



**IMPLEMENTASI SISTEM *TELEMETRY* MONITOR DETAK
JANTUNG DAN LAJU NAPAS UNTUK PENCEGAHAN
OVERTRAINING SAAT BERSEPEDA
MENGUNAKAN LOGIKA FUZZY**

SKRIPSI

Oleh

Andina Maharani

NIM 161910201114

**PROGRAM STUDI STRATA I TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2018**



**IMPLEMENTASI SISTEM *TELEMETRY* MONITOR DETAK
JANTUNG DAN LAJU NAPAS UNTUK PENCEGAHAN
OVERTRAINING SAAT BERSEPEDA
MENGUNAKAN LOGIKA FUZZY**

SKRIPSI

diajukan guna melengkapi skripsi dan memenuhi syarat-syarat
untuk menyelesaikan Program Studi Teknik Elektro (S1)
dan mencapai gelar Sarjana Teknik (ST)

Oleh
Andina Maharani
NIM 161910201114

**PROGRAM STUDI STRATA I TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2018**

PERSEMBAHAN

1. Tuhan Yang Maha Esa Allah SWT syukur Alhamdulillah selalu terucap yang telah memberikan Rahmat, Hidayahnya, serta petunjuk sehingga dapat terselesaikannya laporan skripsi ini.
2. Untuk kedua Orang Tua tercinta '**Gunawan dan Endang D.A**' yang telah memberikan begitu banyak pengorbanan untuk anak-anaknya serta doa yang telah tercurah kepada-Nya.
3. Untuk Adik saya, '**Margian Enjana dan Mardian Drijana**' yang telah memberikan semangat untuk menyelesaikan studi.
4. Seluruh **Dosen Teknik Elektro** yang telah sabar mendidik serta membimbing saya selama kuliah dan selama pengerjaan laporan skripsi ini hingga selesai.
5. Terima kasih untuk saudara-saudara seperjuangan '**INTEL 2013**' yang telah memberikan banyak kenangan indah dan hal-hal yang baru sehingga membuat saya lebih dewasa dan mampu menyelesaikan laporan skripsi hingga selesai.
6. Terima kasih untuk seluruh '**Asisten Laboratorium**' yang telah sabar membimbing dan memberikan ilmunya selama proses belajar di laboratorium sehingga saya mampu menyelesaikan laporan skripsi hingga selesai.
7. Terima kasih untuk seluruh saudara '**Teknik Elektro**' yang telah menempera mental dan mengubah pribadi saya menjadi lebih baik lagi.
8. Almamaterku tercinta '**Fakultas Teknik Universitas Jember**'.

MOTTO

“Pengetahuan tidaklah cukup, kita harus mengamalkannya.

Niat tidaklah cukup, kita harus melakukannya”

(Johann Wolfgang von Goethe)

“Berusaha dan terus berusaha tanpa mengenal lelah.

*Karena hasil yang kita dapat akan setimpal
dengan kerja keras kita”*

(Andina Maharani)

*“Bersabarlah terhadap kerasnya sikap seorang guru. Sesungguhnya gagalnya
mempelajari ilmu karena memusuhinya. Barangsiapa belum merasakan
pahitnya belajar walau sebentar, Ia akan merasakan hinanya
kebodohan sepanjang hidupnya.*

*Dan barangsiapa ketinggalan belajar di masa mudanya, Maka bertakbirlah
untuknya empat kali karena kematiannya. Demi Allah hakekat seorang
pemuda adalah dengan ilmu dan takwa. Bila keduanya tidak
ada maka tidak ada anggapan baginya.”*

(Imam Syafi'i)

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Andina Maharani

NIM : 161910201114

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul: “Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya dan belum pernah diajukan pada institusi mana pun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Juli 2018

Yang menyatakan,

Andina Maharani

NIM 161910201114

LAPORAN SKRIPSI

**IMPLEMENTASI SISTEM *TELEMETRY* MONITOR DETAK
JANTUNG DAN LAJU NAPAS UNTUK PENCEGAHAN
OVERTRAINING SAAT BERSEPEDA
MENGUNAKAN LOGIKA FUZZY**

Oleh
Andina Maharani
NIM 161910201114

Pembimbing

Dosen Pembimbing Utama : Dodi Setiabudi, S.T., M.T.

Dosen Pembimbing Anggota : Khairul Anam, S.T., M.T., Ph.D.

LEMBAR PENGESAHAN

Laporan Skripsi berjudul “Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy” karya Andina Maharani telah diuji dan disahkan oleh Fakultas Teknik Universitas Jember pada:

Hari, tanggal : Senin, 9 Juli 2018

Tempat : Fakultas Teknik

Ketua,

Anggota I,

Dodi Setiabudi, S.T., M.T.
NIP. 198405312008121004

Khairul Anam, S.T., M.T., Ph.D.
NIP. 197804052005011002

Anggota II,

Anggota III,

Catur Suko Sarwono, S.T., M.Si.
NIP. 196801191997021001

Widya Cahyadi, S.T., M.T.
NIP. 198511102014041 001

Mengesahkan
Dekan,

Dr. Ir. Entin Hidayah, M. U.M
NIP. 19661215 199503 2 001

PRAKATA

Puji syukur kami panjatkan atas kehadiran Allah SWT. atas segala berkat, bimbingan, serta rahmat dan hidayah-Nya sehingga penulis bisa menyelesaikan laporan skripsi tentang “Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy”. Laporan proyek akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan Strata Satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Penyusunan laporan proyek akhir ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada :

1. Orang Tua saya Bapak Gunawan dan Ibu Endang D. A.
2. Ibu Dr. Ir. Entin Hidayah, M. U.M selaku Dekan Fakultas Teknik Universitas Jember.
3. Bapak Bambang Sri Kaloko, S.T., M.T. selaku Ketua Jurusan Fakultas Teknik Elektro Universitas Jember.
4. Bapak Dodi Setiabudi, ST., MT, selaku Dosen Pembimbing Utama dan Bapak Khairul Anam, S.T., M.T, Ph.D., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu dan pikiran serta perhatiannya guna memberikan bimbingan dan pengarahan demi terselesaikannya proyek akhir ini.
5. Bapak Suprihadi, S.T., M.T, selaku Dosen Pembimbing Akademik yang telah memberikan bimbingan dan pengarahannya selama dibangku kuliah.
6. Bapak Catur Suko Sarwono, S.T., M.T dan Bapak Widya Cahyadi S.T., M.T selaku Tim Penguji Proyek Akhir yang telah meluangkan waktu dan pikiran serta perhatiannya guna memberikan pengarahan demi terselesaikannya penulisan laporan proyek akhir ini.
7. Sivitas Akademika Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.
8. Seluruh Asisten Laboratorium Teknik Elektro Universitas Jember.

9. Sahabat-sahabatku, Levi, Ifta, Aninda, Tika, Septa dan lainnya yang tidak dapat disebutkan satu persatu yang telah membantu dan memberi semangat selama proses pembuatan laporan ini.
10. Mahasiswa alih jenjang, Dinda, Silvi, Iqbal dan Herlambang yang selalu menemani dan membantu selama proses pembuatan laporan ini.
11. Para master mas Cries dan Rokhim yang telah membagi ilmunya dan bersedia menjadi pembimbing tambahan untuk membantu saya menyelesaikan laporan ini.
12. Almamaterku tercinta Universitas Jember.
13. Seluruh pihak yang turut membantu dalam penyelesaian laporan tugas akhir ini.

Semoga laporan skripsi ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya untuk disiplin ilmu teknik elektro, kritik dan saran diharapkan terus mengalir untuk lebih menyempurnakan skripsi ini dan diharapkan dapat dikembangkan untuk penelitian-penelitian selanjutnya.

Jember, Januari 2018

Penulis

RINGKASAN

“Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* saat Bersepeda Menggunakan Logika Fuzzy”; Andina Maharani 161910201114; 2018: 72 halaman; Program Studi Strata Satu (S1) Teknik , Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.

Overtraining adalah bentuk kronis dari kelelahan patologis dalam olahraga. Keinginan yang berlebihan yaitu melebihi kapasitas fungsional otak adalah pemicu *overtraining*. Untuk mencegah terjadinya *overtraining* maka digunakan metode fuzzy untuk mengambil kesimpulan dari hasil pengukuran kondisi tubuh. *Fuzzy logic* adalah suatu cabang ilmu *Artificial Intelligence*, yaitu suatu pengetahuan yang membuat komputer dapat meniru kecerdasan manusia sehingga diharapkan komputer dapat melakukan hal-hal yang dikerjakan manusia memerlukan kecerdasan. Pada proses pembuatan proyek akhir ini penyusun menggunakan sensor detak jantung dan laju napas sebagai masukan ke Arduino yang di *supply* oleh daya 9 V DC, kemudian keluaran dari Aduino berupa hasil kondisi tubuh dan *output fuzzy* yang ditampilkan di LCD dan disimpan pada memori, kemudian dapat dimonitor dari jarak jauh dengan radio *telemetry*. Parameter kontrol *fuzzy* sensor detak jantung yaitu kategori pelan dengan batas 40 hingga 70 bpm, kategori normal dengan batas 65 hingga 105 bpm, kategori aktivitas ringan dengan batas 100 hingga 135 bpm, kategori aktivitas berat dengan batas 130 hingga 170 bpm, kategori anaerob dengan batas >165 bpm. Parameter kontrol *fuzzy* sensor laju napas yaitu kategori lambat dengan batas <17 bpm, kategori normal dengan batas 13 hingga 20 bpm, kategori cepat dengan batas >17 bpm. Kontrol *output fuzzy* yaitu berhenti dengan nilai 0 sampai 25 dan lanjut dengan nilai 25 sampai 50. Pengujian sensor nilai *error* terbesar yaitu 5,95 % pada detak jantung dan 8 % pada laju napas. Sedangkan nilai *error* terkecil yaitu 1,06 % pada detak jantung dan 0 % pada laju napas. Jarak maksimum pengiriman data menggunakan radio tanpa penghalang sejauh 140 meter dan dengan

penghalang sejauh 96 meter. Hasil pengujian data secara keseluruhan memiliki hasil yang sesuai dimana nilai yang ditampilkan pada alat dan *rule base* sesuai.

Kata kunci: *overtraining*, *fuzzy logic*, sensor detak jantung, sensor laju napas, Arduino UNO, radio *telemetry*.



SUMMARY

“Implementation of Telemetry System Heart Rate and Rate of Breath Monitor for Prevention of Overtraining while Cycling Used Fuzzy Logic”; Andina Maharani 161910201114; 2018: 72 pages; Program Study Strata of One (S1) Engineering, Department of Electrical Engineering, Faculty of Engineering, University of Jember.

Overtraining is a chronic form of pathological fatigue in sports. Excessive desire that exceeds the functional capacity of the brain is the trigger of overtraining. To prevent the occurrence of overtraining the fuzzy method is used to draw conclusions from the results of body condition measurements. Fuzzy logic is a branch of Artificial Intelligence science, which is a knowledge that makes computers can imitate human intelligence so that computer is expected to do things that when done human needs intelligence. In the final project making process the author uses the heartbeat and breath rate sensors as input to the Arduino which is supplied by 9 V DC power, then the output from Aduino is the result of body condition and fuzzy output displayed on LCD and stored in memory, then can be monitored from a distance with radio telemetry. The fuzzy control parameter of the heartbeat is a slow category with a limit of 40 to 70 bpm, a normal category with limits of 65 to 105 bpm, light activity categories with limits of 100 to 135 bpm, heavy activity categories with limits of 130 to 170 bpm, anaerobic category with limits > 165 bpm. The fuzzy control parameter of the respiratory rate sensor is slow category with limit <17 bpm, normal category with 13 to 20 bpm limit, fast category with limit > 17 bpm. The fuzzy output control is stops with values from 0 to 25 and continues with values 25 to 50. The largest sensor error test results is 5,95 % in heart rate and 8 % at respiratory rate. While the smallest error value is 1,06 % in heart rate and 0 % at the rate of breath. The maximum distance of data transmission using radio without obstacle as far as 140 meter and with barrier as far as 96 meter. The results of testing the data as a

whole have a corresponding result where the values displayed on the tool and rule base are appropriate.

Keywords: overtraining, fuzzy logic, heartbeat, breathing rate sensor, Arduino UNO, radio telemetry.



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Jenis olahraga aerobik adalah jenis olahraga yang menggunakan energi dari hasil pembakaran oksigen, dilakukan secara sistematis dengan meningkatkan beban secara bertahap dan terus-menerus tanpa menimbulkan kelelahan. Olahraga aerobik bermacam-macam yaitu jalan, *jogging*, lari, bersepeda dan renang. Tubuh memiliki kemampuan untuk mengatur energi dalam menghadapi keadaan darurat yang mungkin timbul, hal ini disebut kebugaran fisik. Sehingga tubuh tetap dapat melakukan aktivitas dan berfungsi secara efektif. Selama melakukan olahraga aerobik ini perlu diperhatikan intensitas waktu latihan yang perlu direncanakan sebelumnya (Palar, 2015).

Overtraining merupakan gejala serius yang harus didiagnosis, meskipun hal tersebut jarang terjadi selama dilakukannya olahraga. Para dokter olahraga sudah sangat paham akan hal tersebut, tetapi dalam hal ini tidak boleh dikacaukan dengan sebuah “sindroma harapan berlebihan” atau “kelelahan olahraga”. Sehingga perlunya pengembangan akan sebuah metode kesehatan tentang pelaksanaan olahraga yang tepat. Hal ini akan memperkecil kemungkinan adanya latihan olahraga yang membahayakan (Giriwijoyo, 2012).

Olahraga memiliki pengaruh yang sangat besar pada sistem sirkulasi dan pernapasan. Kedua hal tersebut selalu berjalan beriringan sebagai respon dari *homeostasis* yang dilakukan saat olahraga tingkat sedang hingga olahraga berat. Olahraga juga mempengaruhi turunnya tekanan darah sementara dibawah normal, hal ini mungkin disebabkan karena adanya penumpukan metabolit sehingga dalam periode singkat pembuluh otot tetap berdilatasi. Setelah beberapa saat tekanan darah akan kembali normal seperti sebelum olahraga, sedangkan untuk denyut jantung lebih lambat kembali normal (Ganong, 2008).

Diagnosa hasil denyut jantung per menit sangat mempengaruhi kondisi fisik maupun mental seseorang. Hal ini dikarenakan perubahan kondisi fisik dan mental sulit diamati. Perubahan ini yang mengakibatkan perubahan frekuensi denyut jantung tiap menitnya. Kecepatan frekuensi jantung dipengaruhi oleh

banyak hal misalnya seperti saat adanya peningkatan suhu, saat terjadinya olahraga misalnya maka akan melemahkan sistem metabolik di jantung. Perubahan suhu sangat mempengaruhi kinerja jantung, sehingga perlu dilakukan pengaturan suhu tubuh saat berolahraga (Meivita, 2016).

Dari penjelasan sebelumnya maka penelitian yang akan dilakukan menggunakan parameter detak jantung dan laju nafas. Kedua parameter tersebut akan menjadi nilai masukan untuk mengukur kondisi fisik seseorang saat berolahraga. Hasil dari pengukuran kedua parameter tersebut yaitu berupa dua kondisi yaitu lanjutkan dan berhenti saat berolahraga.

Pada bidang kedokteran diperlukan adanya teknologi pemantauan kondisi pasien dari jarak jauh tanpa menggunakan kabel. Dari hasil pemantauan tersebut diharapkan adanya sistem yang dapat memantau kondisi secara keseluruhan dan dapat menganalisanya tanpa harus melihat langsung (Patil, 2015). Teknologi yang digunakan untuk pemantauan kondisi pasien menggunakan modul canggih yang dapat memantau pasien kelainan jantung dan gangguan fisik. Sensor yang digunakan pada pemantauan ini menggunakan sensor detak jantung dan sensor suhu. Sensor yang digunakan harus lebih akurat sehingga tidak diperlukan lagi pengukuran menggunakan alat ukur tradisional seperti termometer atau perangkat lainnya (Kale, 2015).

Dari permasalahan diatas maka peneliti terinspirasi membuat penelitian tentang "Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy" yang membahas tentang kondisi tubuh saat bersepeda sehingga dapat meminimalkan cedera yang diakibatkan oleh *overtraining* tersebut dan dapat dipantau dari jarak jauh hasil pengukurannya.

1.2 Perumusan Masalah

Dalam penelitian ini dirumuskan masalah, yaitu :

1. Bagaimana implementasi logika fuzzy pada pengukuran kondisi tubuh saat bersepeda?
2. Bagaimana membuat sistem *telemetry* untuk pengukuran kondisi tubuh?

1.3 Batasan Masalah

Pada penelitian ini, penyusun membuat batasan masalah dalam penelitian pengukuran kondisi tubuh, antara lain :

1. Menggunakan mikrokontroler berupa Arduino Uno dan modul radio *telemetry* 915 Mhz.
2. Penentuan parameter kondisi tubuh berdasarkan laju napas (bpm) dan detak jantung (bpm) tidak mengikutsertakan pada faktor jenis kelamin, tinggi badan, dan berat badan pada usia antara 20 sampai 25 tahun.
3. Subjek yang diuji tidak boleh memiliki riwayat penyakit jantung atau pernapasan.
4. Metode penelitian menggunakan logika fuzzy Mamdani.

1.4 Tujuan Penelitian

Berdasarkan permasalahan yang diuraikan, tujuan dari penelitian ini yaitu:

1. Dapat mengimplementasikan logika fuzzy pada pengukuran kondisi tubuh saat bersepeda.
2. Dapat membuat alat ukur untuk memonitor kondisi tubuh dengan sistem *telemetry*.

1.5 Manfaat Penelitian

Adapun manfaat penelitian yang ingin dicapai dari penelitian ini antara lain :

1. Dapat mencegah sejak dini penurunan kondisi tubuh yang dialami saat bersepeda.
2. Mempermudah pemantauan pengukuran kondisi tubuh saat bersepeda dari jarak jauh.

BAB II. TINJAUAN PUSTAKA

Pada bab ini membahas tentang materi-materi yang berkaitan dengan penelitian yang akan dilakukan seperti parameter-parameter pengukuran, metode yang digunakan, dan komponen-komponen yang digunakan. Bab ini juga membahas permasalahan yang ada untuk dipelajari lebih lanjut.

2.1 Matrik Permasalahan

Pada Tabel 2.1 membahas tentang beberapa penelitian yang telah dilakukan sebelumnya yang bertujuan untuk pembandingan penelitian yang akan dilakukan.

