TUGAS AKHIR - TF 181801

SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

Akhmad Zainudin
NRP. 02311645000041

Dosen Pembimbing:
Dr. Imam Abadi, S.T., M.T.

DEPARTEMEN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya
2019

“Halaman ini sengaja dikosongkan”

FINAL PROJECT - TF 181801

CONDITION DETECTION SYSTEM FALLS BASED ON POSITION AND SOUND SENSOR USING FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

Akhmad Zainudin
NRP. 02311645000041

Supervisor:
Dr. Imam Abadi, S.T., M.T.

DEPARTMENT OF ENGINEERING PHYSIC
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya
2019

"Halaman ini sengaja dikosongkan"

SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

Nama Mahasiswa : Akhmad Zainudin
NRP : 02311645000041
Departemen : Teknik Fisika FTI-ITS
Dosen Pembimbing : Dr. Imam Abadi, ST, MT

Abstrak

Jatuh adalah faktor kecelakaan yang membahayakan bagi semua kalangan manusia termasuk orang yang lanjut usia(lansia). Lansia yang terjatuh menjadi perhatian yang serius karena bisa mengakibatkan kematian. Dalam mengurangi resiko kematian maka dibuatlah alat yang bisa mendeteksi jatuh orang yang terjatuh. Dengan mengetahui secara dini tentang jatuhnya seseorang akan meminimalisir tingkat kematian seseorang. Dalam peneletian sebelumnya alat pendeteksi jatuh yang dibuat hanya menggunakan 1 sensor yaitu accelerometer serta dalam metode pengambil keputusannya menggunakan *adaptive neuro fuzzy inference system (ANFIS)* dan mempunyai kekurangan yaitu tidak dapat mendeteksi jatuh orang yang jatuh kebelakang. Oleh karena itu dalam penelitian ini alat pendeteksi jatuh yang digunakan menggunakan dua jenis sensor, yaitu sensor accelerometer dan mic. Pada alat pendeteksi ini mempunyai kelebihan dibanding alat pendetsi sebelumnya yaitu bisa mendeteksi orang yang jatuh kebelakang. Untuk Hasil simulasi fuzzy PSO meningkatkan sensitifitas sebesar 7%. Pada pengujian alat dengan percobaan jatuh dengan pengambilan data pada posisi alat berada di dada dan perut menunjukkan nilai sensitifitasnya adalah 100% dan nilai spesitifitasnya adalah 100%. Namun hasilnya berbeda dengan posisi alat berada di paha dengan menghasilkan nilai sensitifitas 72% dan dan nilai spesitifitasnya adalah 100%.

Kata Kunci: Jatuh, Accelerometer, Fuzzy

"Halaman ini sengaja dikosongkan"

CONDITION DETECTION SYSTEM FALLS BASED ON POSITION AND SOUND SENSOR USING FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

Name	**:**	**Akhmad Zainudin**
NRP	**:**	**02311645000041**
Department	**:**	***Engineering Physics FTI-ITS***
Supervisor	**:**	**Dr. Imam Abadi, ST, MT**

Abstract

Falling is an accident factor that is dangerous for all people including the elderly (elderly). Elderly people who fall into serious attention because it can lead to death. In reducing the risk of death a tool can be made that can detect falling people who fall. Knowing early about someone's fall will minimize a person's death rate. In the previous research, the fall detector was made using only 1 sensor, namely the accelerometer, and in the decision-making method using adaptive neuro fuzzy inference system (ANFIS) and had the disadvantage of not being able to detect falling people who fall behind. Therefore in this study the fall detector used uses two types of sensors, namely accelerometer and mic sensors. This detector has advantages over previous detectors which can detect people who fall backwards. For fuzzy PSO simulation results increase sensitivity by 7%. In testing the tool with the fall experiment with data retrieval at the position of the device in the chest and abdomen, the sensitivity value is 100% and the specificity value is 100%. But the results are different from the position of the tool in the thigh by producing a sensitivity value of 72% and the specificity value is 100%.

Keyword: Fall, Accelorometer Fuzzy..

“Halaman ini sengaja dikosongkan”

PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini

Nama	: Akhmad Zainudin
NRP	: 02311645000041
Departemen/Prodi	: Teknik Fisika/S1 Teknik Fisika
Fakultas	: Fakultas Teknologi Industri
Perguruan Tinggi	: Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul "SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)" adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada Tugas Akhir ini maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 19 Jan 2019

Yang membuat pernyataan,

Akhmad Zainudin

NRP. 02311645000041

"Halaman ini sengaja dikosongkan"

LEMBAR PENGESAHAN I
TUGAS AKHIR

SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

Oleh:
Akhmad Zainudin
NRP. 023116450000041

Surabaya, 21 Jan 2019

Menyetujui,
Dosen Pembimbing

Dr. Imam Abadi, S.T., M.T.
NIPN. 19761006 199903 1 002

Mengetahui,
Ketua Departemen
Teknik Fisika FTI-ITS

Agus Muhamad Hatta, S.T., M.Si., Ph.D.
NIPN. 19780902 200312 1 002

“Halaman ini sengaja dikosongkan”

SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)

LEMBAR PENGESAHAN II

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada
Bidang Studi Rekayasa Instrumentasi
Progam Studi S-1 Departemen Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh:

Akhmad Zainudin
NRP. 02311645000041

Disetujui oleh Tim Penguji Tugas Akhir:

1. Dr. Imam Abadi, S.T., M.T. ..... (Pembimbing)
2. Dr. Ir. Ali Musyafa', M.Sc ..... (Penguji I)
3. Dr. Ir. Ronny Dwi Noriyanti, M.Kes ..... (Penguji II)

SURABAYA
Januari, 2019

xiii

"Halaman ini sengaja dikosongkan"

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang senantiasa melimpahkan rahmat dan hidayah-Nya serta shalawat dan salam kepada Nabi Muhammad SAW sehingga dapat terselesaikannya tugas akhir beserta laporan tugas akhir yang berjudul **"SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION)".**

Penulis telah banyak memperoleh bantuan dari berbagai pihak dalam penyelesaian tugas akhir dan laporan tugas akhir ini. Penulis mengucapkan terima kasih kepada:

1. Bapak Agus Muhamad Hatta, S.T., M.Si., Ph.D selaku Ketua Departemen Teknik Fisika ITS yang telah memberikan ilmu, bimbingan serta sarana dan prasarana selama menempuh pendidikan di Teknik Fisika ITS.
2. Bapak Dr. Imam Abadi, S.T., M.T. selaku dosen pembimbing yang telah dengan sabar memberikan petunjuk, ilmu, serta bimbingan dalam pengerjaan tugas akhir ini.
3. Ibu Lizda Johar Mawarani , ST, MT selaku dosen wali yang telah membimbing penulis selama perkuliahan dan memberikan ilmu serta nasihat yang sangat bermanfaat.
4. Seluruh dosen Teknik Fisika ITS atas segala ilmu, bimbingan, petunjuk, dan nasihat yang sangat bermanfaat.
5. Orang tua saya yang saya cintai ibu Siti Fahmi dan ayah (alm) Moh Ghufron. Kakak saya Abdul Aziz Maulana dan adik saya Sakinah Al Rasida
6. Teman kamar kos Achmadi ST. MT. dan teman makan Seno Widya Manggala ST. yang selalu mendukung dan membantu sewaktu pengerjaan tugas akhir ini
7. Teman-teman lab yang selalu membantu pengerjaan alat waktu di lab terutatama Mohammad Musa Bachrowi. ST, Moh. Danni Aditya. ST. dan teman-teman asisten lab fisis 2014, 2015 dan 2016
8. Teman-teman dari angkatan 2016 dan 2017 Lintas Jalur Teknik Fisika ITS

9. Keluarga dan teman- teman seperjuangan, atas segala cinta, kasih sayang, doa, perhatian, serta dukungan moril dan materiil yang telah diberikan.

Penulis sadar bahwa penulisan laporan tugas akhir ini masih terdapat kekurangan. Namun, semoga laporan ini dapat memberikan kontribusi yang berarti dan menambah wawasan yang bermanfaat bagi pembaca, keluarga besar Teknik Fisika ITS khususnya, dan civitas akademika ITS pada umumnya.

Surabaya, 21 Januari 2019

Penulis

DAFTAR ISI

HALAMAN SAMPUL I i
HALAMAN SAMPUL II iii
ABSTRAK v
ABSTRACK vii
PERNYATAAN BEBAS PLAGIARISME ix
LEMBAR PENGESAHAN I xi
LEMBAR PENGESAHAN II xiii
KATA PENGANTAR xv
DAFTAR ISI xvii
DAFTAR GAMBAR xix
DAFTAR TABEL xix
BAB I PENDAHULUAN
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Tujuan 3
1.4 Batasan Masalah 3
1.5 Sistematika Laporan 4
BAB II TINJAUAN PUSTAKA
2.1 Definisi Jatuh 5
2.2 Sensor Accelorometer dan Gyroscope 5
2.3 Modul Mic Max 9814 7
2.4 Modul Wifi 7
2.5 Modul Charger Baterai 8
2.6 Modul Step up Tegangan 5V 9
2.7 Baterai Lithium 3,7 V 10
2.8 STM32F103 10
2.9 Fuzzy 11
2.10 PSO (Particle Swarm Optimization) 15
BAB III METODOLOGI PENELITIAN
3.1 Studi Literatur 17
3.2 Menentukan Kegiatan Pengambilan Data 17
3.3 Perancangan Alat 26
3.4 Realisasi Alat 27
3.5 Interfacing Android 32

3.6 Pengambilan Data 33
3.7 Perancangan Fuzzy 34
3.8 Perancangan PSO 36
3.9 Pengujian Alat 40
BAB IV ANALISA DATA DAN PEMBAHASAN
4.1 Hasil Uji Selisih Waktu Sensor Accelorometer dan Mic 41
4.2 Pengambilan Data Awal Fuzzy 42
4.3 Pembuatan Batas Fuzzy dan Simulasi 44
4.4 Menentukan Batas Baru Dari PSO 48
4.5 Percobaan Alat di Dada 51
4.6 Percobaan Alat di Perut 53
4.7 Percobaan Alat di Paha 55
4.8 Perbandingan Simulasi dan Hasil Uji Alat Posisi Dada 57
4.9 Perbandingan Simulasi dan Hasil Uji Alat Posisi Perut 59
4.10 Perbandingan Simulasi dan Hasil Uji Alat Posisi Paha 61
BAB V KESIMPULAN DAN SARAN
5.1 Kesimpulan 65
5.2 Saran 65

DAFTAR GAMBAR

Gambar 2. 1 Sensor Accelorometer 5
Gambar 2. 2 Prinsip Kerja Accelorometer[13] 6
Gambar 2. 3 Sensor Mic GY MAX 9814 7
Gambar 2. 4 Modul ESP 8266 8
Gambar 2. 5 Modul Charging 8
Gambar 2. 6 Modul *Step up* 9
Gambar 2. 7 Baterai Lithium 10
Gambar 2. 8 Modul STM32F103 11
Gambar 2. 9 Diagaram alir proses fuzzy 11
Gambar 2. 10 fungsi keanggotaan segitiga 13
Gambar 2. 11 fungsi keanggotaan trapesium 13
Gambar 2. 12 Fungsi keanggotaan gaussian 14
Gambar 2. 13 Fungsi keanggotaan *generalized bell* 15
Gambar 2. 14 Pergerakan partikel pada PSO 15
Gambar 2. 15 Diagram alir sistem PSO 16
Gambar 3. 1 Pengujian Sensor 17
Gambar 3. 2 Proses perancangan alat 26
Gambar 3. 3 Skematik Alat 27
Gambar 3. 4 Ki Cad Layout bawah 28
Gambar 3. 5 Desain Bawah 3D 28
Gambar 3. 6 Board dari tampak atas 29
Gambar 3. 7 Hardware setelah pemasangan 29
Gambar 3. 8 Skematik Sistem *Power Supply* 30
Gambar 3. 9 Percobaan *power supply* 30
Gambar 3. 10 Rangkaian sensor dan elektrik 31
Gambar 3. 11 *Hardware* yang sudah di kemas 31
Gambar 3. 12 Pemberitahuan Terjatuh 32
Gambar 3. 13 Interfacing Android 32
Gambar 3. 14 Pencarian Koneksi Serial 33
Gambar 3. 15 Hasil Data Masuk 34
Gambar 3. 16 Batas batas *input* mic 34
Gambar 3. 17 Batas batas *input* accelerometer 35
Gambar 3. 18 Batas batas *output* hasil keputusan 35
Gambar 3. 19 Langkah optimasi PSO 37

Gambar 4. 1 *Input* Mic 45
Gambar 4. 2 *Input* Accelerometer 45
Gambar 4. 3 Hasil Keputusan 45
Gambar 4. 4 *Input* Mic Fuzzy PSO 49
Gambar 4. 5 *Input* Accel Fuzzy PSO 49

DAFTAR TABEL

Tabel 3.1	Percobaan Jatuh	18
Tabel 3.2	Ilustrasi dan penjelasan	22
Tabel 3.3	Rule Base Fuzzy	35
Tabel 3.4	Inisialisasi awal posisi dan kecepatan partikel	37
Tabel 3.5	Hasil perhitungan MSE tiap partikel	38
Tabel 3.6	Hasil Update posisi dan kecepatan partikel	39
Tabel 4.1	Kriteria Orang Pengambilan data	41
Tabel 4.2	Selish Waktu Mic dan Accelorometer	42
Tabel 4.3	Data Awal Jatuh Untuk Fuzzy	43
Tabel 4.4	Data Awal Aktifitas Tidak jatuh Untuk Fuzzy	43
Tabel 4.5	Hasil Simulasi Fuzzy Untuk Data Jatuh	46
Tabel 4.6	Simulasi Fuzzy Ori Untuk Data Dengan Aktifitas Tidak jatuh	47
Tabel 4.7	Hasil Fuzzy PSO Variabel Mic	48
Tabel 4.8	Hasil Fuzzy PSO Variabel Accelorometer	48
Tabel 4.9	Simulasi Fuzzy PSO Untuk Jatuh	50
Tabel 4.10	Simulasi Fuzzy PSO Untuk Kegiatan Tidak jatuh	50
Tabel 4.11	Percobaan Jatuh Hardware di Dada	52
Tabel 4.12	Percobaan Aktifitas Tidak jatuh Hardware di Dada	52
Tabel 4.13	Percobaan Jatuh Hardware di Perut	53
Tabel 4.14	Percobaan Aktifitas Tidak jatuh Hardware di Perut	54
Tabel 4.15	Percobaan Jatuh Hardware di Paha	55
Tabel 4.16	Percobaan Aktifitas Tidak jatuh Hardware di Paha	56
Tabel 4.17	Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi dada	57
Tabel 4.18	Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi dada	58

Tabel 4.19 Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi dada 59
Tabel 4.20 Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi perut 60
Tabel 4.21 Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi perut 61
Tabel 4.22 Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi perut 62

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kehidupan lansia di Indonesia meningkat seiring meningkatnya jumlah kelahiran dan meningkatnya angka harapan hidup . hasil survey oleh BPS jumlah lansia mencapai 22.4 juta atau 8.69 % pada tahun 2015[1] dan diperkirakan oleh BPS pada tahun 2018 bisa mencapai 9.3% dari penduduk seluruh indonesia atau 24.7 juta jiwa. Hal ini akan berakibat fatal apabila kehidupan lansia tidak dijaga dengan baik. Menjaga kesehatan lansia dalam usia yang sudah rentan baik di sebuah rumah atau gedung sangat perlu kehati-hatian. Contoh resiko jatuh lansia pada rumahan di Krasakan Lumbungrejo Tempel Sleman Yogyakarta dengan jumlah koresponden 100 orang, hasil yang di dapat yaitu 16 orang (41%) responden memiliki risiko jatuh sedang, 15 orang (38,5%) memiliki risiko jatuh rendah dan yang terakhir 8 orang (20,5%) memiliki risiko jatuh tinggi[2]. Namun peningakatan resiko jatuh menjadi lebih besar jika lansia terdapat pada satu gedung atau rumah panti jompo. Contoh saja penelitian di panti social Tresna Werdha Budi Mulia 4 Margagguna, Jakarta Selatan dengan jumlah korespondsi 51 orang. Resiko jatuh dengan aktifitas fisik 73.7%[3]. Sehingga perhatian pada lansia harus sangat diperhatikan dan apabila lengah sedikit saja para lansia bisa saja terjatuh dan berakibat fatal. Dalam hal ini perhatian untuk di rumah di panti sosial menjadi kurang, ini bisa dilihat resiko jatuh pada lansia adalah 73.7%

Sehubungan dengan pentingnya akan pendeteksi jatuh pada lansia maka alat yang tersedia terus berkembang mulai dari hardware sampai *software* yang digunakan. Contohnya saja pada tahun 2012 telah dibuat penelitian pendeteksi jatuh pada lansia menggunakan sistem fuzzy dan apabila terdeteksi dikomunikasikan melalui *wireless xbee*[4] dan hasilnya sistem ini mempunyai akurasi minimal 86,6%. Lalu pada tahun 2013 juga dibuat sistem pendeteksi jatuh dengan menggunakan fuzzy namun

tidak mengirimkan data dan sekedar alarm di tubuh objek. Untuk akurasinya mencapai 92%[5]. Pada tahun 2014 untunk penelitian pendeteksi jatuh namun menggunakan type 2 fuzzy namun keakuratan justru hanya 80%[6]. Untuk tahun 2015 juga telah dibuat penelitian deteksi sistem jatuh menggunakan high level fuzzy petri net[7]. Dan untuk sistem ini akurasiannya mencapai diatas 90%. Untuk tahun 2016 dilakukan penelitian tentang teknik deteksi untuk jatuh menggunakan fuzzy model takagi sugeno dengan sensor accelerometer [8] lalu pada tahun 2018 terdapat penelitian menggunakan metode non-intrusive deteksi jatuh pada lansia menggunakan logika fuzzy[9] yang terdapat dua sensor untuk mendeteksi jatuh. Dua sensor ini yaitu sensor accelerometer dan sensor suara. Apabila terdeteksi maka akan dikirimkan melalui IoT ke perawat yang menjaga. Untuk sistem ini tingkta ke akurasian mencapai 80%.

Pada perkembangan alat pendeteksi jatuh dengan sensor accelerometer dan gyroscope yang dibuat oleh Luthfi Fathurrahman sebelum memakai sistem *adaptive neuro fuzzy inference system* (*ANFIS*) hanya mempunyai sensitifitas 80% dan spesitifitas 88%. Ketika memakai sistem *adaptive neuro fuzzy inference system* (*ANFIS*)[10] mempunyai kenaikan ketika nilai *range of influence* sebesar 0,1 maka sensitiitas menjadi 100% dan spesitifitas menjadi 91.67%. Untuk *range of influence* sebesar 0,05 maka sensitifitas dan spesitifitas menjadi 100%. Dalam sistem sistem *adaptive neuro fuzzy inference system* (*ANFIS*) ini mempunyai kekurangan yaitu belum bisa mendeteksi ketika objek terjatuh ke belakang sehingga nilai inilah yang harus di optimasi.

Memaksimalkan keakurasi pendeteksian objek ketika terjatuh merupakan h hal yang wajib yang harus dilakukan. Salah satu langkah dalam pendeteksian seperti menambah sensor. Sensor yang dimaksud adalah sensor suara, seperti pada jurnal yang dibuat oleh Poi Voon Er dan Kok Kiong Tan[9] kesalahan bisa ditekan sampai 20% dan penambahan fitur optimasi untuk mengurangi kesalahan. Optimasi merupakan proses untuk memaksimalkan suatu perhitungan tertentu[11]. Memaksimalkan suatu perhitungan tertentu maksudnya sistem fuzzy dalam penelitian sebelumnya

tidak bisa mendeteksi jatuh kebelakang. Sehingga harapannya optimasi berjalan maksimal dan tidak ada kekurangan untuk logaritma fuzzy. Optimasi sendiri harus dilakukan dengan penuh percobaan dan perhitungan agar nilai hasil benar-benar merupakan nilai maksimal yang bisa dilakukan.

1.2 Rumusan Masalah

Adapun masalah dalam melakukan penelitian Tugas Akhir ini adalah sebagai berikut:

a. Bagaimanakah cara menentukan karakteristik kondisi orang saat terjatuh berdasarkan kegiatan yang sudah ditentukan?
b. Bagaimana cara menentukan performance dari sistem pendeteksi kondisi terjatuh berbasis accelerometer menggunakan fuzzy *PSO (Particle Swarm Optimization)*?

1.3 Tujuan

Adapun tujuan penelitian yang ingin dicapai dalam pengerjaan tugas akhir ini adalah sebagai berikut:

a. Menentukan karakteristik kondisi orang saat terjatuh berdasarkan kegiatan yang sudah ditentukan
b. Menentukan performance dari sistem pendeteksi kondisi terjatuh berbasis accelerometer menggunakan f*uzzy PSO (Particle Swarm Optimization)*

1.4 Batasan Masalah

Adapun batasan masalah yang digunakan dalam pengerjaan tugas akhir ini adalah sebagai berikut :

a. Sensor yang digunakan adalah accelerometer dan mic modul
b. Metode yang digunakan dalam sistem pendeteksi terjatuh adalah *PSO (Particle Swarm Optimization)*
c. Jenis kegiatan/ aktifitas objek untuk tujuan pengambilan data sudah ditentukan seperti berjalan, jogging, berdiri dan lain-lain

d. Penempatan sensor untuk tujuan pengambilan data berada pada tiga posisi yaitu dada, perut dan paha
e. Komunikasi hardware dan android melalui koneksi wifi

1.5 Sistematika Laporan

Sistematika penulisan laporan tugas akhir adalah sebagai berikut:

a. Bab I Pendahuluan
   Pada bab I ini terdiri dari latar belakang, rumusan masalah, tujuan, batasan masalah, dan sistematika laporan.
b. Bab II Teori Penunjang
   Pada bab II ini dibahas mengenai teori-teori yang berkaitan dengan penelitian yang akan dilakukan, seperti teori accelerometer, mic, modul wifi, arduino dan android.
c. Bab III Metodologi Penelitian
   Pada bab III ini berisi mengenai rancangan dari penelitian yang dilakukan, metode, dan langkah-langkah dalam penelitian.
d. Bab IV Analisis Data Dan Pembahasan
   Pada bab IV ini berisi tentang analisis hasil perancangan sistem fuzzy PSO
e. Bab V Kesimpulan Dan Saran
   Pada bab V ini diberikan kesimpulan tentang tugas akhir yang telah dilakukan berdasarkan data-data yang diperoleh serta diberikan saran sebagai penunjang maupun pengembangan untuk tugas akhir yang selanjutnya.

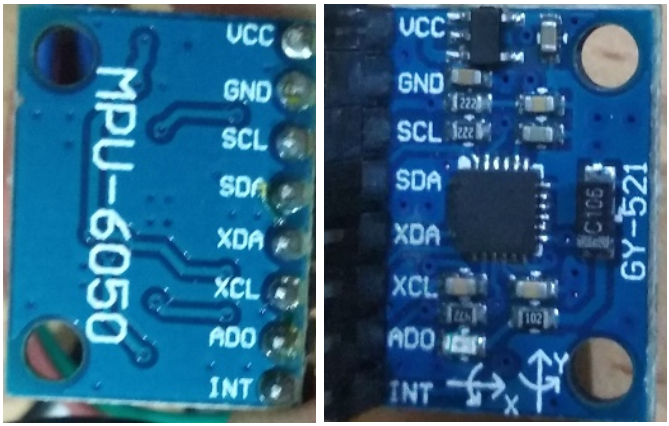
BAB II
TINJAUAN PUSTAKA

2.1 Definisi Jatuh

Menurut Kamus Besar Bahasa Indonesia atau yang biasa disebut adalah jatuh/ja·tuh/ yaitu (terlepas dan) turun atau meluncur ke bawah dengan cepat karena gravitasi bumi (baik ketika masih dalam gerakan turun maupun sesudah sampai ke tanah dan sebagainya. Dalam kecelekaan, jatuh adalah suatu keadaan dimana seseorang terdorong di lantai karena suatu kondisi badan yang lemah karena stroke, kejang, pingsan, ataupun karena dorongan hal lain seperti benda atau manusia atau sekelilingnya.

