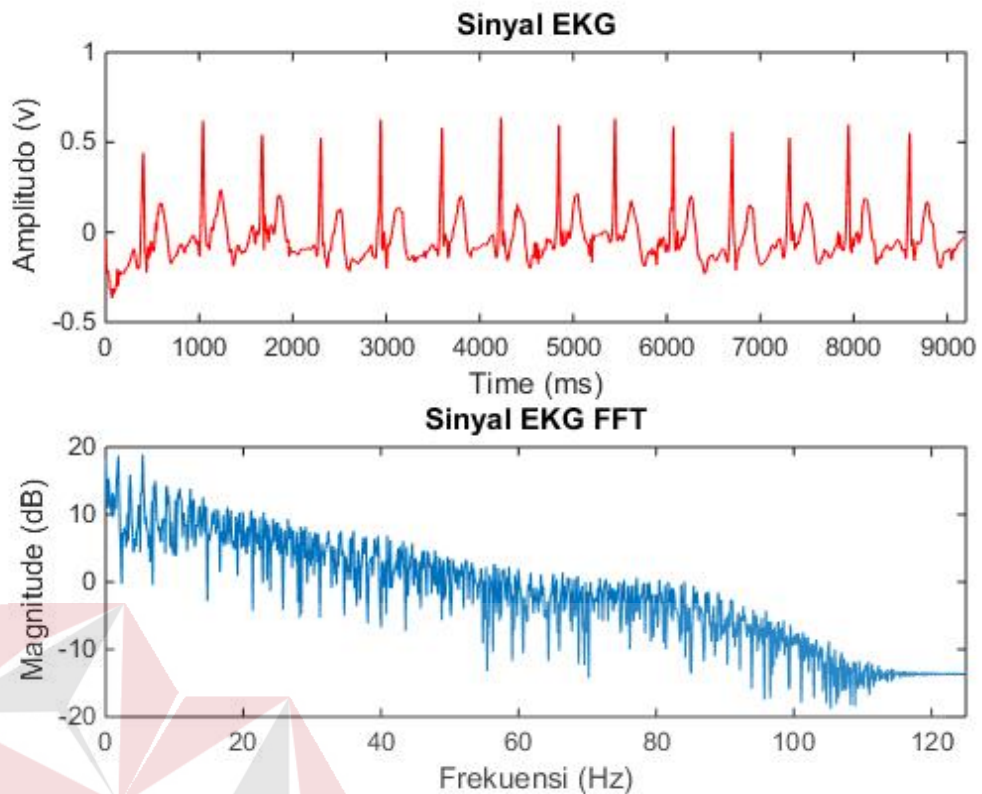
Gambar 4.10 Program GUI Pengujian FFT

#### 4.3.4 Hasil Pengujian FFT

Hasil pengujian dari prosedur yang dilakukan, terdapat 6 sampel data uji. Data dari *file txt* yang di impor dalam *software* MATLAB akan diubah dari *domain* waktu ke *domain* frekuensi. Berikut di bawah ini adalah hasil akhir pengujian proses anonimasi pada 6 sampel data uji:

##### a. Pengujian Data Uji 1

Pengujian dilakukan dengan waktu pengambilan data selama 10 detik, dengan waktu *sampling* selama 4 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 250 Hz. Hasil dari pengujian proses FFT pada data uji 1 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 1 yang dapat dilihat pada Gambar 4.11.

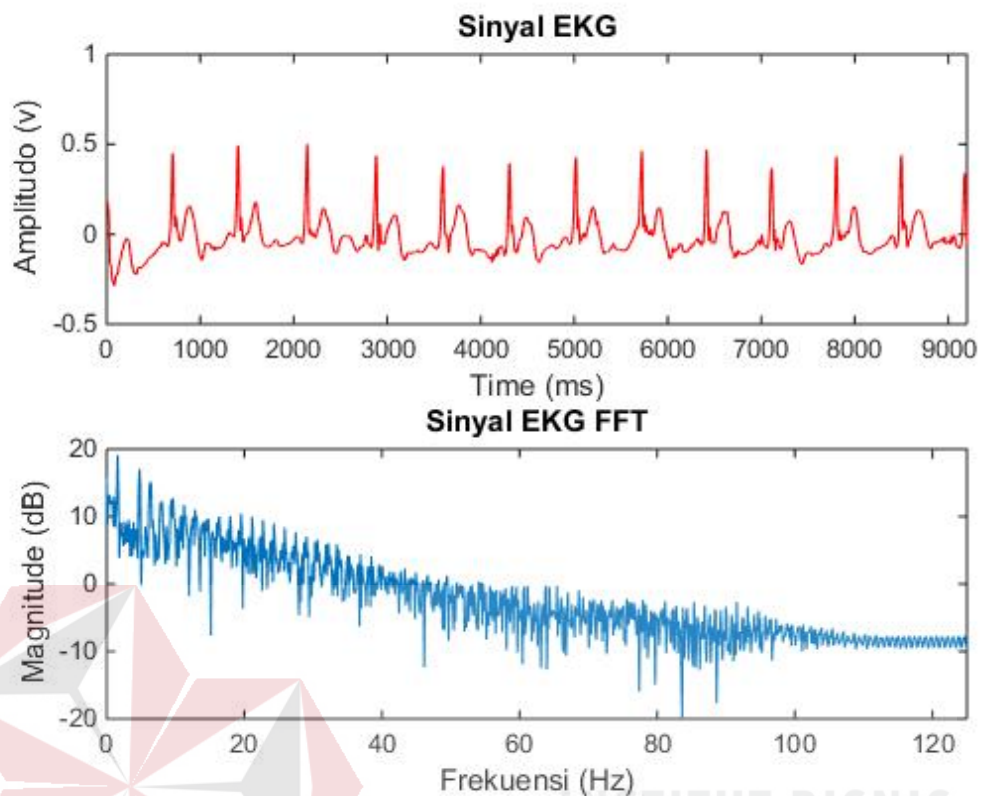


Gambar 4.11 Tampilan Hasil FFT Data Uji 1

Gambar 4.11 adalah tampilan hasil proses FFT pada data uji 1 yang telah divisualisasikan menjadi grafik. Data uji 1 berisi 2226 jumlah data. Dalam pengujian, data yang diambil sebanyak 2048 agar proses perhitungan FFT menjadi efisien.

#### b. Pengujian Data Uji 2

Pengujian dilakukan dengan waktu pengambilan data selama 10 detik, dengan waktu *sampling* selama 4 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 250 Hz. Hasil dari pengujian proses FFT pada data uji 2 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 2 yang dapat dilihat pada Gambar 4.12.