Tabel 2.1 Matrik Permasalahan

No	Judul	Nama/Tahun	Masalah	Metode	Hasil
1	Rancang Bangun Alat Ukur Kondisi Kesehatan Pada Pendaki Gunung Berbasis <i>Fuzzy Logic</i>	Dewi Nurhaji Meivita, dkk (2016)	Belum dapat dimonitor	Menggunakan metode logika Fuzzy	Bagus karena sudah menerapkan metode fuzzy
2	Implementasi Regulator Oksigen Otomatis berdasarkan Tingkat Pernapasan menggunakan Logika Fuzzy	Sugondo Hadiyoso, dkk (2015)	Belum dapat dimonitor	Menggunakan metode logika Fuzzy	Bagus karena sudah menerapkan metode fuzzy
3	<i>Arduino Based Wireless Biomedical Parameter Monitoring System Using Zigbee</i>	Harshavardhan B.Patil, Prof.V.M.Umale (2015)	Hanya melakukan pengukuran parameter	Tidak menggunakan metode apapun dalam pengukuran parameter	Tidak dapat mengetahui kondisi tubuh dari hasil pengukuran tersebut.
4	<i>GSM Based Heart</i>	Aniket V. Kale,	Hanya	Tidak	Tidak dapat

<i>Rate and Temperature Monitoring System</i>	dkk (2015)	melakukan pengukuran parameter	menggunakan metode apapun dalam pengukuran parameter	mengetahui kondisi tubuh dari hasil pengukuran tersebut.
---	------------	--------------------------------	--	--

Dari keempat penelitian yang telah dilakukan maka terlihat bahwa dua penelitian milik Dewi N.M dkk dengan judul rancang bangun alat ukur kondisi kesehatan pada pendaki gunung berbasis *fuzzy logic* dan milik Sugondo Hadiyoso, dkk dengan judul implementasi regulator oksigen otomatis berdasarkan tingkat pernapasan menggunakan logika fuzzy, sudah cukup bagus namun metode yang digunakan masih sederhana dan belum ada sistem pengiriman data sehingga tidak dapat di monitor dari jarak jauh. Sedangkan dua penelitian lainnya yaitu milik Harshavardhan B.Patil, Prof.V.M.Umale, dan Aniket V. Kale, dkk kedua penelitian ini hanya memonitor hasil pengukuran dari parameter tanpa adanya metode yang digunakan pada penelitian tersebut tetapi pada penelitian ini telah dapat dimonitor hasil pengukurannya dari jarak jauh. Sehingga dapat ditarik kesimpulan bahwa perlu adanya sebuah metode untuk penelitian yang akan dilakukan yaitu salah satunya menggunakan metode logika fuzzy dan penerapan sistem pengiriman data yang dapat dimonitor dari jarak jauh.

2.2 Jantung

Ukuran jantung bervariasi sesuai ukuran tubuh. Rata-rata jantung orang dewasa sekitar 5,5 inci (14 sentimeter) panjangnya dan 3,5 inci (9 sentimeter) lebarnya, atau kurang lebih seukuran kepalan tangan. Jantung terletak tepat di atas diafragma antara paru-paru kanan dan kiri. Seperti dari jantung terletak pada sisi kanan dada, sementara dua pertiganya terletak di sisi kiri dada.

Jantung terbagi menjadi dua ruang atas yang disebut atria (tunggal, atrium) dan dua ruang bawah yang disebut ventrikel. Atria adalah ruang penerima, di mana darah dikirim melalui pembuluh besar dan ventrikel adalah ruang pemompa, di mana darah dipompa keluar dari jantung melalui arteri besar.

Jantung manusia berdetak 130 kali per menit pada bayi dan melambat hingga 90 kali per menit pada saat berusia 10 tahun. Pada saat dewasa, jantung

melambat hingga rata-rata hanya 70 kali per menit pada pria dan 78 kali per menit pada wanita. Jantung berdetak kurang lebih 40 juta kali dalam setahun, atau sekitar 3 miliar kali dalam rata-rata kehidupan.

Sekalipun beberapa serangan jantung terjadi secara tiba-tiba, sebagian besar serangan jantung dimulai dengan lambat, dengan sakit ringan. Berikut ini tanda-tanda serangan jantung:

1. Tidak nyaman di dada, biasanya di tengah-tengah dada dan berlangsung lebih dari beberapa menit.
2. Tidak Nyaman di bagian lain dari tubuh bagian atas, seperti salah satu atau Kedua lengan, punggung, leher, rahang atau perut.
3. Napas memendek.
4. Tanda-tanda lain seperti mual, pusing, atau berkeringat dingin.

Olahraga yang teratur meningkatkan jumlah darah yang bisa disemburkan jantung dengan setiap detakan, jadi lebih sedikit detakan per menit yang diperlukan untuk menjaga keluaran jantung. Olahraga bisa meningkatkan keluaran jantung dari 300 hingga 500 persen dan meningkatkan detak jantung hingga 160 detakan per menit. Individu yang berolahraga teratur cenderung memiliki detak jantung istirahat yang lebih rendah (Balaban, 2014).

2.3 Pernapasan

Bernapas atau ventilasi, adalah proses menggerakkan udara masuk dan keluar paru-paru. Kedua tahapannya adalah: 1) penarikan, atau hirupan; 2) pengeluaran, atau penghembusan. Penarikan napas adalah gerakan udara memasuki paru-paru, sementara penghembusan napas adalah gerakan udara keluar paru-paru.

Siklus pernapasan terdiri dari satu tarikan napas diikuti satu hembusan napas. Volume udara yang masuk atau keluar selama satu siklus pernapasan disebut volume tidal. Volume tidal biasanya 500 mililiter, artinya bahwa 500 mililiter udara masuk selama penarikan napas dan jumlah yang sama keluar selama penghembusan.

Seiring dengan bertambahnya usia, pernapasan kita cenderung melambat, sebagaimana seperti yang ditunjukkan Tabel 2.2 . Jenis kelamin juga merupakan faktor dalam kecepatan bernapas.

Tabel 2.2 Jumlah Pernapasan Normal

Usia (tahun)	Napas (per menit)
Bayi	40-60
5	24-60
15	20-22
25 (pria)	14-18
25 (wanita)	16-20

(Sumber: Balaban, 2014)

Sesudah usia 25, kecepatan bernapas merata. Dalam rata-rata hidup, seseorang menghirup lebih dari 600 juta tarikan udara. Rata-rata orang bernapas sekitar 16 kali per menit dan menghirup satu pint udara (0,5 liter) dalam setiap tarikan napas. Rata-rata dalam hidup kita bernapas lebih dari 75 juta galon (2,85 triliyun liter) udara.

Volume menit pernapasan adalah jumlah udara yang bergerak setiap menit. Jumlah tersebut dihitung dengan mengalikan kecepatan pernapasan (rata-rata 12 tarikan per menit) dengan volume tidal (500 mililiter per menit) atau 6.000 mililiter (6 liter) per menit (Balaban, 2014).

2.4 Kelelahan dan *Overtraining*

2.4.1 Kelelahan

Kelelahan dibagi dalam 2 tipe yaitu kelelahan mental dan kelelahan fisik. Memang sulit atau bahkan tidak mungkin memilah secara tegas kedua bentuk kelelahan ini. Konsentrasi mental dan pengendalian emosi adalah dua faktor yang menyertai kejadian kelelahan oleh kerja, yang juga terjadi pada olahraga. Secara umum telah dikenal bahwa kerja otot yang berlebihan dapat merusak mental. Tetapi bahwa kerja mental yang berlebihan dapat mengganggu fungsi otot, merupakan hal yang baru bagi orang kebanyakan.

Pada kerja sedang dan kerja berat, keduanya masih dengan intensitas yang penggunaan O_2 nya di bawah VO_{2max} , sehingga secara fisiologi beban kerjanya masih dalam zona “*normal load*” atau “*submaximal load*”. Artinya beban kerja masih dapat dilakukan dalam kondisi mantap (*steady state*). Pada beban kerja yang masih dapat dilakukan dengan kondisi mantap, perubahan yang terjadi di dalam darah hanyalah sedikit; asam laktat dan cadangan alkali tidak berubah; denyut jantung, volume respirasi dan sistem sirkulasi berubah secara linier sesuai dengan meningkatnya olahdaya (*metabolisme*).

Penyebab utama kelelahan fisik maupun mental haruslah berupa kegiatan yang menggunakan daya (energi), karena tidak akan terjadi kelelahan bila sama sekali tidak ada penggunaan daya. Penyebab-penyebab itu adalah:

- a. Sumber daya habis atau tidak dapat diperoleh
- b. Tertimbunnya sampah olahdaya di dalam tubuh
- c. Terganggunya keseimbangan elektrolit/asam-basa di dalam cairan tubuh
- d. Terganggunya keseimbangan pemasukan dan pengeluaran air di dalam tubuh

Orang yang bekerja berat dengan durasi panjang, kelelahannya dapat ditunda bila selama bekerja ia diberi air minum dengan banyak gula. Sebaliknya orang yang dengan kondisi kekurangan makan/kelaparan, tidak akan mampu bekerja berat dengan durasi yang panjang.

Kelelahan juga dapat terjadi oleh karena terganggunya lingkungan hidup sel. Hal ini dapat terjadi oleh karena terganggunya keseimbangan jumlah air di dalam tubuh atau karena terganggunya penataan keseimbangan garam-garam/elektrolit. Orang yang tersesat di padang pasir tanpa memperoleh air, merupakan contoh dari orang yang kelelahan dan menjadi tidak berdaya oleh kekurangan air, sementara air terus-menerus keluar secara penguapan melalui permukaan kulitnya. Kelelahan yang terjadi karena kehilangan air dan garam ini dapat bersifat ringan sampai kepada ketidakberdayaan (Giriwijoyo, 2012).

2.4.2 Overtraining

Overtraining adalah bentuk kronis dari kelelahan patologis dalam olahraga. Pada tahun 1950 Krestovnikov memasukkannya sebagai bentuk

neurosis olahraga (*sports neurosis*). Keinginan yang berlebihan yaitu melebihi kapasitas fungsional otak adalah pemicu *overtraining* ini. Keinginan berlebihan ini disebabkan oleh:

- a. Proses perangsangan yang berlebihan yang disebabkan oleh volume, intensitas dan kompleksitas latihan dari olahraga tersebut.
- b. Proses penghambatan yang berlebihan dari gerakan-gerakan yang tidak diperlukan pada saat membentuk gerakan-gerakan baru dan halus atau terjadinya pengaruh diferensiasi rangsangan.
- c. Mobilitas proses saraf yang berlebihan atau perubahan-perubahan pada “*stereotype* yang dinamis”.

Semua proses-proses ini dapat menyebabkan terjadinya *exhaustion* dari otak setelah melakukan latihan yang berlebihan secara tersendiri maupun dalam kaitan dengan faktor-faktor lain. Ciri-ciri utamanya adalah kelelahan yang tinggi, respon terhadap latihan yang tidak ekonomis dan tidak seimbang, pemulihan yang lambat sekalipun terhadap latihan yang ringan, labilitas sistem vegetatif dan sistem endokrin, adanya masalah dalam proses olahdaya, dan proses hormonalnya, adanya gejala neurose psikis; hiperaktifitas, depresi atau *euphoria*, *anxietas*, dan menurunnya *output* olahraga.

Overtraining merupakan akibat latihan dengan dosis atau intensitas yang berlebihan yang menyebabkan terjadinya gejala-gejala *overtraining*. Gejala-gejala *overtraining* ini hakikatnya adalah akibat gangguan *homeostasis* karena pemulihan yang tidak kuat. Gejala-gejala *overtraining* meliputi gejala-gejala yang bersifat psikologis, fisiologis dan patologis sebagai berikut:

- a. Insomnia (susah tidur) dan sakit kepala
- b. Sulit memusatkan perhatian (konsentrasi)
- c. Gairah dan motivasi menurun
- d. Lesu, letih dan lemah sehingga menjadi rentan cedera
- e. Rasa lelah > 24 jam
- f. *Anorexia* (mual)
- g. Gangguan fungsi pencernaan – diare
- h. Berat badan menurun

- i. Haus dan banyak minum di malam hari
- j. Tekanan darah menurun dan terjadi *orthostasis*
- k. Nadi istirahat meningkat > 10 denyut dan nadi terhadap standar latihan
- l. Tungkai terasa berat
- m. Dosis latihan tak habis
- n. Nyeri otot dan sendi
- o. Rentan terhadap alergi dan infeksi
- p. Penyembuhan luka lambat
- q. Radang kelenjar getah bening
- r. Hemolisis meningkat sehingga dapat terjadi anemia

Overtraining adalah *polysymptomatik* dengan gejala yang bermacam-macam, sehingga diagnosis yang tepat memerlukan pengertian mengenai metodologi latihan, serta dengan melakukan *anamnesis* yang teliti. *Overtraining* adalah penyakit olahraga dan prevalensi gejala-gejala objektifnya (biasanya muncul lebih lambat dari gejala-gejala subjektifnya) mengacu kepada organ-organ atau sistem tertentu, hanyalah merupakan perwujudan dari penyakitnya dan tidak mengindikasikan patologi lokalnya (Giriwijoyo, 2012).

2.5 Arduino

Arduino didefinisikan sebagai sebuah *platform* elektronik yang *open source* berbasis pada *software* dan *hardware* yang fleksibel dan mudah digunakan. Nama Arduino di sini tidak hanya dipakai untuk menamai *board* rangkaiannya saja, tetapi juga untuk menamai bahasa dan *software* pemrogramannya, serta IDE-nya (*Integrated Development Environment*) atau lingkungan pemrogramannya.

2.5.1 Bagian-Bagian Arduino

Sebenarnya Arduino memiliki berbagai macam versi. Beberapa versi yang cukup banyak tersedia di pasaran lokal, diantara adalah versi Arduino Duemilanove yang disebut sebagai Arduino 2009, versi Arduino Uno yang disebut sebagai Arduino 2010, dan Arduino Mega yang memiliki kapasitas memori dan kaki I/O yang lebih banyak. Pada dasarnya, bagian komponen dari

Arduino Duemilanove dan Arduino Uno sama, hanya berbeda pada IC converter USB ke serialnya. Apabila Arduino Duemilanove IC FTDI, maka Arduino Uno menggunakan IC ATmega8U2 sebagai converter USB ke serialnya.

2.5.2 Contoh Pertama: Led Berkedip

Cara belajar yang paling tepat adalah dengan mempelajari contoh yang diberikan. Software IDE Arduino telah menyediakan begitu banyak contoh program atau sketch, sehingga memudahkan bagi pemula untuk memahami bahasa pemrograman Arduino dengan lebih cepat.

Salah satu contoh sketch yang populer adalah membuat LED berkedip atau *blink* berikut langkah-langkah pembuatannya.

- a. Hubungkan board Arduino dengan port USB computer.
- b. Jalankan software IDE Arduino dengan mengklik dua kali ikon Arduino.
- c. Buka menu **file** dan pilih **examples, basics**, kemudian **blink**.
- d. tekan tombol **verify** untuk mengompilasi sketch menjadi kode mesin, dan tunggu hingga muncul tulisan “**done compiling**”.
- e. Setelah kompilasi selesai, pastikan tipe **board** dan **serial port** di menu **tool**, sesuai dengan yang digunakan.
- f. Tekan tombol **upload** untuk memasukkan kode mesin ke board Arduino, dan tunggu hingga muncul tulisan “**Done Uploading**”.
- g. Setelah proses upload selesai, board Arduino akan menjalankan program. Perhatikan bahwa LED di kaki 13 (berlabel L) sekarang berkedip, yaitu menyala dan padam secara bergantian.

2.5.3 Kaki I/O Arduino

Kebalikan dari output adalah input. Apabila kaki output bertugas mengeluarkan sinyal, maka kaki input bertugas memasukkan sinyal. Tipe input output yang lain, selain digital adalah analog. Analog artinya memiliki lebih dari dua kondisi dalam bentuk sinyal bisa berarti semua sinyal dengan tegangan antara 0 sampai 5V. Baik Arduino Duemilanove maupun Arduino Uno memiliki kaki I/O digital sebanyak 14 kaki, yaitu di kaki D0 sampai D13, dan memiliki 6 kaki I/O analog, yaitu di kaki A0 sampai A5. Khusus untuk kaki-kaki I/O digital, setiap

kakinya bisa dijadikan sebagai input atau output. Namun untuk kaki I/O analog, hanya bisa dijadikan sebagai input saja.

Sebagai input analog, keenam kaki ini memiliki tambahan rangkaian ADC di dalamnya, dengan resolusi ADC sebesar 10 bit. ADC atau Analog to Digital Converter ini adalah mengubah sinyal analog menjadi digital. Resolusi 10 bit artinya untuk tegangan masukan 0 sampai 5V, nilai digital yang dihasilkan memiliki jangkauan nilai dari 0 sampai 1023.

Arduino tidak memiliki kaki output analog. Untuk menghasilkan sinyal analog, digunakan sebuah metode PWM (pulse width modulation), yaitu sebuah cara untuk menghasilkan sinyal analog dari sinyal digital, dengan cara mengatur lebar pulsa sinyal digital pada frekuensi yang tinggi, sehingga rata-rata tegangannya menghasilkan nilai yang bervariasi antara 0 sampai 5V, untuk nilai PWM antara 0 sampai 255. Karena dibuat dari sinyal digital, maka kaki output analog dari PWM ini berada di kaki I/O digital, yang jumlahnya sebanyak 6 kaki, yaitu kaki D3, D5, D6, D9, D10 dan D11.

2.5.4 Bahasa Pemrograman Arduino

Untuk mempelajari bahasa pemrograman Arduino, pembaca dapat membuka **Reference** di menu **help**. Ada tiga bagian utama dalam bahasa pemrograman Arduino, yaitu Struktur, Variabel, Dan Fungsi. Berikut penjelasan masing-masing bagian.

2.5.5 Struktur Program Arduino

Bagian struktur program Arduino ini meliputi kerangka program, sintak program, control aliran program, dan operator. Berikut uraian singkat dari keempat hal tersebut.

2.5.6 Kerangka Program

Kerangka program Arduino sangat sederhana, yaitu terdiri atas dua blok. Blok pertama adalah **void setup()**, dan blok kedua adalah **void loop()**. Berikut keterangan masing-masing:

Blok **void setup()** : berisi kode program yang hanya dijalankan sekali sesaat setelah Arduino dihidupkan atau di-reset. Merupakan bagian persiapan atau inisialisasi program.

Blok **void loop()** : berisi kode program yang akan dijalankan terus menerus. Merupakan tempat program utama.

2.5.7 Sintaks Program

Baik blok **void setup()**, **void loop()** maupun blok **function** harus diberi tanda kurung kurawal buka “{” sebagai tanda awal program di blok itu dan kurung kurawal tutup “}” sebagai tanda akhir program.

- Tanda kurung kurawal buka dan kurung kurawal tutup tersebut juga digunakan pada blok control program, seperti **if**, **if-else**, **for-loop**, **while-loop** dan **do-while-loop**.
- Untuk menandai akhir sebuah baris kode program digunakan tanda titik koma “;”.