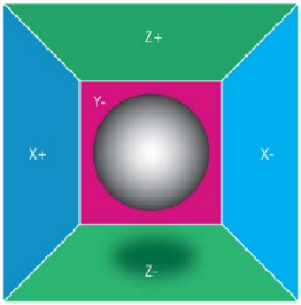
2.2 Sensor Accelorometer dan Gyroscope

Sensor Accelorometer adalah sensor yang digunakan untuk mengukur percepatan, getaran dan gravitasi[12]. Sensor Acceloromoter punya banyak type IC yang digunakan, salah satunya adalah IC MPU 6050. Sensor ini mempunyai 3 sumbu sebagai pengukurannya yaitu sumbu X, sumbu Y dan sumbu Z. dan untuk satuan dari pengukuran accelorometer ini adalah Untuk sensor modulnya bisa dilihat pada gambar 2.1



Gambar 2. 1 Sensor Accelorometer

Cara kerjanya untuk sensor accelorometer ini adalah adalah menggunakan lempengan. Lempengan tersebut dialiri aliran listrik. Ketika terjadi perubahan gerakan pada bagaian-bagaiannya maka bagaian itu terjadi perubahan kapasistif. Lalu perubahan kapasitif inilah yang menjadi tegangan *output*. Untuk gambaran cara kerja untuk sensor acceloromoter bisa dilihat pada gambar 2.2



Gambar 2. 2 Prinsip Kerja Accelorometer[13]

Pada gambar 2.2 menggambarkan suatu bola pada kotak yang mempunyai sumbu X, Y dan Z. Setiap ada gerakan tertentu bolah pasti menyentuh beberapa sumbu tersebut. Sentuhan bisa pada sumbu X, Y atau Z atau bisa sentuhan tersebut pada sumbu positif atau negatif.

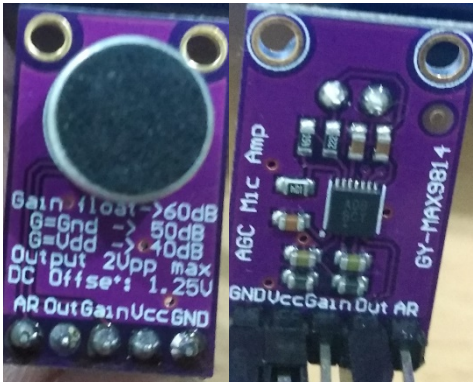
Pada gambar 2.1 GY521 karena sudah terintegrasi dengan accelerometer maka modul ini mempunyai 3 sumbu yaitu X, Y dan Z. untuk IC yang digunakan adalah MPU 6050. Chip ini mempunyai banyak fitur salah satunya adalah sudah memiliki digital motion prosesor (DMP) yang mengolah data mentah dari masing-masing sensor dan dapat menimalisasi error yang dihasilkan.[14]

Accelerometer adalah alat sensor yang peka terhadap gerak linier terhadap 3 sumbu gerakan yakni kanan-kiri, atas-bawah dan depan belakang. Aplikasi Accelerometer sudah banyak di sekeliling kita contohnya pada sebuah perangkat handphone yang sekarang ini menggunakan accelerometer untuk secara otomatis mengubah arah layar (menjadi miring ataupun terbalik) sesuai dengan arah monitor tersebut ditegakkan (portrait atau landscape). Selain itu terdapat

juga sejumlah merk handphone yang menggunakan accelerometer untuk mengubah lagu yang dimainkan (*Track Switching*).

2.3 Modul Mic Max 9814

Mic adalah suatu perangkat yang mengubah gelombang suara menjadi gelombang elektrik. Cara kerja mic ini adalah terdapat membran dan membran inilah yang menghasilkan listrik. Kecepatan bergeraknya membran inilah yang menghasilkan listrik yang dihasilkan. Listrik yang dihasilkan sesuai datasheet adalah 3mV sampai dengan 2,45V. Dari tegangan tersebut masuk ke ADC *microcontroller*. Untuk modul MAX 9814 dapat dilihat pada gambar 2.3



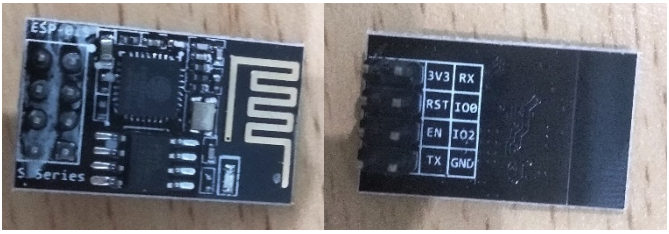
Gambar 2. 3 Sensor Mic GY MAX 9814

Pada gambar 2.3 adalah modul mic yang menerapkan penguatan internal. Sensor mic GY MAX 9814 tegangan yang dibutuhkan dalam pengoperasian yaitu 2,7 V~5,5V. Operasi tegangan ini lebih dari cukup untuk diberi tegangan dalam *microcontroller*. Tegangan *supply* dari *microcontroller* lebih dari cukup untuk mengoperasikan mic. Dalam penggunaan untuk gain internal lebih dari cukup untuk merpresentasikan data.

2.4 Modul Wifi

Modul WI-Fi ESP 8266 Arduino. Wi-Fi adalah kepanjangan dari *wireless fidility* adalah suatu media tanpa kabel yang bisa menghubungkan suatu perangkat atau beberapa perangkat ke jaringan internet atau *local area network* (LAN).

Perangkat ini yang menghubungkan antara modul Wi-Fi arduino dengan ponsel android. Sinyal transfer dari wifi android lalu melalui router dan diterima oleh ponsel.



Gambar 2. 4 Modul ESP 8266

Dalam modul ESP 8266 mempunyai dua tipe fungsi. Fungsi pertama digunakan sebagai akses point atau sebagai jalur internet untuk modul mikrokontroller. Fungsi kedua adalah melalui jalur wifi ke alat lain.

2.5 Modul *Charger* Baterai

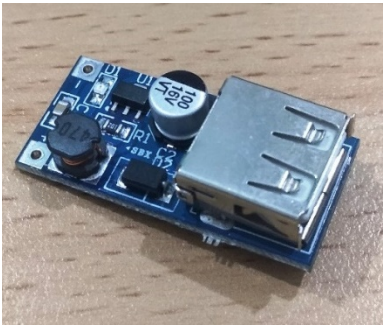
Modul *charger* baterai adalah adalah sebuah rangkaian yang berfungsi untuk mengubah tegangan masukan menjadi tegangan charging. Tegangan *input* adalah 5VDC sedangkan tegangan charging adalah 4,2 VDC dengan arus maksimal datasheet adalah 700 mA[15]. Dalam modul ini ada 3 *integrated circuit* penting.

Gambar 2. 5 Modul Charging

Dalam IC tersebut mempunyai peranan penting. IC utama adalah IC LTC 4056-4,2V adalah mempunyai peranan penting dalam mengubah tegangan dari 5V menjadi 4,2V. IC kedua adalah IC dengan tipe DWG01-G adalah menjaga arus keadaan baterai[16]. Untuk IC dengan tipe S-8205A/B adalah menjaga tegangan[17] yanga masuk ke baterai agar daya tahan baterai tahan lama.

2.6 Modul *Step up* Tegangan 5V

Modul *step up* ini berfungsi untuk mengubah tegangan baterai ke tegangan normal system. Cara kerjanya adalah seperti pada trafo yaitu arus yang ada dilakukan konversi sehingga menambahan tegangan. Tegangan baterai adalah 3,7VDC sedangan tegangan normal system adalah 5V. Namun dalam pengoperasiannya ada tegangan 3,3 V. Dalam modul terdapat ada 1 IC penting dan 1 tipe dioda dalam modul ini.



Gambar 2. 6 Modul *Step up*

IC dalam modul ini adalah IC tipe SOT23. Fungsi IC tersebut adalah mengubah tegangan dari 3,7VDC menjadi tegangan 5VDC[18]. Sedangkan dioda adalah berfungsi untuk keselamatan [19] tegangan keluaran agar apabila salah kutub tidak terjadi hubung singkat. Jika terjadi kutub yang tidak seharusnya maka dioda langsung melakukan pemutusan arus sehingga tidak ada komponen yang terjadi kerusakan.

2.7 Baterai Lithium 3,7 V

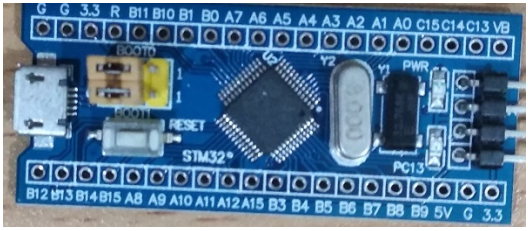
Baterai adalah suatu benda yang didalamnya terdapat bahan kimia yang berfungsi menyimpan tenaga listrik. Baterai mempunyai beberapa tipe jenis baterai seperti baterai Zinc carbon, baterai alkaline, baterai lithium, baterai silver oxide, baterai zinc air cell. Untuk baterai yang dapat diisi ulang adalah baterai rechargeable alkaline, baterai Ni-CD, baterai NI-MH, baterai lithium Ion, baterai lithium polimer, baterai liad acid.

Gambar 2. 7 Baterai Lithium

Dalam gambar tersebut adalah baterai lithium dengan kapasitas penyimpanan 1300 mAh. mAh artinya jika baterai tersebut mengalirkan arus 1,3 A maka tenaga listrik akan habis dalam 1 jam. Untuk tegangan baterai adalah 3,7 V. Sehingga diperlukan penambahan modul seperti *step up* yang berfungsi untuk menaikkan ke sistem umum tegangan. Pada modul *step down* digunakan pada waktu *charge* yang merubah tegangan yang sebelunya 5V menjadi tegangan 3,7V.

2.8 STM32F103

STM32F103 adalah salah satu jenis *microcontroller* dari ARM. Dalam minimum system ini Tidak jatuh disebut blue pill. Perbedaan mendasar STM dengan arduino adalah arduino Tidak jatuh untuk rangkaian atau sistem yang simple sampai menengah sedengkan untuk STM Tidak jatuhnya dari simple sampai tinggi. Hal ini terlihat dari sistem dalam *microcontroller* antara STM dengan arduino. Untuk STM bisa melakukan multi proses sedangkan untuk arduino hanya bisa 1 proses namun berulang. Sehingga proses pada pengambilan data lebih fleksibel.

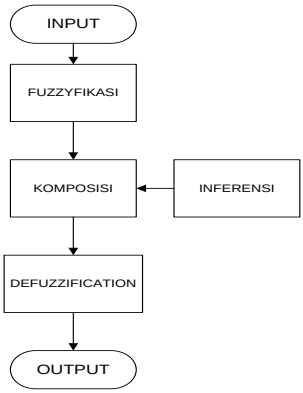


Gambar 2. 8 Modul STM32F103

Dalam STM ini mempunyai core di ARM® 32-bit Cortex®-M3[20] CPU Core dengan fitur single-cycle multiplication and hardware. Division memories 64 or 128 Kbytes of Flash memory dan 20 bytes of SRAM untuk fitur tambahnnya yaitu Clock, reset and *supply* management untuk powernya adalah 2.0 to 3.6 V

2.9 Fuzzy

Fuzzy adalah peningkatan konsep dari logika boolean yang mengenalkan konsep kebenaran sebagaian. Konsep ini menyatakan segala hal bisa di istilahkan dalam binary yaitu 1 dan 0. Dalam hal ini logika fuzzy menggantikan teori boolean dalam tingkat kebenaran. Dalam logika fuzzy nilai keanggotaan dapat dikelompokkan seperti rendah, sedang dan sangat tinggi.Untuk lebih jelasnya proses fuzzyfikasi bisa dilihat pada gambar 2.9



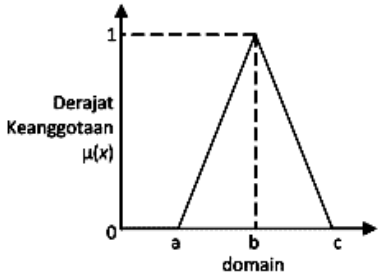
Gambar 2. 9 Diagaram alir proses fuzzy

a. *Input* adalah nilai awal variable untuk logika fuzzy yang nanti berupa bilangan *crisp*
b. Fuzzyfication adalah suatu proses berguna untuk mengkonversi bilangan crisp menjadi suatu bilangan fuzzy (variable linguistik) menggunakan operator fuzzifikasi. Proses fuzzifikasi merupakan ciri utama dalam penalaran logika fuzzy, karena terdapat fungsi keanggotaan fuzzy (membership function) yang berfungsi untuk menentukan letak keberadaan elemen dalam suatu himpunan fuzzy.
c. Rule Base & Inference adalah suatu system yang berisikan aturan-aturan dasar logika fuzzy yang berfungsi untuk membuat hasil keputusan yang dimuat dalam aturan "jika" dan "maka"
d. Defuzzyfication adalah suatu keputusan yang dihasilkan dari proses penalaran masih dalam bentuk fuzzy, yaitu berupa derajat keanggotaan keluaran. Hasil ini harus diubah kembali menjadi bilangan *crisp* melalui proses defuzzyfikasi.
e. *Output* adalah suatu bentuk mitigasi yang harus dilakukan

Pada Gambar 2.9 terlihat mekanisme dari proses dari *input* yang dibutuhkan untuk fuzzy logic dari data awal sampai dari hasil fuzzy logic. Untuk yang pertama data-data yang dibutuhkan adalah data batasan untuk setiap kelompok cluster sehingga nantinya dikelompokan menurut kategorinya.

Sebuah himpunan fuzzy dapat dikarakterisasikan seluruhnya melalui fungsi keanggotaaanya. Fungsi keanggotaan pada satu dimensi atau single-*input* dapat direpresentasikan ke dalam 4 bentuk, yaitu fungsi keanggotaan segitiga, fungsi keanggotaan trapesium, fungsi keanggotaan Gaussian, dan fungsi keanggotaan generalized Bell.

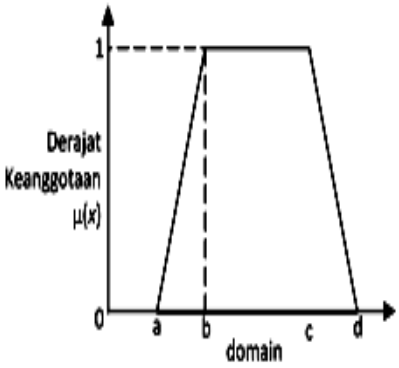
a. Fungsi Keanggotaan Segitiga adalah suatu fungsi keanggotaan segitiga dibentuk oleh 3 parameter {a, b, c} yang dapat dideskripsikan oleh Persamaan (2.1)



Gambar 2. 10 fungsi keanggotaan segitiga

$$\mu[x] = 0; x \leq a \; atau \; x \geq c$$
$$\mu[x] = \frac{(x-a)}{(b-a)}; a < x \leq b$$
$$\mu[x] = \frac{(c-x)}{(c-b)}; b < x \leq c$$
$$\mu[x] = \max\left(min\left(\frac{(x-a)}{(b-a)}, \frac{(c-x)}{(c-b)}\right), 0\right) \quad \text{.....(2.1)}$$

b. Fungsi Keanggotaan Trapesium fungsi keanggotaan trapesium dibentuk oleh 4 parameter {a, b, c, d} yang dapat dideskripsikan oleh Persamaan (2.2)



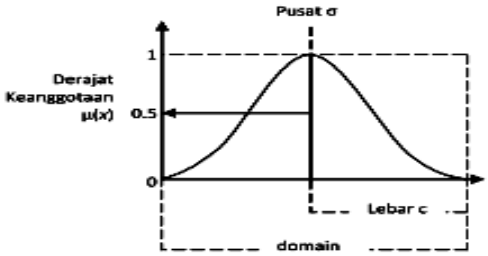
Gambar 2. 11 fungsi keanggotaan trapesium

$$\mu[x] = 0; x \leq a \; atau \; x \geq c$$

$$\mu[x] = \frac{(x-a)}{(b-a)}; a < x \leq b$$
$$\mu[x] = 1; \quad b \leq x \leq c$$
$$\mu[x] = \frac{(d-x)}{(d-b)}; c < x \leq d$$
$$\mu[x] = \max\left(min\left(\frac{(x-a)}{(b-a)}, 1, \frac{(d-x)}{(d-c)}\right), 0\right) \quad \text{(2.2)}$$

c. Fungsi keanggtaan gaussian adalah fungsi keanggotaan Gaussian dibentuk oleh 2 parameter {c, σ}. c merepresentasikan pusat fungsi keanggotaan dan σ merepresetasikan lebar ruang fungsi keanggotaan. Fungsi keanggotaan Gaussian dapat dideskripsikan oleh Persamaan (2.6)

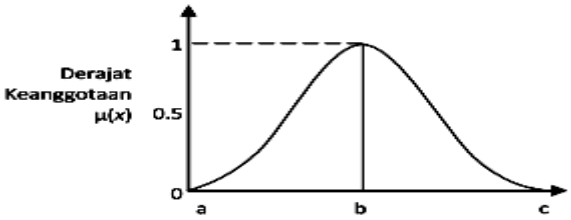
$$\mu[x] = e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2} \quad \text{(2.3)}$$



Gambar 2. 12 Fungsi keanggotaan gaussian

d. Fungsi keanggotaan *generalized bell* Fungsi keanggotaan generalized bell dibentuk oleh 3 parameter {a, b, c} dimana nilai b bernilai positif dan nilai b negatif mengakibatkan kurva terbuka ke atas. Fungsi keanggotaan *generalized bell* dapat dideskripsikan oleh Persamaan (2.4)

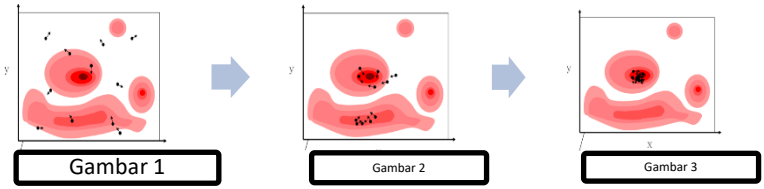
$$\mu[x] = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \quad \text{(2.4)}$$



Gambar 2. 13 Fungsi keanggotaan *generalized bell*

2.10 PSO (Particle Swarm Optimization)

PSO yang mempunyai kepanjangan Particle Swarm Optimization adalah sistem yang menggunakan salah satu sistem optimasi yang terinspirasi oleh kumpulan burung dan ikan[21]. Dalam cara kerjanya partikel atau keanggotaan ini mencari kecepatan dan tempat terdekat untuk makannya. Dalam hal ini makanannya berarti tujuan dari partikel-partikel tersebut.



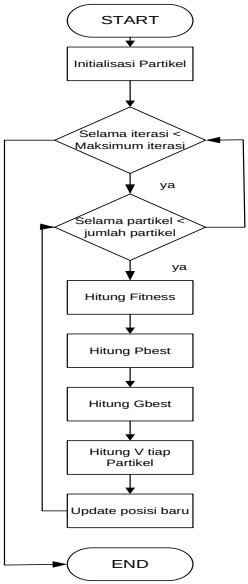
Gambar 2. 14 Pergerakan partikel pada PSO

Pada gambar 2.14 menjelaskan tentang fungsi minimal pada pergerakan partikel. Pada dasarnya sistem optimasi ini dituntut untuk mencari kecepatan terbaik dan jarak terpendek pada tujuan yag telah ditentukan. Suatu individu atau partikel ditentukan dan posisi diartikan sebagai vector. Jadi pada PSO digunakan untuk mencari nilai titik optimum pada data yang ada. Untuk diagram alir bisa dilihat pada gambar 2.15

Optimasi variable setiap data pasti berubah sehingga partikel akan mengingat posisi terbaik dan kecepatan terbaik berdasarkan informasi yang diterima. Tidak jatuhnya dalam kawanan burung

akan mengikuti batasan seperti seekor burung tidak akan terlalu dekat atau terlalu jauh dengan yang lain. Setiap burung akan mengarahkan mengikuti arah kawanan burung yang lain.

PSO merupakan adalah metode optimasi artificial intelligence yang mengadopsi suatu perilaku sosial kawanan burung atau ikan. Sistem sosial dari organisme tersebut baik individu maupun kawanan (swarm) dijadikan sebagai dasar dalam merancang algoritma PSO. Setiap solusi dapat dianggap sebagai partikel atau seekor burung.



Gambar 2. 15 Diagram alir sistem PSO

Pada diagram alir 2.15 merupakan optimisasi dari PSO. Cara kerjanya yaitu nilai yang ada akan Burung akan mencari makanannya melalui usahanya sendiri dan kerja sama sosial dengan kawanannya. Algoritma PSO pertama kali diperkenalkan oleh R. Eberhard dan J. Kennedy pada tahun 1995

BAB III
METODOLOGI PENELITIAN

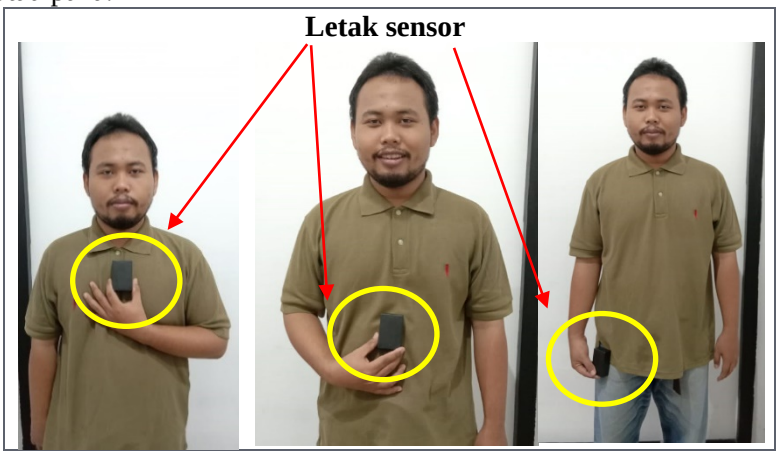
Tahapan-tahapan yang akan dilaksanakan pada tugas akhir ini adalah sebagai berikut :

3.1 Studi Literatur

Studi literatur digunakan untuk meningkatkan pemahaman peneliti mengenai topik yang dipilih. Studi literatur dilakukan dengan mempelajari hal – hal yang mendukung selama penelitian seperti Fuzzy dan sistem optimasi. Hal ini digunakan untuk mendetailkan dan gambaran tugas akhir kedepannya sehingga dalam perjalanan

3.2 Menentukan Kegiatan Pengambilan Data

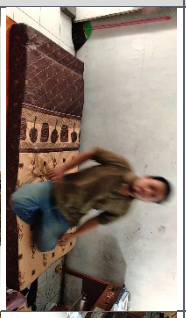
Dalam melakukan pengujian alat maka dilakukan pengujian denganan 3 macam penempatan. Penempatan pertama di dada, posisi kedua diperut dan ketiga adalah dipaha. Pengujian ini dilakukan dimana posisi terbaik dari akurasi sensor terbaik. Melakukan pencarian posisi terbaik dengan cara membadingkan hasil sensitifitas dan spesitiftas dari peletakan sensor di dada, perut atau paha.