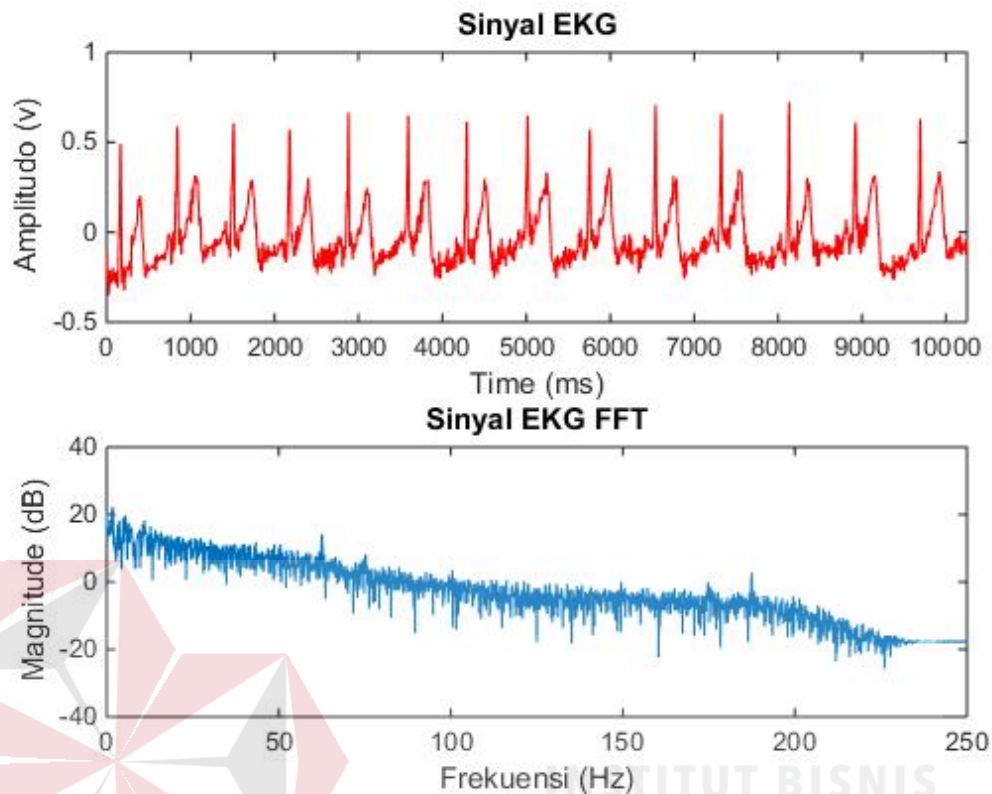


Gambar 4.12 Tampilan Hasil FFT Data Uji 2

Gambar 4.12 adalah tampilan hasil proses FFT pada data uji 2 yang telah divisualisasikan menjadi grafik. Data uji 2 berisi 2221 jumlah data. Dalam pengujian, data yang diambil sebanyak 2048 agar proses perhitungan FFT menjadi efisien.

### c. Pengujian Data Uji 3

Pengujian dilakukan dengan waktu pengambilan data selama 11 detik, dengan waktu *sampling* selama 2 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 500 Hz. Hasil dari pengujian proses FFT pada data uji 3 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 3 yang dapat dilihat pada Gambar 4.13.



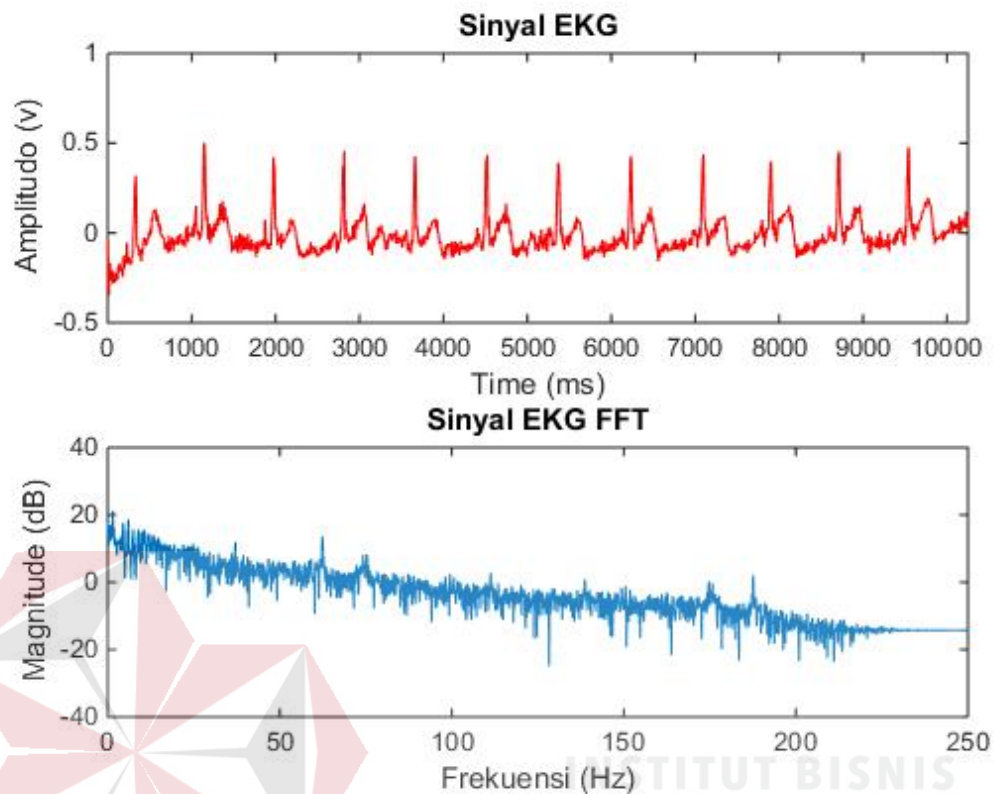
Gambar 4.13 Tampilan Hasil FFT Data Uji 3

Gambar 4.13 tampilan hasil proses FFT pada data uji 3 yang telah divisualisasikan menjadi grafik. Data uji 3 berisi 4390 jumlah data. Dalam pengujian, data yang diambil sebanyak 4096 agar proses perhitungan FFT menjadi efisien.

#### d. Pengujian Data Uji 4

Pengujian dilakukan dengan waktu pengambilan data selama 11 detik, dengan waktu *sampling* selama 2 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 500 Hz. Hasil dari pengujian proses FFT pada data uji 4 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 4 yang dapat dilihat pada Gambar 4.14.



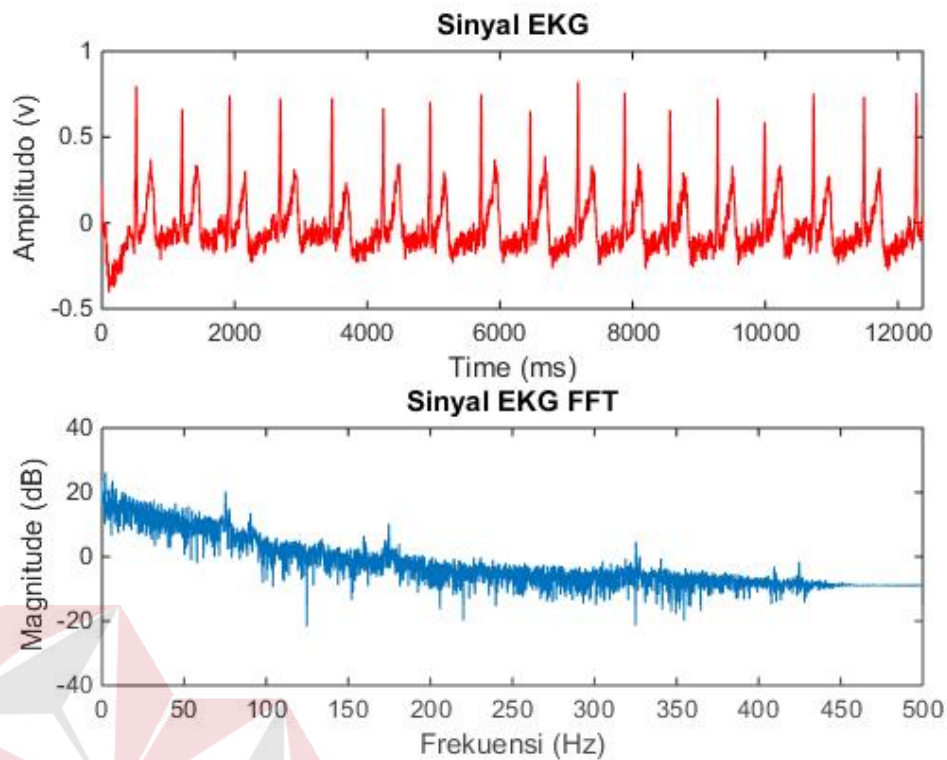


Gambar 4.14 Tampilan Hasil FFT Data Uji 4

Gambar 4.14 adalah tampilan hasil proses FFT pada data uji 4 yang telah divisualisasikan menjadi grafik. Data uji 4 berisi 4390 jumlah data. Dalam pengujian, data yang diambil sebanyak 4096 agar proses perhitungan FFT menjadi efisien.

#### e. Pengujian Data Uji 5

Pengujian dilakukan dengan waktu pengambilan data selama 13 detik, dengan waktu *sampling* selama 1 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 1000 Hz. Hasil dari pengujian proses FFT pada data uji 5 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 5 yang dapat dilihat pada Gambar 4.15.