2.5.8 Control Aliran Program

Control aliran program ini meliputi instruksi-instruksi yang digunakan untuk membuat percabangan dan perulangan. Instruksi percabangan di antaranya adalah **if**, **if-else**, **switch case**, **break**, **continue**, **return** dan **goto**. Sedangkan instruksi perulangan di antaranya adalah **for-loop**, **while-loop**, **do-while-loop**.

2.5.9 Operator

Berikut ini beberapa operator pada program Arduino:

- Operator aritmatika di Arduino meliputi perkalian (*), pembagian (/), penjumlahan (+), pengurangan (-), dan modulo (%). Modulo adalah perhitungan untuk mendapatkan sisa hasil pembagian.
- Hasil operator aritmatika ini tergantung pada tipe data operand. Contoh, untuk pembagian operand 9 dengan operand 4 akan menghasilkan nilai 2 apabila tipe data kedua operand adalah **int**, atau menghasilkan nilai 2,25 apabila tipe data kedua operand adalah **float**.
- Jika operand-nya berbeda tipe datanya, maka data yang paling besar yang digunakan. Sebagai contoh, apabila operand 9 bertipe data **int** dan operand 4 bertipe data **float**, maka hasil perhitungan akan bertipe **int** atau 2.
- Karena operasi aritmatika tergantung pada tipe data operand-nya, maka pilihlah tipe data operand yang cukup besar untuk menampung hasil

operasinya. Jika tipe data tidak cukup besar, maka hasil operasi bisa menjadi salah karena terjadi *rollover* (nilai datanya berputar kembali).

- Operator perbandingan di Arduino meliputi berikut ini:

```
x == y;      // x sama dengan y
x != y;      // x tidak sama dengan y
x < y;       // x lebih kecil dari y
x > y;       // x lebih besar dari y
x <= y;      // x lebih kecil sama dengan y
x >= y;      // x lebih besar sama dengan y
```

- Operator Boolean di Arduino meliputi logika AND (&&), OR (||), dan NOT (!), dengan penggunaan seperti contoh berikut ini:

```
If (x > 0 && x < 5) // logika AND
If (x > 0 || y > 0) // logika OR
If (!x > 0)         // logika NOT
```

- Untuk melakukan operasi perhitungan pada level bit (biner) dari sebuah variabel, gunakan operator-operator *Bitwise* berikut ini:

```
&    (bitwise and)
|    (bitwise or)
^    (bitwise xor)
~    (bitwise not)
<<   (bitshift left)
>>   (bitshift right)
```

- Untuk mempersingkat penulisan instruksi, pembaca dapat menggunakan operator *Compound* berikut:

```
x++;      // sama seperti x=x+1
x--;      // sama seperti x=x-1
x += y ;  // sama seperti x= x+y
x -= y;   // sama seperti x=x-y
x *= y;   // sama seperti x=x*y
x /= y;   // sama seperti x=x/y
x &= y;   // sama seperti x=x&y
```

$x | = y$; // sama seperti $x=x|y$

(Artanto, 2012).

2.6 Fuzzy Logic

2.6.1 Manajemen Ketidakpastian Dengan Logika Fuzzy

Seperti yang telah diketahui, Prof. Lotfi Zadeh merupakan pencetus pertama yang memperkenalkan pengolahan ketidakpastian yang sekarang dikenal sebagai *fuzzy logic* tersebut. Beliau merupakan orang Iran yang menjadi guru besar di *University of California*. Awalnya harus ad aide tentang penyajian bentuk fuzzy. Bentuk penyajian harus dibuat sederhana agar mudah dalam komputasinya tetapi harus tetap menggambarkan ke-fuzzy-an.

Untuk menyajikan bilangan riil (nyata) ke dalam himpunan fuzzy akan ada banyak kemungkinan. Oleh karena itu perlu penegasan anggota yang masuk dalam kriteria dan tidak. Maka tercetuslah ide untuk membuat derajat keanggotaan (*grade of membership*) yang ditentukan dari masing-masing anggota. Pembuatan fungsi keanggotaan (*membership function*) diperlukan untuk penyajian himpunan fuzzy tersebut. Dengan penyajian fungsi keanggotaan maka penyajian suatu variabel akan lebih alamiah.

2.6.2 Himpunan Fuzzy

Dalam suatu himpunan fuzzy (*fuzzy set*) nilai keanggotaan suatu item A dalam sebuah semesta pembicaraan (*universe of discourse*) Z dapat disebut sebagai fungsi keanggotaan (*membership function*) berupa μ_A , yang nilainya sekitar [0,1]. Sehingga dapat dirumuskan sebagai berikut:

$$\mu_A : Z \rightarrow [0,1]$$

Nilai keanggotaan item A dalam sebuah semesta pembicaraan Z merupakan elemen-elemen yang berpasangan yaitu z (z anggota Z) dan derajat keanggotaan (*grade of membership*) dapat dirumuskan sebagai berikut:

$$A = \{ (z, \mu_A(z) / z \in Z) \}$$

tanda '/' merupakan penghubung elemen dengan derajat keanggotaannya. Jika Z adalah diskrit maka A bisa dinyatakan dengan :

$$A = \mu_A(z_1) / z_1 + \dots + \mu_A(z_n) / z_n$$

atau

$$A = \sum_{i=1}^n \mu_A(z_i) / z_i$$

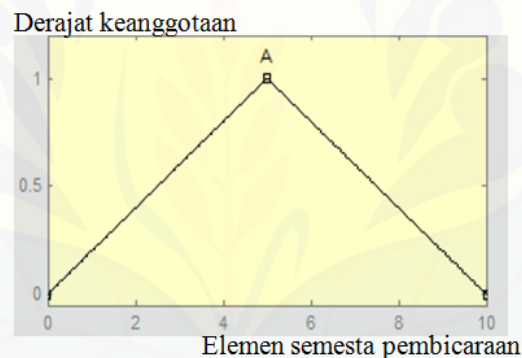
dan jika Z adalah kontinu, maka himpunan fuzzy bisa dinyatakan dengan:

$$A = \int_Z \mu_A(z) / z$$

tanda '+', ' \sum ', dan ' \int ' menyatakan operator union (gabungan). Sebagai contoh, untuk semesta pembicaraan 'bilangan cacah yang kurang dari 10' dan himpunan fuzzy A yang didefinisikan sebagai 'bilangan cacah yang dekat dengan 5', dinyatakan:

$$A = 0/0 + 0.2/1 + 0.4/2 + 0.6/3 + 0.8/4 + 1.0/5 + 0.8/6 + 0.6/7 + 0.4/8 + 0.2/9$$

dapat dipresentasikan dalam bentuk diagram dari himpunan fuzzy A pada Gambar 2.1.



Gambar 2.1 Definisi Himpunan Fuzzy A

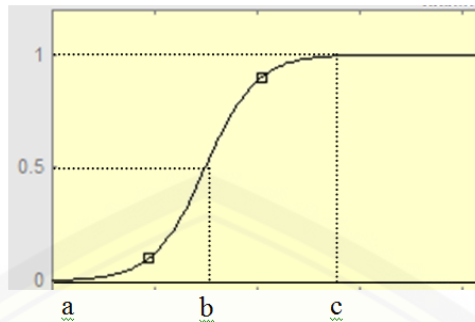
Fuzzyfikasi (*fuzzification*) merupakan proses penentuan nilai masukan derajat keanggotaan dari suatu variabel numerik non-fuzzy menjadi himpunan fuzzy. Penentuan tersebut tidak dibatasi oleh aturan-aturan tertentu. Berikut contoh tiga macam fungsi keanggotaan yang dinyatakan sebagai S , π , dan T (*triangular*).

1. *S-function*

Definisi *S-function* adalah sebagai berikut:

$$S(z; a, b, c) = \begin{cases} 0 & ; z < a \\ 2 \left(\frac{z-a}{c-a} \right) & ; a \leq z \leq b \\ 1 - 2 \left(\frac{z-a}{c-a} \right) & ; b \leq z \leq c \\ 1 & ; z > c \end{cases}$$

Bentuk diagramatik *S-function* ditunjukkan pada Gambar 2.2.



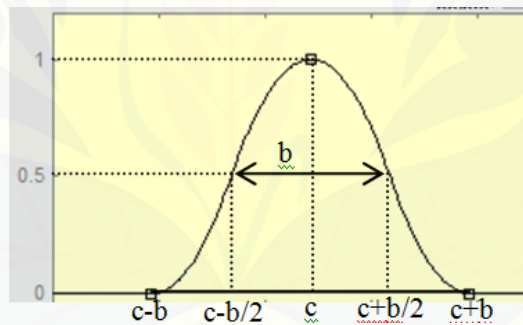
Gambar 2.2 *S-function*

2. π -function

Definisi π -function adalah sebagai berikut:

$$\pi(z; b, c) = \begin{cases} S(z; c - b, c - \frac{b}{2}, c) & ; z \leq c \\ 1 - S(z; c, c + \frac{b}{2}, c + b) & ; z \geq c \end{cases}$$

Bentuk diagramatik π -function ditunjukkan pada Gambar 2.3.



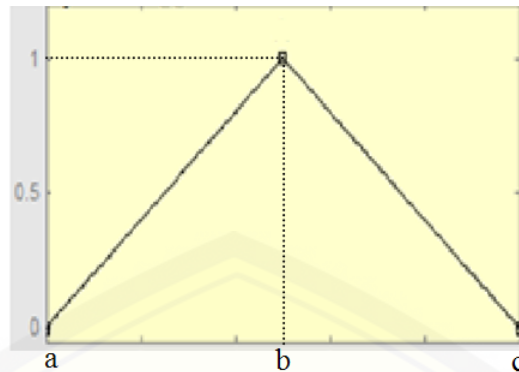
Gambar 2.3 π -function

3. *T-function*

Definisi *T-function* adalah sebagai berikut:

$$T(z; a, b, c) = \begin{cases} 0 & ; z < a \\ \frac{z-a}{b-a} & ; a \leq z \leq b \\ \frac{c-z}{c-b} & ; b \leq z \leq c \\ 0 & ; z > c \end{cases}$$

Bentuk diagramatik *T-function* ditunjukkan pada Gambar 2.4.



Gambar 2.4 T-function

2.6.3 Operasi Himpunan Fuzzy

Dasar operasi himpunan fuzzy pada himpunan fuzzy A dan B pada semesta pembicaraan Z antara lain:

1. *Equality*

$$\mu_A(z) = \mu_B(z), z \in Z$$

2. *Union*

$$\mu_{(A \cup B)}(z) = \max \{ \mu_A(z), \mu_B(z) \}, z \in Z$$

3. *Intersection*

$$\mu_{(A \cap B)}(z) = \min \{ \mu_A(z), \mu_B(z) \}, z \in Z$$

4. *Complement*

$$\mu_{\bar{A}}(z) = 1 - \mu_A(z), z \in Z$$

Jika himpunan A dan \bar{A} adalah komplemen maka berlaku aturan sebagai berikut:

$$\mu_{(A \cap \bar{A})}(z) = \min \{ \mu_A(z), \mu_{\bar{A}}(z) \} \leq 0,5$$

$$\mu_{(A \cup \bar{A})}(z) = \max \{ \mu_A(z), \mu_{\bar{A}}(z) \} \geq 0,5$$

5. *Normalization*

$$\mu_{\text{NORMAL}(A)}(z) = \mu_A(z) / \max(\mu_A(z)), z \in Z$$

6. *Concentration*

$$\mu_{\text{CONV}(A)}(z) = (\mu_A(z))^2, z \in Z$$

7. *Dilation*

$$\mu_{\text{DIL}(A)}(z) = (\mu_A(z))^{0,5}, z \in Z$$

8. *Algebraic product*

$$\mu_{(A \cdot B)}(z) = \mu_A(z) \cdot \mu_B(z), z \in Z$$

9. *Bounded sum*

$$\mu_{(A \oplus B)}(z) = \min \{1, \mu_A(z) + \mu_B(z)\}, z \in Z$$

10. *Bounded product*

$$\mu_{(A \otimes B)}(z) = \max \{0, \mu_A(z) + \mu_B(z) - 1\}, z \in Z$$

11. *Intensification*

$$\mu_{\text{INT}(A)}(z) = \begin{cases} 2(\mu_A(z)) & ; 0 \leq \mu_A(z) < 0,5 \\ 1 - 2(1 - \mu_A(z))^2 & ; 0,5 \leq \mu_A(z) \leq 1 \end{cases}$$

12. *Drastic product*

$$\mu_A(z) \otimes \mu_B(z) = \begin{cases} \mu_A(z) & ; \mu_B(z) = 1 \\ \mu_B(z) & ; \mu_A(z) = 1 \\ 0 & ; \mu_A(z), \mu_B(z) < 1 \end{cases}$$

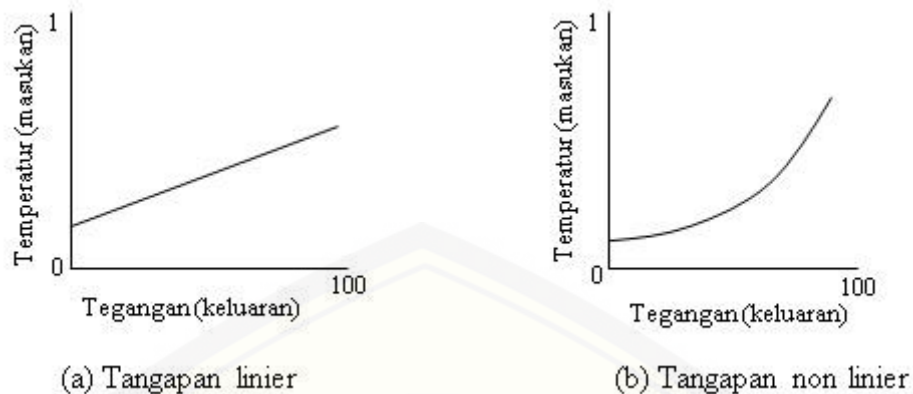
Variabel fuzzy bisa dikatakan sebagai variabel linguistik dalam penjelasan kali ini didefinisikan sebagai z . Himpunan pada z memiliki masing-masing fungsi keanggotaan normal (memiliki harga maksimum sama dengan 1) dan *convex* pada Z . R (*rule*) adalah aturan sintatik yang menghasilkan nilai-nilai pada z dan S (semesta) adalah aturan yang menghubungkan tiap nilai dengan artinya (Kuswandi, 2007).

2.7 Sensor

Sensor merupakan komponen yang dapat mengetahui suatu kondisi tertentu seperti perubahan energi (listrik, fisika, kimia, biologi, mekanik dan sebagainya). Untuk pemilihan kualitas sensor perlu diperhatikan hal-hal berikut ini:

a. Linieritas

Sinyal keluaran yang dihasilkan oleh sensor kebanyakan berubah menjadi kontinyu sebagai tanggapan terhadap masukan yang berubah secara kontinyu. Sebagai contoh, panas yang mempengaruhi besar tegangan yang dihasilkan. Dalam hal ini maka dapat dianalisa perbandingan keluaran dengan masukan berupa sebuah grafik.



Gambar 2.7 Linieritas Sensor

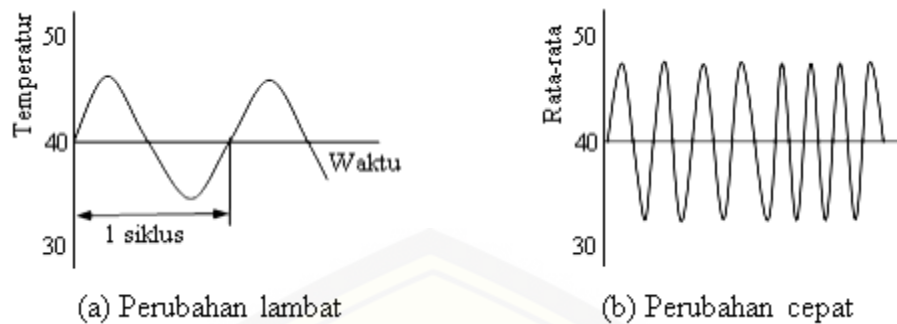
(Sumber: Daryanto, 2011)

b. Sensitivitas

Sensitivitas merupakan ukuran seberapa besar kepekaan sensor tersebut terhadap kuantitas yang diukur. Sensitivitas bisa juga dinyatakan dengan bilangan yang menyebutkan “perbandingan yang dihasilkan oleh perubahan antara keluaran dan masukan”. Beberapa sensor panas kepekaannya dinyatakan dalam satuan “satu volt per derajat”, yang berarti kenaikan satu volt adalah kenaikan satu derajat pada keluarannya. Ada juga sensor panas yang memiliki kepekaan “dua volt per derajat”, sensor jenis ini lebih peka. Sensitivitas juga dipengaruhi oleh linieritas sensor. Semakin linier tanggapan sensor maka semakin sensitif pula pengukurannya.

c. Tanggapan Waktu

Tanggapan waktu pada sensor merupakan seberapa cepat tanggapannya saat ada perubahan masukan. Contoh tanggapan yang jelek yaitu thermometer merkuri. Temperature adalah masukan dan posisi merkuri adalah keluaran. Frekuensi adalah jumlah siklus dalam satu detik dan dinyatakan dalam satuan Hertz (Hz). Frekuensi dianggap rendah jika perubahan masukan lambat (Daryanto, 2011).



Gambar 2.8 Tanggapan Waktu Sensor

(Sumber: Daryanto, 2011)

2.7.1 Sensor Detak Jantung (*Pulse Sensor*)

Saat ini diperlukan penelitian menggunakan teknik optik pada pembuluh darah halus yang menempel pada permukaan kulit yaitu misalnya ujung jari, cuping telinga dll. Pada permukaan kulit tersebut suplai darah bagus karena mengalir bergantian tepat waktu seiring dengan detak jantung. Penggunaan cahaya LED atau semacamnya bisa mendeteksi perubahan aliran darah tersebut meskipun terdapat variasi warna kulit. Ini menggunakan prinsip transmisi dan reflektansi. Ini adalah metode non-invasif untuk menemukan detak jantung pada tubuh (Mallick, 2016).

Detak jantung dapat diperoleh dengan menggabungkan LED dengan pendeteksi cahaya (LDR). Bentuk pulsa yang didapatkan dari cahaya yang dipantulkan di tangan akan diterima oleh LDR dan diproses pada mikrokontroler. Sinyal tersebut dikirimkan oleh pemancar dengan format digital. Rangkaian ini menggunakan pengkodean untuk menghindari jeda antara satu dan nol. Perhitungan detak jantung dengan menghitung waktu yang dibutuhkan dari pulsa pertama hingga pulsa kedua (Singh, 2013).