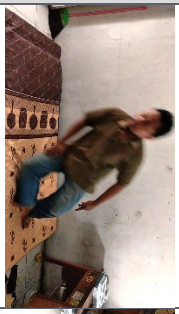
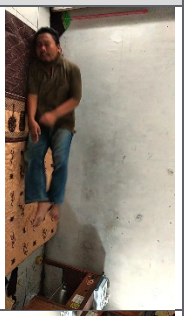
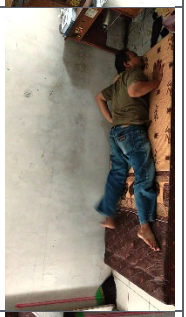
Gambar 3. 1 Pengujian Sensor

Dalam melakukan rekam karakteristik terjatuh dilakukan dokumentasi percobaan terjatuh. Rekam karakterisitik adalah nilai variable sensor yang memenuhi nilai jatuh. Variabel sensor yang dimaksud adalah nilai sensor pada accelerometer dan nilai ADC mic. Dari hasil rekam mic dan accelerometer dilakukan analisa nilai minimal yang bisa dikatakan jatuh. Dalam pengujian jatuh ada 5 jenis jatuh yang dilakuakn untuk percobaan jatuh seperti pada tabel 3.1

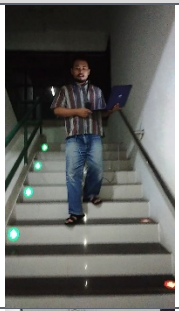
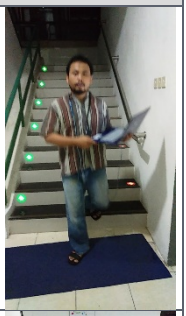
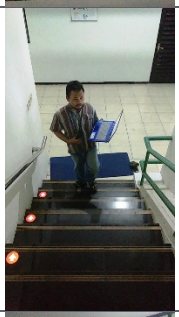
Tabel 3. 1 Percobaan Jatuh

No	Aktifitas	Gambar		
		1	2	3
1	Jatuh Vertikal			
2	Jatuh ke kiri			

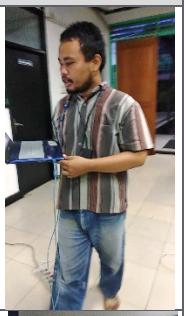
Tabel 3.1 Lanjutan

No	Aktifitas	Gambar		
		1	2	3
3	Jatuh ke kanan			
4	Jatuh Ke depan			
5	Jatuh ke Belakang			

Tabel 3.1 Lanjutan

No	Aktifitas	Gambar		
		1	2	3
6	Turun Tangga			
7	Naik Tangga			
8	Duduk Berdiri			

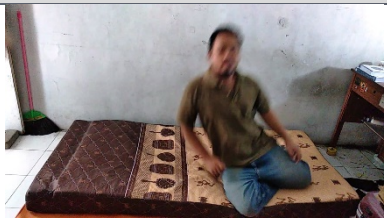
Tabel 3.1 Lanjutan

No	Aktifitas	Gambar		
		1	2	3
10	Berjalan			
11	Berdiri Membungkuk			
12	Berbaring Berdiri			

Pada tabel 3.1 dilakukan juga pengujian pada aktifitas pada keaadaan aktifitas harian. Aktifitas harian seperti naik tangga, turun tangga, duduk lalu berdiri. Lalu berjalan dalam keadaan kecepatan Tidak jatuh lalu ditambahkan dengan aktifitas ekstrim pada manula seperti berlari kecil. Tidak lupa seperti mengambil barang di bawah maka dilakukan penambahan aktifitas seperti berdiri lalu

membungkuk dan berdiri lagi. Pada bangun tidur dilakukan pengujian pada aktifitas pada tidur lalu duduk dan melanjutkan berdiri. Sedangkan penjelasan detail pada tabel 3.2

Tabel 3. 2 Ilustrasi dan penjelasan

No	Aktifitas	Repetisi	Ilustrasi dan Penjelasan
1	Jatuh Vertikal	3x	 Posisi tegak lurus lalu jatuh dominan vertikal kebawah sehingga objek duduk sementara dan diakhiri dengan posisi berbaring
2	Jatuh ke kiri	3x	Posisi tegak lurus lalu jatuh kea rah kiri sehingga objek berbaring dengan bahu kiri menyentuh lantai dan posisi kaki agak tertekuk
3	Jatuh ke kanan	3x	Posisi tegak lurus lalu jatuh ke arah kekanan sehingga objek berbaring dengan bahu kanan menyentuh lantai dan posisi kaki agak tertekuk

Tabel 3. 2 Lanjutan

No	Aktifitas	Repetisi	Ilustrasi dan Penjelasan
4	Jatuh Ke depan	3x	Posisi tegak lurus lalu jatuh kedepan kebawah sehingga objek duduk sementara dan diakhiri dengan posisi berbaring bagian tubuh bagian depan (muka, dada dan perut) menghadap tanah
5	Jatuh ke Belakang	3x	Posisi tegak lurus lalu jatuh kebelakang sehingga objek dengan posisi berbaring bagian tubuh bagian depan (muka, dada dan perut) menghadap atas (langit)
6	Turun Tangga	3x	Menuruni tangga dari lantai dua menuju satu

Tabel 3. 2 Lanjutan

No	Aktifitas	Repetisi	Ilustrasi dan Penjelasan
7	Naik Tangga	3x	Menaiki tangga dari lantai satu menuju dua
8	Duduk Berdiri	3x	Duduk dari kursi sekitar 50 cm lalu berdiri
9	Berlari Kecil	3x	Berjalan cepat sekitar 10 meter

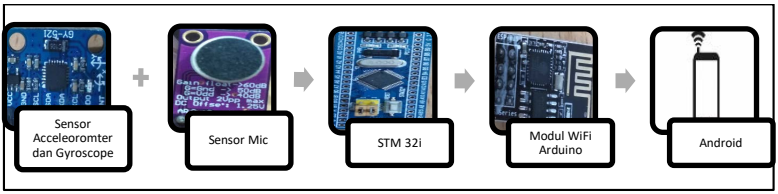
Tabel 3. 2 Lanjutan

No	Aktifitas	Repetisi	Ilustrasi dan Penjelasan
10	Berjalan	3x	Berjalan tenang sekitar 10 meter
11	Berdiri Membungkuk	3x	Berdiri tegak lurus lalu membungkuk seperti mengambil sesuatu dan berdiri lagi
12	Berbaring Berdiri	3x	Berbaring dengan posisi tidur lalu duduk sementara dan berdiri.

Dalam aktifitas seperti pada tabel 3.2dilakukan pengujian 3 kali agar apabila terjadi kesalahan pengukuran maka kesalahan yang dilakukan tidak terlalu besar. Dalam melakukan pengujian dilakukan gerakan senatural mungkin agar mendekati aktifitas seperti kehidupan normal orang biasanya.

3.3 Perancangan Alat

Pada perancangan alat untuk deteksi jatuh dibutuhkan sensor accelerometer dan gyroscope. Arduino, modul Wifi dan Android. Fungsi dari accelerometer adalah fungsi mendeteksi arah dan kecepatan perubahan gaya vektor. Pada mic adalah berfungsi untuk mendeteksi suara yang masuk dan mengubahnya menjadi tegangan. Lalu pada STM32 melakukan proses kontrol fuzzy dan mengirim data ke android melalui modul wifi. Untuk bagan seperti pada gambar dibawah ini.

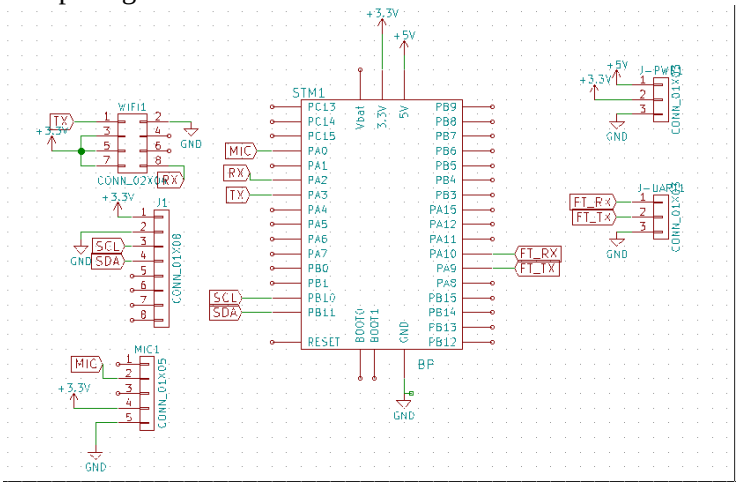


Gambar 3. 2 Proses perancangan alat

Pada Gambar 3.2 penggunaan sensor digunakan sebagai posisi pada sumbu X, Y dan Z. posisi ini digunakan sebagai apakah objek dalam posisi berdiri, duduk, tertidur ataupun posisi lainnya seperti jongkok ditambahkan dengan sensor suara sehingga penambahan sensor suara ini bisa mepertambah ke akurasian untuk sensor jatuh ini.Lalu nilai yang keluar dari sensor diteruskan dan diproses oleh arduino. Arduino tersebut akan memproses secara fuzzy PSO. Setelah pemrosesan dalam fuzzy PSO maka apakah posisi objek tersebut terjadi jatuh secara valid atau tidak. Jika posisi objek jatuh dan dinyatakan valid maka arduino mengirimkan sinyal ke ke modul Wi-Fi dan dan dikomunasikan ke ponsel android yang terhubung.

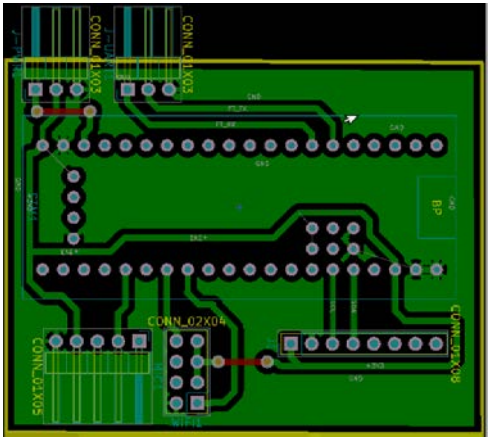
3.4 Realisasi Alat

Pada proses ini adalah teknis pembuatan alat. Sensor mic dan accelerometer dirakit dan modul wifi dijadikan satu berbentuk *shield* seperti pada arduino. Pada pembuatan *schematic* ini menggunakan *software* ki-cad. Ki-cad adalah *software* untuk melakukan design PCB. Namun pada *software* ini adalah gratis. Awal Pembuatan ini dimulai dari di wiring dahulu melalui *schematic*. Sinyal sensor dimulai didahulukan daripada untuk wiring power dan semua wiring dan disimpan. Untuk gambar bisa dilihat pada gambar 3.3



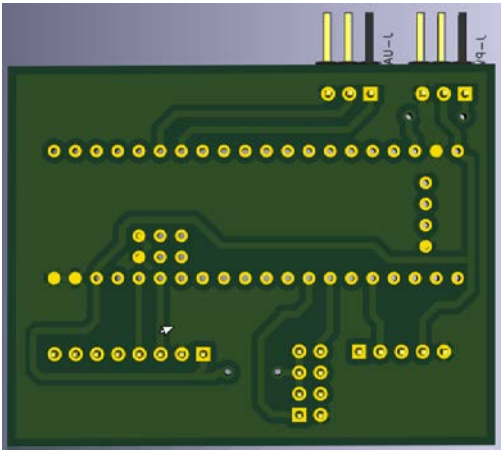
Gambar 3. 3 Skematik Alat

Pada gambar 3.3 terlihat pada keluaran mic pada area bawah kiri dihubungkan pada STM di pin A0 yang difungsikan sebagai penerima ADC. Untuk accelerometer terdiri dari dua *output* yaitu SCL dan SDA. SCL dihubungkan pada STM maka masuk ke pin B10 dan di SDA masuk ke B11. Pada modul wifi pada komunikasi penerima pada wifi dilakukan secara silang. Pada TX wifi dihubungkan pada RX STM dan untuk TX STM dihubungkan pada RX modul wifi. Untuk tegangan sistem dihubungkan ke 3,3 V dan untuk ground nya diberikan tegangan 0V. Setelah dihubungkan semua maka bisa dilakukan schematic PCB board seperti pada gambar 3.4

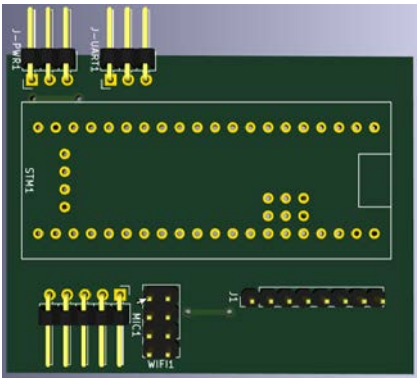


Gambar 3. 4 Ki Cad Layout bawah

Setelah masuk ke mode board maka dilakukan penyambunaggan seperti garis garis bantuan lalu tugas kita adalah menyambungkkan seperti selera kita. Paling penting adalah batas antara jalur satu dengan jalur lain tidak terlalu dekat agar waktu dalam pembuatan PCB nya tidak tercampur atau short. Berikut adalah simulasi 3D untuk board dari tampak atas dan bawah pada gambar 3.5 dan 3.6

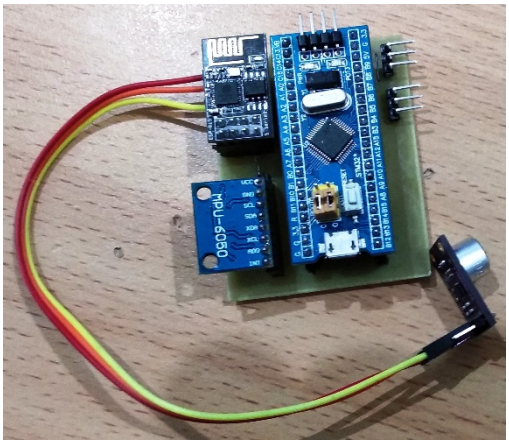


Gambar 3. 5 Desain Bawah 3D



Gambar 3. 6 Board dari tampak atas

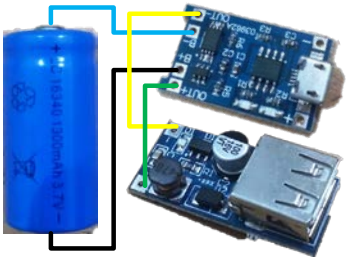
Setelah pembuatan PCB selesai maka dilakukan pensolderan pada PCB. Pensolderan komponen yang dimaksud adalah pin header female, pin header bengkok dan pin header lurus. Setelah semua selesai tinggalah pemasangan pada blue pill, ESP 8266, MPU 6050 dan mic.



Gambar 3. 7 Hardware setelah pemasangan

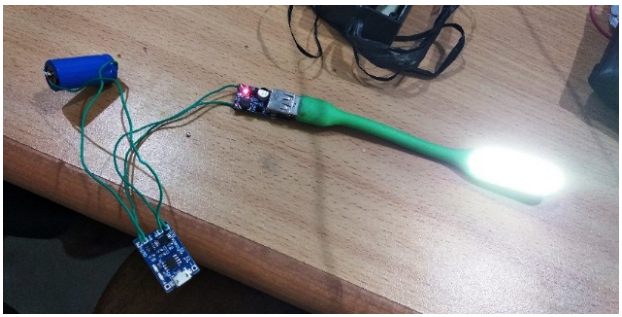
Setelah pemasangan dari sensor, alat komunikasi dan *microcontroller* maka diperlukan alat yang dapat menunjang sistem kelistrikan. Kelistrikan yang dimaksud adalah alat dari kesuluruhan sistem adalah tanpa kabel sehingga diperlukan kelistrikan dari internal sistem itu sendiri. Maka dari itu

kelistrikan yang dibuat dipasok dari baterai lithium 3,7 V dan diberikan pengangkat tegangan dari 3,7 V ke 5 V. 5V diperluakan untuk operasional sistem *microcontroller* dan sensor. Apabila baterai habis bisa dilakukan charging. Untuk skematik bisa dilihat pada gamabar 3.8



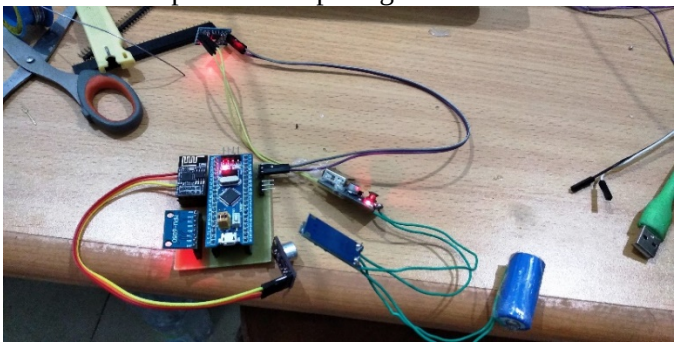
Gambar 3. 8 Skematik Sistem *Power Supply*

Dari hasil skematik terlihat baterai dihubungkan parallel antara modul charging dengan mosul *step up*. Modul charging mempunyai dua *output*. *Output* yang pertama adalah masuk ke baterai. *Output* yang kedua adalah masuk ke sistem tegangan. Modul charging pada fungsinya untuk merubah teganngan yang diperlukan ke tegangan baterai. Dari tegangan baterai masuk ke terimal *output* dan masuk ke modul *step up* ke 5V. Tegangan 5V masuk ke tegangan sistem. Untuk hardware bisa dilihat pada gambar 3.9



Gambar 3. 9 Percobaan *power supply*

Setelah percobaan pada tegangan hasil berupa 5V. maka bisa dilakukan pengujian pada hardware dan sistem. Berikut adalah percobaan elektrik pada sistem pada gambar 3.10.



Gambar 3. 10 Rangkaian sensor dan elektrik

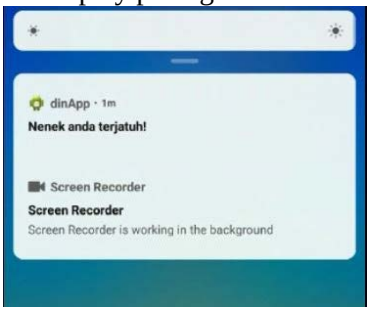
Dari percobaan elektrik tidak ada masalah. Tegangan yang diperlukan bisa di *supply* dengan sempurna. Tegangan yang diperlukan adalah 5 V dan 3,3V. Dalam mengalirkan dan pengoperasian tegangan juga masih stabil karena dalam modul elektrik ada penstabil tegangan.

Gambar 3. 11 *Hardware* yang sudah di kemas

Pada gambar 3.11 adalah sebuah kotak yang didalamnya sudah dimasukkan dari rangkaian sensor, microcontroller dan rangkaian elektrik didalamnya. Untuk pada mic sudah dipasangkan pada tubuh kotak agar suara yang diterima lebih maksimal. Pada saklar sudah dipatenkan dalam kotak bagaian luar agar penggua lebih mudah menjangkau dan mudah mengatur kebutuhan alat.

3.5 Interfacing Android

Pemberitahuan informasi sangat penting mengingat data yang penting harus segera terkirim. Begitu pula hardware dan *software* harus bisa tersambung. Untuk pembuatan *software* android menggunakan intelej IDEA yang hanya bisa digunakan di sistem operasi linux. Tersambungny komunikasi data dan diproses sehingga bisa mucul di display pada gambar 3.12



Gambar 3. 12 Pemberitahuan Terjatuh

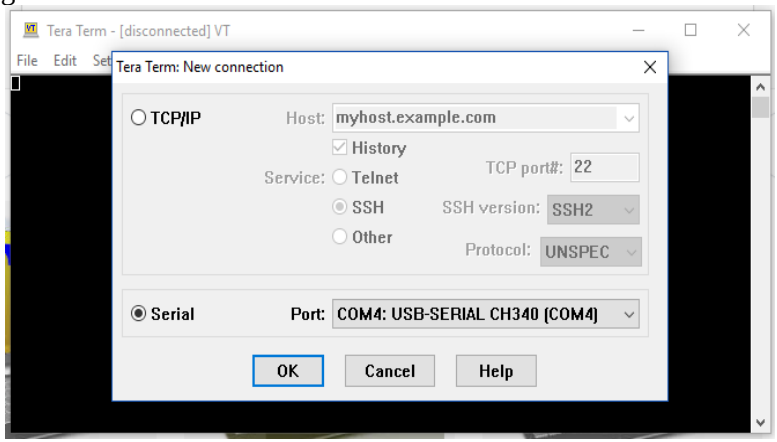
Gambar 3. 13 Interfacing Android

Gambar 3.13 Menampilkan monitoring kondisi sensor mic dan accelerometer. Jika kondisi sudah terjatuh maka alat akan mengirimkan data yang sebelumnya ponsel terlebih dahulu

meminta data. Jika kondisi pada accelerometer dan mic sudah sudah keadaan jatuh setelah proses fuzzy maka data akan dikirimkan ke ponsel melalui koneksi wifi. Setelah terjadi pengiriman data jatuh maka dari android untuk mengunci data dan suara pemberitahuan akan mucul terus sampai *hardware* dan *software* android di reset.

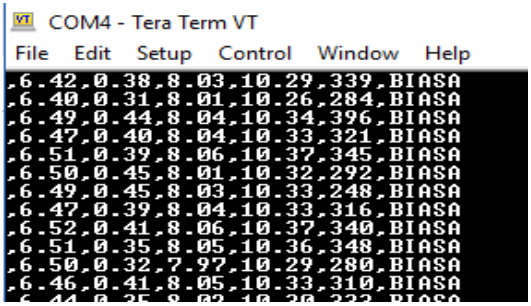
3.6 Pengambilan Data

Tahap ini adalah tahap pengambilan data waktu sistem sedang jalan. pengujian seperti jatuh kedepan, belakang, kiri dan kanan. Hal ini dilakukan agar bisa dilakukan analisa data dan bisa dilakukan pungujian sistem. Pada pengambialan data diperlukan *software* bernama teraterm. *Software* ini berfungsi merekam data serial yang masuk ke PC. Data yang masuk ke serial langsung dikonversikan ke excel. Berikut adalah *software* teraterm pada gambar 3.14



Gambar 3. 14 Pencarian Koneksi Serial

Pada *software* tersebut sangat mudah pemasangannya dan pengaturannya. Cukup menentukan serial yang masuk dan direkam lalu dimana file yang dijadikan hasil perekaman. Waktu perekaman hanya menampilkan angka yang masuk di serial. Berikut adalah tampilan serial yang masuk pada gambar 3.14

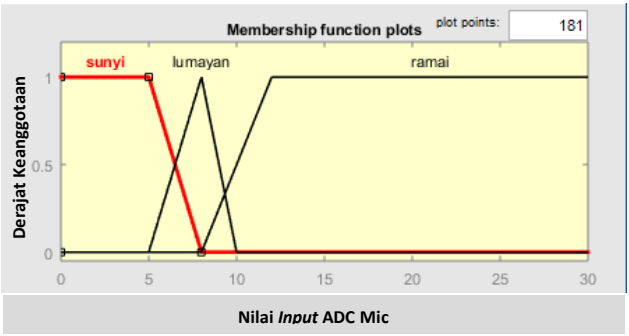


Gambar 3. 15 Hasil Data Masuk

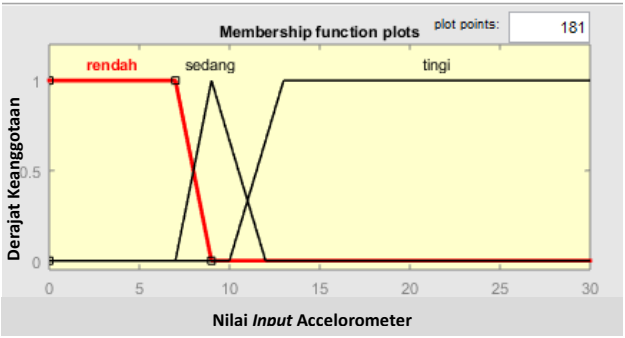
Pada gambar 3.15 adalah data serial yang masuk. Numerik masih mengikuti perhitungan internasional. Sehingga symbol koma jika dikonversikan ke Indonesia adalah titik dan titik adalah koma. Sehingga perbedaan variable dibedakan oleh koma. Jika diurutkan dari paling kiri adalah nilai X, Y, Z, Total XYZ, nilai mic dan yang terakhir hasil fuzzy.

3.7 Perancangan Fuzzy

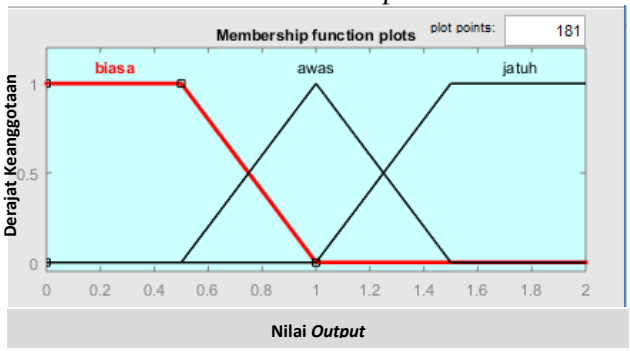
Pada perancangan fuzzy diperlukan data awal. Data awal diperlukan untuk menentukan titik-titik batas jatuh dan tidak. Berikut adalah contoh batas batas sepert gambar 3.16, 3.17 dan 3.18



Gambar 3. 16 Batas batas *input* mic



Gambar 3. 17 Batas batas *input* accelerometer



Gambar 3. 18 Batas batas *output* hasil keputusan

Setelah dibuat batas-batas fuzzy maka dibuatalah rule fuzzy yang diperlukan sebagai pengaturan jika nilai mic dan accelerometer masuk. Berikut adalah tabel maka dan jika pada tabel 3.3

Tabel 3. 3 Rule Base Fuzzy

		Axcellorometer		
		Rendah	Sedang	Tinggi
Mic	Sunyi	Tidak jatuh (1)	Tidak jatuh (4)	Awas (7)
	Lumayan	Tidak jatuh (2)	Awas (5)	Awas (8)
	Ramai	Tidak jatuh (3)	Awas (6)	Jatuh (9)

Pada tabel 3.4 menjelaskan dua variable. Variable pertama adalah accelerometer dan yang kedua adalah mic. Sedangkan ditengah-tengah kolom dan baris adalah hasil keputusan atau keluaran. Peraturan yang dibuat adalah jika pada accelerometer dan mic bernilai ramai maka hasil keputusan jatuh. Jika salah satu nilai sensor dibawah itu maka dipastikan keluaran jatuh tidak dapat muncul.