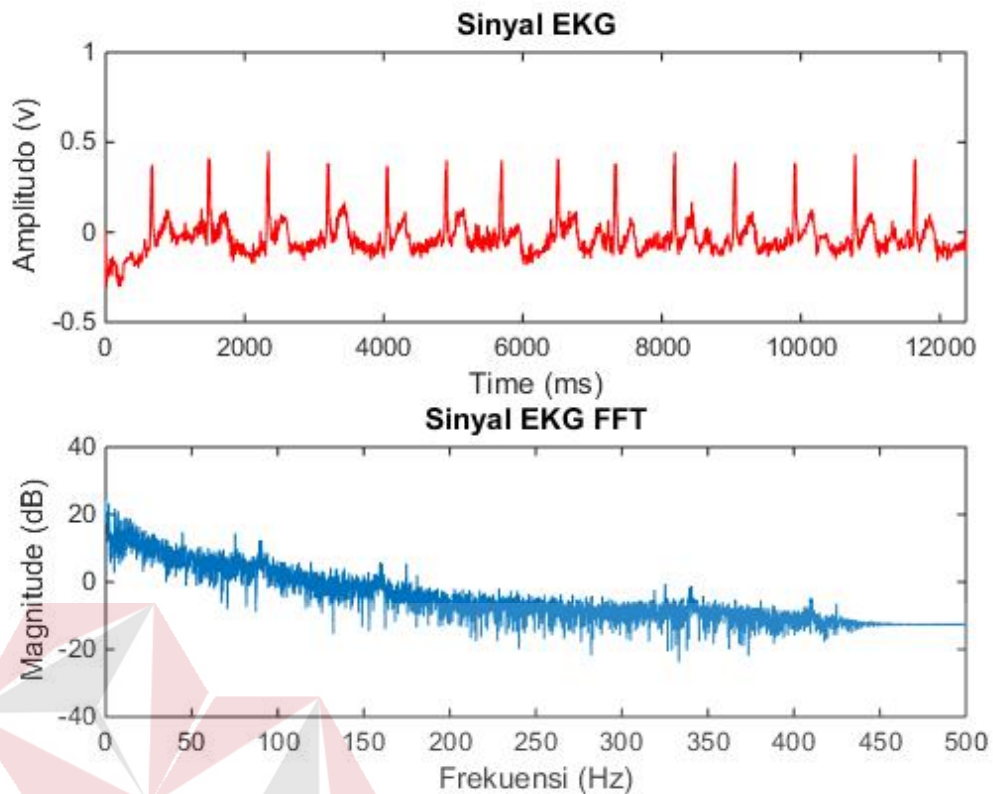


Gambar 4.15 Tampilan Hasil FFT Data Uji 5

Gambar 4.15 adalah tampilan hasil proses FFT pada data uji 5 yang telah divisualisasikan menjadi grafik. Data uji 5 berisi 8595 jumlah data. Dalam pengujian, data yang diambil sebanyak 8192 agar proses perhitungan FFT menjadi efisien.

#### f. Pengujian Data Uji 6

Pengujian dilakukan dengan waktu pengambilan data selama 13 detik, dengan waktu *sampling* selama 1 ms. Frekuensi *sampling* yang diperoleh dari pengujian ini sebanyak 1000 Hz. Hasil dari pengujian proses FFT pada data uji 6 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 6 yang dapat dilihat pada Gambar 4.16.



Gambar 4.16 Tampilan Hasil FFT Data Uji 6

Gambar 4.16 adalah tampilan hasil proses FFT pada data uji 6 yang telah divisualisasikan menjadi grafik. Data uji 6 berisi 8595 jumlah data. Dalam pengujian, data yang diambil sebanyak 8192 agar proses perhitungan FFT menjadi efisien.

#### 4.4 Pengujian Anonimasi

Pengujian anonimasi dilakukan menggunakan *software* MATLAB. Pengujian anonimasi meliputi partisi, pemisahan, modifikasi, enkripsi, dan rekonstruksi pada sinyal EKG.

#### 4.4.1 Tujuan Pengujian Anonimasi

Pengujian ini dilakukan untuk memberikan keamanan pada sinyal EKG. Keamanan yang diberikan bertujuan untuk merubah data sinyal EKG menjadi bentuk yang tersamar.

#### 4.4.2 Kebutuhan Pengujian Anonimasi

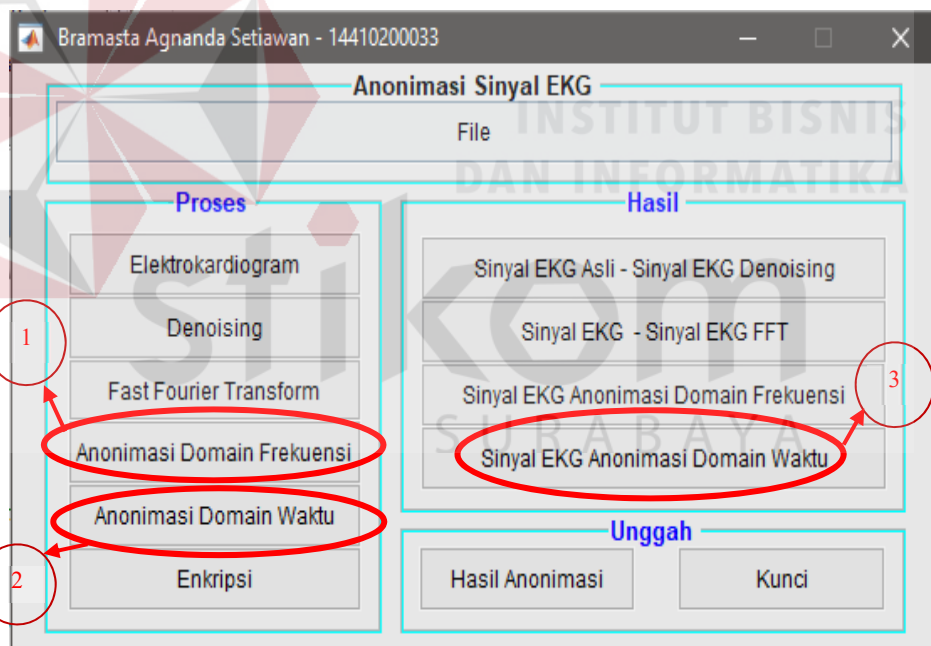
Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software* MATLAB
- c. Data sinyal EKG berekstensi txt

#### 4.4.3 Prosedur Pengujian Anonimasi

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.
- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 4.2.
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms seperti yang dijelaskan dalam sub Bab 4.3.
- e. Partisi *domain* frekuensi rendah dan frekuensi tinggi.
- f. Pisahkan bagian frekuensi tinggi untuk dimodifikasi.
- g. Hitung nilai dari  $\Gamma$  seperti yang dijelaskan dalam sub Bab 3.6.6.

- h. Modifikasi sinyal frekuensi tinggi dengan  $\Gamma$ . Pengujian anonimasi pada tahap modifikasi, dapat dilakukan dengan memilih *button* lingkaran merah nomor 1 di program GUI seperti pada Gambar 4.17.
- i. Rekontruksi sinyal EKG yang telah termodifikasi. Pengujian anonimasi pada tahap rekontruksi dapat dilakukan dengan memilih *button* lingkaran merah nomor 2 di program GUI seperti pada Gambar 4.17.
- j. Bandingkan dengan sinyal EKG asli yang sudah di-*denosing* dengan hasil akhir anonimasi. Hasil akhir pengujian anonimasi dapat dilakukan dengan memilih *button* lingkaran merah nomor 3 di program GUI seperti pada Gambar 4.18.