2.7.2 Sensor Laju Nafas (*Airflow Sensor*)

Airflow sensor adalah sensor yang digunakan untuk mendeteksi adanya pernapasan baik saat fase inspirasi maupun ekspirasi. Sensor ini dipasang pada sekitaran hitung pasien dengan tetap memperhatikan kenyamanan penggunanya. Untuk mengetahui ada tidaknya pernapasan yaitu dengan mendeteksi perubahan suhu yang ada disekitan lubang hidung karena pada saat ekspirasi, suhu

meningkat jika dibandingkan pada saat fasa inspirasi. Oleh karena itu, Airflow sensor dibuat dari bahan yang peka terhadap perubahan suhu yaitu menggunakan resistor yang nilai resistansinya berubah terhadap suhu. Selanjutnya dengan menambahkan rangkaian pembagi tegangan dan pengkondisi sinyal berupa penguat dan filter sinyal keluaran disesuaikan dengan kebutuhan. Keluaran blok sensor ini berupa tegangan analog yang nantinya di konversi menjadi data digital oleh mikrokontroler (Hadiyoso, 2015).

2.8 Dasar-Dasar Radio

Radiasi elektromagnetik meliputi gelombang radio, gelombang mikro (*microwaves*), radiasi inframerah, cahaya yang terlihat, gelombang ultraviolet, sinar x, dan sinar gamma. Semua radiasi diatas memiliki persamaan, yaitu semua membentuk spektrum elektromagnetik dan bergerak dengan kecepatan cahaya. Yang membedakan adalah panjang gelombang. Panjang gelombang adalah jarak yang ditempuh sebuah gelombang untuk membentuk satu siklus yang sempurna. Panjang gelombang terkait dengan jumlah energi yang dibawa oleh gelombang. Semakin pendek panjang gelombang, semakin besar energinya.

2.8.1 Gelombang radio

Gelombang radio merambat seperti dipermukaan air. Gelombang radio merambat dari permukaan bumi dan merambat ke angkasa dengan berbagai arah (membentuk sudut) relative terhadap permukaan bumi. Ketika gelombang radio merambat, energy dipancarkan ke segala arah. Ada dua komponen gelombang radio, yaitu puncak dan lembah. Kedua komponen ini bergerak menjauhi radio satu-persatu secara berurutan dengan kecepatan yang konsisten. Jarak antara puncak gelombang inilah yang disebut panjang gelombang dan biasanya diwakili dengan lambang berupa huruf kecil bahasa Yunani, Lamda (λ).

2.8.2 Frekuensi

Gelombang radio memancarkan sinyal-sinyal radio. Gelombang ini mempunyai panjang gelombang dari *range* yang sangat kecil (kurang dari 1 cm) sampai puluhan hingga ratusan meter. Frekuensi gelombang radio sama dengan

jumlah siklus yang sempurna (komplit) yang terjadi dalam satu detik. Semakin lama waktu siklusnya, semakin panjang gelombang dan semakin pendek frekuensinya. Semakin pendek siklus waktunya, semakin pendek panjang gelombangnya dan semakin tinggi frekuensinya.

Satuan frekuensi dinyatakan dalam Hertz (Hz). Biasanya diukur dan dinyatakan dalam ribuan Hertz (kilohertz (kHz)), jutaan Hertz (MHz), bahkan dalam milyaran Hertz (GHz).

1 Hz = 1 siklus per detik

1 kHz = 1000 siklus per detik

1 MHz = 1.000.000 siklus per detik

1 GHz = 1000.000.000 siklus per detik

Untuk alasan praktis, kecepatan gelombang radio dinyatakan dengan sebuah angka yang konstan, sesuai dengan frekuensi atau amplitudo gelombang yang dipancarkan. Untuk mencari frekuensi ketika diketahui panjang gelombang, bagilah kecepatan dengan panjang gelombang.

$$\text{Frekuensi (Hertz)} = \frac{300.000.000 \text{ (m/detik)}}{\text{panjang gelombang (m)}}$$

Untuk mencari panjang gelombang ketika diketahui frekuensi adalah dengan membagi kecepatan dengan frekuensi.

$$\text{Panjang gelombang (m)} = \frac{300.000.000 \text{ (m/detik)}}{\text{Frekuensi (Hertz)}}$$

(Wowok, 2008).

BAB 3 METODOLOGI

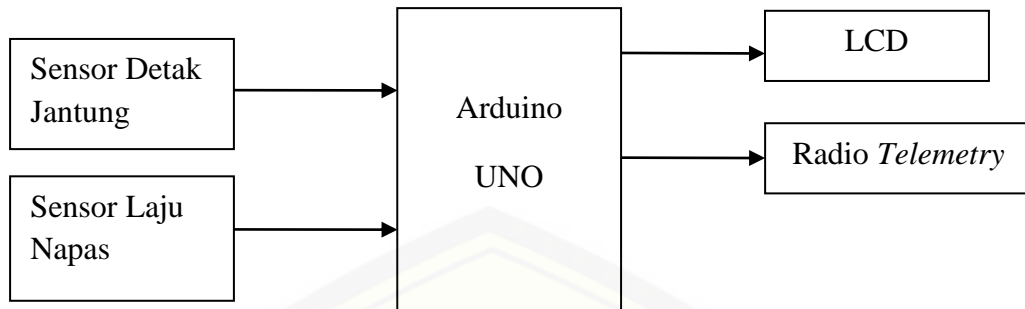
Bab ini membahas tentang hal yang berkaitan dengan penelitian yang akan dilakukan, seperti waktu dan tempat pembuatan alat, tahap perancangan, *flowchart*, serta langkah-langkah dalam pengambilan data dan manajemen penelitian di lapangan. Dalam penelitian ini dirancang sebuah alat pendeteksi yang dapat memonitor kondisi tubuh. Alat yang akan dibuat ini menggunakan frekuensi radio untuk pengiriman data hasil kondisi kesehatan tersebut dari jarak jauh sehingga tanpa menggunakan kabel (*wireless*) sehingga kondisi tubuh berupa detak jantung dan laju napas dapat dimonitor secara langsung (*real time*).

Pembuatan skripsi dengan judul "Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy" ini dimulai pada bulan September 2017. Pembuatan skripsi ini dalam pembuatan *hardware* maupun *software* dilakukan di Laboratorium Elektronika Terapan, Jurusan Teknik Elektro Fakultas Teknik, Universitas Jember di Jalan Slamet Riyadi nomor 62 Patrang, Jember.

3.1 Perancangan Sistem

3.1.1 Perancangan *Hardware*

Pada alat yang telah dibuat merupakan alat yang dapat mendeteksi kelelahan manusia berdasarkan dua tanda vital yang diukur. Untuk mengetahui kelelahan tersebut dilakukan pengukuran tanda vital berupa detak jantung dan laju napas manusia. Dari hasil pengukuran tanda vital tersebut maka dapat diketahui kondisi tubuh saat berolahraga seperti bersepeda untuk mencegah terjadinya *overtraining*. Kondisi tubuh tersebut diolah dengan sebuah metode yaitu logika fuzzy. Logika fuzzy dapat menyimpulkan hasil pengukuran kondisi tubuh saat lelah. Dengan adanya logika fuzzy alat akan memberitahu untuk dapat melanjutkan olahraga atau harus berhenti.



Gambar 3.1 Blok Diagram Sistem

Pada proses pembuatan skripsi ini penyusun menggunakan rangkaian sensor detak jantung dan sensor laju napas, kedua sensor tersebut sebagai masukan ke mikrokontroler berupa Arduino UNO, kemudian keluaran dari mikrokontroler ditampilkan pada LCD dan data dikirim melalui frekuensi radio untuk dapat diterima dan dibaca di laptop maupun *handphone (support OTG)*.

Prinsip kerja alat ini yaitu mengukur parameter dari detak jantung dan laju napas manusia dalam satu menit yang kemudian diproses oleh Arduino untuk menentukan kesimpulan kondisi tubuh menggunakan fuzzy. Hasil pengukuran parameter tersebut dan hasil kesimpulan ditampilkan pada LCD dan disimpan pada *SD card*, hasil pengukuran tersebut dapat dimonitor dari jarak jauh dengan menggunakan *radio telemetry*.

3.1.2 Alat dan Bahan

Pada tahap ini diperlukan beberapa komponen yang dibutuhkan selama pembuatan alat yaitu dibedakan menjadi perangkat keras, perangkat lunak dan bahan pendukung:

Perangkat Keras:

- a. Sensor detak jantung (*pulse sensor*)
- b. Sensor laju napas (MPX2010DP)
- c. *Radio Telemetry* (3DR V2 915 MHz)
- d. Arduino Uno
- e. ECG monitor
- f. LCD 16x2

- g. Laptop
- h. Avometer
- i. Oximeter
- j. *USB downloader*
- k. Adaptor 9V
- l. *Micro SD card*
- m. *Slot Memory*
- n. DC-DC XL6009E1 (*step up radio*)
- o. Modul I2C

Perangkat lunak:

- a. Matlab 2013
- b. IDE Arduino 1.8.3
- c. *Serial USB Terminal*

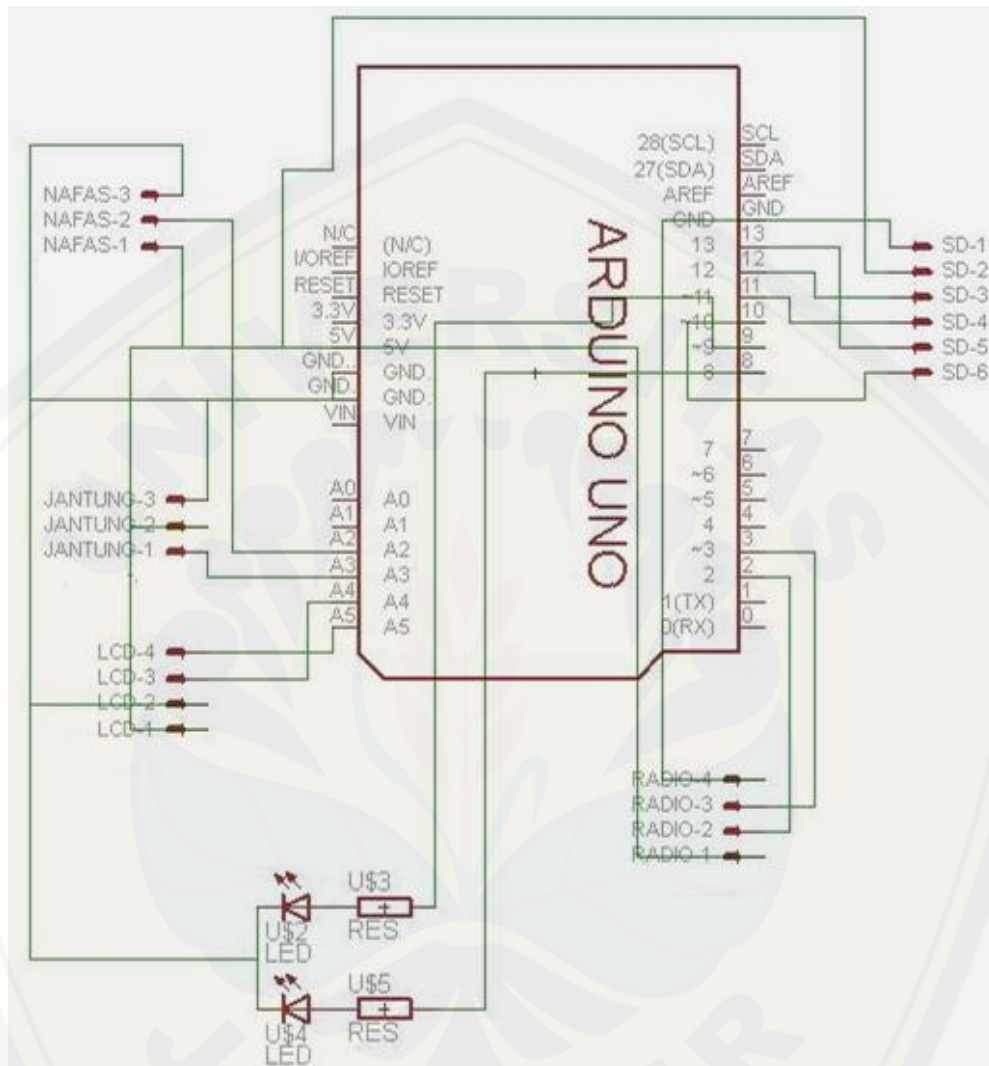
Bahan pendukung:

- a. PCB polos
- b. Larutan *Ferri Clorida*
- c. Akrilik
- d. Kabel
- e. Timah
- f. Baut dan mur
- g. Lem
- h. Setrika
- i. Solder
- j. Tang potong
- k. Bor
- l. Pinset
- m. Obeng

3.1.3 Skema Perancangan Alat

Pada desain rancangan alat ini merupakan hasil rancangan alat yang telah dibuat. Hasil rancangan tersebut merupakan hasil rangkaian beberapa komponen utama yaitu sensor detak jantung, sensor laju napas, LCD, *SD card* dan radio

telemetry yang semua dihubungkan pada pin-pin di Arduino. Semua komponen menggunakan tegangan *input* sebesar 5V.



Gambar 3.2 Skema Perancangan Alat

Pada Tabel 3.1 akan dijelaskan penggunaan pin-pin pada Arduino untuk pembuatan alat pencegah *overtraining* yang telah dibuat. Penggunaan pin-pin tersebut sesuai pada Gambar 3.2. Pin yang digunakan merupakan pin analog dan digital sesuai kebutuhan komponen. Sedangkan tegangan yang digunakan semua menggunakan tegangan sumber sebesar 5 volt.

Tabel 3.1 Koneksi pin Arduino

No.	Pin	Koneksi
1	GND	<i>Ground</i>
2	5V	Tegangan 5V
3	A2	Pin analog A2
4	A3	Pin analog A3
5	A4	Pin analog A4
6	A5	Pin analog A5
7	2	Pin digital 2
8	3	Pin digital 3
9	8	Pin digital 8
10	9	Pin digital 9
11	10	Pin digital 10
12	11	Pin digital 11
13	12	Pin digital 12
14	13	Pin digital 13

Fungsi dari pin-pin yang digunakan yaitu pertama pin *ground* merupakan pin untuk mencegah kerusakan jika ada tegangan yang tidak diinginkan. Ke dua pin 5V yaitu pin untuk mensuplai tegangan sebesar 5 Volt. Ke tiga yaitu pin A2 dan A3 yang merupakan pin analog yang digunakan oleh sensor untuk *input* data karena sensor merupakan sensor analog. Ke empat pin A4 (SDA) dan A5 (SCL) merupakan pin yang digunakan pada komunikasi LCD menggunakan *wire library*. Ke lima pin 2 dan 3 yaitu pin yang digunakan untuk mengaktifkan program *interrupt*. Ke enam pin 8 dan 9 yaitu pin yang digunakan sebagai *output* indikator LED untuk detak jantung dan laju napas. Ke tujuh pin 10 (SS), 11 (MOSI), 12 (MISO), dan 13 (SCK) mendukung komunikasi SPI dengan menggunakan SPI *library*.

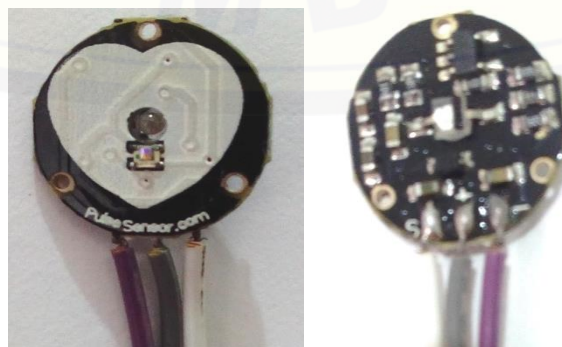
3.2 Perancangan Sensor

3.2.1 Sensor Detak Jantung (*Pulse Sensor*)

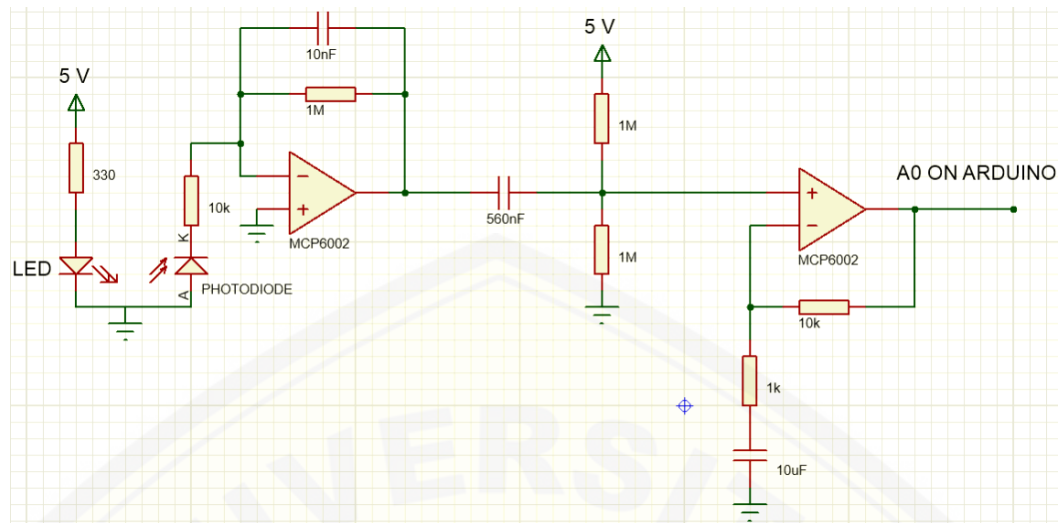
Sensor detak jantung merupakan sensor yang dapat membaca jumlah denyut pada jantung tiap menitnya dengan satuan BPM (*beats per minute*). Sensor detak jantung ini menggunakan *pulse sensor* yang telah mendukung Arduino dengan satuan BPM (*beats per minute*). Sensor ini merupakan sensor yang diletakkan di jari dengan membaca sinyal yang telah dikuatkan amplitudonya lalu kembali ke titik referensi. Sensor ini berbentuk sebuah komponen kecil yang terdiri dari LED pemantul cahaya dan sensor penerima pantulan (*photodiode*) serta dibaliknya terdapat rangkaian penguat amplitudo. Sensor ini bekerja pada tegangan DC sebesar 5V, *output* tegangannya maksimal 5V dan pada arus 100 mA. Sensor ini merupakan sensor analog. Terdapat tiga pin pada sensor ini yaitu +5V, *output* dan *ground*. Berikut hasil sampling sensor menggunakan *fingertip oximeter*.