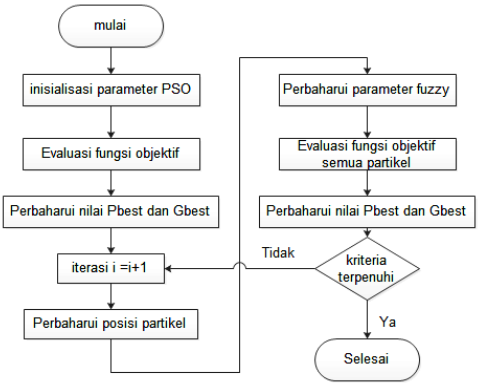
3.8 Perancangan PSO

Perancangan PSO adalah suatu perancangan yang mempergunakan optimasi atau detailnya menggunakan sistem particle swarm optimization (PSO). Algoritma dimulai dengan memberikan parameter awal diantaranya jumlah patikel (n), iterasi maksimal (imax), nilai inersia pemberat awal (ω_{max}), nilai *inersia* pemberat akhir (ω_{min}), faktor *cognitive rate* (C1), faktor *social rate* (C2). Nilai dari masing-masing parameter adala sebagai berikut:

max=10;
n=10;
wmax=0.7;
wmin=0.3;
c1=0.7;
c2=0.3;

Batas-batas yang sudah ditentukan sebelumnya akan digunakan sebagai gbest awal. Nilai a adalah $gbest^1$, b adalah $gbest^2$, dan c adalah $gbest^3$, seperti berikut :
$Gbest^1 = -1$
$Gbest^2 = 0$
$Gbest^3 = 1$

Langkah berikutnya adalah menginisiasi random posisi awal (S^o) untuk tiap dimensi dan juga melakukan inisialisasi nilai ecepatan awal (V^o) untuk tiap partikel pada tiap-tiap dimensi



Gambar 3. 19 Langkah optimasi PSO

Tabel 3. 4 Inisialisasi awal posisi dan kecepatan partikel

S^{01}	S^{02}	S^{03}	V^{01}	V^{02}	V^{03}
-0,070	0,020	0,090	0	0	0
-0,020	-0,010	0,070	0	0	0
-0,100	-0,020	0,089	0	0	0
-0,080	0,040	0,082	0	0	0
-0,095	0,015	0,087	0	0	0
-0,092	-0,012	0,088	0	0	0
-0,076	-0,030	0,090	0	0	0
-0,088	0,030	0,079	0	0	0
-0,086	0,020	0,095	0	0	0
-0,085	0,000	0,085	0	0	0

Untuk iterasi 0, pbest untuk tiap-tiap partikel menggunakan posisi awal (s^0) sebagai perbandingan untuk iteras-iterasi berikutnya. Langkah berikutnya adalah menghitung MSE yang dihasilkan oleh posisi partikel dengan melihat nilai μ yang dihasilkan oleh posisi partikel menggunakan batas-batas yang

udah ditentukan sebelumnya. Sebagai contoh, dihitung menggunakan partikel pertama :

$$\mu^{01} = \frac{x-a}{b-a} = \frac{0{,}07-(-0{,}1)}{0-(-0{,}1)} = 0{,}3$$

$$\mu^{02} = \frac{c-x}{c-b} = \frac{0{,}1-(0{,}02)}{0{,}1-(0)} = 0{,}8$$

$$\mu^{03} = \frac{c-x}{c-b} = \frac{0{,}1-(0{,}09)}{0{,}1-(0)} = 0{,}1$$

Langkah berikutnya menghitung nilai MSE (*mean square error)* yang dicapai :

$$\text{MSE} = \frac{(\mu^{01}-y_1)^2 + (\mu^{02}-y_2)^2 + (\mu^{01}-y_1)^2}{y_1^2+y_2^2+y_3^2}$$

$$= \frac{(0.3-0)^2+(0.8-1)^2+(0.1-0)^2}{0^2+1^2+0^2} = 0.189$$

Proses Perhitungan MSE dilakukan pada setiap partikel, dan didapatkan hasil sebagai berikut:

Tabel 3. 5 Hasil perhitungan MSE tiap partikel

μ^{01}	μ^{02}	μ^{03}	MSE
0,3	0,8	0,1	0,189
0,8	0,9	0,3	0,480
0	0,8	0,11	0,080
0,2	0,6	0,18	0,537
0,05	0,85	0,13	0,056
0,08	0,88	0,12	0,044
0,24	0,7	0,1	0,283
0,12	0,7	0,21	0,271
0,14	0,8	0,05	0,094
0,15	1	0,15	0,043

Dari perhitungan nilai MSE masing-masing partikel didapatkan bahwa nilai MSE terkecil dimiliki oleh partikel 10 dengan nilai MSE = 0,043. Sehingga kondisi terbaik yang pernah

dicapai oleh semua partikel adalah pada posisi (-0,085 ; 0 ; 0,085) dan akan dijadikan Gbest baru di iterasi selanjutnya.Langkah selanjutnya menghitung nilai inersia (ω) untuk iterasi ke-0 menggunakan persamaan 2.8

$$w = (w_{max} - w_{min}) \times \frac{iter_{max} - Iter}{iter_{max}} + w_{min}$$
$$= (0{,}07 - 0{,}03) \times \frac{10-0}{10} + 0.03 = 0.07$$

Langkah berikutnya adalah menghitung kecepatan (v) dan posisi (s) baru partikel untuk iterasi berikutnya menggunakan Persamaan berikut :

$$v^1 = (\omega v^0 + c_1 r_1) \times (Pbest - s^0) + c_2 r_2 \times (Gbest - s^0)$$
$$s^1 = (s^0 + v^1)$$

Sehingga diperoleh hasil seperti tabel 3.7 :

Tabel 3. 6 Hasil *Update* posisi dan kecepatan partikel

S^{11}	S^{12}	S^{13}	V^{11}	V^{12}	V^{13}
-0,092	0,006	0,097	-0,022	-0,014	0,007
-0,078	-0,003	0,092	-0,058	0,007	0,022
-0,100	-0,006	0,097	0,000	0,014	0,008
-0,094	0,011	0,095	-0,014	-0,029	0,013
-0,099	0,004	0,096	-0,004	-0,011	0,009
-0,098	-0,003	0,097	-0,006	0,009	0,009
-0,093	-0,008	0,097	-0,017	0,022	0,007
-0,097	0,008	0,094	-0,009	-0,022	0,015
-0,096	0,006	0,099	-0,010	-0,014	0,004
-0,096	0,000	0,096	-0,011	0,000	0,011

Setelah memperbaharui nilai kecepatan (v) dan nilai posisi (s) maka dilanjutkan dengan iterasi selanjutnya yaitu iterasi ke-1. Cara yang sama dilakukan dengan terlebih dahulu menghitung nilai MSE seperti pada iterasi ke 0 untuk iterasi selanjutnya. Sehingga diperoleh hasil dari optimasi fungsi keanggotaan *Input* eror, deror

dan *output* duty cycle baru dan fungsi keanggotaan ini yang akan di *input*kan ke mikrokontroller *microcontroller*.

3.9 Pengujian Alat

DalamPengambilan data ini sangat penting untuk melakukan ujihardware dan *software* agar kinerjanya sesuai dengan harapan. Pengujiannnya untuk sistem berfungsi dari angka 0 sampai 100% mempunyai 2 kategori. Penilaian pertama yaitu sensitifitas. Sensitifitas adalah seberapa besar untuk mengklasifisasikan pengambilan keputusan tersebut dengan benar. Sensitifitas dapat dihitung dengan rumus dengan persamaan 1.1

$$\text{Sensitifitas} = \frac{\text{Positif benar}}{\text{Positif benar+Negatif Palsu}} \text{X100\%} \quad \text{.....(3.1)}$$

Pada persamaan 1.1 yang dimaksud dengan positif benar adalah keputusan atau notifikasi yang benar akan keadaan yang ada. Contohnya ketika sistem memberi notifikasi terjadinya objek yang jatuh dan dil lapangan terjadi objek yang jatuh maka hal termasuk positif benar. Namun ketika terjadi notifikasi objek yang jatuh tetapi pada dilapangan ternyata objek tidak terjatuh (objek melakukan kegiatan sehari-hari) maka hal ini disebut negative palsu.

Spesitifitas adalah adalah tidak ada notifikasinya jatuh dan memang benar dilapangan objek tersebut memang tidak jatuh. Untuk rumus untuk spesitifitas dirumuskan pada persamaan 3.2

$$\text{Spesitifitas} = \frac{\text{Negatif Benar}}{\text{Positif Palsu+Negatif Benar}} \text{X100\%} \quad \text{.....(3.2)}$$

Pada persamaan 3.2 maksud dari negatif benar adalah kejadian yang tidak jatuh dan benar sistem mendeteksi kejadian tersebut termasuk kegiatan sehari-hari dan bukan kegiatan terjatuh. Untuk positif palsu adalah saat objek benar-benar terjatuh namun dalam sistem tidak ada notifikasi

BAB IV
ANALISA DATA DAN PEMBAHASAN

Pada penelitian ini dilakukan pengambilan data di tiga tempat berbeda yaitu pada posisi 1 (dada), posisi 2 (perut) dan posisi 3 (paha) dengan menggunakan bantuan orang orang dengan kriteria sebagai berikut:

Tabel 4. 1 Kriteria Orang Pengambilan data

No	Nama	Jurusan	Estimasi Tinggi Badan (cm)	Estimasi Berat Badan (Kg)
1	Akhmad Zainudin	S1 Teknik Fisika ITS	175	80
2	Muhammad Musa Bahrowi	Alumni S1 Teknik Fisika ITS	178	50
3	Zanuar suryanullah	S1 Teknik Fisika ITS	170	54

4.1 Hasil Uji Selisih Waktu Sensor Accelorometer dan Mic

Pada percobaan ini digunakan sebagai acuan untuk mengetahui selisih nilai waktu antara 2 sensor yang digunakan. Sensor yang digunakan adalah accelerometer dan mic. Terdepat perubahan waktu karena nilai tersebut muncul karena efek yang berbeda. Nilai accelerometer muncul karena waktu perubahan posisi sedangkan mic muncul karena terdapat suara yang terjatuh. Untuk membandingkan selisih bisa dilihat pada tabel 4.2

Tabel 4. 2 Selish Waktu Mic dan Accelorometer

Jatuh ke	ACC	ADC mic	Selisih (ms)
Depan	58.45	41.62	37
	65.24	40.69	111
	64.1	40.35	188
Belakang	58.86	41.13	27
	65.81	40.54	106
	53.99	40.05	9
Kanan	53.76	42.2	447
	60.33	48.94	30
	60.67	40.47	108
kiri	56.79	41.04	48
	53.82	40.86	86
	53.74	40.27	88
Vertikal	54.45	40.1	31
	59.09	40.39	35
	63	40.43	31

Dari tabel 4.2 data yang ada diambil dari 5 jenis jatuh dan setiap jatuh diulangi sampai 3 kali. Jatuh yang di ambil datanya adalah jatuh kedepan, jatuh kekiri, jatuh kekanan, jatuh kebelakang dan yang terakhir jatuh vertikal. Dari hasil pengambilan data nilai selisih terkecil adalah 9 ms. Untuk yang terbesar adalah 447 ms. Namun pada akhirnya diambil per 100 ms dan diambil rata-rata.

4.2 Pengambilan Data Awal Fuzzy

Pengambilan data ini digunakan sebagai data awal untuk fuzzy. Data awal digunakan digunakan untuk mengetahui sifat atau karakteristik suatu kegiatan tertentu. Karakteristik digunakan untuk mengetahui lebih lanjut suatu hasil produk tertentu. Pengambilan data ini dilakukan dua jenis. Jenis pertama adalah jatuh, jatuh yang digunakan ada lima arah yaitu depan, belakang, kiri, kanan dan vertikal. Untuk hasil data bisa dilihat pada tabel 4.3

Tabel 4. 3 Data Awal Jatuh Untuk Fuzzy

Jatuh ke	ADC mic	Accelorometer
Depan	12.54	21.55
	9.02	10.27
	15.04	21.17
Belakang	10.37	14.62
	10.37	19.54
	11.97	16.91
Kanan	12.86	18.6
	16.53	19.27
	10.61	20.37
Kiri	15.16	20.91
	14.19	16.54
	13.71	16.87
Vertikal	11.35	14.15
	15.42	23.85
	16.07	20.87

Pada tabel 4.3 menghasilkan data dari sensor mic dan accelerometer. Pada nilai tertinggi dari accelerometer menghasilkan nilai 23.85 sedangkan untuk sensor mic manghasilkan nilai 16.53. Untuk terendah nilai dari sensor accelerometer adalah 10,27 sedangkan untuk sensor mic adalah 9,42. Untuk kebutuhan fuzzy agar terjatuh ditetapkan nilai terendah dari mic dan accelerometer sebesar 10,27 dari accelerometer dan untuk mic sebesar 9,42.

Pengambilan data jenis kedua adalah kegiatan Tidak jatuh yang tidak terjatuh. Kegiatan aktifitas ini Tidak jatuh dilakukan sehari-hari. Aktifitas tersebut meliputi berbaring berdiri, berdiri membungkuk, berjalan, berlari kecil, duduk berdiri, naik tangga dan turun tangga. Untuk data yang dihasilkan bisa dilihat pada gambar 4.4

Tabel 4. 4 Data Awal Aktifitas Tidak jatuh Untuk Fuzzy

Aktifitas	ADC mic	Accelorometer
Berbaring berdiri	2.59	0.05
	2.19	6.13
	2.15	4.22

Tabel 4. 4 Lanjutan

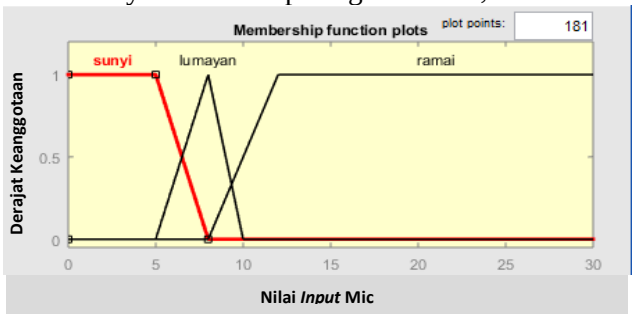
Aktifitas	ADC mic	Accelorometer
Berdiri membungkuk	1	2.33
	0.67	6.26
	0.52	4.28
Bejalan	1.48	0.11
	1.03	1.64
	1.86	0.27
berlari kecil	1.72	2.34
	1.35	2.44
	2.32	6.47
duduk berdiri	1.03	0
	1.26	3.96
	1.69	2.33
naik tangga	3.07	2.63
	2.07	3.86
	2.34	3.13
turun tangga	2.44	1.27
	2.21	0.16
	1.57	1.23

Pada tabel 4.4 Menghasilkan nilai tertinggi dan terendah. Pada nilai terendah untuk nilai mic menghasilkan nilai 0,52 sedangkan untuk accelerometer menunjukkan nilai 0. Untuk nilai tertinggi dihasilkan untuk sensor mic sebesar 2,59 sedangkan untuk nilai accelerometer sebesar yaitu 6,26. Pada sistem fuzzy nilai maksimal batas untuk sistem fuzzy. Jadi kegiatan tersebut hanya memberikan rentang nilai tersebut.

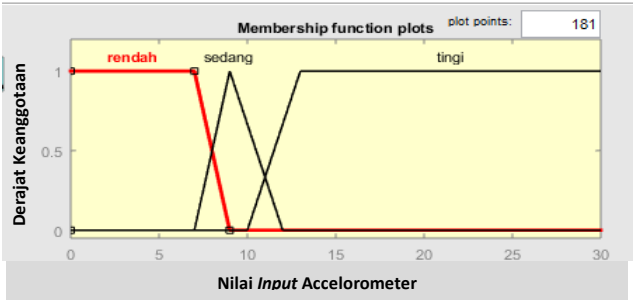
4.3 Pembuatan Batas Fuzzy dan Simulasi

Pada sub bab sebelumnya telah diambil data awal dan menentukan batas atas dan batas bawah. Batas batas tersebut diambil sesuai kebutuhan. Kebutuhan yang dimaksud adalah kebutuhan yang sesuai kepergunaannya. Contohnya adalah untuk menentukan jatuh maka diperlukan batas bawah nilai dari data rekam terjatuh baik untuk sensor accelerometer dan mic. Data jatuh yang diambil adalah jenis jatuh kedepan, belakang, kanan, kiri dan

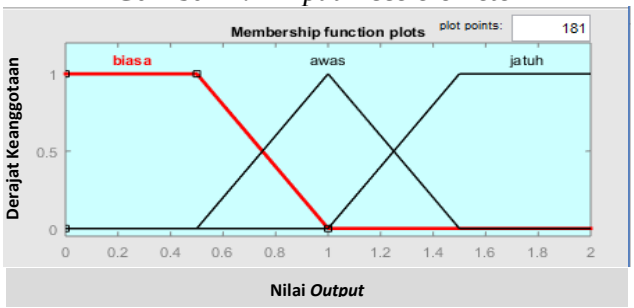
vertikal. Sedangkan untuk menentukan indikator tidak jatuh diperlukan data atas dari data rekam dari aktifitas Tidak jatuh atau kegiatan sehari-hari. Kegiatan yang diambil datanya dari kegiatan sehari-hari adalah seperti berbaring berdiri, berdiri membungkuk, berjalan, berlari kecil, duduk berdiri, naik tangga dan turun tangga. Untuk batas fuzzy bisa dilihat pada gambar 4.1, 4.2 dan 4.3



Gambar 4. 1 *Input* Mic



Gambar 4. 2 *Input* Accelerometer



Gambar 4. 3 Hasil Keputusan

Pada gambar 4.1 adalah *Input* variable untuk mic. Variabel ini bertugas menerima data dan dikategorikan di membership function di sunyi, lumayan atau ramai. Pengkondisian ini tergantung data yang masuk. Untuk pembatasan membership adalah sebagai berikut. Untuk sunyi memmpunyai batasan nilai (0,5,8) sedangkan lumayan mempunyai batasan nilai (5,8,10), Untuk ramai memmpunyai batasan (8,12,30).

Pada Gambar 4.2 tentang *input* accelerometer menjelaskan tentang variable *input* untuk accelerometer mempunyai 3 membership function yaitu rendah, sedang dan tinggi. Batasan nilainya adalah untuk rendah adalah (0,7,9) sedangkan pada sedang mempunyai batasan nilai (7,9,12) dan pada membership function tinggi mempunyai batasan nilai (10,13,30).

Pada gambar 4.3 Untuk hasil keputusan dimasukkan pada jenis *output*. *Output* ini mempunyai 3 membership function yaitu Tidak jatuh, awas dan jatuh. Pada membership function untuk Tidak jatuh mempunyai batas nilai [0,(0,5),1]. Pada membership function awas mempunyai batas nilai [(0,5),1,(1,5)]. Untuk membership function jatuh mempunyai batas [1,(1,5), 2]. Untuk hasil simulasi hasil fuzzy terdapat pada tabel 4.5

Tabel 4. 5 Hasil Simulasi Fuzzy Untuk Data Jatuh

ADC mic	Accelorometer	Nilai	Fuzzy
12.54	21.55	1.62	Jatuh
9.02	10.27	1.28	Awas
15.04	21.17	1.62	jatuh
10.15	14.62	1.58	Jatuh
10.37	19.54	1.57	Jatuh
11.97	16.91	1.62	Jatuh
12.86	18.6	1.62	Jatuh
16.53	19.27	1.62	Jatuh
10.61	20.37	1.59	Jatuh
15.16	20.91	1.62	Jatuh
14.19	16.54	1.62	Jatuh
13.71	16.87	1.62	Jatuh
11.35	14.15	1.6	Jatuh
15.42	23.85	1.62	Jatuh
16.07	20.87	1.62	Jatuh

Pada Tabel 4.5 adalah hasil simulasi dengan data jatuh. Data tersebut sebagai masukkan nilai variable untuk sensor mic dan accelerometer. Dari tabel tersebut menghasilkan nilai yang bervariasi dari nilai yang terendah sampai yang tertinggi. Angka terendah adalah 1,28 yang masuk kategori awas atau belum jatuh sedangkan pada nilai 1,57 atau keatasnya masuk aktegori jatuh. Dengan hasil ini maka terdapat kesalahan pengambilan keputusan sehingga sensitifitasnya hanya 93%. Untuk simulasi dengan aktifitas Tidak jatuh bisa dilihat pada tabel 4.6

Tabel 4. 6 Simulasi Fuzzy Untuk Aktifitas Tidak jatuh

ADC mic	Accelorometer	Nilai	Hasil
2.59	0.05	0.384	Tidak jatuh
2.19	6.13	0.384	Tidak jatuh
2.15	4.22	0.384	Tidak jatuh
1	2.33	0.384	Tidak jatuh
0.67	6.26	0.384	Tidak jatuh
0.52	4.28	0.384	Tidak jatuh
1.48	0.11	0.384	Tidak jatuh
1.03	1.64	0.384	Tidak jatuh
1.86	0.27	0.384	Tidak jatuh
1.72	2.34	0.384	Tidak jatuh
1.35	2.44	0.384	Tidak jatuh
2.32	6.47	0.384	Tidak jatuh
1.03	0	0.384	Tidak jatuh
1.26	3.96	0.384	Tidak jatuh
1.69	2.33	0.384	Tidak jatuh
3.07	2.63	0.384	Tidak jatuh
2.07	3.86	0.384	Tidak jatuh
2.34	3.13	0.384	Tidak jatuh
2.44	1.27	0.384	Tidak jatuh
2.21	0.16	0.384	Tidak jatuh
1.57	1.23	0.384	Tidak jatuh

Tabel 4.6 Menjelaskan tentang hasil simulasi fuzzy. Hasil menunjukkan semua nilai yang sama yaitu 0,384. Nilai 0,384 artinya hasilnya Tidak jatuh atau tidak jatuh. Nilai yang

dimasukkan pada tabel tersebut adalah data awal dengan mengambil kegiatan sehari-hari. Kegiatan sehari-hari berarti artinya tidak jatuh dan pada simulasi ini mempunyai spesitifitas sebesar 100% dikarenakan nilai kebenarannya 100%

4.4 Menentukan Batas Baru Dari PSO

Pada hasil simulasi fuzzy menunjukkan hasil yang kurang maksimal. Sensitifitas menghasilkan nilai 93% sedangkan untuk sensitifitasnya adalah 100%. Hal ini mengakibatkan sistem fuzzy membutuhkan optimisasi untuk peningkatan kinerja. Optimisasi PSO yang digunakan mengatur ulang batas-batas fuzzy yang baru sehingga menghasilkan nilai sensitifitas naik. Untuk perubahan batas-batas baru bisa dilihat pada tabel 4.7 dan 4.8

Tabel 4. 7 Hasil Fuzzy PSO Variabel Mic

	Trapesium			Segitiga			Trapesium		
	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
	0	5	8	5	8	10	8	12	30
PSO	0	5	7.57378	5.54647	8	9.31291	7.33903	12	30

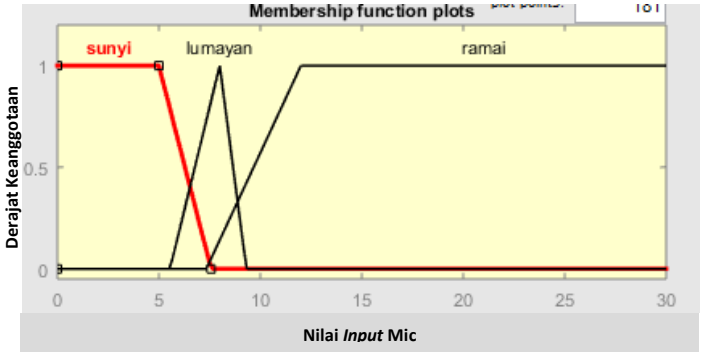
Tabel 4. 8 Hasil Fuzzy PSO Variabel Accelorometer

	Trapesium			Segitiga			Trapesium		
	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
	0	7	9	7	9	12	10	13	30
PSO	0	7	7.78493	7.6109	9	10.3521	9.0384	13	30

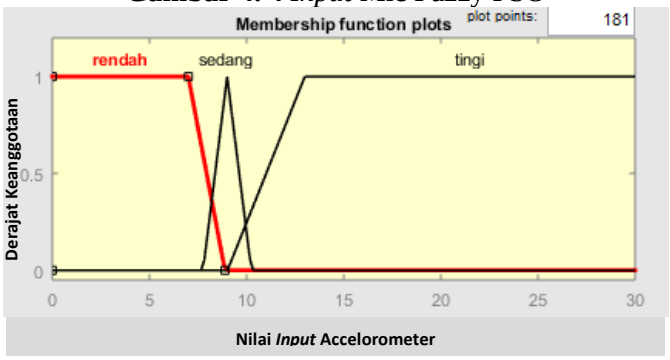
Pada Tabel 4.7 dan 4.8 meruapakan batas-batas baru dari hasil optimisasi PSO. Hasil batas-batas baru dari variable mic. Dalam batas-batas terebut dibagi menjadi 3 membership function yaitu 2 trapesium dan 1 segitiga. Teknisnya segitiga diapit oleh 2 trapesium. Apabila di gambarkan maka pada trapesium kiri merupakan huruf {A,B,C} lalu pada segitiga tengah dilambangkan huruf {D,E,F} pada trapesium kanan.