Gambar 4.17 Program GUI Pengujian Anonimasi

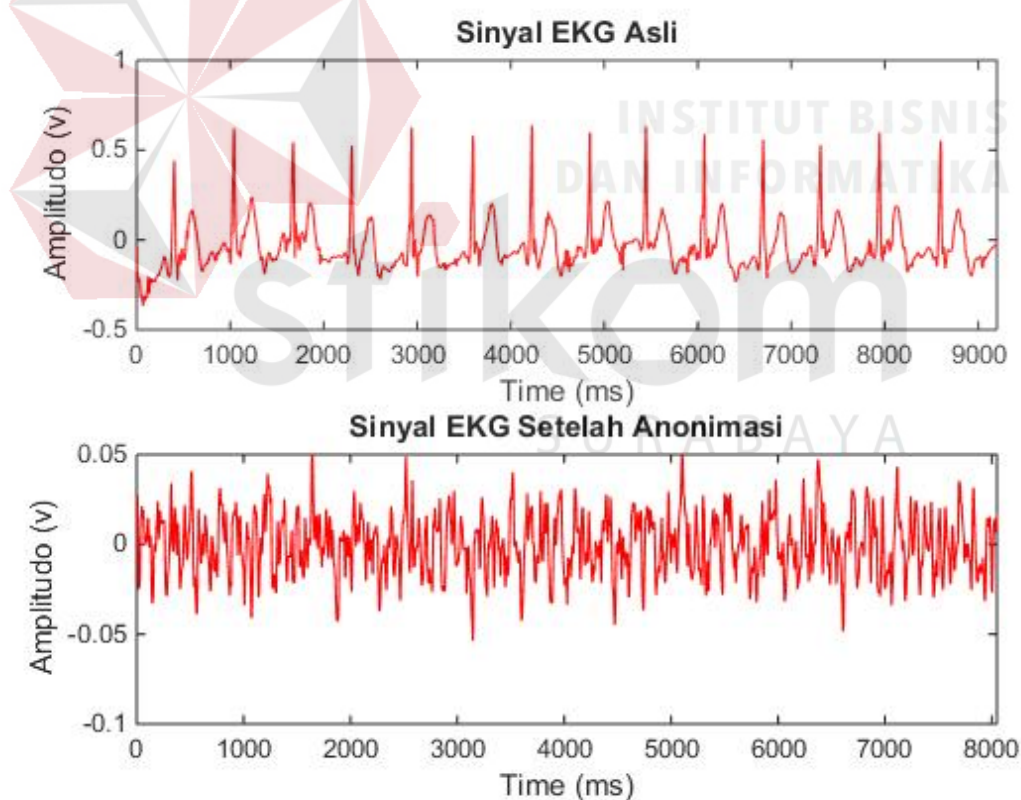
#### 4.4.4 Hasil Pengujian Anonimasi

Pengujian anonimasi pada 6 sampel data uji dilakukan setelah proses FFT. Sinyal EKG dipisah dan dipartisi menjadi 2 bagian antara frekuensi rendah dan

frekuensi tinggi. Bagian frekuensi rendah yang menjadi *key* akan dienkrpsi. Bagian frekuensi tinggi akan dimodifikasi hingga direkontruksi. Dalam tahap rekontruksi sinyal EKG menggunakan metode *inverse* FFT. Hal ini bertujuan agar sinyal EKG dalam *domain* frekuensi dapat diubah kembali ke dalam *domain* waktu. Berikut di bawah ini adalah hasil akhir pengujian proses anonimasi pada 6 sampel data uji:

#### a. Pengujian Data Uji 1

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 1 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 1 yang dapat dilihat pada Gambar 4.18.



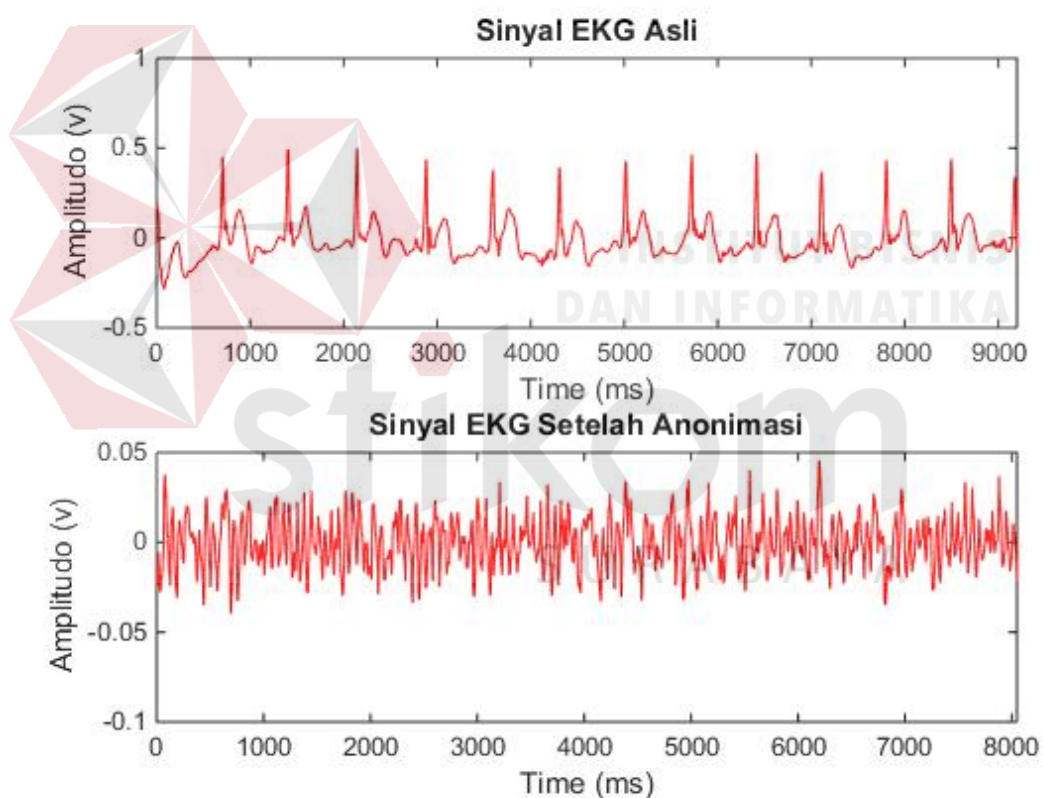
Gambar 4.18 Tampilan Hasil Anonimasi Data Uji 1

Berdasarkan Gambar 4.18, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli.

Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji 1 berisi 2048 data. Data diambil sebanyak 1792 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.

#### b. Pengujian Data Uji 2

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 2 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 2 yang dapat dilihat pada Gambar 4.19.



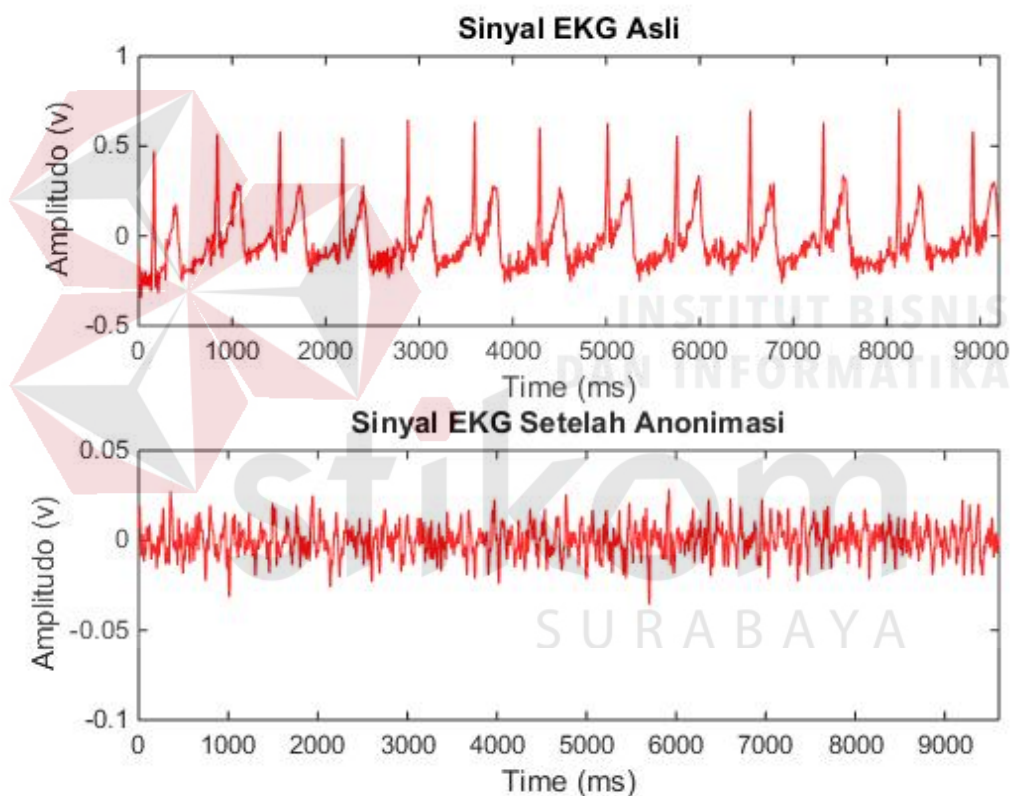
Gambar 4.19 Tampilan Hasil Anonimasi Data Uji 2

Berdasarkan Gambar 4.19, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli. Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji

2 berisi 2048 data. Data diambil sebanyak 1792 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.