Tabel 3.2 *Sampling* Pengukuran Sensor Detak Jantung

No	<i>oximeter</i> (BPM)	detak jantung sensor (BPM)
1	Subjek 1	Subjek 1
2	Subjek 2	Subjek 2
3	Subjek 3	Subjek 3
4	Subjek 4	Subjek 4
5	Subjek 5	Subjek 5

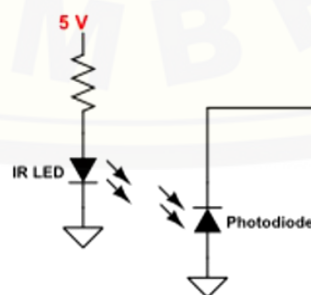


Gambar 3.3 Sensor Detak Jantung

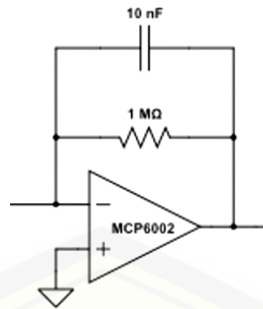
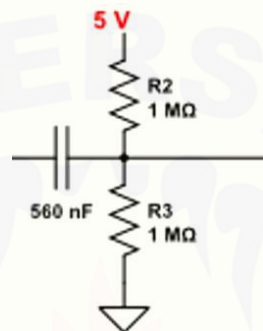
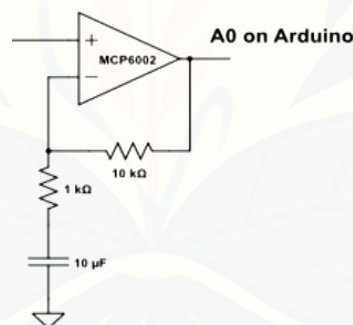


Gambar 3.4 Rangkaian Sensor Detak Jantung

Pada Gambar 3.4 merupakan rangkaian yang ada dibalik sensor detak jantung. Rangkaian tersebut merupakan kombinasi dari rangkaian *finger cuff*, *transimpedance stage*, *high-pass filter*, and *AC gain-stage*. Rangkaian *finger cuff* merupakan rangkaian *input* yaitu terdiri LED dan fotodiode yang membaca plasma darah pada permukaan kulit. Rangkaian *transimpedance stage* merupakan rangkaian konversi dari arus ke tegangan, arus dihasilkan dari fotodiode yang sangat kecil sehingga diperlukan resistor besar ($1\text{ M}\Omega$) sebagai *gain-setting*. Sedangkan kapasitor berfungsi mengurangi frekuensi tinggi. Rangkaian *high-pass filter* merupakan rangkaian yang hanya meloloskan sinyal dengan frekuensi tinggi yang bertujuan untuk pembacaan denyut. Rangkaian *AC gain-stage* merupakan rangkaian *non-inverting*, kapasitor pada rangkaian hanya membiarkan sinyal AC untuk diloloskan. Sinyal AC inilah yang akan dikonversi dari analog ke digital.



Gambar 3.5 Rangkaian *Finger Cuff*

Gambar 3.6 Rangkaian *Transimpedance Stage*Gambar 3.7 Rangkaian *High-Pass Filter*Gambar 3.8 Rangkaian *AC Gain-Stage*

3.2.2 Sensor Laju Napas (MPX2010DP)

Sensor laju napas merupakan sensor yang dapat membaca jumlah respirasi pada paru-paru tiap menitnya dengan satuan BPM (*breaths per minute*). Sensor laju nafas ini dibuat menggunakan sensor tekanan tipe MPX2010DP dengan satuan berupa BPM (*breath per minute*). Sensor ini bekerja pada tegangan DC 5V dengan arus 7 mA. Data yang dikeluarkan berupa data analog. Sensor tekanan ini memiliki 2 *input* yaitu tekanan udara masuk dan tekanan udara keluar. Udara masuk menggunakan tekanan udara pada suntikan yang telah diatur tekanannya dan udara keluar merupakan tarikan napas yang diambil melalui masker oksigen

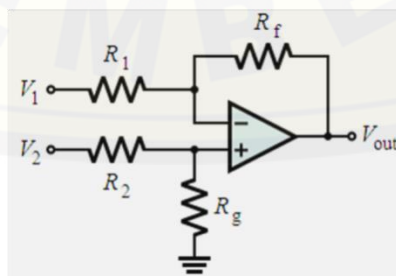
yang diletakkan dihidung. Dimana perbedaan nilai dari udara masuk dan keluar tersebut akan menjadi nilai yang dihitung sebagai perhitungan adanya laju napas. Sehingga jika nilai udara keluar melebihi nilai batas udara masuk (324 dalam nilai ADC) maka dianggap ada laju napas dan nilai tersebut langsung dikonversikan dalam 1 menit. Berikut hasil sampling sensor menggunakan yang dilakukan secara manual.

Tabel 3.3 *Sampling* Pengukuran Sensor Laju Napas

No	Manual (BPM)	Sensor (BPM)
1	Subjek 1	Subjek 1
2	Subjek 2	Subjek 2
3	Subjek 3	Subjek 3
4	Subjek 4	Subjek 4
5	Subjek 5	Subjek 5



Gambar 3.9 Sensor Laju Napas (MPX2010DP)



Gambar 3.10 Rangkaian Penguat Diferensial

Fungsi adanya rangkaian penguat diferensial pada sensor laju napas yang terbuat dari sensor tekanan yaitu mengkonversi variabel dinamik ke dalam sinyal elektrik. Rangkaian ini menggunakan rangkaian *inverting*.

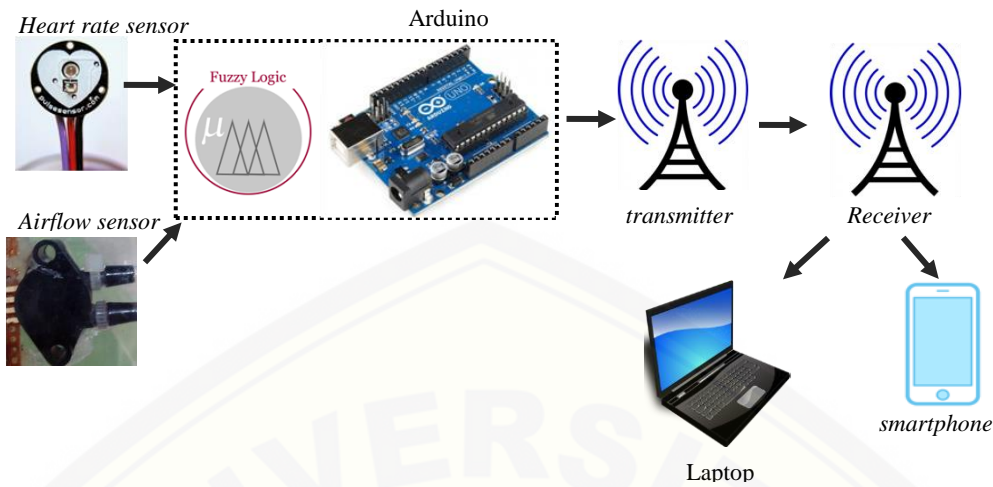
3.3 Perancangan *Output Sistem Telemetry*

Hasil keluaran pada sistem akan ditampilkan di LCD dengan ukuran 16x2. Hasil pengukuran yang ditampilkan yaitu pada baris pertama terdapat hasil pengukuran laju napas yang disingkat menjadi LN dan hasil pengukuran detak jantung yang disingkat menjadi DJ. Pada baris kedua ada hasil kondisi dari *output fuzzy* berupa kondisi lanjut atau berhenti dan nilai pengaturan *threshold* dari sensor laju napas.



Gambar 3.11 Tampilan LCD

Selain tampilan pada LCD adapula hasil *output* yang dikirimkan melalui frekuensi radio (*telemetry*). Data yang dikirim secara *telemetry* yaitu pengukuran laju napas, detak jantung, dan *output fuzzy* saja. Hasil pengukuran ini ditampilkan pada komputer yang *support* dan ter-*instal* Arduino karena radio yang digunakan juga radio yang *support* Arduino. Jika pada Arduino untuk menampilkan hasil *output* dengan cara ditampilkan pada serial monitor yang telah tersedia pada *software tersebut*. Untuk tampilan pada lainnya bisa ditampilkan pada *smartphone* yang telah *support OTG*. Pada *smartphone* diperlukan *software serial USB terminal* dengan kecepatan pengiriman data disesuaikan dengan program pada Arduino yaitu 57600 *baudrate*.



Gambar 3.12 Skema Perancangan Sistem *Telemetry*

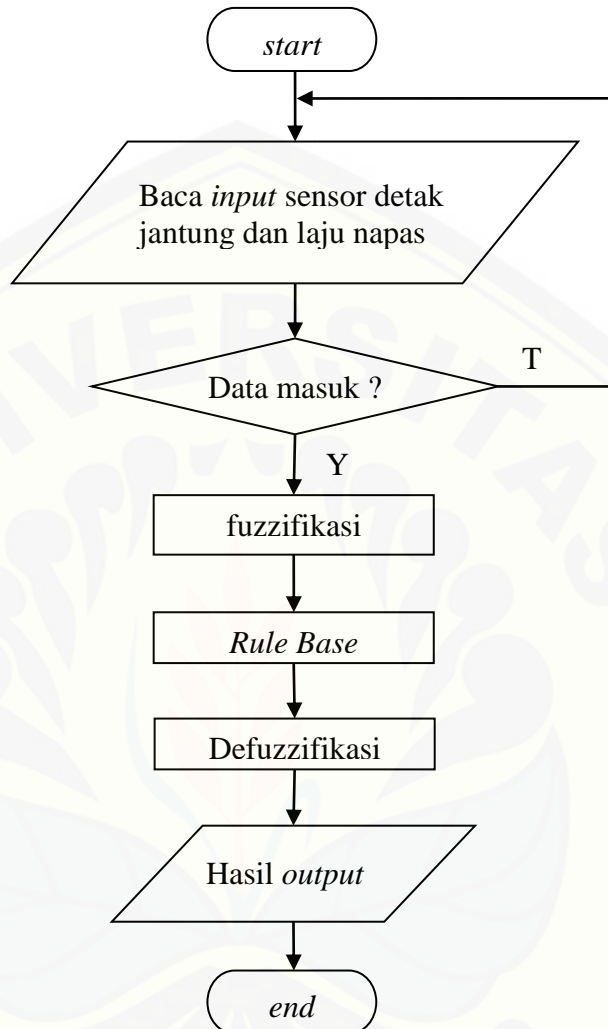
3.4 Perancangan Fuzzy

3.4.1 Desain Fuzzy

Untuk mendesain fuzzy pengukuran kondisi tubuh saat bersepeda diperlukan pemilihan parameter yang akan diuji. Pemilihan parameter ini telah dikonsultasikan dengan dokter yang ahli pada bidang fisiologi yaitu dr. Aris Prasetyo, M.kes. Parameter yang terpilih sebagai *input* yaitu detak jantung dan laju napas manusia, sedangkan *output* yang dihasilkan berupa kondisi lanjut dan berhenti.

Sebelum membahas tentang fuzzy lebih lanjut, maka dijelaskan terlebih dahulu langkah-langkah proses pembuatan logika fuzzy. Pertama proses fuzzifikasi yaitu mengumpulkan informasi data detak jantung dan laju napas. Dari data tersebut kemudian dikategorikan menjadi masing-masing himpunan. Untuk detak jantung terdapat 5 himpunan dan fungsi keanggotaan berupa pelan, normal, aktivitas ringan, aktivitas berat, dan anaerob, sedangkan untuk laju napas terdapat 3 himpunan dan fungsi keanggotaan berupa lambat, normal, dan cepat. Kedua proses pembuatan *rule base* yaitu menentukan aturan-aturan untuk himpunan yang telah didesain. Dari 5 himpunan detak jantung dan 3 himpunan laju napas maka didapatkan 15 aturan untuk dua kondisi yaitu berhenti atau lanjut. Terakhir proses defuzzifikasi yaitu jika sensor mendeteksi suatu nilai maka nilai tersebut akan langsung dihitung derajat keanggotaannya untuk dikonversikan ke dalam suatu

himpunan yang telah dibuat. Proses ini merupakan proses realisasi dari fuzzifikasi.



Gambar 3.13 Flowchart Fuzzy Logic

Dari nilai himpunan yang dihasilkan dari detak jantung dan laju napas dapat ditentukan aturan yang sesuai untuk kondisi tersebut. Berikut hasil desain yang telah dibuat dan dikonsultasikan:

3.4.2 Himpunan dan Fungsi Keanggotaan

a. Sensor Detak Jantung

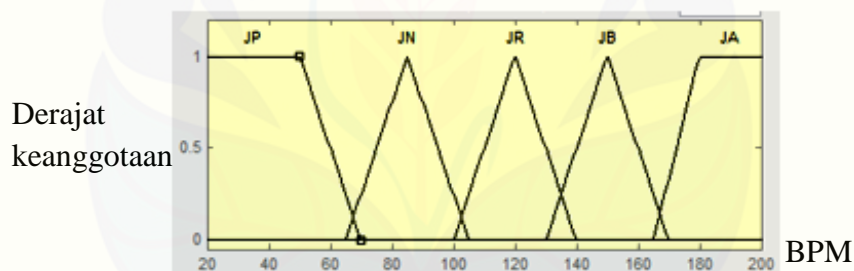
Kontrol *fuzzy* pada detak jantung menggunakan parameter pada Tabel 3.4.

Tabel 3.4 Parameter Sensor Detak Jantung

Parameter	<i>Beat Per Minute</i> (BPM)
Pelan	<70
Normal	65 – 105
Aktivitas Ringan	100 – 140
Aktivitas Berat	130 – 170
Anaerob	>165

(Sumber : Meivita, 2016)

Dari parameter pada Tabel 3.4 dapat direpresentasikan dalam sebuah grafik *input* fuzzy untuk menentukan titik detak jantung yang sesuai. Pada Gambar 3.14 variabel suhu tubuh didefinisikan lima himpunan fuzzy, yaitu, PELAN, NORMAL, AKTIVITAS RINGAN, AKTIVITAS BERAT dan ANAEROB. Dari parameter tersebut dapat direpresentasikan dalam sebuah grafik *input* fuzzy.

Gambar 3.14 Representasi Variabel Himpunan *Fuzzy* Detak Jantung

Pada Gambar 3.14 ada lima variabel dari himpunan detak jantung. Variabel JP (Jantung Pelan) merupakan fungsi keanggotaan dari himpunan detak jantung pelan. Variabel JN (Jantung Normal) merupakan fungsi keanggotaan dari himpunan detak jantung normal. Variabel JR (Jantung Ringan) merupakan fungsi keanggotaan dari himpunan detak jantung aktivitas ringan. Variabel JB (Jantung Berat) merupakan fungsi keanggotaan dari himpunan detak jantung aktivitas berat. Variabel JA (Jantung Anaerob) merupakan fungsi keanggotaan dari himpunan detak jantung anaerob. Adapun sumbu vertikal merupakan derajat keanggotaan dari nilai *input* variabel detak jantung. Berikut sumbu keanggotaan dari variabel detak jantung:

$$\mu_{\text{pelan}} = \begin{cases} 1; z_1 \leq 50 \\ \frac{50-z_1}{20}; 50 \leq z_1 \leq 70 \\ 0; z_1 \geq 70 \end{cases} \quad (\text{Persamaan 3.1})$$

$$\mu_{\text{normal}} = \begin{cases} 0; z_1 \leq 65 \\ \frac{z_1-65}{20}; 65 \leq z_1 \leq 85 \\ \frac{105-z_1}{20}; 85 \leq z_1 \leq 105 \end{cases} \quad (\text{Persamaan 3.2})$$

$$\mu_{\text{ringan}} = \begin{cases} 0; z_1 \leq 100 \\ \frac{z_1-100}{20}; 100 \leq z_1 \leq 120 \\ \frac{140-z_1}{20}; 120 \leq z_1 \leq 140 \end{cases} \quad (\text{Persamaan 3.3})$$

$$\mu_{\text{berat}} = \begin{cases} 0; z_1 \leq 130 \\ \frac{z_1-130}{20}; 130 \leq z_1 \leq 150 \\ \frac{170-z_1}{20}; 150 \leq z_1 \leq 170 \end{cases} \quad (\text{Persamaan 3.4})$$

$$\mu_{\text{anaerob}} = \begin{cases} 0; z_1 \leq 165 \\ \frac{z_1-165}{15}; 165 \leq z_1 \leq 180 \\ 1; z_1 \geq 180 \end{cases} \quad (\text{Persamaan 3.5})$$

keterangan : z_1 = hasil pengukuran detak jantung

b. Sensor Laju Napas

Kontrol *fuzzy* pada laju napas untuk kecepatan napas per menit menggunakan parameter pada Tabel 3.5.

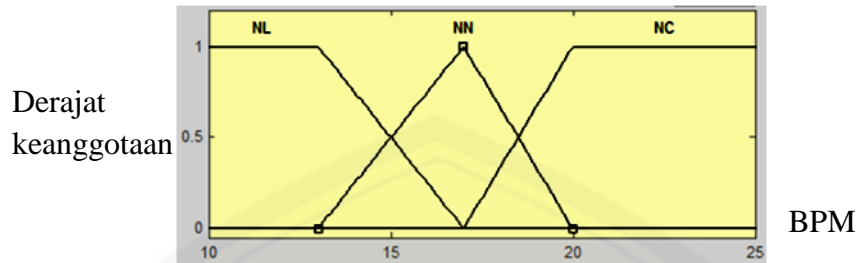
Tabel 3.5 Parameter Sensor Laju Napas

Parameter	Laju Napas
Lambat	<17
Normal	13-20
Cepat	>17

(Sumber : Hadiyoso, 2015)

Dari parameter pada Tabel 3.5 dapat direpresentasikan dalam sebuah grafik *input fuzzy* untuk menentukan laju napas yang sesuai. Pada Gambar 3.15

variabel laju napas didefinisikan tiga himpunan fuzzy, yaitu lambat, normal dan cepat.