Optimisasi dilakukan 3x agar bisa dilakukan perbandingan optimisasi yang terbaik. Cara mencari optimisasi yang terbaik adalah dengan menentukan batas-batas baru dari hasil optimisasi. Hasil yang terbaik terbaik maka batas fuzzy tersebut ditanam pada *microcontroller* dan fungsikan lalu diuji. Simulasi Fuzzy PSO

Simulasi yang kedua ini merupakan hasil pengembangan batas baru yang berbeda dengan batas-batas dari PSO. Batas baru ini adalah perubahan hasil PSO (Particle Swarm Optimization). Penggunaan PSO adalah merubah-rubah batas fuzzy dalam membership functionnya. Merubah-rubah batas fuzzy sehingga nilai akhirnya bisa berubah sesuai target. Berikut adalah hasil perubahan batas fuzzy pada gambar 4.6 dan 4.7



Gambar 4. 4 *Input* Mic Fuzzy PSO



Gambar 4. 5 *Input* Accel Fuzzy PSO

Pada gambar 4.6 merupakan perubahan batas fuzzy untuk *input* mic. Pada perubahan batas di membership function sunyi yaitu { 0,5,(7.5737)} lumayan {(5.5464),8,(9.3129)} ramai { (7.3390),12,30}. Pada gambar 4.7 pada *input* accelerometer merupakan perubahan batas fuzzy pada inpu t accelerometer adalah

rendah {0,7,(7.7849)} sedang { (7.6108),9,(10.3520)} tinggi { (9.0384),13,30}. Untuk simulasi berada pada tabel 4.11

Tabel 4. 9 Simulasi Fuzzy PSO Untuk Jatuh

Jatuh ke	ADC mic	Accelorometer	Nilai	Fuzzy
Depan	12.54	21.55	1.62	Jatuh
	9.02	10.27	1.34	Jatuh
	15.04	21.17	1.62	jatuh
Belakang	10.15	14.62	1.58	Jatuh
	10.37	19.54	1.58	Jatuh
	11.97	16.91	1.62	Jatuh
Kanan	12.86	18.6	1.62	Jatuh
	16.53	19.27	1.62	Jatuh
	10.61	20.37	1.59	Jatuh
Kiri	15.16	20.91	1.62	Jatuh
	14.19	16.54	1.62	Jatuh
	13.71	16.87	1.62	Jatuh
Vertikal	11.35	14.15	1.6	Jatuh
	15.42	23.85	1.62	Jatuh
	16.07	20.87	1.62	Jatuh

Tabel 4. 10 Simulasi Fuzzy PSO Untuk Kegiatan Tidak jatuh

Aktifitas	MIC	Acceloro meter	Nilai	Fuzzy
Berbaring berdiri	2.59	0.05	0.384	Tidak jatuh
	2.19	6.13	0.384	Tidak jatuh
	2.15	4.22	0.384	Tidak jatuh
Berdiri membungkuk	1	2.33	0.384	Tidak jatuh
	0.67	6.26	0.384	Tidak jatuh
	0.52	4.28	0.384	Tidak jatuh
Berjalan	1.48	0.11	0.384	Tidak jatuh
	1.03	1.64	0.384	Tidak jatuh
	1.86	0.27	0.384	Tidak jatuh

Tabel 4. 10 Lanjutan

Aktifitas	MIC	Acceloro meter	Nilai	Fuzzy
berlari kecil	1.72	2.34	0.384	Tidak jatuh
	1.35	2.44	0.384	Tidak jatuh
	2.32	6.47	0.384	Tidak jatuh
duduk berdiri	1.03	0	0.384	Tidak jatuh
	1.26	3.96	0.384	Tidak jatuh
	1.69	2.33	0.384	Tidak jatuh
naik tangga	3.07	2.63	0.384	Tidak jatuh
	2.07	3.86	0.384	Tidak jatuh
	2.34	3.13	0.384	Tidak jatuh
turun tangga	2.44	1.27	0.384	Tidak jatuh
	2.21	0.16	0.384	Tidak jatuh
	1.57	1.23	0.384	Tidak jatuh

Tabel 4.11 menjelaskan tentang hasil simulasi dari PSO2. Hasilnya terdapat 1 nilai salah sehingga menyatakan sensitifitasnya 93%. Dalam hasil PSO2 nilai salah tersebut hilang dan dalam hasil yang diperlihatkan pada tabel 4.11 nilainya sudah benar. Perubahan dalam PSO2 ini nilai spesitifitasnya berubah yang sebelumnya sebelum PSO dan PSO1 nilainya masih 93% namun hasil ini menyatakan 100%. Untuk aktifitas Tidak jatuh hasil simulasi dinilai sudah maksimal yaitu spesitifitas 100%.

4.5 Percobaan Alat di Dada

Pada percobaan ini dilakukan percobaan dengan hasil simulasi terbaik. Simulasi terbaik dari fuzzy yaitu pada fuzzy yang diPSO. Setela sensor diletakkan pada sensor lalu dilakukan pengujian sensitifitas dan spesitifitas. Hasil pengujian diambil melalui interface serial ke computer. Hasil diambil dan disimpan melalui format .csv. Format CSV adalah salah satu format excel yang hanya berbasis data. Data yang disimpan secara otomatis berubah kolom jika terdapat simbol titik. Pada pengujian alat tersebut diletakkan pada dada. Untuk hasil bisa dilihat pada gambar 4.15

Tabel 4. 11 Percobaan Jatuh Hardware di Dada

Jatuh ke	ADC MIC	Accelorometer				Status
		X	Y	Z	total	
Depan	11.37	3.16	6.4	5.03	11.69	Jatuh
	9.71	4.61	8.56	8.75	17.96	Jatuh
	10.16	12.22	5.61	4.07	18.21	Jatuh
Belakang	11.31	4.82	5.84	6.53	12.69	Jatuh
	10.78	7.36	8.6	5.26	16.92	Jatuh
	17.06	13.53	8.05	7.3	22.81	Jatuh
Kanan	11.91	8.64	5.13	7.63	15.62	Jatuh
	10.81	8.17	4.84	4.13	12.95	Jatuh
	14.27	6.46	3.98	8.02	14.34	Jatuh
kiri	11.9	5.23	4.98	9.68	15.94	Jatuh
	9.91	6.82	4.69	7.94	14.17	Jatuh
	12.65	4.94	3	9.22	13.87	Jatuh
Vertikal	10.38	9.02	8.85	8.72	19.93	Jatuh
	13.93	5.34	6.78	13.46	20.17	Jatuh
	8.97	3.72	4.42	5.61	11	Jatuh

Tabel 4. 12 Percobaan Aktifitas Tidak jatuh Hardware di Dada

Aktifitas	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
Berbaring berdiri	4.09	0	2.33	0	2.33	Tidak jatuh
	1.05	0.88	0.02	2.8	3.67	Tidak jatuh
	1.2	0.01	4.63	0.02	4.67	Tidak jatuh
Berdiri membungkuk	2.75	0.99	0.21	0.86	1.86	Tidak jatuh
	2.07	0	1.95	7.28	9.22	Tidak jatuh
	2.4	0	1.95	3.5	5.46	Tidak jatuh
Bejalan	2.12	0	0	3.5	3.5	Tidak jatuh
	2.96	0	0	1.53	1.53	Tidak jatuh
	2.35	0	0	1.14	1.15	Tidak jatuh
berlari kecil	1.4	0	2.27	3.91	6.19	Tidak jatuh
	2.83	0.07	1.15	0.37	1.6	Tidak jatuh
	2.44	0	5.46	2.33	6.88	Tidak jatuh

Tabel 4. 12 Lanjutan

Aktifitas	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
duduk berdiri	1.92	0.07	7.21	0.23	7.38	Tidak jatuh
	2.93	0	3.11	1.16	4.28	Tidak jatuh
	1.84	0.55	0.8	6.19	6.88	Tidak jatuh
naik tangga	1.35	0	10.14	0	10.14	Awas
	0.91	0	5.46	0	5.46	Tidak jatuh
	0.25	0	4.28	0	4.28	Tidak jatuh
turun tangga	2.25	0.55	0.15	4.36	4.57	Tidak jatuh
	0.33	0	2.33	0	2.33	Tidak jatuh
	6.65	5.15	4.07	10.6	14.98	Awas

Pada tabel 4.15 dan 4.16 hasil yang didapat cukup untuk dibilang bagus. Seluruh aktifitas jatuh yang dilakukan dengan hasil dari pengukuran alat bisa sesauai. Percobaan yang dilakukan dari jatuh depan, belakang, kanan, kiri dan vertical menunjukkan hasil jatuh. Sehingga nilai sensitifitasnya nilainya adalah 100%. Untuk hasil spesitifitas pada hasil pengujian adalah 100%. Nilai tersebut karena pada saat saat pengujian nilai yang tertera adalah Tidak jatuh dan awas. Untuk nilai jatuh tidak ada sehingga untuk nilai spesitifitasnya adalah 100%

4.6 Percobaan Alat di Perut

Pada percobaan kedua alat diletakkan pada perut. Hal ini dilakukan perubahan untuk melihat hasil pengujian pada beda possisi. Fuzzy yang ditanam adalah fuzzy PSO 2. Hasil pengujian diambil melalui interface serial ke computer. Hasil diambil dan disimpan melalui format .csv. Untuk alat tersebut diletakkan pada dada. Untuk hasil bisa dilihat pada gambar 4.17 dan tabel 4.18

Tabel 4. 13 Percobaan Jatuh Hardware di Perut

Posisi	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
Depan	10.35	4.55	4.59	7.53	12.57	Jatuh
	11.2	10.6	5.09	6.38	16.37	Jatuh
	10.51	5.61	3.98	3.02	10.21	Jatuh

Tabel 4. 13 Lanjutan

Posisi	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
Belakang	10.35	4.55	4.59	7.53	12.57	Jatuh
	13.26	3	4.9	8	12.65	Jatuh
	14.14	6.94	7.84	8.89	18.23	Jatuh
Kanan	13.15	5.92	2.73	6.36	11.84	Jatuh
	10.74	9.42	2.48	11.01	18.16	Jatuh
	12.66	10.72	4.36	8.27	17.96	Jatuh
kiri	9.87	8.64	7.03	13.8	21.51	Jatuh
	9.27	9.47	6.63	10.56	20.93	Jatuh
	9.54	6.53	3.21	7.9	13.87	Jatuh
Vertikal	9.39	8.42	3.33	4.42	13.72	Jatuh
	12.62	10.01	8.93	9.35	21.56	Jatuh
	10.91	6.36	2.25	11.07	15.41	Jatuh

Tabel 4. 14 Percobaan Aktifitas Tidak jatuh Hardware di Perut

Aktifitas	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
Berbaring berdiri	5.03	0	10.92	0	10.92	Awas
	4.62	0.76	4.59	4.44	8.95	Tidak jatuh
	2.16	0	3.09	0	3.09	Tidak jatuh
Berdiri membungkuk	2.5	0.51	2.3	9.73	11.67	Awas
	0.62	0.05	0.8	4.46	5.02	Tidak jatuh
	1.7	3.46	0	0	3.46	Tidak jatuh
Bejalan	8.64	0.28	5.4	3.32	8.56	Awas
	3.56	0	0	5.84	5.84	Tidak jatuh
	4.28	0	3.11	0.37	3.5	Tidak jatuh
berlari kecil	2.43	0.68	1.54	2.15	3.3	Tidak jatuh
	2.41	0.47	1.12	5.09	6.09	Tidak jatuh
	1.17	0.1	2.72	1.14	3.86	Tidak jatuh
duduk berdiri	1.45	0	0	2.33	2.33	Tidak jatuh
	2.18	0	0	7.78	7.78	Tidak jatuh
	1.84	1.33	0.77	4.98	6.09	Tidak jatuh

Tabel 4. 14 Lanjutan

Aktifitas	ADC MIC	Accelorometer			TOTAL	STATUS
		X	Y	Z		
naik tangga	3.53	0	0.38	4.67	5.07	Tidak jatuh
	5.57	0.03	1.16	0.88	2.04	Tidak jatuh
	1.68	0	3.9	1.15	5.05	Tidak jatuh
turun tangga	4.19	0	2.73	0	2.73	Tidak jatuh
	2.45	0.37	3.78	1.7	5.14	Tidak jatuh
	0.81	0	1.95	1.57	3.51	Tidak jatuh

Pada tabel 4.15 dan 4.16 hasil yang didapat cukup untuk dibilang bagus. Seluruh aktifitas jatuh yang dilakukan dengan hasil dari pengukuran alat bisa sesauai. Percobaan yang dilakukan dari jatuh depan, belakang, kanan, kiri dan vertical menunjukkan hasil jatuh. Sehingga nilai sensitifitasnya nilainya adalah 100%. Untuk hasil spesitifitas pada hasil pengujian adalah 100%. Nilai tersebut karena pada saat saat pengujian nilai yang tertera adalah Tidak jatuh dan awas. Untuk nilai jatuh tidak ada sehingga untuk nilai spesitifitasnya adalah 100%

4.7 Percobaan Alat di Paha

Pada percobaan ketiga, alat diletakkan pada paha dan lebih tepatnya pada paha kanan.. Hal ini dilakukan perubahan untuk melihat hasil pengujian pada beda possisi. Fuzzy yang ditanam adalah fuzzy PSO 2. Hasil pengujian diambil melalui interface serial ke computer. Hasil diambil dan disimpan melalui format .csv. Untuk alat tersebut diletakkan pada dada. Untuk hasil bisa dilihat pada gambar 4.15 dan tabel 4.16

Tabel 4. 15 Percobaan Jatuh Hardware di Paha

Aktifitas	ADC MIC	accelorometer			TOTAL	STATUS
		X	Y	Z		
Depan	6.22	9.96	4.76	10.88	20.07	Awas
	5.72	11.46	6.63	4.61	17.52	Awas
	6.91	5.94	2.85	12.86	17.47	Awas
Belakang	4.27	1.27	0.25	15.56	15.93	Awas
	9.17	7.09	1.83	5.88	11.53	Jatuh
	9.69	7.4	4.26	10.38	15.77	Jatuh

Tabel 4. 15 Lanjutan

Aktifitas	ADC MIC	accelorometer			TOTAL	STATUS
		X	Y	Z		
Kanan	11.32	6.69	2.51	9.81	15.18	Jatuh
	9.75	8.94	5.61	11.55	18.94	Jatuh
	9.57	6.32	4.15	6.28	12.85	Jatuh
kiri	12.41	7.3	3.96	6.28	13.82	Jatuh
	9.33	5.76	1.41	15.27	18.58	Jatuh
	7.99	11.64	0.93	8.93	16.57	Awas
Vertikal	10.01	7.86	6.92	5.65	15.89	Jatuh
	12.53	11.28	7.11	13.13	23.1	Jatuh
	12.73	10.81	6.42	14.19	22.44	Jatuh

Tabel 4. 16 Percobaan Aktifitas Tidak jatuh Hardware di Paha

Aktifitas	ADC MIC	accelorometer			TOTAL	STATUS
		X	Y	Z		
Berbaring berdiri	0.74	0.05	0.77	4.5	5.28	Tidak jatuh
	4.04	9.03	0.93	2.95	10.36	Awas
	5.21	6.96	5.09	6.46	14.63	Awas
Berdiri membungkuk	3.55	0.11	0.75	0.81	1.37	Tidak jatuh
	1.63	0	0.76	0.01	0.79	Tidak jatuh
	1.86	0	1.89	0.07	1.94	Tidak jatuh
Bejalan	2.97	0.86	1.36	4.23	5.69	Tidak jatuh
	2.18	0.09	0.82	4.94	5.78	Tidak jatuh
	4.19	3.26	2.39	7.38	10.83	Awas
berlari kecil	6.29	5.46	2.44	6.59	11.32	Awas
	2.29	0.79	0.51	4.82	5.19	Tidak jatuh
	2.97	2.39	1.5	8.72	10.81	Awas
duduk berdiri	2.33	0.07	0.8	0.07	0.85	Tidak jatuh
	3.3	6.19	0	0.05	6.24	Tidak jatuh
	3.17	1.91	0	0.01	1.94	Tidak jatuh
naik tangga	3.23	2.52	2.15	2.36	5.69	Tidak jatuh
	2.07	1.02	1.21	1.2	3.05	Tidak jatuh
	3.98	4.51	1.51	5.96	9.88	Tidak jatuh
turun tangga	1.87	0.49	1.99	2.27	4.32	Tidak jatuh
	3.44	1.57	2.07	3.77	6.25	Tidak jatuh
	0.77	0	13.26	0	13.26	Awas

Pada tabel 4.15 dan 4.16 hasil yang didapat cukup berbeda dari percobaan . Seluruh aktifitas jatuh yang dilakukan dengan hasil dari pengukuran tidak bisa sesuai. Percobaan yang dilakukan dari jatuh depan, belakang, kanan, kiri dan vertical menunjukkan

beberapa hasil jatuh namun pula terkadang menunjukkan awas sehingga nilai sensitifitasnya nilainya adalah 72%. Untuk hasil spesitifitas pada hasil pengujian adalah 100%. Nilai tersebut karena pada saat saat pengujian nilai yang tertera adalah Tidak jatuh dan awas. Untuk nilai jatuh tidak ada sehingga untuk nilai spesitifitasnya adalah 100%

4.8 Perbandingan Simulasi dan Hasil Uji Alat Posisi Dada

Pada perbandingan untuk ini adalah membandingakan hasil dari uji alat dengan simulasi. Hasil uji alat adalah hasil penunjukkan apakah kegiatan atau aktifitas yang dilakukan menunjukkan hasil tidak jatuh, awas atau terjatuh. Serangkaian kegiatan terjatuh dibagi menjadi terjatuh kedepan, belakang, kiri, kanan dan vertikal. Untuk kegiatan tidak terjatuh dispesifikan menjadi berbaring berdiri, berdiri membungkuk, berjalan, berlari kecil, duduk berdiri, naik tangga dan turun tangga. Untuk simulasi dari hasil fuzzy dilakukan dengan *input* data ADC mic dan accelerometer dari hasil uji alat dan dimasukkan dalam simulasi fuzzy matlab. Untuk perbandingannya adalah pada tabel 4.17 dan tabel 4.18.

Tabel 4. 17 Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi dada

Jatuh ke	ADC MIC	ACC	Hasil Alat	Simulasi Fuzzy	
				Nilai	Hasil
Depan	11.37	11.69	Jatuh	1.58	jatuh
	9.71	17.96	Jatuh	1.57	jatuh
	10.16	18.21	Jatuh	1.58	jatuh
Belakang	11.31	12.69	Jatuh	1.6	jatuh
	10.78	16.92	Jatuh	1.59	jatuh
	17.06	22.81	Jatuh	1.62	jatuh
Kanan	11.91	15.62	Jatuh	1.62	jatuh
	10.81	12.95	Jatuh	1.59	jatuh
	14.27	14.34	Jatuh	1.62	jatuh

Tabel 4. 17 Lanjutan

Jatuh ke	ADC MIC	ACC	Hasil Alat	Simulasi Fuzzy	
				Nilai	Hasil
kiri	11.9	15.94	Jatuh	1.61	jatuh
	9.91	14.17	Jatuh	1.57	jatuh
	12.65	13.87	Jatuh	1.62	jatuh
Vertikal	10.38	19.93	Jatuh	1.58	jatuh
	13.93	20.17	Jatuh	1.62	jatuh
	8.97	11	Jatuh	1.34	jatuh

Tabel 4. 18 Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi dada

Aktifitas	ADC MIC	ACC	HASIL ALAT	Simulasi Fuzzy	
				Nilai	Hasil
Berbaring berdiri	4.09	2.33	Tidak jatuh	0.384	Tidak jatuh
	1.05	3.67	Tidak jatuh	0.384	Tidak jatuh
	1.2	4.67	Tidak jatuh	0.384	Tidak jatuh
Berdiri membungkuk	2.75	1.86	Tidak jatuh	0.401	Tidak jatuh
	2.07	9.22	Tidak jatuh	0.401	Tidak jatuh
	2.4	5.46	Tidak jatuh	0.384	Tidak jatuh
Berjalan	2.12	3.5	Tidak jatuh	0.384	Tidak jatuh
	2.96	1.53	Tidak jatuh	0.384	Tidak jatuh
	2.35	1.15	Tidak jatuh	0.384	Tidak jatuh
berlari kecil	1.4	6.19	Tidak jatuh	0.384	Tidak jatuh
	2.83	1.6	Tidak jatuh	0.384	Tidak jatuh
	2.44	6.88	Tidak jatuh	0.384	Tidak jatuh
duduk berdiri	1.92	7.38	Tidak jatuh	0.403	Tidak jatuh
	2.93	4.28	Tidak jatuh	0.384	Tidak jatuh
	1.84	6.88	Tidak jatuh	0.384	Tidak jatuh
naik tangga	1.35	10.14	Awas	0.459	Tidak jatuh
	0.91	5.46	Tidak jatuh	0.384	Tidak jatuh
	0.25	4.28	Tidak jatuh	0.384	Tidak jatuh
turun tangga	2.25	4.57	Tidak jatuh	0.384	Tidak jatuh
	0.33	2.33	Tidak jatuh	0.384	Tidak jatuh
	6.65	14.98	Awas	0.73	Awas

Dari tabel 4.17 menunjukkan bahwasnya dengan posisi jatuh kemapun alat menunjukkan status terjatuh. Begitu juga untuk simulasi fuzzy menunjukkan nilai minimum 1,34 sampai dengan

1,62 dengan skala 0 sampai dengan 2. Dengan menunjukkan nilai tersebut maka hasil keluaran dari simulasi fuzzy adalah jatuh. Sehingga sensitifitas dari uji alat adalah 100%. Pada sensitifitas dari simulasi fuzzy juga menunjukka 100% karena menunjukkan hasil jatuh pada segala aktifitas jatuh. Pada tabel 4.18 tentang perbandingan simulasi aktifitas biasa juga mempunyai performa yang sangat bagus dengan menunjukka spesitifitas dari uji alat 100% dan spesitifitas dari simulasi 100%.

4.9 Perbandingan Simulasi dan Hasil Uji Alat Posisi Perut

Pada perbandingan untuk ini adalah membandingakan hasil dari uji alat dengan simulasi. Hasil uji alat adalah hasil penunjukkan apakah kegiatan atau aktifitas yang dilakukan menunjukkan hasil tidak jatuh, awas atau terjatuh. Serangkaian kegiatan terjatuh dibagi menjadi terjatuh kedepan, belakang, kiri, kanan dan vertikal. Untuk kegiatan tidak terjatuh dispesifikan menjadi berbaring berdiri, berdiri membungkuk, berjalan, berlari kecil, duduk berdiri, naik tangga dan turun tangga. Untuk simulasi dari hasil fuzzy dilakukan dengan *input* data ADC mic dan accelerometer dari hasil uji alat dan dimasukkan dalam simulasi fuzzy matlab. Untuk perbandingannya adalah pada tabel 4.19 dan tabel 4.20.