### c. Pengujian Data Uji 3

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 3 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 3 yang dapat dilihat pada Gambar 4.20.



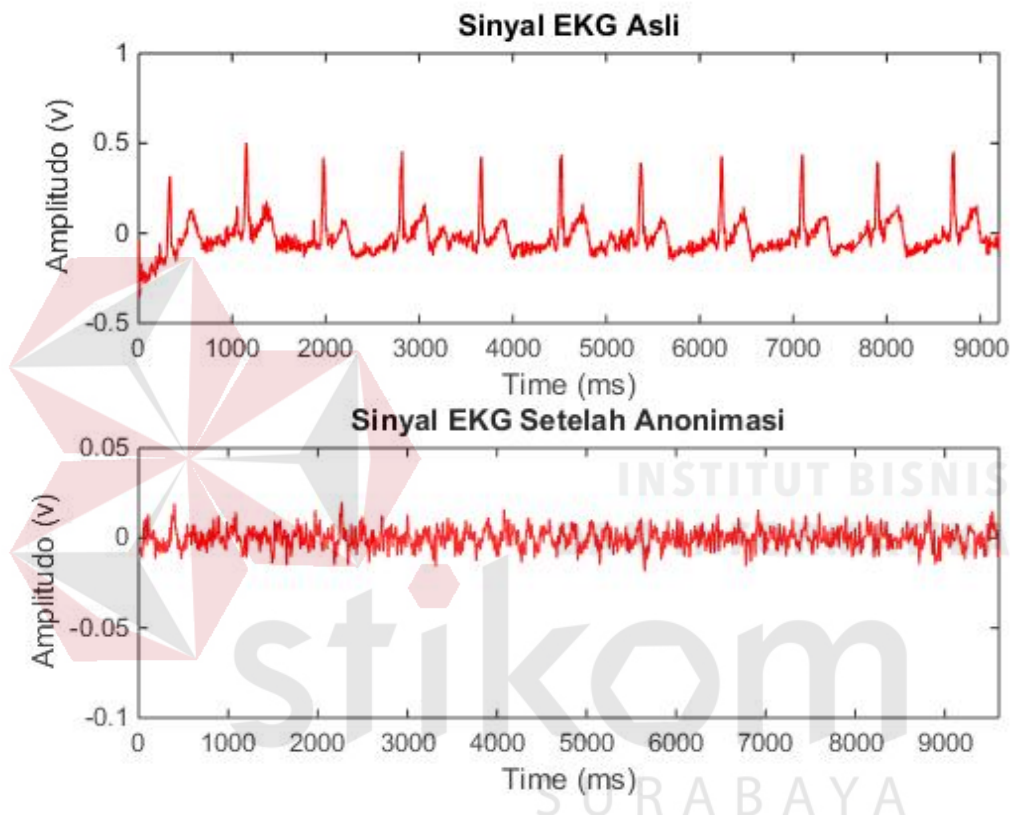
Gambar 4.20 Tampilan Hasil Anonimasi Data Uji 3

Berdasarkan Gambar 4.20, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli. Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji 3 berisi 4096 data. Data diambil sebanyak 3840 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.



#### d. Pengujian Data Uji 4

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 4 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 4 yang dapat dilihat pada Gambar 4.21.

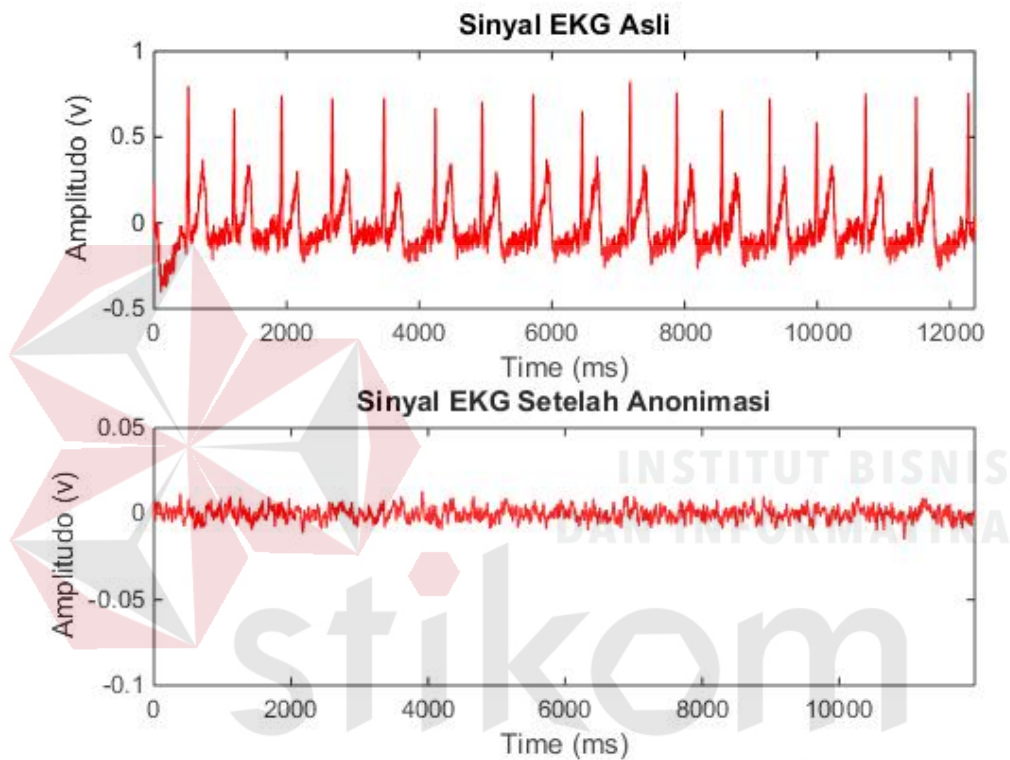


Gambar 4.21 Tampilan Hasil Anonimasi Data Uji 4

Berdasarkan Gambar 4.21, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli. Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji 4 berisi 4096 data. Data diambil sebanyak 3840 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.

### e. Pengujian Data Uji 5

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 5 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 5 yang dapat dilihat pada Gambar 4.22.