Gambar 3.15 Representasi Variabel Himpunan *Fuzzy* Laju Napas

Pada Gambar 3.15 ada tiga variabel dari himpunan laju napas. Variabel NL (Nafas Lambat) merupakan fungsi keanggotaan dari himpunan laju napas lambat. Variabel NN (Nafas Normal) merupakan fungsi keanggotaan dari himpunan laju napas normal. Variabel NC (Nafas Cepat) merupakan fungsi keanggotaan dari himpunan laju napas cepat. Adapun sumbu vertikal merupakan derajat keanggotaan dari nilai *input* variabel laju napas. Berikut sumbu keanggotaan dari variabel laju napas:

$$\mu_{\text{lambat}} = \begin{cases} 1; z_2 \leq 13 \\ \frac{17-z_2}{4}; 13 \leq z_2 \leq 17 \\ 0; z_2 \geq 17 \end{cases} \quad (\text{Persamaan 3.6})$$

$$\mu_{\text{normal}} = \begin{cases} 0; z_2 \leq 13 \\ \frac{z_2-13}{4}; 13 \leq z_2 \leq 17 \\ \frac{20-z_2}{3}; 17 \leq z_2 \leq 20 \end{cases} \quad (\text{Persamaan 3.7})$$

$$\mu_{\text{cepat}} = \begin{cases} 0; z_2 \leq 17 \\ \frac{z_2-17}{3}; 17 \leq z_2 \leq 20 \\ 1; z_2 \geq 20 \end{cases} \quad (\text{Persamaan 3.8})$$

keterangan : z_2 = hasil pengukuran laju napas

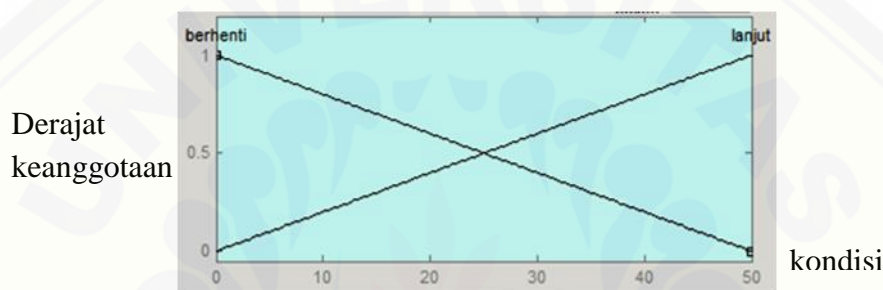
c. Output

Kontrol *fuzzy output*, merupakan hasil keputusan dari *rules* yang telah dibuat yang berdasarkan nilai detak jantung dan laju napas. *Output* tersebut akan dipengaruhi oleh dua *input* yang telah ditentukan dan dapat berubah-ubah berdasarkan nilai ADC yang dikeluarkan oleh dua sensor.

Tabel 3.6 *Output Fuzzy Set*

Parameter	Nilai
Berhenti	0-50
Lanjut	0-50

Dari parameter pada Tabel 3.6 dapat direpresentasikan dalam sebuah grafik *output fuzzy* untuk menentukan keluaran yang sesuai. Pada Gambar 3.16 variabel *output* didefinisikan dua himpunan fuzzy, yaitu BERHENTI, dan LANJUT.

Gambar 3.16 Representasi Variabel Himpunan *Output Fuzzy*

Adapun sumbu vertikal merupakan tingkat keanggotaan dari nilai *input* variabel detak jantung dan laju napas. Berikut sumbu keanggotaan dari variabel berhenti dan lanjut:

$$\mu_{\text{berhenti}} = \begin{cases} 0; & z_3 \geq 50 \\ \frac{50-z_3}{50}; & 0 \leq z_3 \leq 50 \end{cases} \quad (\text{Persamaan 3.9})$$

$$\mu_{\text{lanjut}} = \begin{cases} 0; & z_3 \leq 0 \\ \frac{z_3-0}{50}; & 0 \leq z_3 \leq 50 \end{cases} \quad (\text{Persamaan 3.10})$$

keterangan : z_3 = hasil kondisi dari *output fuzzy*

3.4.3 Rule Base

Dari beberapa data *input* yang didapat, maka dibuat suatu aturan atau *rule* yang akan menghasilkan keputusan dari *fuzzy controller*. Keputusan ini nantinya yang akan berperan sebagai *output*. Dasarnya *rule* ini adalah sebuah *rule if – and – then* yang mudah dimengerti karena hanya merupakan kata-kata terdapat 15 *rule* yang menghasilkan *output* yang menunjukkan hubungan antara detak jantung dan

laju napas yang akan menghasilkan *output* berupa kondisi tubuh yaitu dapat lanjut atau harus berhenti.

1. If JANTUNG is PELAN and LAJU NAPAS is LAMBAT then HASIL is LANJUT.
2. If JANTUNG is PELAN and LAJU NAPAS is NORMAL then HASIL is LANJUT.
3. If JANTUNG is PELAN and LAJU NAPAS is CEPAT then HASIL is BERHENTI.
4. If JANTUNG is NORMAL and LAJU NAPAS is LAMBAT then HASIL is LANJUT.
5. If JANTUNG is NORMAL and LAJU NAPAS is NORMAL then HASIL is LANJUT.
6. If JANTUNG is NORMAL and LAJU NAPAS is CEPAT then HASIL is BERHENTI.
7. If JANTUNG is RINGAN and LAJU NAPAS is LAMBAT then HASIL is LANJUT.
8. If JANTUNG is RINGAN and LAJU NAPAS is NORMAL then HASIL is LANJUT.
9. If JANTUNG is RINGAN and LAJU NAPAS is CEPAT then HASIL is BERHENTI.
10. If JANTUNG is BERAT and LAJU NAPAS is LAMBAT then HASIL is LANJUT.
11. If JANTUNG is BERAT and LAJU NAPAS is NORMAL then HASIL is LANJUT.
12. If JANTUNG is BERAT and LAJU NAPAS is CEPAT then HASIL is BERHENTI.
13. If JANTUNG is ANAEROB and LAJU NAPAS is LAMBAT then HASIL is BERHENTI.
14. If JANTUNG is ANAEROB and LAJU NAPAS is NORMAL then HASIL is BERHENTI.

15. If JANTUNG is ANAEROB and LAJU NAPAS is CEPAT then HASIL is BERHENTI.

Tabel 3.7 Rule Base

Detak Jantung \ Laju Napas	Pelan	Normal	Ringan	Berat	Anaerob
Lambat	Lanjut	Lanjut	Lanjut	Lanjut	Berhenti
Normal	Lanjut	Lanjut	Lanjut	Lanjut	Berhenti
Cepat	Berhenti	Berhenti	Berhenti	Berhenti	Berhenti

3.5 Implementasi Perangkat Keras

Seperti yang telah dijelaskan sebelumnya alat yang telah dibuat menggunakan sensor detak jantung dan laju napas. Untuk mikrokontrolernya menggunakan Arduino Uno. Sistem yang telah dibuat dapat menampilkan hasil pengukuran kondisi detak jantung dan laju napas. Pemilihan dua parameter tersebut untuk dapat mencegah terjadinya *overtraining* saat berolahraga. Dari hasil pengukuran juga dapat langsung diambil kesimpulan berupa kondisi berhenti atau lanjut menggunakan logika fuzzy. Berikut hasil alat yang telah dibuat:



Gambar 3.17 Alat Tampak Atas



Gambar 3.18 Alat Tampak Samping 1



Gambar 3.19 Alat Tampak Samping 2



Gambar 3.20 Antena Radio dan USB Receiver

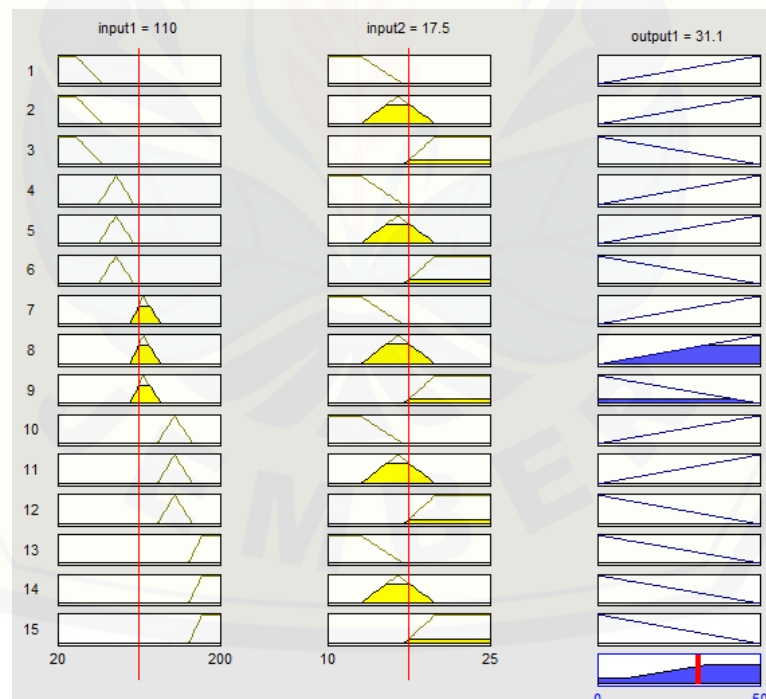
Keterangan:

1. LCD untuk menampilkan data pengukuran dan *output fuzzy*,
2. Sensor detak jantung untuk mengukur detak jantung manusia,
3. Sensor tekanan untuk mendeteksi laju napas manusia dengan perubahan tekanan masuk dan tekanan keluar,
4. Masker oksigen untuk tekanan keluar,
5. Suntikan untuk tekanan masuk,
6. Modul *transmitter* untuk mengirim data menggunakan radio,
7. *Port USB downloader* untuk meng-*upload* program dari komputer ke Arduino,
8. *Port adaptor 9V* untuk memberi daya masukan sebesar 9V,

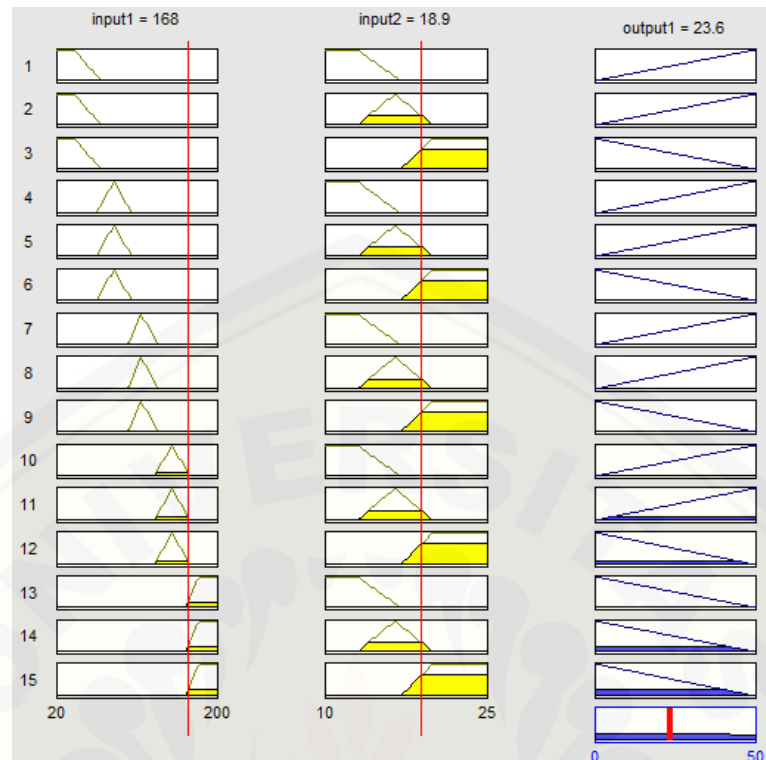
9. *Slot memory* untuk tempat memasukkan *Micro SD card*,
10. Box yang berisi mikrokontroler Arduino dan modul pendukung lainnya,
11. Antena penerima sinyal frekuensi radio,
12. Modul *receiver* untuk memproses data yang diterima oleh antena,
13. Antena pengirim sinyal frekuensi radio.

3.6 Implementasi Fuzzy (Mamdani) pada Sistem

Pada sistem logika fuzzy berfungsi sebagai pengambil keputusan akhir yaitu kondisi lanjut atau berhenti. Seperti yang telah dijelaskan pada penjelasan perancangan fuzzy. Hasil implementasi dari alat pencegah *overtraining* ini dapat dilihat pada Gambar 3.20 yaitu nilai input dari detak jantung dan laju nafas mempengaruhi keputusan yang akan ditentukan. *Input1* merupakan himpunan detak jantung, *input2* merupakan himpunan laju nafas, dan *output1* merupakan himpunan dari kondisi lanjut dan berhenti.



Gambar 3.21 *Sampling* Pengambilan Keputusan Lanjut



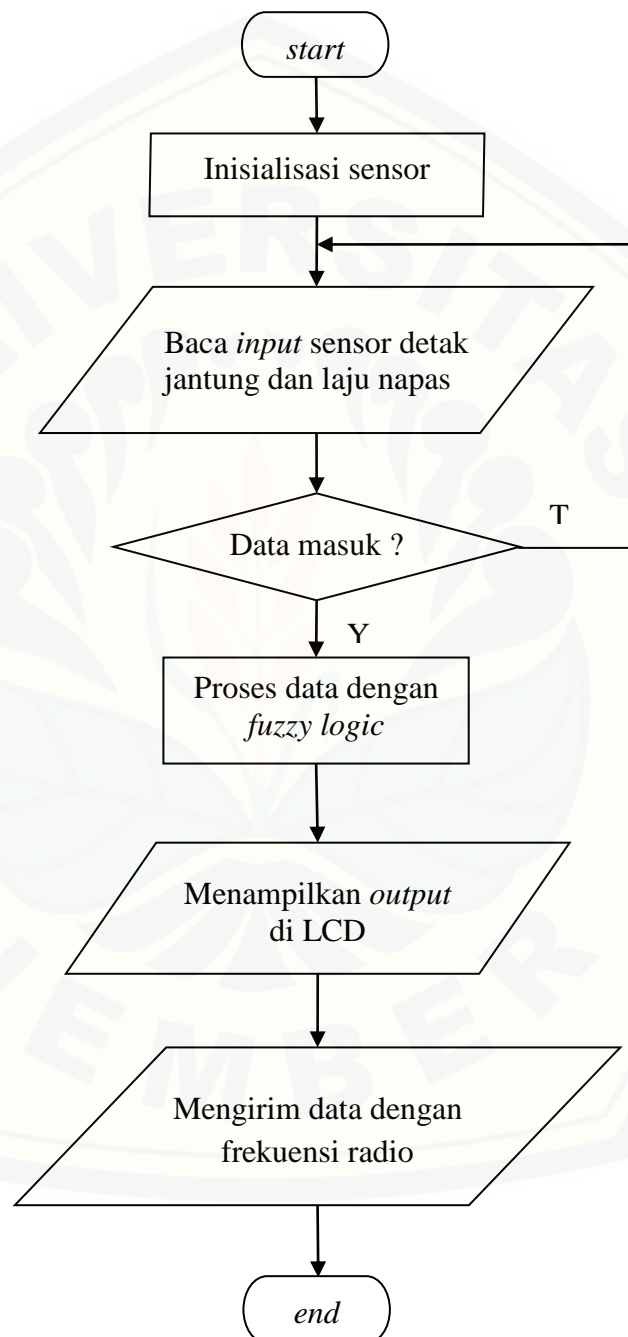
Gambar 3.22 *Sampling Pengambilan Keputusan Berhenti*

Pengambilan keputusan pada fuzzy Mamdani ini menggunakan metode COA (*Center of Area*) atau *min-max* yaitu saat implikasi mengambil nilai derajat keanggotaan terkecil (*min*) dan saat agregasi mengambil nilai terbesar (*max*). Sedangkan untuk menentukan hasil akhirnya dengan menghitung titik tengah area (*Center of Area*) untuk menentukan *output*. Konstanta COA merupakan acuan dasar dalam menentukan area setiap keputusan yang telah ditentukan.

3.7 Flowchart

Pada sistem kontrol logika fuzzy menggunakan detak jantung dan laju napas yang pertama kali dilakukan adalah pengambilan data pada setiap sensor (*input*), yaitu sensor detak jantung dan laju napas yang akan diproses oleh Arduino. Setelah mendapatkan data *input* pada setiap sensor yang digunakan maka Arduino akan memproses data dengan menggunakan metode logika fuzzy. Setelah mendapatkan data *input* pada setiap sensor yang digunakan maka Arduino akan memproses data dengan menggunakan metode *fuzzy logic* jenis COA (*Center Of Area*) dalam menentukan *output* yang berupa keputusan berdasarkan

konstanta COA. *Output* didapatkan dari *rule base* yang telah ditentukan kemudian ditampilkan pada LCD dan dikirim melalui frekuensi radio untuk dapat dimonitor langsung melalui laptop maupun *handphone* (*support OTG*).



Gambar 3.23 Flowchart Sistem

3.8 Model Pengujian

Pada model pengujian ini ditujukan untuk mempermudah proses pengambilan data agar terstruktur. Model pengujian ini dibedakan menjadi tiga jenis pengujian sebagai berikut:

3.8.1 Model Pengujian Parameter

Pada pengujian parameter ini terdiri dari dua parameter yang akan diuji yaitu detak jantung dan laju napas. Kedua parameter tersebut pengujiannya yaitu dengan cara kalibrasi sensor lalu membandingkan hasil pengukuran sensor dengan hasil pengukuran alat ukur sesungguhnya. Hasil pengujian akan dibandingkan dengan hasil pengujian menggunakan alat yang dipakai di rumah sakit yaitu ECG monitor. Pada alat ECG monitor hasil pengukuran yang ditampilkan yaitu detak jantung, saturasi oksigen dalam tubuh dan respirasi paru-paru. Dari hasil pengukuran tersebut maka didapatkan perbandingan dan nilai *error* yang dihasilkan.

3.8.2 Model Pengujian Jarak Pengiriman Radio

Pada pengujian jarak pengiriman radio ini yaitu terdiri dari beberapa pengujian yang akan diuji untuk mengetahui jarak maksimum untuk pengiriman data. Hasil pengujian transmisi ini akan digunakan juga untuk pengambilan data pada jarak yang berbeda-beda serta pengaruh pengiriman saat di dalam ruangan dan di luar ruangan. Kecepatan pengiriman data diatur pada kecepatan 57600 *baudrate* yang disesuaikan dengan kecepatan pengiriman data pada Arduino. Pengukuran jarak ini dilakukan menggunakan *smartphone* untuk mengukur jarak maksimal pengiriman data.