Tabel 4. 19 Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi perut

JATUH KE	ADC MIC	ACC	HASIL ALAT	SIMULASI FUZZY	
				NILAI	HASIL
Depan	10.35	12.57	Jatuh	1.58	Jatuh
	11.2	16.37	Jatuh	1.6	Jatuh
	10.51	10.21	Jatuh	1.5	Jatuh
Belakang	10.35	12.57	Jatuh	1.58	Jatuh
	13.26	12.65	Jatuh	1.61	Jatuh
	14.14	18.23	Jatuh	1.62	Jatuh
Kanan	13.15	11.84	Jatuh	1.59	Jatuh
	10.74	18.16	Jatuh	1.59	Jatuh
	12.66	17.96	Jatuh	1.62	Jatuh

Tabel 4. 19 Lanjutan

JATUH KE	ADC MIC	ACC	HASIL ALAT	SIMULASI FUZZY	
				NILAI	HASIL
kiri	9.87	21.51	Jatuh	1.57	Jatuh
	9.27	20.93	Jatuh	1.52	Jatuh
	9.54	13.87	Jatuh	1.56	Jatuh
Vertikal	9.39	13.72	Jatuh	1.52	Jatuh
	12.62	21.56	Jatuh	1.62	Jatuh
	10.91	15.41	Jatuh	1.59	Jatuh

Tabel 4. 20 Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi perut

Aktifitas	ADC MIC	ACC	HASIL UJI ALAT	SIMULASI FUZZY	
				NIlai	Hasil
Berbaring berdiri	5.03	10.92	Awas	0.438	Tidak jatuh
	4.62	8.95	Tidak jatuh	0.387	Tidak jatuh
	2.16	3.09	Tidak jatuh	0.384	Tidak jatuh
Berdiri membungkuk	2.5	11.67	Awas	0.417	Tidak jatuh
	0.62	5.02	Tidak jatuh	0.384	Tidak jatuh
	1.7	3.46	Tidak jatuh	0.384	Tidak jatuh
Bejalan	8.64	8.56	Awas	0.61	Tidak jatuh
	3.56	5.84	Tidak jatuh	0.384	Tidak jatuh
	4.28	3.5	Tidak jatuh	0.384	Tidak jatuh
berlari kecil	2.43	3.3	Tidak jatuh	0.384	Tidak jatuh
	2.41	6.09	Tidak jatuh	0.384	Tidak jatuh
	1.17	3.86	Tidak jatuh	0.384	Tidak jatuh
duduk berdiri	1.45	2.33	Tidak jatuh	0.384	Tidak jatuh
	2.18	7.78	Tidak jatuh	0.426	Tidak jatuh
	1.84	6.09	Tidak jatuh	0.384	Tidak jatuh
naik tangga	3.53	5.07	Tidak jatuh	0.384	Tidak jatuh
	5.57	2.04	Tidak jatuh	0.405	Tidak jatuh
	1.68	5.05	Tidak jatuh	0.384	Tidak jatuh
turun tangga	4.19	2.73	Tidak jatuh	0.384	Tidak jatuh
	2.45	5.14	Tidak jatuh	0.384	Tidak jatuh
	0.81	3.51	Tidak jatuh	0.384	Tidak jatuh

Dari tabel 4.19 menunjukkan bahwasnya dengan posisi jatuh kemapun alat menunjukkan status terjatuh. Begitu juga untuk

simulasi fuzzy menunjukkan nilai minimum 1,5 sampai dengan 1,62 dengan skala 0 sampai dengan 2. Dengan menunjukkan nilai tersebut maka hasil keluaran dari simulasi fuzzy adalah jatuh. Sehingga sensitifitas dari uji alat adalah 100%. Pada sensitifitas dari simulasi fuzzy juga menunjukka 100% karena menunjukkan hasil jatuh pada segala aktifitas jatuh. Pada tabel 4.18 tentang perbandingan simulasi aktifitas biasa juga mempunyai performa yang sangat bagus dengan menunjukka spesitifitas dari uji alat 100% dan spesitifitas dari simulasi 100%.

4.10 Perbandingan Simulasi dan Hasil Uji Alat Posisi Paha

Pada perbandingan untuk ini adalah membandingakan hasil dari uji alat dengan simulasi. Hasil uji alat adalah hasil penunjukkan apakah kegiatan atau aktifitas yang dilakukan menunjukkan hasil tidak jatuh, awas atau terjatuh. Serangkaian kegiatan terjatuh dibagi menjadi terjatuh kedepan, belakang, kiri, kanan dan vertikal. Untuk kegiatan tidak terjatuh dispesifikan menjadi berbaring berdiri, berdiri membungkuk, berjalan, berlari kecil, duduk berdiri, naik tangga dan turun tangga. Untuk simulasi dari hasil fuzzy dilakukan dengan *input* data ADC mic dan accelerometer dari hasil uji alat dan dimasukkan dalam simulasi fuzzy matlab. Untuk perbandingannya adalah pada tabel 4.17 dan tabel 4.18.

Tabel 4. 21 Perbandingan simulasi fuzzy kondisi terjatuh pada sensor posisi paha

Aktifitas	ADC MIC	ACC	HASIL UJI ALAT	SIMULASI FUZZY	
				NILAI	HASIL
Depan	6.22	20.07	Awas	0.604	Tidak jatuh
	5.72	17.52	Awas	0.457	Tidak jatuh
	6.91	17.47	Awas	0.805	Awas
Belakang	4.27	15.93	Awas	0.384	Tidak jatuh
	9.17	11.53	Jatuh	1.45	Jatuh
	9.69	15.77	Jatuh	15.77	Jatuh
Kanan	11.32	15.18	Jatuh	1.6	Jatuh
	9.75	18.94	Jatuh	1.57	Jatuh
	9.57	12.85	Jatuh	1.56	Jatuh
kiri	12.41	13.82	Jatuh	1.62	Jatuh
	9.33	18.58	Jatuh	1.56	Jatuh
	7.99	16.57	Awas	1.1	Awas

Tabel 4. 21 Lanjutan

Aktifitas	ADC MIC	ACC	HASIL UJI ALAT	SIMULASI FUZZY	
				NILAI	HASIL
Vertikal	10.01	15.89	Jatuh	1.57	Jatuh
	12.53	23.1	Jatuh	1.62	Jatuh
	12.73	22.44	Jatuh	1.62	Jatuh

Tabel 4. 22 Perbandingan simulasi fuzzy kondisi aktifitas biasa pada sensor posisi pada

Aktifitas	ADC MIC	ACC	STATUS	SIMULASI FUZZY	
				NILAI	HASIL
Berbaring berdiri	0.74	5.28	Tidak jatuh	0.384	Tidak jatuh
	4.04	10.36	Awas	0.454	Tidak jatuh
	5.21	14.63	Awas	0.391	Tidak jatuh
Berdiri membungkuk	3.55	1.37	Tidak jatuh	0.384	Tidak jatuh
	1.63	0.79	Tidak jatuh	0.384	Tidak jatuh
	1.86	1.94	Tidak jatuh	0.384	Tidak jatuh
Bejalan	2.97	5.69	Tidak jatuh	0.384	Tidak jatuh
	2.18	5.78	Tidak jatuh	0.384	Tidak jatuh
	4.19	10.83	Awas	0.44	Tidak jatuh
berlari kecil	6.29	11.32	Awas	0.624	Tidak jatuh
	2.29	5.19	Tidak jatuh	0.384	Tidak jatuh
	2.97	10.81	Awas	0.439	Tidak jatuh
duduk berdiri	2.33	0.85	Tidak jatuh	0.384	Tidak jatuh
	3.3	6.24	Tidak jatuh	0.384	Tidak jatuh
	3.17	1.94	Tidak jatuh	0.384	Tidak jatuh
naik tangga	3.23	5.69	Tidak jatuh	0.384	Tidak jatuh
	2.07	3.05	Tidak jatuh	0.384	Tidak jatuh
	3.98	9.88	Tidak jatuh	0.459	Tidak jatuh
turun tangga	1.87	4.32	Tidak jatuh	0.384	Tidak jatuh
	3.44	6.25	Tidak jatuh	0.384	Tidak jatuh
	0.77	13.26	Awas	0.384	Tidak jatuh

Dari tabel 4.20 menunjukkan bahwasnya dengan posisi jatuh kemapun alat menunjukkan status terjatuh. Begitu juga untuk simulasi fuzzy menunjukkan nilai minimum 0,34 sampai dengan 1,62 dengan skala 0 sampai dengan 2. Dengan menunjukkan nilai tersebut maka hasil keluaran dari simulasi fuzzy adalah bervariasi dari tidak jatuh sampai dengan jatuh sesuai dengan keadaan hasil

dari keluaran uji alat. Sehingga sensitifitas dari uji alat adalah 72%. Pada sensitifitas dari simulasi fuzzy juga menunjukkna 72% jarena sama-sama menunujukkan nilai yang sama sehingga kesalahan untuk perbandingan sensititifitas dari uji alat dan simulasi adalah 0%. Pada tabel 4.22 tentang perbandingan simulasi aktifitas biasa juga mempunyai performa yang sangat bagus dengan menunjukka spesitifitas dari uji alat 100% dan spesitifitas dari simulasi 100%.

"Halaman ini sengaja dikosongkan"

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil dari tugas akhir ini adalah sebagai berikut:

1. Terdapat karakteristik jatuh yang menghasilkan perbedaan waktu nilai accelerometer dan nilai ADC dengan nilai suara dengan nilai selisih terkecil adalah 9 ms dan yang terbesar adalah 447 ms sehingga pada akhirnya dilakukan pengumpulan data per 100ms dan dilakukan nilai rata-rata untuk dijadikan nilai keluaran sebelum dimasukkan ke proses fuzzy.
2. Hasil uji simulasi fuzzy sebelum PSO adalah sebesar 93% untuk spesitifitas dan 100% untuk sensitifitas. Setelah dilakukan PSO meningkat menjadi 100% spesitifitas dan 100% untuk sensitifitas.
3. Hasil fuzzy PSO menghasilkan 3 fuzzy PSO baru dan hasil pengujian fuzzy PSO 1 dan 3 masih menghasilkan nilai yang sama saat sebelum dilakukan fuzzy PSO yaitu sebesar 93% untuk spesitifitas dan 100% untuk sensitifitas.
4. Dalam percobaan pengambilan data dengan posisi alat di bagaian tubuh berbeda yaitu di dada, perut dan paha. Pada hasil uji posisi alat pada bagaian dada dan perut menunjukkan nilai sensitifitasnya adalah 100% dan nilai spesitifitasnya adalah 100%. Hasilnya berbeda dengan posisi alat diperut dengan menghasilkan nilai sensitifitas 72% dan dan nilai spesitifitasnya adalah 100%.
5. Perbandingan dari uji simulasi dan uji alat untuk sensitifitasnya dan spesifitasnya mempunyai kesalahan 0%.

5.2 Saran

Adapun saran untuk hasil riset kedepannya adalah sebagai berikut

1. Untuk riset selanjutnya jika terdapat penggunaan sistem komunikasi yang memuat permintaan data maka untuk *microcontroller* nya disarankan menggunakan STM32 daripada arduino nano dikarenakan untuk STM32 menggunakan multi proses sedangan untuk arduino nano menggunakan satu proses.
2. Penggunaan sensor suara pada alat detector jatuh kurang memenuhi syarat. Syarat yang dimaksud adalah suara terjatuh terkadang pelan dan suara yang ditangkap terkadang suara lingkungan sehingga untuk selanjutnya bisa penggantian di sensor getar.

DAFTAR PUSTAKA

[1] departemen kesehatan, "Jumlah lansia," *2018*. [Online]. Available: http://www.depkes.go.id/article/view/18050900001/lansia-sejahtera-masyarakat-bahagia-.html.

[2] N. Utami, "Hubungan Dukungan Keluarga Dengan Risiko Jatuh Pada Lansia Di Desa Krasakan Lumbungrejo Tempel Sleman Yogyakarta," *Univ. Aisyiyah, Yogjakarta*, 2017.

[3] P. H. Ashar, "Gambaran persepsi faktor risiko jatuh pada lansia di panti werdha budi mulia 4 margaguna Jakarta Selatan," *Skripsi Fak. Kedokt. dan Ilmu Kesehat. Univ. Islam Negeri Syarif Hidayatullah*, 2016.

[4] W. Putchana, S. Chivapreecha, and T. Limpiti, "Wireless intelligent fall detection and movement classification using fuzzy logic," *5th 2012 Biomed. Eng. Int. Conf. BMEiCON 2012*, 2012.

[5] J. Surana, C. Hemalatha, V.Vaidehi, S. Palavesam, and M. J. A. Khan, "Adaptive learning based human activity and fall detection using fuzzy frequent pattern mining," *2013 Int. Conf. Recent Trends Inf. Technol.*, pp. 744–749, 2013.

[6] Y. P. Huang and J. Y. Chen, "Identifying elderly activity types by interval type-2 fuzzy models," *2014 IEEE Int. Conf. Syst. Man, Cybern.*, pp. 1702–1707, 2014.

[7] V. R. L. Shen, H. Y. Lai, and A. F. Lai, "The implementation of a smartphone-based fall detection system using a high-level fuzzy Petri net," *Appl. Soft Comput. J.*, vol. 26, pp. 390–400, 2015.

[8] B. Kwolek and M. Kepski, "Fuzzy inference-based fall detection using kinect and body-worn accelerometer," *Appl. Soft Comput. J.*, vol. 40, pp. 305–318, 2016.

[9] P. V. Er and K. K. Tan, "Non-intrusive fall detection monitoring for the elderly based on fuzzy logic," *Meas. J. Int. Meas. Confed.*, vol. 124, no. April, pp. 91–102, 2018.

[10] L. FATHURRAHMAN, *Sistem Pendeteksi Kondisi Terjatuh Berbasis Accelerometer Dan Gyroscope*

Menggunakan Adaptive Neuro Fuzzy Inference System. Surabaya: Institut Teknologi Sepuluh Nopember, 2018.

[11] S. K. U. P. K. Melati, A. Kuntari, S. Octarina, and E. S. Cahyono, "Optimasi Produksi Dan Analisis Sensitivitas Menggunakan Algoritma Titik Interior (Studi Kasus: UP2K Melati, Prabumulih)," *J. Mat. Integr.*, vol. 11, no. 1, pp. 75–84, 2015.

[12] M. Riyadi *et al.*, "Pendeteksi Posisi Menggunakan Sensor Accelerometer MMA7260Q Berbasis Mikrokontroler Atmega 32," *Semarang, Tek. Elektro Univ. Diponegoro*, vol. 12, no. 2, pp. 76–81, 2010.

[13] A. U. Darajat, M. Komarudin, and S. R. S, "SISTEM TELEMETRI Unmanned Aerial Vehicle (UAV) BERBASIS Inertial Measurement Unit (IMU)," *Univ. Lampung, Bandar Lampung*, no. 3, pp. 169–177, 2012.

[14] D. Yacchirema, J. S. de Puga, C. Palau, and M. Esteve, "Fall detection system for elderly people using IoT and Big Data," *Procedia Comput. Sci.*, vol. 130, pp. 603–610, 2018.

[15] Linear Technology Corporation, "LTC4056 Linear Li-Ion *Charger* with Termination in ThinSOT," no. V, pp. 1–16.

[16] R. For, "Purpose : Fo r R ef er en nl y Fo r R There are three parts to this survey : er en ce nl y," *Learning*, pp. 1–6, 2011.

[17] B. P. Ic, "S-8205A/B Series BATTERY PROTECTION IC," pp. 2010–2016, 2016.

[18] "DATASHEET," pp. 1–10.

[19] S. M. A. Do-ac, "Ss12, ss13, ss14, ss15, ss16," pp. 15–18.

[20] STMicroeletronics, "STM32F103x8 STM32F103xB: Medium-density performance line ARM®-based 32-bit MCU with 64 or 128 KB Flash, USB, CAN, 7 timers, 2 ADCs, 9 com. interfaces [datasheet]," no. August, pp. 1–117, 2015.

[21] G. Yu, "PSO-based Fuzzy Control of a Self-Balancing Two-Wheeled Robot," *IEEE*, pp. 1–5, 2017.

BIODATA PENULIS

Nama penulis yaitu Akhmad Zainudin. Penulis lahir di Gresik, 10 Juli 1993. Pada tahun 2005 penulis menyelesaikan pendidikan Sekolah Dasar di Minu Salafiyyah, Bedilan, Gresik.

pada tahun 2008 menyelesaikan pendidikan Sekolah Menengah Pertama di SMP Semen Gresik, Gresik, Pada tahun 2011 menyelesaikan pendidikan di jurusan Instalasi listrik, Sekolah Menengah Kejuruan di SMK Semen Gresik, Gresik. Pada tahun 2014 Menyelesaikan pendidikan diploma 3 diprodi D3 Metrologi Instrumentasi, Institut Teknologi Sepuluh Nopember. Setelah lulus dari diploma 3, penulis bekerja di proyek pembangunan pabrik semen di Rembang, Jawa Tengah lalu pada tahun 2016 penulis membuka outlet Sego Njamoer dikawasan Jalan Kartini, Gresik dan memulai pendidikan strata satu atau lintas jalur di Departemen Teknik Fisika, Institut Teknologi Sepuluh Nopember namun ditengah pendidikannya penulis dipercaya menjadi evaluator teknis elektrikal dan instrumentasi dan komersial di proyek pembangunan pabrik semen di Aceh.

Konsentrasi yang didalami pada tugas akhir adalah bidang Rekayasa Instrumentasi dan Kontrol. Pada bulan Desember 2018 penulis telah menyelsaikan Tugas Akhir dengan Judul **SISTEM PENDETEKSI KONDISI TERJATUH BERBASIS SENSOR POSISI DAN SUARA MENGGUNAKAN FUZZY PSO (PARTICLE SWARM OPTIMIZATION).**

Bagi pembaca yang memiliki kritik, saran atau ingin berdiskusi lebih lanjut mengenai tugas akhir ini, anda dapat menghubungi melalui email: akhmadzainudin24@gmail.com.

“Halaman ini sengaja dikosongkan

LISTING KODE ANDROID

```
package com.example.dinApp;

import android.annotation.SuppressLint;
import android.app.Activity;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.text.method.ScrollingMovementMethod;
import android.view.Gravity;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.io.DataInputStream;
import java.io.IOException;
import java.net.Socket;

public class MyActivity extends Activity {
    /**
     * Called when the activity is first created.
     */

    Button btnReq;
    Button btnNotif;
    Button btnStart;
    TextView txtAx;
    TextView txtAy;
    TextView txtAz;
    TextView txtMag;
    TextView txtMic;
    TextView txtFall;
    TextView txtFuzz;
    TextView txtOut;

    int notif = 0;
    private Runnable datreq;
    Handler hreq;
```

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    btnReq = (Button) findViewById(R.id.btnReq);
    btnNotif = (Button) findViewById(R.id.btnNotif);
    btnStart = (Button) findViewById(R.id.btnStart);
    txtAx = (TextView) findViewById(R.id.txtAx);
    txtAy = (TextView) findViewById(R.id.txtAy);
    txtAz = (TextView) findViewById(R.id.txtAz);
    txtMag = (TextView) findViewById(R.id.txtMag);
    txtMic = (TextView) findViewById(R.id.txtMic);
    txtFall = (TextView) findViewById(R.id.txtFall);
    txtFuzz = (TextView) findViewById(R.id.txtFuzz);
    txtOut = (TextView) findViewById(R.id.txtOut);
    txtOut.setMovementMethod(new ScrollingMovementMethod());

    hreq = new Handler();
    datreq = new Runnable() {
        @Override
        public void run() {
            txtOut.append("Trying to Request\n");
            MyClientTask mycl = new MyClientTask("192.168.4.1");
            mycl.execute();
            hreq.postDelayed(this,300);
        }
    };

    btnReq.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            txtOut.append("Trying to Request\n");
            MyClientTask mycl = new MyClientTask("192.168.4.1");
            mycl.execute();
        }
    });

    btnNotif.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            showNotification();
        }
    });

    btnStart.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            txtOut.append("...Data Request Start...\n");
```

```
            hreq.post(datreq);
            btnStart.setEnabled(false);
            btnNotif.setEnabled(false);
            btnReq.setEnabled(false);
        }
    });

    txtOut.append("Ready\n");
}

private void Pesan(String txtpesan) {
    Context context = getApplicationContext();
    int duration = Toast.LENGTH_LONG;

    Toast toast = Toast.makeText(context, txtpesan, duration);

    toast.setGravity(Gravity.CENTER, 0, 0);
    toast.show();
}

private void showNotification() {

    NotificationManager notificationManager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);

    String MyText = "Peringatan";
    Notification mNotification = new Notification(R.drawable.ic_launcher, MyText,
System.currentTimeMillis() );

    String MyNotificationTitle = "Nenek anda terjatuh!";
    String MyNotificationText  = "Tolong lihat Nenek anda";

    Intent MyIntent = new Intent(Intent.ACTION_VIEW);
    PendingIntent StartIntent =
PendingIntent.getActivity(getApplicationContext(),0,MyIntent,
PendingIntent.FLAG_CANCEL_CURRENT);

    mNotification.setLatestEventInfo(getApplicationContext(), MyNotificationTitle,
MyNotificationText, StartIntent);

    int NOTIFICATION_ID = 1;
    notificationManager.notify(NOTIFICATION_ID , mNotification);

    try {
        Uri notifSnd =
RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);
        Ringtone r = RingtoneManager.getRingtone(getApplicationContext(), notifSnd);
        r.play();
    } catch (Exception e) {
        e.printStackTrace();
```

```
    }

}


@SuppressLint("StaticFieldLeak")
public class MyClientTask extends AsyncTask<Void, Void, Void> {

    String hostIP;
    int str;
    StringBuilder strData = new StringBuilder();

    MyClientTask(String strIP) {
        hostIP = strIP;
    }

    @Override
    protected Void doInBackground(Void... arg0){
        Socket socket = null;
        DataInputStream dataInputStream = null;

        try {
            socket = new Socket(hostIP,80);

            dataInputStream = new DataInputStream(socket.getInputStream());

            while ((str = dataInputStream.read()) != -1) {
                strData.append((char) str);
            }

        } catch (IOException e) {
            Pesan("e");
        } finally {
            if (socket != null) {
                try {
                    socket.close();
                } catch (IOException e) {
                    Pesan("e");
                }
            }
            if (dataInputStream != null) {
                try {
                    dataInputStream.close();
                } catch (IOException e) {
                    Pesan("e");
                }
            }
        }
        return null;
    }
```

```
        @Override
        protected void onPostExecute(Void result) {
            txtOut.append(strData.toString() + "\n");

            String resData = strData.toString().trim();

            if (resData.length() > 0) {
                String[] arrStrData = resData.split(":");

                if (arrStrData.length > 6) {
                    txtMic.setText(arrStrData[0].trim());
                    txtAx.setText(arrStrData[1].trim());
                    txtAy.setText(arrStrData[2].trim());
                    txtAz.setText(arrStrData[3].trim());
                    txtMag.setText(arrStrData[4].trim());
                    txtFuzz.setText(arrStrData[5].trim());
                    txtFall.setText(arrStrData[6].trim());

                    if(Integer.parseInt(arrStrData[6].trim())==1){
                        if(notif==0) {
                            showNotification();
                            notif=1;
                        }
                    }

                }
            }

            super.onPostExecute(result);
        }
    }
}
```

LISTING KODE STM 32 SENSOR MIC

```
/**
 * @file    din_adc.c
 * @brief   ADC (for Mic) routine code.
 *
 * @addtogroup ADC
 * @{
 */

#include "din_adc.h"
#include "din_mpu.h"
```

```
/**
 * @brief   ADC buffer variable.
 */
static adcsample_t samples[ADC_GRP_NUM_CHANNELS *
ADC_GRP_BUF_DEPTH];
adcsample_t adc0,adc00,adc01;
uint32_t sum_adc0;
float v_adc;

void adccb(ADCDriver *adcp, adcsample_t *buffer, size_t n){
  (void) buffer; (void) n;
  int i;
  if (adcp->state == ADC_COMPLETE) {
    sum_adc0=0;
    for(i=0;i<ADC_GRP_BUF_DEPTH;i++){
       sum_adc0=sum_adc0+samples[0+(i*ADC_GRP_NUM_CHANNELS)];
    }
    adc0=sum_adc0/ADC_GRP_BUF_DEPTH;
   }
 }