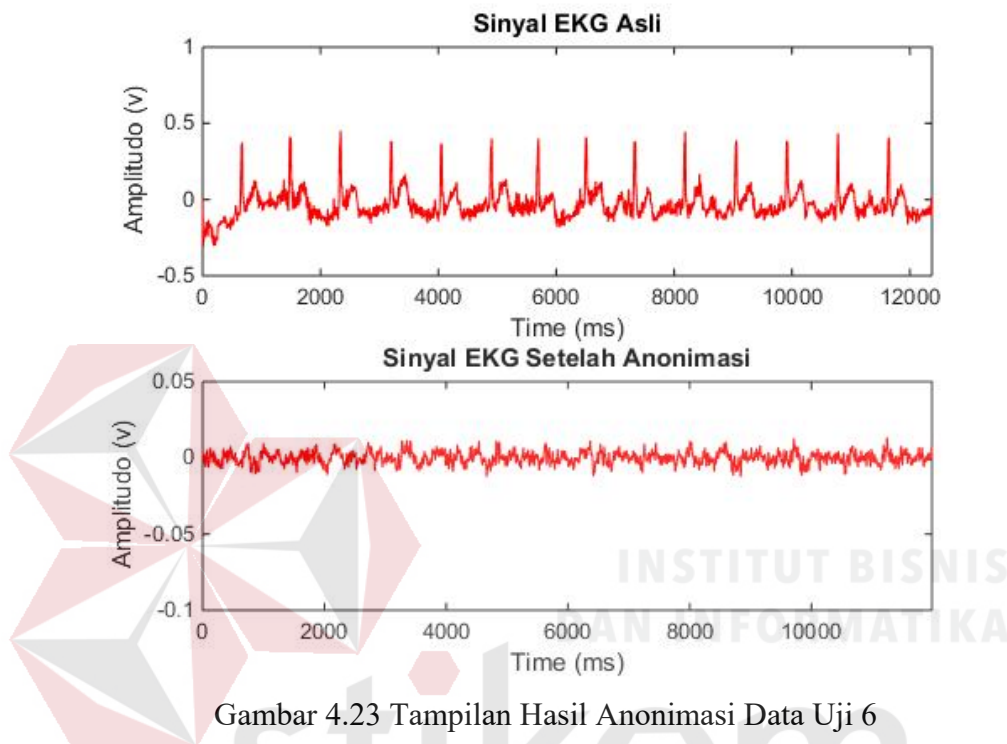


Gambar 4.22 Tampilan Hasil Anonimasi Data Uji 5

Berdasarkan Gambar 4.22, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli. Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji 5 berisi 8192 data. Data diambil sebanyak 7936 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.

#### f. Pengujian Data Uji 6

Hasil dari pengujian proses anonimasi sinyal jantung EKG pada data uji 6 telah divisualisasikan menjadi grafik. Berikut adalah tampilan hasil anonimasi data uji 6 yang dapat dilihat pada Gambar 4.23.



Gambar 4.23 Tampilan Hasil Anonimasi Data Uji 6

Berdasarkan Gambar 4.23, dapat dibuktikan bahwa bentuk dari sinyal EKG setelah mengalami proses anonimasi berbeda dengan bentuk sinyal EKG yang asli. Hal ini menunjukkan proses anonimasi sinyal jantung EKG telah berhasil. Data uji 6 berisi 8192 data. Data diambil sebanyak 7936 digunakan untuk proses anonimasi dan sisanya sebagai panjang *key* untuk proses enkripsi.

#### 4.5 Pengujian Enkripsi

Pengujian enkripsi dilakukan menggunakan *software* MATLAB. Pengujian ini dilakukan pada *key* yang sudah terenkripsi untuk selanjutnya dikirim ke *email medical center*.

#### 4.5.1 Tujuan Pengujian Enkripsi

Pengujian ini dilakukan untuk memberikan keamanan pada sinyal EKG. Keamanan yang diberikan bertujuan agar bilangan kompleks pada *key* diubah dalam nilai *magnitude* dan nilai sudut.

#### 4.5.2 Kebutuhan Pengujian Enkripsi

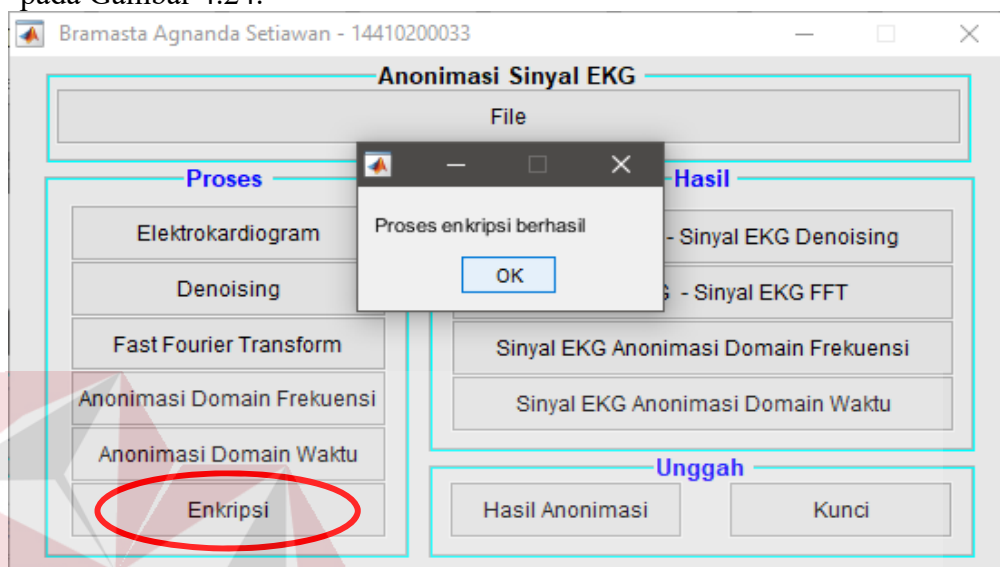
Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software* MATLAB
- c. Data sinyal EKG berekstensi txt

#### 4.5.3 Prosedur Pengujian Enkripsi

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.
- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 4.2.
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms seperti yang dijelaskan dalam sub Bab 4.3.
- e. Partisi *domain* frekuensi rendah dan frekuensi tinggi.
- f. Tentukan panjang nilai *key*, agar perhitungan lebih efisien panjang *key* adalah kelipatan dari  $2^n$ .

- g. Pisahkan bagian frekuensi rendah dengan ukuran yang sama dengan panjang *key* yang telah ditentukan untuk dienkripsi. Pengujian enkripsi dapat dilakukan dengan memilih *button* lingkaran merah di program GUI seperti pada Gambar 4.24.




Gambar 4.24 Program GUI Pengujian Enkripsi

#### 4.5.4 Hasil Pengujian Enkripsi

Pengujian enkripsi pada 6 sampel data uji dilakukan setelah proses FFT. Sinyal EKG akan dipisah dan dipartisi menjadi 2 bagian antara frekuensi rendah dan frekuensi tinggi. Bagian frekuensi tinggi akan dimodifikasi hingga direkonstruksi. Bagian frekuensi rendah yang menjadi *key* akan dienkripsi. Panjang *key* yang digunakan pada pengujian ini adalah 128 data. Nilai dari *key* yaitu bilangan kompleks, terdiri atas bilangan *real* dan bilangan *imaginer*. Sebelum dalam tahap enkripsi, dilakukan perhitungan untuk memperoleh nilai *magnitude* dan nilai sudut pada MATLAB. Untuk mencari nilai *magnitude* menggunakan fungsi `abs` pada MATLAB. Sedangkan untuk mencari nilai sudut menggunakan fungsi `angle` pada MATLAB.

Dalam proses enkripsi, nilai *magnitude* dan nilai sudut dari bilangan kompleks dimasukkan dalam perhitungan rumus seperti yang dijelaskan dalam sub Bab 3.6.7. Berikut di bawah ini adalah perbandingan hasil *key* sebelum dienkripsi dan setelah dienkripsi pada Gambar 4.25 dan Gambar 4.26.



```

1 -66.038+0i
2 -17.245+0.42742i
3 -5.248-11.191i
4 -33.676+8.0279i
5 -28.437+4.8053i
6 0.18453+12.674i
7 -15.646+4.0196i
8 -8.0438+5.1692i
9 -20.418+5.1044i
10 2.1664+8.9103i
11 -10.353+7.1582i
12 -18.898+2.8408i
13 -10.97+4.6909i
14 -32.459-2.3868i
15 -40.906+8.6566i
16 73.952+4.7248i
17 13.829+16.02i
18 6.2476+1.0146i
19 2.4261+1.6438i
20 -0.26646+0.90108i

```

Gambar 4.25 Key Sebelum Dienkripsi

```

1 473.26, 32.991
2 131.75, 32.818
3 97.521, -3.0651
4 253.34, 31.353
5 212.88, 31.819
6 99.73, 21.894
7 124.08, 31.231
8 77.931, 28.993
9 158.33, 31.276
10 75.19, 20.326
11 99.105, 28.757
12 144.77, 31.947
13 94.518, 30.163
14 238.83, -10.477
15 303.68, 31.531
16 529.72, 11.447
17 159.14, 17.011
18 55.306, 12.127
19 31.514, 15.169
20 17.578, 24.008

```

Gambar 4.26 Key Setelah Dienkripsi

Pada Gambar 4.25 adalah *key* yang belum dienkrpsi, dimana pada kolom pertama adalah nilai bilangan *real* dan pada kolom kedua adalah nilai bilangan *imaginer*. Pada Gambar 4.26 adalah *key* yang sudah dienkrpsi, dimana pada kolom pertama adalah nilai *magnitude* dan pada kolom kedua adalah nilai sudut.