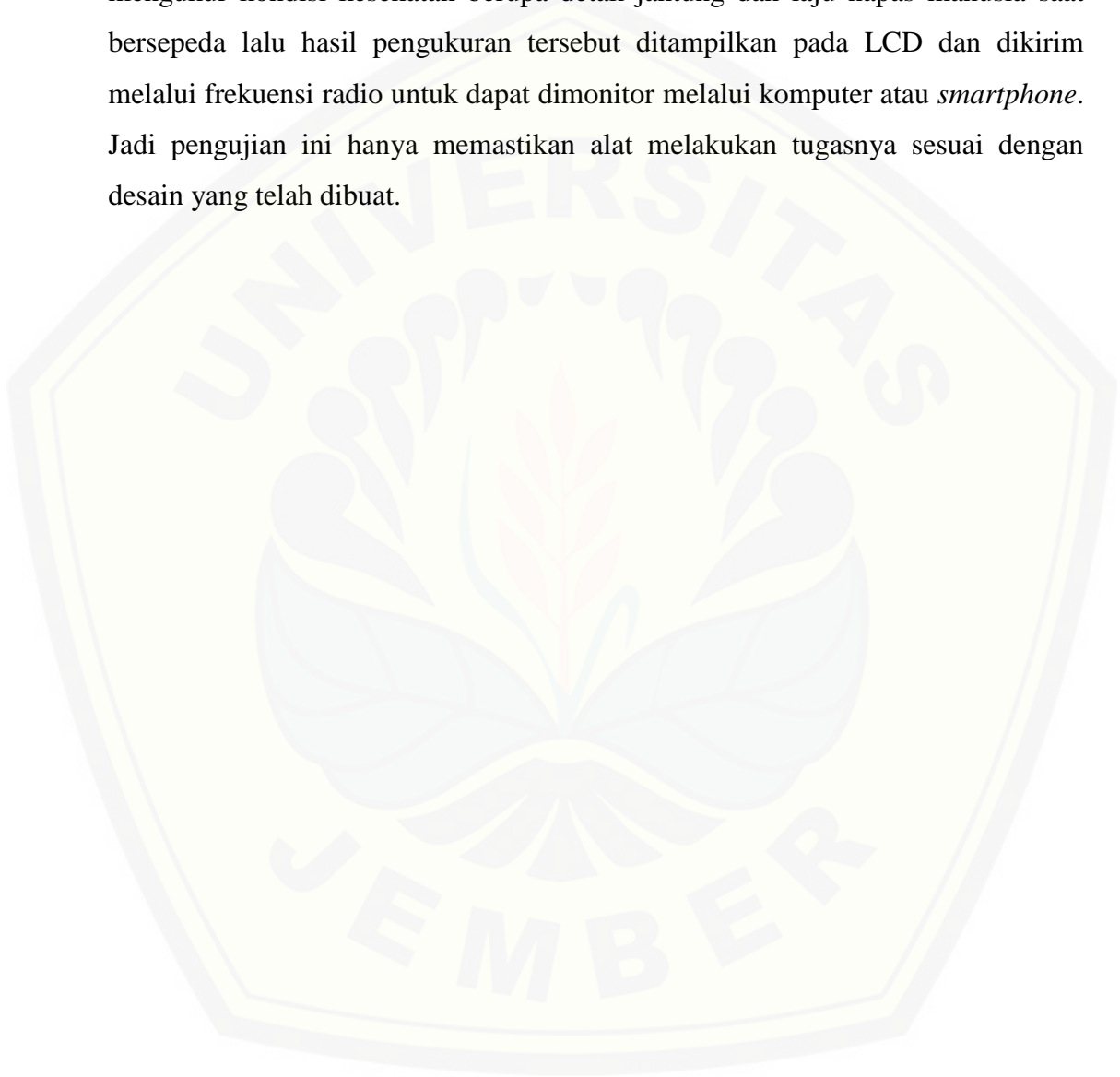
3.8.3 Model Pengujian Logika Fuzzy

Pada pengujian ini membahas tentang perbandingan hasil keputusan yang telah dibuat yaitu *rule base* dengan hasil keputusan yang dihasilkan oleh alat. Dengan parameter detak jantung dan laju napas sebagai *input*. Jika hasil pengukuran belum sesuai maka akan dilakukan pemrograman ulang hingga hasil yang dihasilkan sesuai dengan *rule base* yang telah dibuat. Hasil yang diinginkan yaitu suatu keputusan kondisi saat berolahraga yaitu kondisi lanjut atau harus

berhenti sehingga mencegah *overtraining* yang mengakibatkan cedera atau hal yang tidak diinginkan.

3.8.4 Model Pengujian Sistem Keseluruhan

Pada pengujian sistem keseluruhan ini yaitu pengujian alat yang akan mengukur kondisi kesehatan berupa detak jantung dan laju napas manusia saat bersepeda lalu hasil pengukuran tersebut ditampilkan pada LCD dan dikirim melalui frekuensi radio untuk dapat dimonitor melalui komputer atau *smartphone*. Jadi pengujian ini hanya memastikan alat melakukan tugasnya sesuai dengan desain yang telah dibuat.



BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan pengujian yang telah dilakukan dari penelitian dengan judul “Implementasi Sistem *Telemetry* Monitor Detak Jantung dan Laju Napas untuk Pencegahan *Overtraining* Saat Bersepeda Menggunakan Logika Fuzzy” didapatkan beberapa kesimpulan :

1. Pengujian sensor nilai *error* terbesar yaitu 5,95 % pada detak jantung dan 8 % pada laju napas. Sedangkan nilai *error* terkecil yaitu 1,06 % pada detak jantung dan 0 % pada laju napas.
2. Jarak maksimum pengiriman data menggunakan radio tanpa penghalang sejauh 140 meter dan dengan penghalang sejauh 96 meter.
3. Hasil pengujian fuzzy telah sesuai dengan *rule base* yang telah dibuat dan telah dibuktikan dengan perhitungan manual dan matlab. Hasil perhitungan dengan nilai *error* terkecil yaitu 0,32% dan nilai *error* terbesar yaitu 0,46%.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, maka penulis memberikan saran untuk pengembangan penelitian ini berikutnya, antara lain sebagai berikut :

1. Rancangan mekanik alat belum tahan guncangan sehingga diharapkan alat selanjutnya dapat membuat mekanik yang lebih tahan guncangan.
2. Untuk mendapatkan hasil *output* yang lebih baik lagi disarankan menggunakan metode lain atau kombinasi dari dua metode.

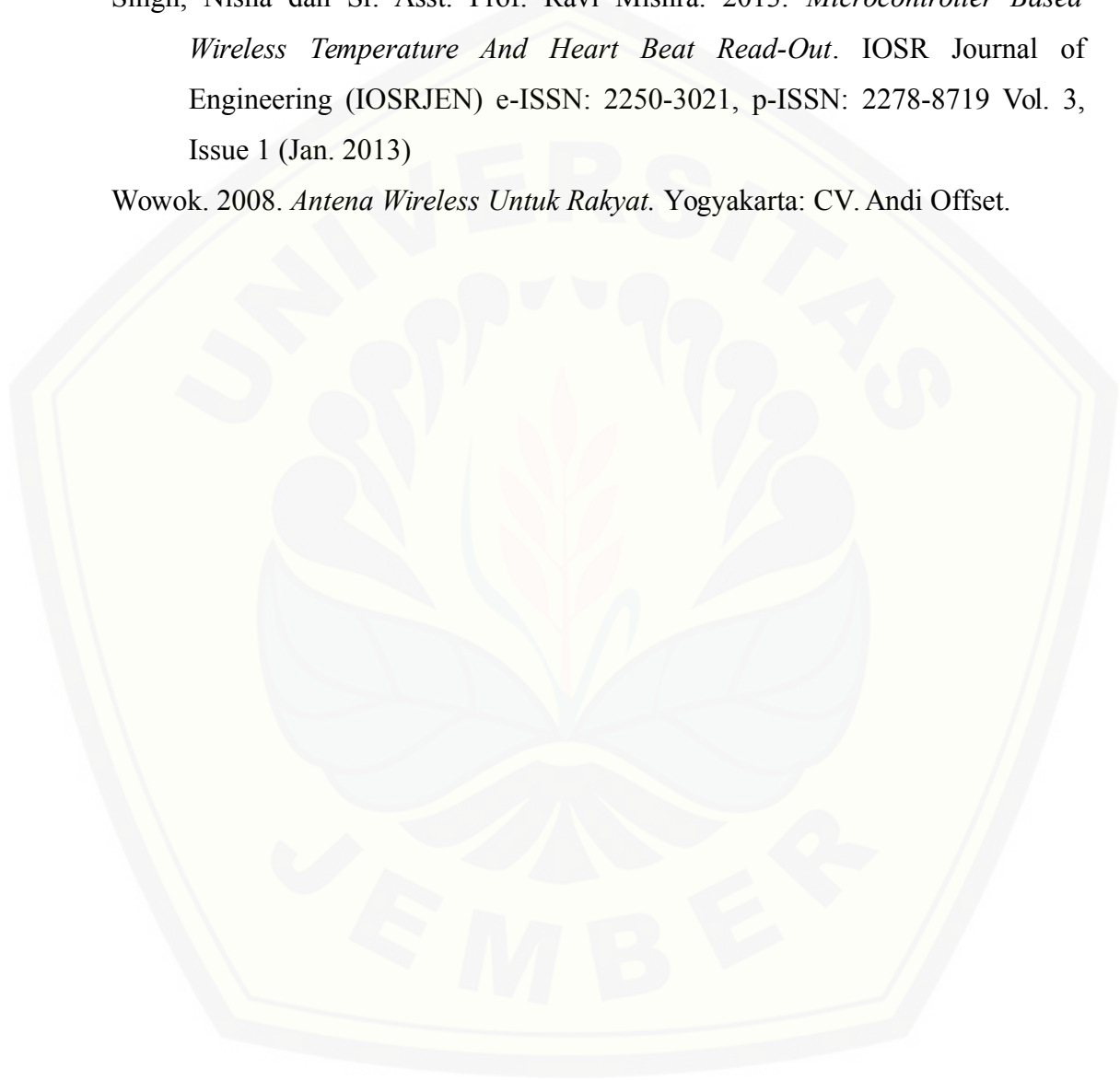
DAFTAR PUSTAKA

- Artanto, Dian. 2012. *Interaksi Arduino dan LabVIEW*. Jakarta: PT. Elex Media Komputindo.
- Balaban, Naomi E dan James E. Bobick. 2014. *Seri Ilmu Pengetahuan Anatomi Dan Fisiologi*. Jakarta: PT. Indeks
- Daryanto. 2011. *Teknik Mekatronika*. Bandung: PT. Sarana Tutorial Nurani Sejahtera.
- Ganong, William F. 2008. *Buku Ajar Fisiologi Kedokteran*. Jakarta: Buku Kedokteran EGC.
- Giriwijoyo, Prof. H.Y.S Santosa dan Dr. Didik Z.S, M.Pd. 2012. *Ilmu Kesehatan Olahraga*. Bandung: PT. Remaja Rosdakarya.
- Hadiyoso, Sugondo dkk. 2015. *Implementasi Regulator Oksigen Otomatis berdasarkan Tingkat Pernapasan Menggunakan Logika Fuzzy*. Teknik Elektro Itenas Jurnal ELKOMIKA ISSN: 2338-8323 No. 1 Vol. 3 Januari - Juni 2015
- Kale, Aniket V. dkk. 2015. *GSM Based Heart Rate and Temperature Monitoring System*. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 4 Issue 04.
- Kuswandi, Son. 2007. *Kendali Cerdas*. Yogyakarta: CV. Andi Offset
- Mallick, Bandana dan Ajit Kumar Patro. 2016. *Heart Rate Monitoring System Using Finger Tip Through Arduino And Processing Software*. International Journal of Science, Engineering and Technology Research (IJSETR), ISSN: 2278 – 7798 Volume 5, Issue 1, January 2016
- Meivita, Dewi Nurhaji. 2016. *Rancang Bangun Alat Ukur Kondisi Kesehatan Pada Pendaki Gunung Berbasis Fuzzy Logic*. Yogyakarta, Seminar Nasional Aplikasi Teknologi Informasi (SNATI), ISSN:1907-5022
- Palar, Chrisly M dkk. 2015. *Manfaat Latihan Olahraga Aerobik Terhadap Kebugaran Fisik Manusia*. Jurnal e-Biomedik (eBm), Volume 3, Nomor 1, Januari-April 2015

Patil, Harshavardhan B dan Prof.V.M.Umale. 2015. *Arduino Based Wireless Biomedical Parameter Monitoring System Using Zigbee*. International Journal of Engineering Trends and Technology (IJETT) ISSN: 2231-5381 Volume 28 Number 7.

Singh, Nisha dan Sr. Asst. Prof. Ravi Mishra. 2013. *Microcontroller Based Wireless Temperature And Heart Beat Read-Out*. IOSR Journal of Engineering (IOSRJEN) e-ISSN: 2250-3021, p-ISSN: 2278-8719 Vol. 3, Issue 1 (Jan. 2013)

Wowok. 2008. *Antena Wireless Untuk Rakyat*. Yogyakarta: CV. Andi Offset.



LAMPIRAN

Pada lampiran ini terdiri dari lampiran gambar-gambar hasil pengujian alat, gambar hasil penyimpanan data pada *micro SD card*, gambar hasil penyimpanan data pada Android, gambar hasil tampilan monitor pada Android, hasil tampilan monitor pada laptop dan *listing* program. Berikut hasil dokumentasi gambar dan *listing* program:

-Listing Program

```
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <SPI.h>
#include <SD.h>
#include <stdlib.h>

LiquidCrystal_I2C lcd(0x3f, 16, 2);
SoftwareSerial mySerial(3,2);
const int chipSelect = 10;
float sensorValue[6];
int nafas;
float tress=326;
int t1,t2;
int times, last;

int pulsePin = 3;
int blinkPin = 9;

volatile int BPM;
volatile int Signal;
volatile int IBI = 600;
volatile boolean Pulse = false;
volatile boolean QS = false;

float atas =0, bawah = 0, zx = 0;
float apred[16], z[16], mf [5][3];
// rule
int L [] = {1,2,4,5,7,8,10,11};//8
int B [] = {3,6,9,12,13,14,15};//7

void setup(){

    mySerial.begin(57600);
```



```
interruptSetup();
lcd.begin();
lcd.backlight();
pinMode(8,OUTPUT);
pinMode(9,OUTPUT);
if (!SD.begin(chipSelect)) {

    return;
}

File dataFile = SD.open("data.txt", FILE_WRITE);
if (dataFile) {
    dataFile.println("=====");
    dataFile.close();
}

}

void loop(){
    if (QS == true) { QS = false; }
    lajuNafas();
    digitalWrite(8, t1);
    fuzzyRutin(nafas,BPM);
    serial();
    lcd1();
}
void lcd1(){
    lcd.setCursor(0,0);
    lcd.print("LN: ");
    lcd.print(nafas);
    lcd.print(" ");
    lcd.setCursor(7,0);
    lcd.print("DJ: ");
    lcd.print(BPM);
    lcd.print(" ");
    lcd.setCursor(0,1);
    if(zx>=25){lcd.print("LANJUT ");}
    else      {lcd.print("BERHENTI");}
    lcd.setCursor(13,1);
    lcd.print((int)sensorValue[0]);
    lcd.print(" ");
}
void serial(){
```

```
mySerial.print(nafas);
mySerial.print("\t");
mySerial.print(BPM);
mySerial.print("\t");
mySerial.println(zx);
saveData(nafas, BPM, zx);
}

void lajuNafas(){
  sensorValue[5] =sensorValue[4];
  sensorValue[4] =sensorValue[3];
  sensorValue[3] =sensorValue[2];
  sensorValue[2] =sensorValue[1];
  sensorValue[1] =sensorValue[0];
  sensorValue[0] =
(analogRead(A2)+sensorValue[1]+sensorValue[2]+sensorValue[3]
+sensorValue[4]+sensorValue[5])/6;
  t2=t1;
  t1 = ((sensorValue[0]>tress) ? 1 : 0 );
  if(t1==0 && t2==1){ // falling
    times=millis()-last;
    last=millis();
    if (times<2000){times=2000;}
    nafas= (0.787*(60000.0/times))+4.131;
  }
}

void fuzzyRutin(float LN, float DJ){
  //member function DETAK JANTUNG
  mf[0][0] = FuzzyMemberTrap1(DJ,40, 40, 70); //pelan
  mf[1][0] = FuzzyMember(DJ,65, 90, 105); //Normal
  mf[2][0] = FuzzyMember(DJ,100, 117, 135); //aktifitas
  ringan
  mf[3][0] = FuzzyMember(DJ,130, 150, 170); //aktifitas
  berat
  mf[4][0] = FuzzyMemberTrap2(DJ,165, 180, 180); //anaerob
  //mamber function LAJU NAFAS
  mf[0][1] = FuzzyMemberTrap1(LN,12, 12, 17); //lambat
  mf[1][1] = FuzzyMember(LN,13, 15, 20); //normal
  mf[2][1] = FuzzyMemberTrap2(LN,17, 22, 22); //cepat

  //mencari alfa predikat FUZZY
  int x1 = 1;
  for(int a1 = 0; a1 <=4; a1 = a1 +1)
  for(int a2 = 0; a2 <=2; a2 = a2 +1)
```

```
{
    apred[x1] = min(mf[a1][0],mf[a2][1]);
    x1 = x1 + 1;
}

//Mencari nilai z output

for(int ri = 1; ri <= 25; ri = ri + 1)
{
    switch(ri){
        case 1 : {lanjut(ri);break;}
        case 2 : {lanjut(ri);break;}
        case 4 : {lanjut(ri);break;}
        case 5 : {lanjut(ri);break;}
        case 7 : {lanjut(ri);break;}
        case 8 : {lanjut(ri);break;}
        case 10 : {lanjut(ri);break;}
        case 11 : {lanjut(ri);break;}

        case 3 : {berhenti(ri);break;}
        case 6 : {berhenti(ri);break;}
        case 9 : {berhenti(ri);break;}
        case 12 : {berhenti(ri);break;}
        case 13 : {berhenti(ri);break;}
        case 14 : {berhenti(ri);break;}
        case 15 : {berhenti(ri);break;}
    }
}

//Defuzifikasi
atas =0, bawah = 0, zx = 0;
for (int pi = 1; pi <= 15; pi = pi +1)
{
    atas = atas + (apred[pi] * z[pi]);
    bawah = bawah + apred[pi];
}

zx = atas / bawah;

}

float FuzzyMember(float a, float b, float c, float d){
    float hasil;
```

```
if(a <= b || a >= d){hasil = 0;}
else if(a > b && a<=c){hasil = (a - b)/(c-b);}
else if(a > c && a < d){hasil = (d - a)/(d-c);}

return hasil;

}
float FuzzyMemberTrap1(float xz, float az, float bz, float
cz) {
float result1;
if (xz <= bz){result1 = 1;}
else if (xz > bz && xz < cz){result1 = (cz - xz) / (cz -
bz);}
else if (xz >= cz){result1 = 0;}
return result1;
}

float FuzzyMemberTrap2(float xz, float az, float bz, float
cz) {
float result2;
if (xz <= az){result2 = 0;}
else if (xz > az && xz <= bz){result2 = (xz - az) / (bz -
az);}
else if (xz >= bz){result2 = 1;}
return result2;
}

void berhenti(int x){

z[x]= 25 - (apred[x] * 25); // KURVA TURUN

}
void lanjut(int x){
z[x] = 20 + (apred[x] * 25); // KURVA NAIK

}

void saveData(float x0,float x1,float x2)
{
String dataString = "";
dataString += String(x0)+"\t";
dataString += String(x1)+"\t";
dataString += String(x2)+"\t";

File dataFile = SD.open("data.txt", FILE_WRITE);
```

```
    if (dataFile) {
        dataFile.println(dataString);
        dataFile.close();
    }

}

volatile int rate[10]; // array to hold
last ten IBI values
volatile unsigned long sampleCounter = 0; // used
to determine pulse timing
volatile unsigned long lastBeatTime = 0; // used
to find IBI
volatile int P = 512; // used to find
peak in pulse wave, seeded
volatile int T = 512; // used to find
trough in pulse wave, seeded
volatile int thresh = 512; // used to find
instant moment of heart beat, seeded
volatile int amp = 100; // used to hold
amplitude of pulse waveform, seeded
volatile boolean firstBeat = true; // used to seed
rate array so we startup with reasonable BPM
volatile boolean secondBeat = false; // used to seed
rate array so we startup with reasonable BPM

void interruptSetup(){
    // Initializes Timer2 to throw an interrupt every 2mS.
    TCCR2A = 0x02; // DISABLE PWM ON DIGITAL PINS 3 AND
11, AND GO INTO CTC MODE
    TCCR2B = 0x06; // DON'T FORCE COMPARE, 256 PRESCALER
    OCR2A = 0x7C; // SET THE TOP OF THE COUNT TO 124 FOR
500Hz SAMPLE RATE
    TIMSK2 = 0x02; // ENABLE INTERRUPT ON MATCH BETWEEN
TIMER2 AND OCR2A
    sei(); // MAKE SURE GLOBAL INTERRUPTS ARE
ENABLED
}

// THIS IS THE TIMER 2 INTERRUPT SERVICE ROUTINE.
// Timer 2 makes sure that we take a reading every 2
milliseconds
ISR(TIMER2_COMPA_vect){ // triggered
when Timer2 counts to 124
```

```

cli(); // disable
interrupts while we do this
Signal = analogRead(pulsePin); // read the
Pulse Sensor
sampleCounter += 2; // keep track
of the time in mS with this variable
int N = sampleCounter - lastBeatTime; // monitor the
time since the last beat to avoid noise

// find the peak and trough of the pulse wave
if(Signal < thresh && N > (IBI/5)*3){ // avoid
dichrotic noise by waiting 3/5 of last IBI
if (Signal < T){ // T is the
trough
T = Signal; // keep track of
lowest point in pulse wave
}
}

if(Signal > thresh && Signal > P){ // thresh
condition helps avoid noise
P = Signal; // P is the peak
} // keep track of
highest point in pulse wave

// NOW IT'S TIME TO LOOK FOR THE HEART BEAT
// signal surges up in value every time there is a pulse
if (N > 250){ // avoid
high frequency noise
if ( (Signal > thresh) && (Pulse == false) && (N >
(IBI/5)*3) ){
Pulse = true; // set the
Pulse flag when we think there is a pulse
digitalWrite(blinkPin,HIGH); // turn on
pin 13 LED
IBI = sampleCounter - lastBeatTime; // measure
time between beats in mS
lastBeatTime = sampleCounter; // keep
track of time for next pulse

if(secondBeat){ // if this is
the second beat, if secondBeat == TRUE
secondBeat = false; // clear
secondBeat flag