/**
 * @brief   ADC configuration variable.
 */
static const ADCConversionGroup adcgrpcfg = {
    FALSE,
    ADC_GRP_NUM_CHANNELS,
    adccb,
    NULL,
    /* HW dependent part.*/
    0,
    0,
    ADC_SMPR2_SMP_AN0(ADC_SAMPLE_239P5),
    0,
    ADC_SQR1_NUM_CH(ADC_GRP_NUM_CHANNELS),
    0,
    ADC_SQR3_SQ1_N(ADC_CHANNEL_IN0)
};

static THD_WORKING_AREA(waADC, 128);
static THD_FUNCTION(thdADC, arg) {

    (void)arg;
```

```
   chRegSetThreadName("ADC Conversion");
   while (true) {
      chThdSleepMicroseconds(10);
      adcStartConversion(&ADCD1, &adcgrpcfg, samples,
ADC_GRP_BUF_DEPTH);
   }
}

/**
 * @brief   ADC start function.
 */
void d_adc_start(void){
   palSetPadMode(GPIOA,0,PAL_MODE_INPUT_ANALOG);
   adcStart(&ADCD1, NULL);
   chThdCreateStatic(waADC, sizeof(waADC), NORMALPRIO, thdADC,
NULL);
}

/** @} */
```

KODING MPU 6050

```
/**
 * @file    din_mpu.c
 * @brief   MPU Sensor routine code.
 *
 * @addtogroup MPU60
 * @{
 */

#include "din_mpu.h"

extern adcsample_t adc0,adc00,adc01;

////////////////////////////////////// Rutin MPU60x0 ////////////////

/**
 * @brief   Global variable for Accel vector X.
 */
float ax,ax0,ax1;

/**
 * @brief   Global variable for Accel vector Y.
 */
```

```
float ay,ay0,ay1;

/**
 * @brief   Global variable for Accel vector Z.
 */
float az,az0,az1;

/**
 * @brief   Global variable for Accel vector Magnitude.
 */
float mag;

/**
 * @brief   Global variable for Normalization at 2G.
 */
float rangePerDigit = .000061f;

/**
 * @brief   Global variable for Sample Rate Divisor.
 */
static u_int8_t smplrt_div= 0;

/**
 * @brief   Global variable for MPU Configuration.
 */
static u_int8_t mpu_config = 0;

/**
 * @brief   Global variable for Gyro Configuration.
 */
static u_int8_t gyro_config = 0;

/**
 * @brief   Global variable for Accel Configuration.
 */
static u_int8_t accel_config = 0;

/**
 * @brief   Global variable for User Control Configuration.
 */
static u_int8_t user_control = 0x00;

/**
 * @brief   Global variable for Power Management.
```

```
 */
static u_int8_t power_mgmt1 = 0x00;

//static u_int8_t fifo_enable = 0x00;
//static u_int8_t int_pin_config = 0x00;
//static u_int8_t int_pin_enable = 0x00;
//static u_int8_t signal_path_reset = 0x00;
//static u_int8_t power_mgmt2  = 0x00;
//static u_int8_t aux_vddio = 0x00;

/**
 * @brief   Defines value for SMPRT_DIV register.
 * @note    Sample Rate = Gyroscope Output Rate / (1 + SMPLRT_DIV)
 * @param[in] samplerate_divisor value of divisor
 */
static u_int8_t d_mpu_setSampleRate(u_int8_t samplerate_divisor){
   smplrt_div = samplerate_divisor;
#if MPU_DEBUG
   chprintf((BaseSequentialStream *)&SD1,"Sample rate value: 0x%x\r\n",
smplrt_div);
#endif
   return smplrt_div;
}

/**
 * @brief   Defines register controls FSYNC and bandwidth of gyro and
accelerometer.
 * @param[in] ext_sync_set typical @see EXT_SYNC_SET0
 * @param[in] dlpf_cfg typical @see DLPF_CFG0
 */
static u_int8_t d_mpu_configRegister(u_int8_t ext_sync_set, u_int8_t
dlpf_cfg){
   mpu_config = 0x00;
   mpu_config = ext_sync_set | dlpf_cfg;
#if MPU_DEBUG
   chprintf((BaseSequentialStream *)&SD1,"CONFIG Register value:
0x%x\r\n", mpu_config);
#endif
   return mpu_config;
}

/**
 * @brief   Defines register controls self test and  range of gyroscopes.
 * @param[in] xgyro_st typical @see XG_ST_EN
```

```
 * @param[in] ygyro_st typical @see YG_ST_EN
 * @param[in] zgyro_st typical @see ZG_ST_EN
 * @param[in] gyro_range typical @see FS_SEL250
 */
static u_int8_t d_mpu_setGyro(u_int8_t xgyro_st, u_int8_t ygyro_st, u_int8_t
zgyro_st, u_int8_t gyro_range){
    gyro_config = 0x00;
    gyro_config = xgyro_st | ygyro_st | zgyro_st | gyro_range;
#if MPU_DEBUG
    chprintf((BaseSequentialStream *)&SD1,"Gyro Config value: 0x%x\r\n",
gyro_config);
#endif
    return gyro_config;
}

/**
 * @brief   Defines register controls self test, accelerometer range and DHPF for
accelerometer.
 * @param[in] xaccel_st typical @see XA_ST_EN or @see XA_ST_DIS
 * @param[in] yaccel_st typical @see YA_ST_EN or @see YA_ST_DIS
 * @param[in] zaccel_st typical @see ZA_ST_EN or @see ZA_ST_DIS
 * @param[in] accel_range typical @see AFS_SEL0
 * @param[in] dhpf_accel typical @see ACCEL_HPF0
 */
static u_int8_t d_mpu_setAccel(u_int8_t xaccel_st, u_int8_t yaccel_st, u_int8_t
zaccel_st, u_int8_t accel_range, u_int8_t dhpf_accel){
    accel_config = 0x00;
    accel_config = xaccel_st | yaccel_st | zaccel_st | accel_range | dhpf_accel;
#if MPU_DEBUG
    chprintf((BaseSequentialStream *)&SD1,"Accelerometer Config value:
0x%x\r\n", accel_config);
#endif
    return accel_config;
}

/*
 * This function defines value for FIFO_EN register. This register controls
which sensor output to be
 * written in fifo regiser.
 * Typical funciton call: set_mpu_fifo_register(TEMP_FIFO_EN/DIS,
XG_FIFO_EN/DIS, YG_FIFO_EN/DIS, ACCLE_FIFO_EN/DIS,
SLVx_FIFO_EN/DIS...)
 */
```

```
//static u_int8_t d_mpu_fifoRegister(u_int8_t temperature_fifo, u_int8_t
xg_fifo, u_int8_t yg_fifo, u_int8_t zg_fifo, u_int8_t accel_fifo, u_int8_t
slv2_fifo, u_int8_t slv1_fifo,u_int8_t slv0_fifo){
//    fifo_enable = 0x00;
//    fifo_enable = temperature_fifo | xg_fifo | yg_fifo | zg_fifo | accel_fifo |
slv2_fifo | slv1_fifo | slv0_fifo;
//#if MPU_DEBUG
//    chprintf((BaseSequentialStream *)&SD1,"FIFO Enable value: 0x%x\r\n",
fifo_enable);
//#endif
//    return fifo_enable;
//}

/*
 * This function defines value for INT_PIN_CFG register. This controls
behavior of Interrupt PIN
 */
//static u_int8_t d_mpu_intBehavior(u_int8_t int_level, u_int8_t int_pin_mode,
u_int8_t latch_int, u_int8_t int_status_bits, u_int8_t fsync_level, u_int8_t
fsync_enable, u_int8_t i2c_bypass, u_int8_t clock){
//    int_pin_config = 0x00;
//    int_pin_config = int_level | int_pin_mode | latch_int | int_status_bits |
fsync_level | fsync_enable | i2c_bypass |clock;
//#if MPU_DEBUG
//    chprintf((BaseSequentialStream *)&SD1,"Interrupt behavior value:
0x%x\r\n", int_pin_config);
//#endif
//    return int_pin_config;
//}

/*
 * This function defines value for INT_ENABLE register. This register controls
source of interrupt.
 * Typical function call: set_mpu_interrupt_sources(FF_EN/DIS,
MOT_EN/DIS,.....,DATA_RDY_EN/DIS)
 */
//static u_int8_t d_mpu_intSource(u_int8_t free_fall, u_int8_t motion_threshold,
u_int8_t zero_motion, u_int8_t fifo_overflow, u_int8_t i2c_mst, u_int8_t
data_ready){
//    int_pin_enable = 0x00;
//    int_pin_enable = free_fall | motion_threshold | zero_motion | fifo_overflow |
i2c_mst | data_ready;
//#if MPU_DEBUG
```

```
//    chprintf((BaseSequentialStream *)&SD1,"Interrupt Pin Source value value:
0x%x\r\n", int_pin_enable);
//#endif
//    return int_pin_enable;
//}

/*
 * This function defines value for SIGNAL_PATH_RESET register. This
register can reset gyro, accelerometer and
 * temperature sensors' digital and analog signal path.
 * Typical function call: reset_mpu_signal_path(GYRO_RESET_EN/DIS,
ACCEL_RESET_EN/DIS, TEMP_RESET_EN/DIS)
 */
//static u_int8_t d_mpu_Reset(u_int8_t gyro_reset, u_int8_t accel_reset,
u_int8_t temperature_reset){
//    signal_path_reset = 0x00;
//    signal_path_reset = gyro_reset | accel_reset | temperature_reset;
//#if MPU_DEBUG
//    chprintf((BaseSequentialStream *)&SD1,"Signal Path Reset value:
0x%x\r\n", signal_path_reset);
//#endif
//    return signal_path_reset;
//}

/**
 * @brief   Defines value for USER_CTRL register.
 * @param[in] fifo_operation typical @see USER_FIFO_EN or @see
USER_FIFO_DIS
 * @param[in] aux_i2c typical @see I2C_MST_EN or @see I2C_MST_DIS
 * @param[in] bus_select typical @see I2C_IF_EN or @see I2C_IF_DIS
 * @param[in] fifo_reset typical @see FIFO_RESET_EN or @see
FIFO_RESET_DIS
 * @param[in] i2c_reset typical @see I2C_MST_EN or @see I2C_MST_DIS
 * @param[in] signal_cond_reset typical @see SIG_COND_RESET_EN or
@see SIG_COND_RESET_DIS
 */
static u_int8_t d_mpu_userControl(u_int8_t fifo_operation, u_int8_t aux_i2c,
u_int8_t bus_select, u_int8_t fifo_reset, u_int8_t i2c_reset, u_int8_t
signal_cond_reset){
   user_control = 0x00;
   user_control = fifo_operation | aux_i2c | bus_select | fifo_reset | i2c_reset |
signal_cond_reset;
#if MPU_DEBUG
```

```
   chprintf((BaseSequentialStream *)&SD1,"User Control value: 0x%x\r\n",
user_control);
#endif
   return user_control;
}

/**
 * @brief   Defines register device reset, sleep mode, cycle between different
mode and clock source.
 * @param[in] device_reset typical @see DEVICE_RESET_EN or @see
DEVICE_RESET_DIS
 * @param[in] sleep typical @see SLEEP_EN or @see SLEEP_DIS
 * @param[in] cycle typical @see CYCLE_EN or @see CYCLE_DIS
 * @param[in] temperature typical @see TEMPERATURE_EN or @see
TEMPERATURE_DIS
 * @param[in] clock_source typical @see CLKSEL_EN or @see CLKSEL_DIS
 */
static u_int8_t d_mpu_powerMGMT1(u_int8_t device_reset, u_int8_t sleep,
u_int8_t cycle, u_int8_t temperature, u_int8_t clock_source){
   power_mgmt1 = 0x00;
   power_mgmt1 = device_reset | sleep | cycle | temperature | clock_source;
#if MPU_DEBUG
   chprintf((BaseSequentialStream *)&SD1,"Power Managemen1 value:
0x%x\r\n", power_mgmt1);
#endif
   return power_mgmt1;
}

/**
 * @brief   Write to @see SMPRT_DIV register
 */
static void d_mpu_writeSampleRate(void){
    d_mpu_i2cWrite(SMPRT_DIV, smplrt_div);
}

/*
 * This function writes value of configuration into CONFIG register.
 */
//static void d_mpu_writeConfigRegister(){
//    d_mpu_i2cWrite(CONFIG, mpu_config);
//}

/**
 * @brief   Write to @see GYRO_CONFIG register
```

```
 */
static void d_mpu_writeGyro(void){
   d_mpu_i2cWrite(GYRO_CONFIG, gyro_config);
}

/**
 * @brief   Write to @see ACCEL_CONFIG register
 */
static void d_mpu_writeAccel(void){
   d_mpu_i2cWrite(ACCEL_CONFIG, accel_config);
}

/**
 * @brief   Write to @see POWER_MGMT1 register
 */
static void d_mpu_writePowerMGMT1(void){
   d_mpu_i2cWrite(PWR_MGMT_1, power_mgmt1);
}

/*
 * This function writes value of user_control into USER_CTRL register.
 */
//static void d_mpu_writeUserControl(void){
//    d_mpu_i2cWrite(USER_CTRL, user_control);
//}

/**
 * @brief   Write to I2C bus
 * @param[in] addr Address to write
 * @param[in] value Value to write
 */
void d_mpu_i2cWrite(u_int8_t addr, u_int8_t value){
   u_int8_t mpu_txbuf[10], mpu_rxbuf[10];
   mpu_txbuf[0] = addr;
   mpu_txbuf[1] = value;

#if MPU_DEBUG
   chprintf((BaseSequentialStream *)&SD1,"Address: 0x%x Value: 0x%x\r\n",
mpu_txbuf[0],mpu_txbuf[1]);
#endif

   i2cAcquireBus(&I2CD2);
   i2cMasterTransmit(&I2CD2, MPU_ADDR, mpu_txbuf, 2, mpu_rxbuf, 0);
   i2cReleaseBus(&I2CD2);
```

```
    chThdSleepMilliseconds(100);
}

/**
 * @brief   Write then Read to/from I2C bus
 * @param[in] addr Address to write. Typical 0x3B
 * @param[in] length Length of response. Typical 14
 * @note First three values are accelerometer.
 * @note Fourth one is temperature sensor.
 * @note Last three are gyro values.
 * @note Havent explored FIFO and interrupt part yet.
 */
void d_mpu_i2cReadData(u_int8_t addr, u_int8_t length){
    u_int8_t mpu_txbuf[20], mpu_rxbuf[20], i = 0;
    u_int16_t mpu_val[10];

    mpu_txbuf[0] = addr;
    for(i=0;i<length;i++)mpu_rxbuf[i] = 0x00;

    i2cAcquireBus(&I2CD2);
    i2cMasterTransmit(&I2CD2, MPU_ADDR, mpu_txbuf, 1, mpu_rxbuf,
length);
    i2cReleaseBus(&I2CD2);

    mpu_val[0] = (mpu_rxbuf[0] << 8) + mpu_rxbuf[1];
    mpu_val[1] = (mpu_rxbuf[2] << 8) + mpu_rxbuf[3];
    mpu_val[2] = (mpu_rxbuf[4] << 8) + mpu_rxbuf[5];
    mpu_val[3] = (mpu_rxbuf[6] << 8) + mpu_rxbuf[7];
    mpu_val[4] = (mpu_rxbuf[8] << 8) + mpu_rxbuf[9];
    mpu_val[5] = (mpu_rxbuf[10] << 8) + mpu_rxbuf[11];
    mpu_val[6] = (mpu_rxbuf[12] << 8) + mpu_rxbuf[13];

    ax1 = mpu_val[0] * rangePerDigit * 9.80665f;
    ay1 = mpu_val[1] * rangePerDigit * 9.80665f;
    az1 = mpu_val[2] * rangePerDigit * 9.80665f;

    ax = abs(ax1-ax0);
    ay = abs(ay1-ay0);
    az = abs(az1-az0);

    ax0 = ax1;
    ay0 = ay1;
    az0 = az1;
```

```
    mag = d_mpu_vectorMag(ax,ay,az);

#if MPU_DEBUG
    chprintf((BaseSequentialStream *)&SD1, "Result:");
    chprintf((BaseSequentialStream *)&SD1,
"\t%5.2f\t%5.2f\t%5.2f\t%5.2f",ax,ay,az,mag);
    chprintf((BaseSequentialStream *)&SD1, "\r\n");
#endif

}

void d_mpu_i2cReadFirst(u_int8_t addr, u_int8_t length){
    u_int8_t mpu_txbuf[20], mpu_rxbuf[20], i = 0;
    u_int16_t mpu_val[10];

    mpu_txbuf[0] = addr;
    for(i=0;i<length;i++)mpu_rxbuf[i] = 0x00;

    i2cAcquireBus(&I2CD2);
    i2cMasterTransmit(&I2CD2, MPU_ADDR, mpu_txbuf, 1, mpu_rxbuf,
length);
    i2cReleaseBus(&I2CD2);

    mpu_val[0] = (mpu_rxbuf[0] << 8) + mpu_rxbuf[1];
    mpu_val[1] = (mpu_rxbuf[2] << 8) + mpu_rxbuf[3];
    mpu_val[2] = (mpu_rxbuf[4] << 8) + mpu_rxbuf[5];
    mpu_val[3] = (mpu_rxbuf[6] << 8) + mpu_rxbuf[7];
    mpu_val[4] = (mpu_rxbuf[8] << 8) + mpu_rxbuf[9];
    mpu_val[5] = (mpu_rxbuf[10] << 8) + mpu_rxbuf[11];
    mpu_val[6] = (mpu_rxbuf[12] << 8) + mpu_rxbuf[13];

    ax0 = mpu_val[0] * rangePerDigit * 9.80665f;
    ay0 = mpu_val[1] * rangePerDigit * 9.80665f;
    az0 = mpu_val[2] * rangePerDigit * 9.80665f;
    mag = d_mpu_vectorMag(ax0,ay0,az0);

#if MPU_DEBUG
    chprintf((BaseSequentialStream *)&SD1, "Result:");
    chprintf((BaseSequentialStream *)&SD1,
"\t%5.2f\t%5.2f\t%5.2f\t%5.2f",ax,ay,az,mag);
    chprintf((BaseSequentialStream *)&SD1, "\r\n");
#endif
```

```
   adc00 = adc0;
}

////////////////////////////////////////////////////////////////////////////////

/**
 * @brief   I2C Config Variable
 */
static const I2CConfig i2cfg = {
   OPMODE_I2C,
   400000,
   FAST_DUTY_CYCLE_2,
};

/**
 * @brief   Write to I2C bus
 * @param[in] vx Value of Accel at vector X
 * @param[in] vy Value of Accel at vector Y
 * @param[in] vz Value of Accel at vector Z
 * @return Magnitude of vector resultant
 */
float d_mpu_vectorMag(float vx, float vy, float vz){
   return sqrt( pow(vx,2) + pow(vy,2) + pow(vz,2) );
}

/**
 * @brief   Test ID of MPU60
 * @note    result must be either 0x68 or 0x69
 */
void d_mpu_whoAmI(void){
   u_int8_t tx_buff[8];
   u_int8_t rx_buff[8];

#if MPU_DEBUG
   i2cflags_t errors = 0;
   msg_t status = MSG_OK;
#endif

   tx_buff[0] = 0x75;

   i2cAcquireBus(&I2CD2);
#if MPU_DEBUG
   status = i2cMasterTransmit(&I2CD2, MPU_ADDR, tx_buff, 1, rx_buff, 2);
//rx_buff length 1 failed
```

```
#else
   i2cMasterTransmit(&I2CD2, MPU_ADDR, tx_buff, 1, rx_buff, 2); //rx_buff
length 1 failed
#endif
   i2cReleaseBus(&I2CD2);

#if MPU_DEBUG
   if (status != MSG_OK){
      errors = i2cGetErrors(&I2CD2);
      chprintf((BaseSequentialStream *)&SD1,"s=%i | e=%i\r\n",status,errors);
   }
   else{
      chprintf((BaseSequentialStream *)&SD1,"id: 0x%x\r\n",rx_buff[0]);
      chprintf((BaseSequentialStream *)&SD1,"i2c accel oke\r\n");
   }
#endif

}

/**
 * @brief   Setup for MPU60 module
 */
void d_mpu_setup(void){
   d_mpu_setSampleRate(9);
   d_mpu_configRegister(EXT_SYNC_SET0,DLPF_CFG0);
   d_mpu_setGyro(XG_ST_DIS,YG_ST_DIS,ZG_ST_DIS,FS_SEL_250);

d_mpu_setAccel(XA_ST_DIS,YA_ST_DIS,ZA_ST_DIS,AFS_SEL_2g,ACCE
L_HPF0);

d_mpu_powerMGMT1(DEVICE_RESET_DIS,SLEEP_DIS,CYCLE_DIS,TEM
PERATURE_EN,CLKSEL_XG);
   d_mpu_userControl(USER_FIFO_DIS, I2C_MST_DIS, I2C_IF_DIS,
FIFO_RESET_DIS, I2C_MST_RESET_DIS, SIG_COND_RESET_DIS);

   d_mpu_writePowerMGMT1();
   d_mpu_writeGyro();
   d_mpu_writeAccel();
   d_mpu_writeSampleRate();
}

/**
 * @brief   Start I2C Peripheral and MPU60 module
 */
```

```
void d_mpu_start(void){
   palSetPadMode(GPIOB, 10,
PAL_MODE_STM32_ALTERNATE_OPENDRAIN );   /* SCL */
   palSetPadMode(GPIOB, 11,
PAL_MODE_STM32_ALTERNATE_OPENDRAIN );   /* SDA */
   i2cStart(&I2CD2, &i2cfg);

   chThdSleepMilliseconds(200);
   d_mpu_setup();

   chThdSleepMilliseconds(200);
   d_mpu_whoAmI();

   chThdSleepMilliseconds(200);
   d_mpu_i2cReadFirst(0x3B,14);
}
/** @} */
```

KODING FUZZY

```
/**
 * @file    din_fuzzy.c
 * @brief   Fuzzy Logic code.
 *
 * @addtogroup Fuzzy
 * @{
 */

#include "din_fuzzy.h"

/**
 * @brief   Accelerometer fuzzy array variable
 */
static float f_accel[3];

/**
 * @brief   Mic fuzzy array variable
 */
static float f_mic[3];

/**
```

```
 * @brief   Fuzzy rule array variable
 */
static float f_rule[RULE_SIZE];

/**
 * @brief   Accelerometer low membership array variable
 */
static float vf_accel_rendah[2] = {7, 7.78493};

/**
 * @brief   Accelerometer mid membership array variable
 */
static float vf_accel_sedang[3] = {7.6109, 9, 10.3521};

/**
 * @brief   Accelerometer high membership array variable
 */
static float vf_accel_tinggi[2] = {9.0384, 13};

/**
 * @brief   Mic low membership array variable
 */
static float vf_mic_rendah[2] = {5, 7.57378};

/**
 * @brief   Mic mid membership array variable
 */
static float vf_mic_sedang[3] = {5.5467, 8, 9.31291};

/**
 * @brief   Mic high membership array variable
 */
static float vf_mic_tinggi[2] = {7.33903,12 };

/**
 * @brief   Accelerometer membership fuzzyfication
 * @param[in] input_accel Input from Accelerometer
 */
static void Accel(float input_accel){
   // fuzzy RENDAH
```

```
    if(input_accel <= vf_accel_rendah[0]){
        f_accel[0] = 1;
    }
    else if((input_accel > vf_accel_rendah[0]) && (input_accel <=
vf_accel_rendah[1])){
        f_accel[0] = (vf_accel_rendah[1] -
input_accel)/(vf_accel_rendah[1] - vf_accel_rendah[0]);
    }
    else if(input_accel > vf_accel_rendah[1]){
        f_accel[0] = 0;
    }

    // fuzzy SEDANG
    if(input_accel <= vf_accel_sedang[0]){
        f_accel[1] = 0;
    }
    else if((input_accel > vf_accel_sedang[0]) && (input_accel <=
vf_accel_sedang[1])){
        f_accel[1] = (input_accel -
vf_accel_sedang[0])/(vf_accel_sedang[1] - vf_accel_sedang[0]);
    }
    else if((input_accel > vf_accel_sedang[1]) && (input_accel <=
vf_accel_sedang[2])){
        f_accel[1] = (vf_accel_sedang[2] -
input_accel)/(vf_accel_sedang[2] - vf_accel_sedang[1]);
    }
    else if(input_accel > vf_accel_sedang[2]){
        f_accel[1] = 0;
    }