#### **4.6 Pengujian Perubahan Panjang Key**

Pengujian perubahan panjang *key* dilakukan menggunakan *software* MATLAB. Pengujian ini dilakukan dengan merubah panjang *key* dari pengujian sebelumnya seperti yang dijelaskan dalam sub Bab 4.5.4 .

##### **4.6.1 Tujuan Pengujian Perubahan Panjang Key**

Pengujian ini dilakukan untuk mengetahui pengaruh perubahan panjang *key* yang ditentukan pada sinyal EKG anonimasi.

##### **4.6.2 Kebutuhan Pengujian Perubahan Panjang Key**

Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software* MATLAB
- c. Data sinyal EKG berekstensi txt

##### **4.6.3 Prosedur Pengujian Perubahan Panjang Key**

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.

- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 4.2
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms seperti yang dijelaskan dalam sub Bab 4.3.
- e. Partisi *domain* frekuensi rendah dan frekuensi tinggi.
- f. Tentukan panjang nilai *key*, agar perhitungan lebih efisien panjang *key* adalah kelipatan dari  $2^n$ .

#### 4.6.4 Hasil Pengujian Perubahan Panjang *Key*

Sinyal EKG akan dipisah dan dipartisi menjadi 2 bagian antara frekuensi rendah dan frekuensi tinggi. Bagian frekuensi tinggi akan dimodifikasi hingga direkonstruksi. Bagian frekuensi rendah yang menjadi *key* akan dienkripsi. Pengujian perubahan panjang *key* dilakukan untuk mengetahui pengaruh panjang *key* pada waktu pemrosesan. Panjang *key* yang digunakan pada pengujian ini adalah 128 data dan 256 data. Berikut ini adalah hasil dari pengujian perubahan panjang *key* terhadap sinyal EKG seperti pada Tabel 4.2.

Tabel 4.2 Pengujian Perubahan *Key*

Key	Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6
128	0.0054	0.0054	0.0059	0.0059	0.0067	0.0067
256	0.0054	0.0054	0.0059	0.0059	0.0067	0.0067



Pada Tabel 4.2 adalah nilai rata-rata dari 100 kali uji coba yang telah dilakukan. Hal ini menunjukkan bahwa dari pengujian perubahan *key* tidak berpengaruh pada waktu pemrosesan anonimasi sinyal EKG pada MATLAB.

#### **4.7 Pengujian Unggah Data Pada Server**

Pengujian pengiriman data pada *server* dilakukan menggunakan *software* MATLAB. Pengujian ini dilakukan pada data sinyal EKG hasil anonimasi.

##### **4.7.1 Tujuan Pengujian Unggah Data Pada Server**

Pengujian ini dilakukan untuk mengirimkan data ke *server*. Data yang dikirim adalah data dari hasil anonimasi sinyal EKG.

##### **4.7.2 Kebutuhan Pengujian Unggah Data Pada Server**

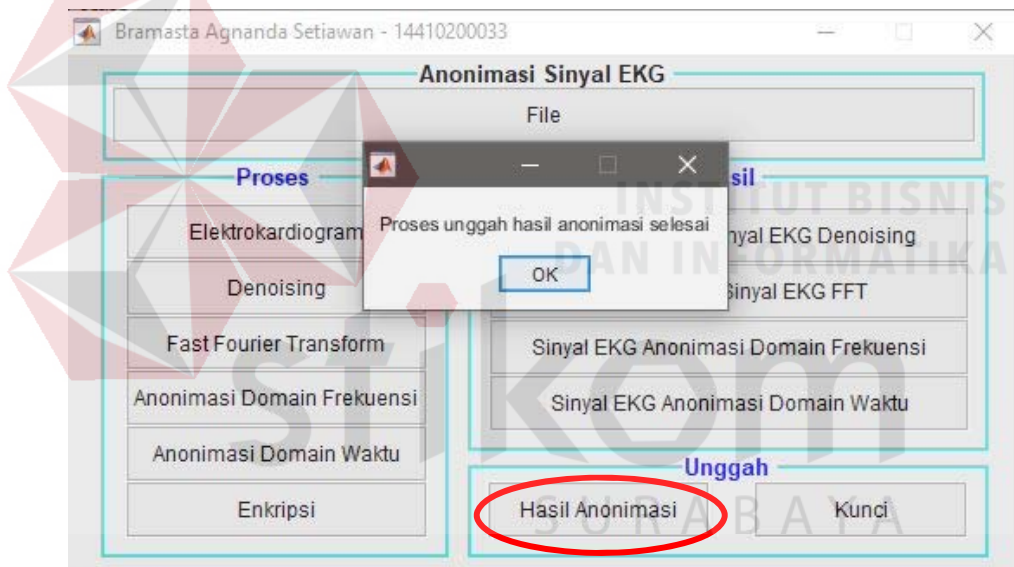
Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software* MATLAB
- c. Data sinyal EKG berekstensi txt
- d. *Software FileZilla Server*

##### **4.7.3 Prosedur Pengujian Unggah Data Pada Server**

- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.

- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 4.2.
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms seperti yang dijelaskan dalam sub Bab 4.3.
- e. Lakukan proses anonimasi seperti yang dijelaskan pada sub Bab 4.4.
- f. Unggah data hasil anonimasi dan cek status pengunggahan pada *server*.  
Pengujian unggah data pada *server* dapat dilakukan dengan memilih *button* lingkaran merah di program GUI seperti pada Gambar 4.27.



Gambar 4.27 Program GUI Pengujian Unggah Data Pada Server

#### 4.7.4 Hasil Pengujian Unggah Data Pada Server

Pengujian pengiriman data pada *server* dilakukan setelah proses anonimasi. *Server* yang digunakan pada pengujian ini adalah *software FileZilla server*. Protokol yang digunakan dalam pengujian ini adalah FTP. Sebelum data dikirim pada *server*, hasil dari proses anonimasi diubah menjadi *file txt* dengan

menggunakan fungsi `dlmwrite` di MATLAB. Berikut di bawah ini adalah hasil dari pengujian bahwa data berhasil terkirim pada *server* seperti pada Gambar 4.28.

```

Warning: FTP over TLS is not enabled, users cannot securely log in.
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> Connected on port 21, sending welcome message...
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> 220-FileZilla Server 0.9.60 beta
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> 220 Please visit https://filezilla-project.org/
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> USER EKG
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> 331 Password required for ekg
(000001)10/07/2018 19.27.14 - (not logged in) (192.168.137.125)> PASS *****
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 230 Logged on
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> PWD
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 257 "/" is current directory.
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> STAT
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 500 Syntax error, command unrecognized.
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> PWD
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 257 "/" is current directory.
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> TYPE I
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 200 Type set to I
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> PORT 192,168,137,125,9,52
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 200 Port command successful
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> STOR hasilanonimasi.txt
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 150 Opening data channel for file upload to server of "/hasilanonimasi.txt"
(000001)10/07/2018 19.27.14 - ekg (192.168.137.125)> 226 Successfully transferred "/hasilanonimasi.txt"
(000001)10/07/2018 19.27.25 - ekg (192.168.137.125)> disconnected.

```

Gambar 4.28 Data Hasil Anonimasi Berhasil Diunggah

## 4.8 Pengujian Unggah Data Pada *Email*

Pengujian pengiriman data pada *email* dilakukan menggunakan *software* MATLAB. Pengujian ini dilakukan pada data *key* dari sinyal EKG setelah dienkripsi.