```

```

        for(int i=0; i<=9; i++){           // seed the
running total to get a realistic BPM at startup
        rate[i] = IBI;
    }
}

    if(firstBeat){                       // if it's the
first time we found a beat, if firstBeat == TRUE
        firstBeat = false;               // clear
firstBeat flag
        secondBeat = true;               // set the
second beat flag
        sei();                            // enable
interrupts again
        return;                            // IBI value is
unreliable so discard it
    }

    // keep a running total of the last 10 IBI values
    word runningTotal = 0;                // clear the
runningTotal variable

    for(int i=0; i<=8; i++){              // shift data
in the rate array
        rate[i] = rate[i+1];             // and drop
the oldest IBI value
        runningTotal += rate[i];         // add up the
9 oldest IBI values
    }

    rate[9] = IBI;                        // add the
latest IBI to the rate array
    runningTotal += rate[9];              // add the
latest IBI to runningTotal
    runningTotal /= 10;                   // average the
last 10 IBI values
    BPM = (0.9832*(60000/runningTotal))+2.054;
// how many beats can fit into a minute? that's BPM!
    QS = true;                            // set
Quantified Self flag
    // QS FLAG IS NOT CLEARED INSIDE THIS ISR
}
}

```

```
    if (Signal < thresh && Pulse == true){ // when the
values are going down, the beat is over
        digitalWrite(blinkPin,LOW); // turn off pin
13 LED
        Pulse = false; // reset the
Pulse flag so we can do it again
        amp = P - T; // get amplitude
of the pulse wave
        thresh = amp/2 + T; // set thresh at
50% of the amplitude
        P = thresh; // reset these
for next time
        T = thresh;
    }

    if (N > 2500){ // if 2.5 seconds
go by without a beat
        thresh = 512; // set thresh
default
        P = 512; // set P default
        T = 512; // set T default
        lastBeatTime = sampleCounter; // bring the
lastBeatTime up to date
        firstBeat = true; // set these to
avoid noise
        secondBeat = false; // when we get
the heartbeat back
    }

    sei(); // enable
interrupts when youre done!
} // end isr
```


A. Dokumentasi Pengujian Sensor



Gambar A1. Pengujian Sensor ke 1



Gambar A2. Pengujian ECG Monitor ke 1



Gambar A3. Pengujian Sensor ke 2



Gambar A4. Pengujian ECG Monitor ke 2



Gambar A5. Pengujian Sensor ke 3



Gambar A6. Pengujian ECG Monitor ke 3



Gambar A7. Pengujian Sensor ke 4



Gambar A8. Pengujian ECG Monitor ke 4



Gambar A9. Pengujian Sensor ke 5



Gambar A10. Pengujian ECG Monitor ke 5



Gambar A11. Pengujian Sensor ke 6



Gambar A12. Pengujian ECG Monitor ke 6



Gambar A13. Pengujian Sensor ke 7



Gambar A14. Pengujian ECG Monitor ke 7



Gambar A15. Pengujian Sensor ke 8



Gambar A16. Pengujian ECG Monitor ke 8



Gambar A17. Pengujian Sensor ke 9



Gambar A18. Pengujian ECG Monitor ke 9



Gambar A19. Pengujian Sensor ke 10



Gambar A20. Pengujian ECG Monitor ke 10



Gambar A21. Pengujian Sensor ke 11



Gambar A22. Pengujian ECG Monitor ke 11



Gambar A23. Pengujian Sensor ke 12



Gambar A24. Pengujian ECG Monitor ke 12



Gambar A25. Pengujian Sensor ke 13



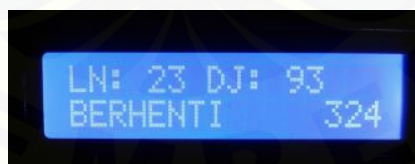
Gambar A26. Pengujian ECG Monitor ke 13



Gambar A27. Pengujian Sensor ke 14



Gambar A28. Pengujian ECG Monitor ke 14



Gambar A29. Pengujian Sensor ke 15



Gambar A30. Pengujian ECG Monitor ke 15

B. Dokumentasi Pengujian Alat



Gambar B1. Pengukuran Kondisi Pertama



Gambar B2. Pengukuran Kondisi Kedua



Gambar B3. Pengukuran Kondisi Ketiga

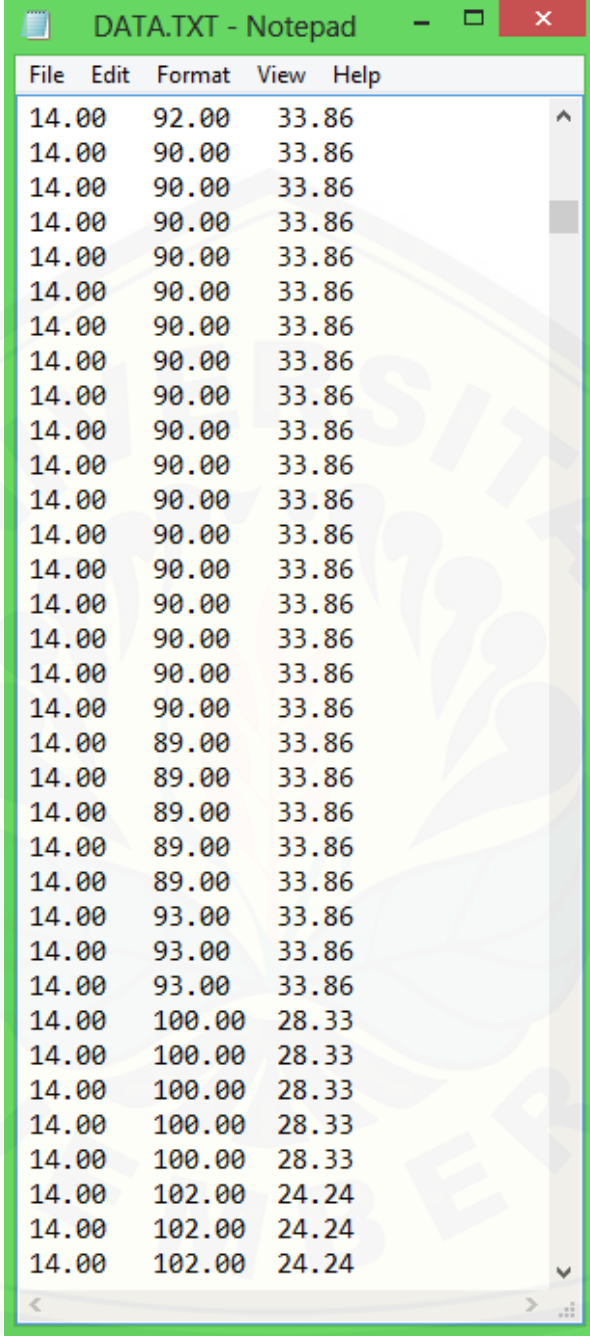


Gambar B4. Pengukuran Kondisi Keempat



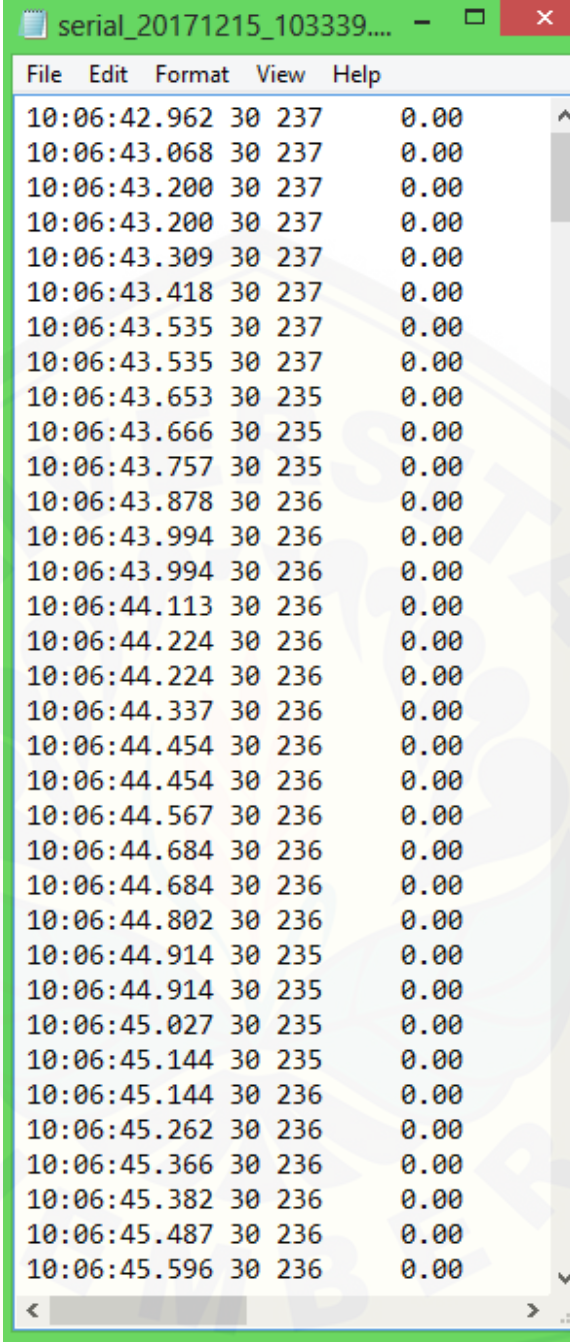
Gambar B5. Pengukuran Kondisi Kelima

C. Dokumentasi Penyimpanan



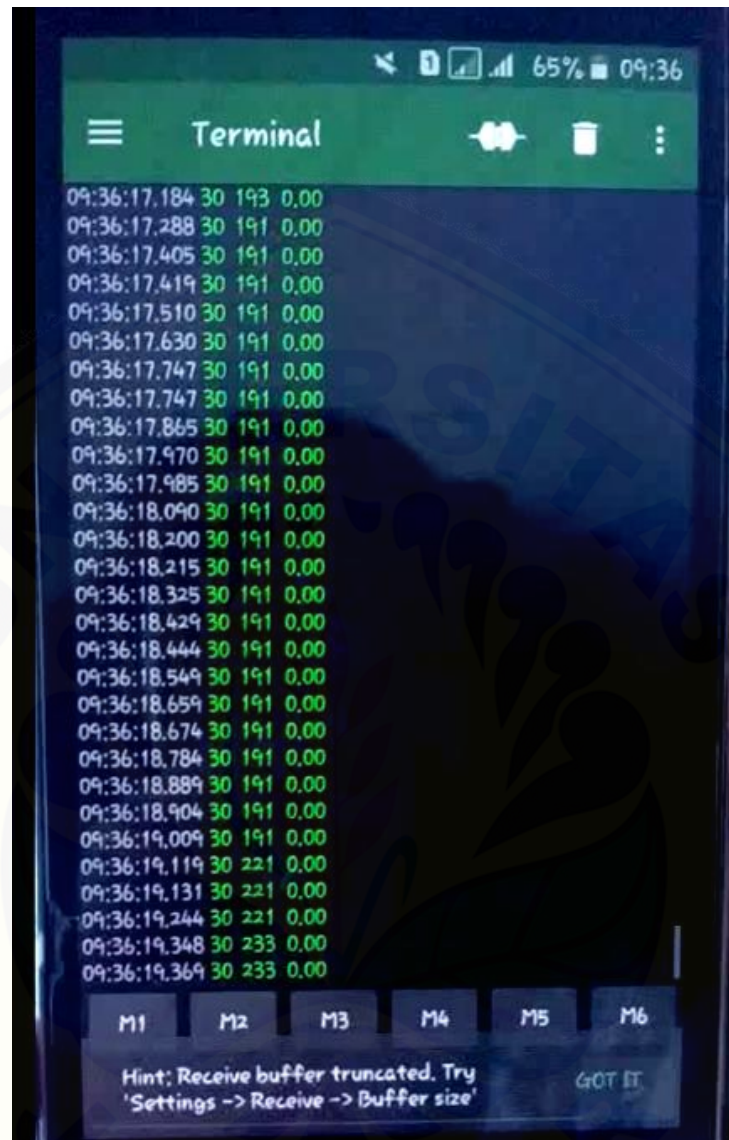
File	Edit	Format	View	Help
14.00	92.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	90.00	33.86		
14.00	89.00	33.86		
14.00	89.00	33.86		
14.00	89.00	33.86		
14.00	89.00	33.86		
14.00	89.00	33.86		
14.00	89.00	33.86		
14.00	93.00	33.86		
14.00	93.00	33.86		
14.00	93.00	33.86		
14.00	100.00	28.33		
14.00	100.00	28.33		
14.00	100.00	28.33		
14.00	100.00	28.33		
14.00	100.00	28.33		
14.00	102.00	24.24		
14.00	102.00	24.24		
14.00	102.00	24.24		

Gambar C1. Hasil Penyimpanan *Micro SD Card*

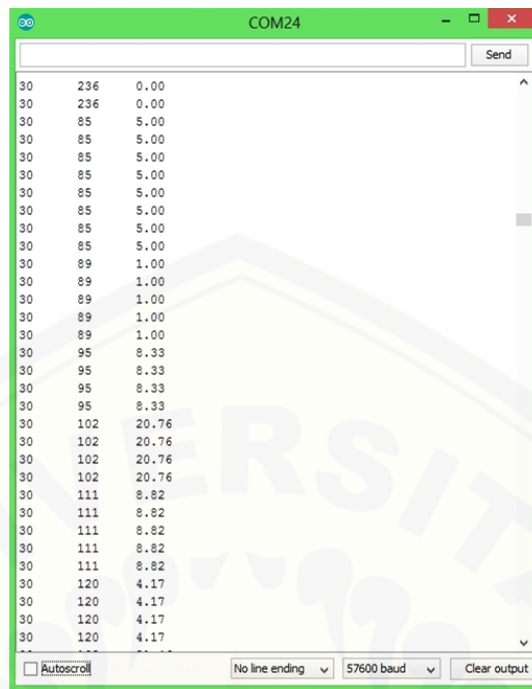


File	Edit	Format	View	Help
10:06:42.962	30	237	0.00	
10:06:43.068	30	237	0.00	
10:06:43.200	30	237	0.00	
10:06:43.200	30	237	0.00	
10:06:43.309	30	237	0.00	
10:06:43.418	30	237	0.00	
10:06:43.535	30	237	0.00	
10:06:43.535	30	237	0.00	
10:06:43.653	30	235	0.00	
10:06:43.666	30	235	0.00	
10:06:43.757	30	235	0.00	
10:06:43.878	30	236	0.00	
10:06:43.994	30	236	0.00	
10:06:43.994	30	236	0.00	
10:06:44.113	30	236	0.00	
10:06:44.224	30	236	0.00	
10:06:44.224	30	236	0.00	
10:06:44.337	30	236	0.00	
10:06:44.454	30	236	0.00	
10:06:44.454	30	236	0.00	
10:06:44.567	30	236	0.00	
10:06:44.684	30	236	0.00	
10:06:44.684	30	236	0.00	
10:06:44.802	30	236	0.00	
10:06:44.914	30	235	0.00	
10:06:44.914	30	235	0.00	
10:06:45.027	30	235	0.00	
10:06:45.144	30	235	0.00	
10:06:45.144	30	236	0.00	
10:06:45.262	30	236	0.00	
10:06:45.366	30	236	0.00	
10:06:45.382	30	236	0.00	
10:06:45.487	30	236	0.00	
10:06:45.596	30	236	0.00	

Gambar C2. Hasil Penyimpanan Monitor Android

D. Dokumentasi Tampilan Sistem *Telemetry*

Gambar D1. Tampilan Monitor pada Android



Gambar D2. Tampilan Monitor pada Laptop