### 4.8.1 Tujuan Pengujian Unggah Data Pada *Email*

Pengujian ini dilakukan untuk mengirimkan data ke *email*. Data yang dikirim adalah data *key* dari sinyal EKG yang sudah dienkripsi.

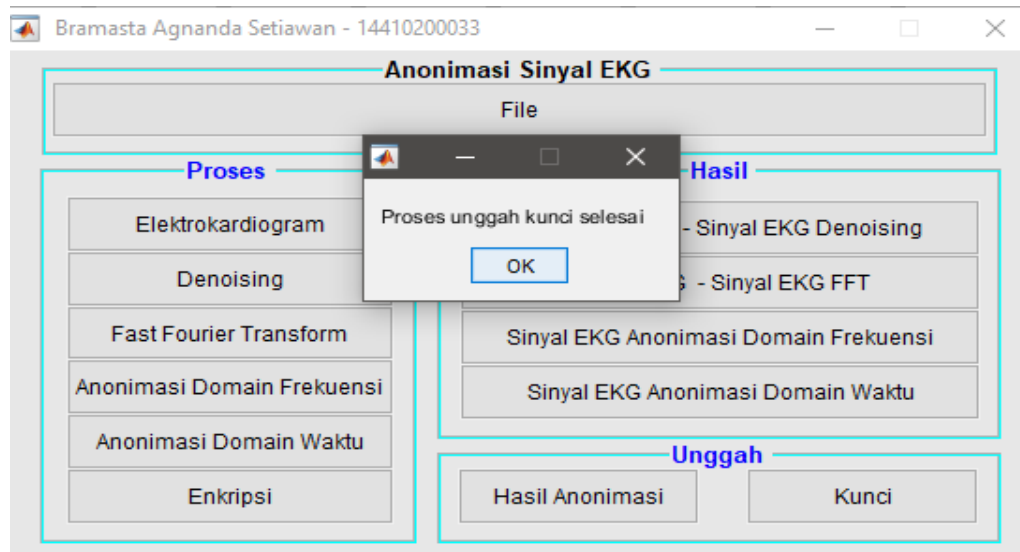
### 4.8.2 Kebutuhan Pengujian Unggah Data Pada *Email*

Kebutuhan yang diperlukan untuk melakukan pengujian antara lain:

- a. *Personal Computer/Laptop*
- b. *Software MATLAB*
- c. Data sinyal EKG berekstensi txt
- d. *Email* pengirim
- e. *Email* penerima (*medical center*)

#### 4.8.3 Prosedur Pengujian Unggah Data Pada *Email*

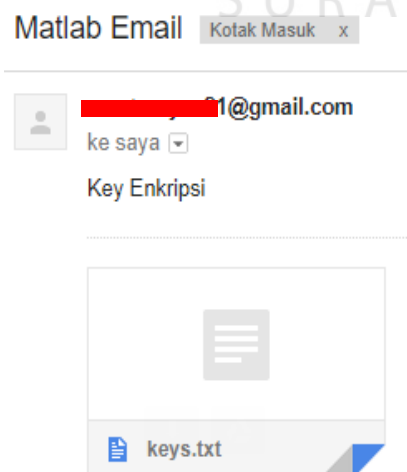
- a. Buka *software* MATLAB.
- b. Impor data sinyal EKG berekstensi *file* txt di *software* MATLAB, seperti yang dijelaskan dalam sub Bab 3.6.2.
- c. Lakukan *denoising* sinyal untuk menghilangkan *noise* pada 6 sampel data uji, seperti yang telah dijelaskan dalam sub Bab 4.2
- d. Lakukan transformasi FFT pada data sinyal EKG dari 2 subjek uji dengan masing-masing waktu *sampling* 1 ms, 2 ms, 4 ms seperti yang dijelaskan dalam sub Bab 4.3.
- e. Lakukan proses enkripsi seperti yang dijelaskan dalam sub Bab 4.5. Unggah data hasil enkripsi dan cek status unggah pada *email*. Pengujian unggah data pada *email* dapat dilakukan dengan memilih *button* lingkaran merah di program GUI seperti pada Gambar 4.29.



Gambar 4.29 Program GUI Pengujian Unggah Data Pada Email

#### 4.8.4 Hasil Pengujian Unggah Data Pada *Email*

Pengujian pengiriman data pada *email* dilakukan setelah proses enkripsi. Protokol yang digunakan dalam pengujian ini adalah SMTP. Sebelum data dikirim pada *email*, hasil dari proses enkripsi diubah menjadi *file* txt dengan menggunakan fungsi `dlmwrite` di MATLAB. Berikut di bawah ini adalah hasil dari pengujian bahwa data berhasil terkirim pada *email medical center* seperti pada Gambar 4.30.



Gambar 4.30 Data *Key* Berhasil Diunggah

## BAB V

### PENUTUP

Berdasarkan dari hasil pengujian anonimasi sinyal EKG (Elektrokardiogram) untuk keamanan transmisi data pada sebuah *node* sensor ini, maka dapat diambil beberapa kesimpulan dan saran sebagai berikut:

#### 5.1 Kesimpulan

Adapun kesimpulan yang dapat dituliskan setelah melakukan analisa dari hasil pengujian sistem yang telah dibuat antara lain sebagai berikut:

1. Berdasarkan hasil pengujian didapatkan bahwa hasil algoritma Jusak-Seedahmed dapat digunakan untuk melakukan anonimasi sinyal EKG, dalam penelitian ini digunakan 3 macam sinyal EKG dengan frekuensi *sampling* 250 Hz, 500 Hz, 1000 Hz.
2. Berdasarkan hasil pengujian dengan menggunakan variasi panjang *key* yang berbeda algoritma Jusak-Seedahmed dapat melakukan anonimasi sinyal EKG. Untuk perubahan panjang *key* 128 dan *key* 256 dengan algoritma Jusak-Seedahmed tidak mempengaruhi waktu pemrosesan di MATLAB.
3. Berdasarkan hasil pengujian didapatkan bahwa sinyal EKG dapat dienkripsi dengan baik.

4. Berdasarkan hasil pengujian didapatkan bahwa *key* dapat diunggah ke *email medical center* menggunakan protokol SMTP, sedangkan hasil anonimasi dapat diunggah ke *server* menggunakan protokol FTP.

## 5.2 Saran

Saran yang dapat diberikan untuk mengembangkan sistem ini sebagai berikut:

1. Algoritma ini dapat diimplementasikan pada *node* sensor, sehingga algoritma ini dapat di implementasikan pada transmisi sinyal EKG nyata.
2. Hasil anonimasi sinyal EKG dapat diimplementasikan pada perangkat *mobile*.



## DAFTAR PUSTAKA

- Arduino.cc.* (2017). Dipetik September 28, 2017, dari <https://www.arduino.cc/en/Guide/Arduino101>
- Cahyono, B. (2013). Penggunaan Software Matrix Laboratory (Matlab) Dalam Pembelajaran Aljabar Linier. *Phenomenon*, 1.
- Daso, F., Jusak, & Puspasari, I. (2015). Analisis Sinyal Suara Jantung Dengan Menggunakan Analisis Spektrum. *JCONES*, 92-99.
- Dianputra, R. (2014). Implementasi Algoritma Fast Fourier Transform Untuk Pengolahan Sinyal Digital Pada Tuning Gitar Dengan Open String. *Teknologi Informasi*, 10.
- Jusak, J., & Mahmoud, S. S. (2018). A Novel and Low Processing Time ECG Security. *International Journal of Communication Networks and Information Security (IJCNIS)*, 213.
- Sipasulta, R. Y. (2014). Simulasi Sistem Pengacak Sinyal Dengan Metode FFT (Fast Fourier Transform). *Teknik Elektro dan Komputer*.
- Sugiarto, W. R., Jusak, & Puspasari, I. (2016). Rancang Bangun Alat Elektrokardiograf Untuk Visualisasi, Perekaman, dan Penyimpanan Sinyal Jantung. *JCONES*, 38-46.



Wibowo, P. A., Jusak, & Puspasari, I. (2016). Analisis Time-Frequency Pada Sinyal ECG (Electrocardiogram) dengan Menggunakan Continuous Wavelet Transform. *JCONES*, 47-54.