    // fuzzy TINGGI
    if(input_accel <= vf_accel_tinggi[0]){
        f_accel[2] = 0;
    }
    else if((input_accel > vf_accel_tinggi[0]) && (input_accel <=
vf_accel_tinggi[1])){
        f_accel[2] = (input_accel - vf_accel_tinggi[0])/(vf_accel_tinggi[1] -
vf_accel_tinggi[0]);
    }
    else if(input_accel > vf_accel_tinggi[1]){
```

```
        f_accel[2] = 1;
    }
}

/**
 * @brief   Mic membership fuzzyfication
 * @param[in] input_mic Input from Mic
 */
static void Mic(unsigned int input_mic){
    // fuzzy RENDAH
    if(input_mic <= vf_mic_rendah[0]){
        f_mic[0] = 1;
    }
    else if((input_mic > vf_mic_rendah[0]) && (input_mic <=
vf_mic_rendah[1])){
        f_mic[0] = (vf_mic_rendah[1] - input_mic)/(vf_mic_rendah[1] -
vf_mic_rendah[0]);
    }
    else if(input_mic > vf_mic_rendah[1]){
        f_mic[0] = 0;
    }

    // fuzzy SEDANG
    if(input_mic <= vf_mic_sedang[0]){
        f_mic[1] = 0;
    }
    else if((input_mic > vf_mic_sedang[0]) && (input_mic <=
vf_mic_sedang[1])){
        f_mic[1] = (input_mic - vf_mic_sedang[0])/(vf_mic_sedang[1] -
vf_mic_sedang[0]);
    }
    else if((input_mic > vf_mic_sedang[1]) && (input_mic <=
vf_mic_sedang[2])){
        f_mic[1] = (vf_mic_sedang[2] - input_mic)/(vf_mic_sedang[2] -
vf_mic_sedang[1]);
    }
    else if(input_mic > vf_mic_sedang[2]){
        f_mic[1] = 0;
    }
```

```
    // fuzzy TINGGI
    if(input_mic <= vf_mic_tinggi[0]){
        f_mic[2] = 0;
    }
    else if((input_mic > vf_mic_tinggi[0]) && (input_mic <=
vf_mic_tinggi[1])){
        f_mic[2] = (input_mic - vf_mic_tinggi[0])/(vf_mic_tinggi[1] -
vf_mic_tinggi[0]);
    }
    else if(input_mic > vf_mic_tinggi[1]){
        f_mic[2] = 1;
    }
}

/**
 * @brief   Evaluasi Fuzzy
 * @return Index maximum membership of fuzzy array
 */
static u_int16_t Evaluasi(void){
    unsigned int m,n;
    unsigned int i,idx=0;
    float v_max;

    for(m=0;m<=2;m++){
        for(n=0;n<=2;n++){
            f_rule[3*m + n]=f_accel[m]*f_mic[n];
        }
    }

    v_max = f_rule[0];

    for(i=1;i<RULE_SIZE;i++){
        if(f_rule[i] > v_max){
            v_max = f_rule[i];
            idx = i;
        }
    }

    return idx;
}
```

```
/**
 * @brief  Decision based on fuzzy rule array
 * @return Index of Response fuzzy logic
 */
static u_int16_t Decision(unsigned int index){
   unsigned int out_fuzzy=0;

   switch (index){
   case 0:
      out_fuzzy = OUT_BIASA; //rendah,sunyi
      break;
   case 1:
      out_fuzzy = OUT_BIASA; //rendah,lumayan
      break;
   case 2:
      out_fuzzy = OUT_BIASA; //rendah,ramai
      break;
   case 3:
      out_fuzzy = OUT_BIASA; //sedang,sunyi
      break;
   case 4:
      out_fuzzy = OUT_AWAS; //sedang,lumayan
      break;
   case 5:
      out_fuzzy = OUT_AWAS; //sedang,ramai
      break;
   case 6:
      out_fuzzy = OUT_AWAS; //tinggi,sunyi
      break;
   case 7:
      out_fuzzy = OUT_AWAS; //tinggi,lumayan
      break;
   case 8:
      out_fuzzy = OUT_JATUH; //tinggi,ramai
      break;
   }

   return out_fuzzy;
}
```

```
/**
 * @brief  Main Fuzzy routine
 * @return Response fuzzy logic
 */
u_int8_t d_fuzzy(float vmag, float vmic){
   Accel(vmag);
   Mic(vmic);

   return Decision(Evaluasi());
}
/** @} */
```

KODING WIFI

```
/**
 * @file    din_web.c
 * @brief   Web server routine code.
 *
 * @addtogroup Communication
 * @{
 */

#include "din_web.h"

extern adcsample_t adc0,adc00,adc01;
extern float v_adc;
extern float ax,ay,az,mag;

float s_ax,s_ay,s_az;
float s_v_adc,s_mag;
float m_ax,m_ay,m_az;
float m_v_adc,m_mag;

u_int8_t fuz_res;
u_int8_t fall_stt=0;

static THD_WORKING_AREA(waData, 256);
static THD_FUNCTION(thdData, arg) {
```

```
    (void)arg;

    uint8_t i=0;

    s_ax = 0;
    s_ay = 0;
    s_az = 0;
    s_v_adc = 0;
    s_mag = 0;

    chRegSetThreadName("Data Process");
    while (true) {
      d_mpu_i2cReadData(0x3B, 14);

      adc01 = adc0;
      v_adc = ADC_SCALE * abs(adc01-adc00);
      adc00 = adc01;

      s_ax = s_ax + ax;
      s_ay = s_ay + ay;
      s_az = s_az + az;
      s_v_adc = s_v_adc + v_adc;
      s_mag = s_mag + mag;

      i++;

      if(i==100){
        m_ax = s_ax/100;
        m_ay = s_ay/100;
        m_az = s_az/100;
        m_v_adc = s_v_adc/100;
        m_mag = s_mag/100;

        fuz_res = d_fuzzy(m_mag,m_v_adc);
        if(fuz_res==1){
          fall_stt = 1;
        }
//        chprintf((BaseSequentialStream
*)&SD1,"%5.2f:%5.2f:%5.2f:%5.2f:%5.2f:%1i\r\n",m_v_adc,m_ax,m_
ay,m_az,m_mag,fuz_res);
```

```
            i = 0;
            s_ax = 0;
            s_ay = 0;
            s_az = 0;
            s_v_adc = 0;
            s_mag = 0;
        }

        chThdSleepMilliseconds(1);
    }
}

/**
 * @brief   Shell handler variable.
 */
static thread_t *shelltp = NULL;

#if SERVER_NOTIF
/**
 * @brief   Server request notification to SD1.
 */
static void request_notif(u_int8_t chan){
    chprintf((BaseSequentialStream
*)&SD1,"%i:%5.2f:%5.2f:%5.2f:%5.2f:%5.2f:%1i:%1i\r\n",chan,m_v_
adc,m_ax,m_ay,m_az,m_mag,fuz_res,fall_stt);
}
#endif

/**
 * @brief   Text sending function.
 */
static void text_http(void){
    chThdSleepMilliseconds(100);
    chprintf((BaseSequentialStream
*)&SD2,"%5.2f:%5.2f:%5.2f:%5.2f:%5.2f:%1i:%1i\r\n",m_v_adc,m_a
x,m_ay,m_az,m_mag,fuz_res,fall_stt);
    chThdSleepMilliseconds(100);
    chprintf((BaseSequentialStream *)&SD2,"\r\n");
    chThdSleepMilliseconds(100);
```

```
}

/**
 * @brief   Channel 0 handler function.
 */
static void cmd_send0(BaseSequentialStream *chp, int argc, char
*argv[]) {

  (void)argv;
  (void)argc;
  (void)chp;

#if SERVER_NOTIF
    request_notif(0);
#endif
  chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=0,34\r\n");
  text_http();
  chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=0\r\n");
}

/**
 * @brief   Channel 1 handler function.
 */
static void cmd_send1(BaseSequentialStream *chp, int argc, char
*argv[]) {

    (void)argv;
    (void)argc;
    (void)chp;

#if SERVER_NOTIF
    request_notif(1);
#endif
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=1,34\r\n");
    text_http();
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=1\r\n");
}

/**
 * @brief   Channel 2 handler function.
```

```
 */
static void cmd_send2(BaseSequentialStream *chp, int argc, char
*argv[]) {

   (void)argv;
   (void)argc;
   (void)chp;

#if SERVER_NOTIF
   request_notif(2);
#endif
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=2,34\r\n");
   text_http();
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=2\r\n");
}

/**
 * @brief   Channel 3 handler function.
 */
static void cmd_send3(BaseSequentialStream *chp, int argc, char
*argv[]) {

   (void)argv;
   (void)argc;
   (void)chp;

#if SERVER_NOTIF
   request_notif(3);
#endif
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=3,34\r\n");
   text_http();
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=3\r\n");
}

/**
 * @brief   Channel 4 handler function.
 */
static void cmd_send4(BaseSequentialStream *chp, int argc, char
*argv[]) {
```

```
    (void)argv;
    (void)argc;
    (void)chp;

#if SERVER_NOTIF
    request_notif(4);
#endif
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=4,34\r\n");
    text_http();
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=4\r\n");
}

/**
 * @brief   Channel 5 handler function.
 */
static void cmd_send5(BaseSequentialStream *chp, int argc, char
*argv[]) {

    (void)argv;
    (void)argc;
    (void)chp;

#if SERVER_NOTIF
    request_notif(5);
#endif
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=5,34\r\n");
    text_http();
    chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=5\r\n");
}

/**
 * @brief   Channel 6 handler function.
 */
static void cmd_send6(BaseSequentialStream *chp, int argc, char
*argv[]) {

    (void)argv;
    (void)argc;
    (void)chp;
```

```
#if SERVER_NOTIF
   request_notif(6);
#endif
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=6,34\r\n");
   text_http();
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=6\r\n");
}

/**
 * @brief   Channel 7 handler function.
 */
static void cmd_send7(BaseSequentialStream *chp, int argc, char
*argv[]) {

   (void)argv;
   (void)argc;
   (void)chp;

#if SERVER_NOTIF
   request_notif(7);
#endif
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPSEND=7,34\r\n");
   text_http();
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPCLOSE=7\r\n");
}

/**
 * @brief   Shell commands enumeration.
 */
static const ShellCommand commands[] = {
   {"0,CONNECT", cmd_send0},
   {"1,CONNECT", cmd_send1},
   {"2,CONNECT", cmd_send2},
   {"3,CONNECT", cmd_send3},
   {"4,CONNECT", cmd_send4},
   {"5,CONNECT", cmd_send5},
   {"6,CONNECT", cmd_send6},
   {"7,CONNECT", cmd_send7},
   {NULL, NULL}
};
```

```
/**
 * @brief   Shell configuration variable.
 */
static const ShellConfig shell_cfg = {
   (BaseSequentialStream *)&SD2,
   commands
};

/**
 * @brief   Web server start function.
 */
void d_web_start(void){

palSetPadMode(GPIOA,2,PAL_MODE_STM32_ALTERNATE_PUSH
PULL); //TX
   palSetPadMode(GPIOA,3,PAL_MODE_INPUT); //RX
   sdStart(&SD2,NULL);

   chThdSleepMilliseconds(100);
   d_web_setup();

   shellInit();
}

/**
 * @brief   Web server shell function.
 */
void d_web_term(void){
   if (!shelltp)
     shelltp = shellCreate(&shell_cfg, SHELL_WA_SIZE,
NORMALPRIO);
   else if (chThdTerminatedX(shelltp)) {
     chThdRelease(shelltp);
     shelltp = NULL;
   }
}

/**
 * @brief   Web server setup function.
```

```
 */
void d_web_setup(void){
   chprintf((BaseSequentialStream *)&SD2,"AT+RST\r\n");
   chThdSleepMilliseconds(1000);
   chprintf((BaseSequentialStream *)&SD2,"ATE0\r\n");
   chThdSleepMilliseconds(100);
   chprintf((BaseSequentialStream *)&SD2,"AT+CWMODE=2\r\n");
   chThdSleepMilliseconds(100);
   chprintf((BaseSequentialStream *)&SD2,"AT+CIPMUX=1\r\n");
   chThdSleepMilliseconds(100);
   chprintf((BaseSequentialStream
*)&SD2,"AT+CIPSERVER=1,80\r\n");
   chThdSleepMilliseconds(100);
}

void d_web_data(void){
   chThdCreateStatic(waData, sizeof(waData), NORMALPRIO,
thdData, NULL);
}
/** @} */
```

LISTING KODE MATLAB

```
clc;
clear;
close all;
%--------------------------------------------------------------------------
% Optimasi Fuzzy MPPT menggunakan Linear Decreasing Weight Particle
Swarm Optmization
% case: Optimasi 5 membership function (2 trapesium & 3 segitiga)
%--------------------------------------------------------------------------

% step1:Menentukan parameter-parameter PSO
% jumlah partikel(n),iterasi maks(imax),inersia awal(wmax),inersia akhir
% (wmin),konstanta pemberat(c1),konstanta pemberat(c2)
imax=10;      %iterasi maks(imax)
n=10;         %jumlah partikel
wmax=0.6;     %inersia awal(wmax)
wmin=0;       %inersia akhir(wmin)
c1=0.2;       %konsatanta pemberat
```

```
c2=0.1;          %konstanta pemberat

for i=1
   % step2:
% Inisialisasi batas fungsi keanggotaan yang sudah ditentukan sebelumnya
% yang akan digunakan sebagai gbest awal

% NB MF : a=batas kiri=(tetap); b=batas tengah(berubah); c=batas
kanan=gbest1
% negative big membership function (trapesium)
a(:,i)=0;
b(:,i)=5;
c(:,i)=8;
% NS MF : d=batas kiri=gbest2;  e=batas tengah(tetap); f=batas kanan=gbest3
% negative small - function (segitiga)
d(:,i)=5;
e(:,i)=8;
f(:,i)=10;

% PB MF : n=batas kiri=gbest8;  o=batas tengah(tetap); p=batas kanan=(tetap)
% positive big membership function (trapesium)
o(:,i)=7;
p(:,i)=12;
q(:,i)=30;

% step3:
% Inisialisasi posisi awal (x) secara acak untuk tiap dimensi dan juga
% kecepatan awal (v=0) untuk tiap partikel pada tiap dimensi
% NB:
x1(:,i)=(c(:,i)-(rand(n,1)*(c(:,i)-b(:,i)))); %constraint: b<=x1<=c
v1(:,i)=zeros(n,1);
% NS:
x2(:,i)=(e(:,i)-(rand(n,1)*(e(:,i)-d(:,i)))); %constraint: d<=x2<=e
v2(:,i)=zeros(n,1);
x3(:,i)=(f(:,i)-(rand(n,1)*(f(:,i)-e(:,i)))); %constraint: e<=x3<=f
v3(:,i)=zeros(n,1);
% PB:
x8(:,i)=(p(:,i)-(rand(n,1)*(p(:,i)-o(:,i)))); %constraint: o<=x8<=p
v8(:,i)=zeros(n,1);



% step4:
% Menghitung MSE[mean square error] yang dihasilkan oleh setiap partikel
```

```
% (per baris)yaitu dengan menghitung nilai "myu" [derajat keanggotaan] tiap
dimensi
myu1(:,i)=1;                    % myu1=derajat keanggotaan PB batas kiri (a)
(tetap)
myu2(:,i)=1;                    % myu2=derajat keanggotaan PB batas tengah
(b) (tetap)
myu3(:,i)=(c(:,i)-x1(:,i))/(c(:,i)-b(:,i)); % myu3=derajat keanggotaan PB batas
kanan (c)
y1=1;                          % y1=derajat keanggotaan ideal PB batas kiri (a)
adalah 1 (tetap)
y2=1;                          % y2=derajat keanggotaan ideal PB batas tengah
(b) adalah 1 (tetap)
y3=0;                          % y3=derajat keanggotaan ideal PB batas kanan
(c) adalah 0

myu4(:,i)=(x2(:,i)-d(:,i))/(e(:,i)-d(:,i)); % myu4=derajat keanggotaan PS batas
kiri (d)
myu5(:,i)=1;                    % myu5=derajat keanggotaan PS batas tengah
(e) (tetap)
myu6(:,i)=(f(:,i)-x3(:,i))/(f(:,i)-e(:,i)); % myu6=derajat keanggotaan PS batas
kanan (f)
y4=0;                          % y4=derajat keanggotaan ideal PS batas kiri (d)
adalah 0
y5=1;                          % y5=derajat keanggotaan ideal PS batas tengah
(e) adalah 1 (tetap)
y6=0;                          % y6=derajat keanggotaan ideal PS batas kanan
(f) adalah 0

myu13(:,i)=(x8(:,i)-o(:,i))/(p(:,i)-o(:,i)); % myu13=derajat keanggotaan PB batas
kiri (o)
myu14(:,i)=1;                     % myu14=derajat keanggotaan PB batas
tengah (p) (tetap)
myu15(:,i)=1;                     % myu15=derajat keanggotaan PB batas
kanan (q) (tetap)
y13=0;                         % y13=derajat keanggotaan ideal PB batas kiri
(o) adalah 0
y14=1;                         % y14=derajat keanggotaan ideal PB batas
tengah (p) adalah 1 (tetap)
y15=1;                         % y15=derajat keanggotaan ideal PB batas kanan
(q) adalah 1 (tetap)

mse(:,i)=(((y1-myu1(:,i)).^2)+((y2-myu2(:,i)).^2)+((y3-myu3(:,i)).^2)+((y4-
myu4(:,i)).^2)+((y5-myu5(:,i)).^2)+...
```

```
   ((y6-myu6(:,i)).^2)+((y13-myu13(:,i)).^2)+((y14-myu14(:,i)).^2)+((y15-
myu15(:,i)).^2))/15;
% step5:
% Menghitung Inersia, fungsi inersia adalah untuk mengurangi kecepatan
% perhitungan nilai minimum fungsi obyektif pada iterasi berikutnya supaya
% lebih teliti sehingga tidak ada nilai yang terlewatkan
w(:,i)=wmin-((wmax-wmin)/imax)*(i-1);

% step6:Membangkitkan nilai parameter r1 dan r2 secara acak
r1(:,i)=rand();
r2(:,i)=rand();

% step7:Menghitung kecepatan terbaru partikel (update kecepatan)
pbest1(:,i)=x1(:,i);
gbest1(:,i)=c(:,i);
v1(:,i+1)=(w(:,i)*v1(:,i))+(c1*r1(:,i)*(pbest1(:,i)-
x1(:,i)))+(c2*r2(:,i)*(gbest1(:,i)-x1(:,i)));

pbest2(:,i)=x2(:,i);
gbest2(:,i)=d(:,i);
v2(:,i+1)=(w(:,i)*v2(:,i))+(c1*r1(:,i)*(pbest2(:,i)-
x2(:,i)))+(c2*r2(:,i)*(gbest2(:,i)-x2(:,i)));
pbest3(:,i)=x3(:,i);
gbest3(:,i)=f(:,i);
v3(:,i+1)=(w(:,i)*v3(:,i))+(c1*r1(:,i)*(pbest3(:,i)-
x3(:,i)))+(c2*r2(:,i)*(gbest3(:,i)-x3(:,i)));

pbest8(:,i)=x8(:,i);
gbest8(:,i)=o(:,i);
v8(:,i+1)=(w(:,i)*v8(:,i))+(c1*r1(:,i)*(pbest8(:,i)-
x8(:,i)))+(c2*r2(:,i)*(gbest8(:,i)-x8(:,i)));

% step8:Menghitung posisi terbaru partikel (update posisi)
x1(:,i+1)=x1(:,i)+v1(:,i+1);
x2(:,i+1)=x2(:,i)+v2(:,i+1);
x3(:,i+1)=x3(:,i)+v3(:,i+1);
x8(:,i+1)=x8(:,i)+v8(:,i+1);

% step9:Mencari batas baru a dan c berdasarkan nilai minimum MSE;
[value(:,i),index(:,i)]=min(mse(:,i));
a(:,i+1)=a(:,i);
b(:,i+1)=b(:,1);
c(:,i+1)=pbest1(index(:,i));
```

```
d(:,i+1)=pbest2(index(:,i));
e(:,i+1)=e(:,1);
f(:,i+1)=pbest3(index(:,i));


o(:,i+1)=pbest8(index(:,i));
p(:,i+1)=p(:,1);
q(:,i+1)=q(:,1);


end

for i=2:imax
   myu1(:,i)=1;                    % myu1=derajat keanggotaan PB batas kiri
(a) (tetap)
myu2(:,i)=1;                    % myu2=derajat keanggotaan PB batas tengah
(b) (tetap)
myu3(:,i)=(c(:,i)-x1(:,i))/(c(:,i)-b(:,i)); % myu3=derajat keanggotaan PB batas
kanan (c)
y1=1;                           % y1=derajat keanggotaan ideal PB batas kiri (a)
adalah 1 (tetap)
y2=1;                           % y2=derajat keanggotaan ideal PB batas tengah
(b) adalah 1 (tetap)
y3=0;                           % y3=derajat keanggotaan ideal PB batas kanan
(c) adalah 0

myu4(:,i)=(x2(:,i)-d(:,i))/(e(:,i)-d(:,i)); % myu4=derajat keanggotaan PS batas
kiri (d)
myu5(:,i)=1;                    % myu5=derajat keanggotaan PS batas tengah
(e) (tetap)
myu6(:,i)=(f(:,i)-x3(:,i))/(f(:,i)-e(:,i)); % myu6=derajat keanggotaan PS batas
kanan (f)
y4=0;                           % y4=derajat keanggotaan ideal PS batas kiri (d)
adalah 0
y5=1;                           % y5=derajat keanggotaan ideal PS batas tengah
(e) adalah 1 (tetap)
y6=0;                           % y6=derajat keanggotaan ideal PS batas kanan
(f) adalah 0

myu13(:,i)=(x8(:,i)-o(:,i))/(p(:,i)-o(:,i)); % myu13=derajat keanggotaan PB batas
kiri (o)
myu14(:,i)=1;                   % myu14=derajat keanggotaan PB batas
tengah (p) (tetap)
```

```
myu15(:,i)=1;                    % myu15=derajat keanggotaan PB batas
kanan (q) (tetap)
y13=0;                           % y13=derajat keanggotaan ideal PB batas kiri
(o) adalah 0
y14=1;                           % y14=derajat keanggotaan ideal PB batas
tengah (p) adalah 1 (tetap)
y15=1;                           % y15=derajat keanggotaan ideal PB batas kanan
(q) adalah 1 (tetap)

mse(:,i)=(((y1-myu1(:,i)).^2)+((y2-myu2(:,i)).^2)+((y3-myu3(:,i)).^2)+((y4-
myu4(:,i)).^2)+((y5-myu5(:,i)).^2)+...
   ((y6-myu6(:,i)).^2)+((y13-myu13(:,i)).^2)+((y14-myu14(:,i)).^2)+((y15-
myu15(:,i)).^2))/15;

w(:,i)=wmax-(((wmax-wmin)/imax)*(imax-1)); %MODIFIED W dikondisikan
di sini W = (wmax-wmin)*((imax-i)/imax)+ wmax
r1(:,i)=rand();
r2(:,i)=rand();

% cek pbest
pbest1(:,i)=((mse(:,i)<mse(:,i-1)).*x1(:,i))+((mse(:,i-1)<mse(:,i)).*x1(:,i-1));
pbest2(:,i)=((mse(:,i)<mse(:,i-1)).*x2(:,i))+((mse(:,i-1)<mse(:,i)).*x2(:,i-1));
pbest3(:,i)=((mse(:,i)<mse(:,i-1)).*x3(:,i))+((mse(:,i-1)<mse(:,i)).*x3(:,i-1));
pbest8(:,i)=((mse(:,i)<mse(:,i-1)).*x8(:,i))+((mse(:,i-1)<mse(:,i)).*x8(:,i-1));

gbest1(:,i)=c(:,i);
gbest2(:,i)=d(:,i);
gbest3(:,i)=f(:,i);
gbest8(:,i)=o(:,i);

v1(:,i+1)=(w(:,i)*v1(:,i))+(c1*r1(:,i)*(pbest1(:,i)-
x1(:,i)))+(c2*r2(:,i)*(gbest1(:,i)-x1(:,i)));
v2(:,i+1)=(w(:,i)*v2(:,i))+(c1*r1(:,i)*(pbest2(:,i)-
x2(:,i)))+(c2*r2(:,i)*(gbest2(:,i)-x2(:,i)));
v3(:,i+1)=(w(:,i)*v3(:,i))+(c1*r1(:,i)*(pbest3(:,i)-
x3(:,i)))+(c2*r2(:,i)*(gbest3(:,i)-x3(:,i)));
v8(:,i+1)=(w(:,i)*v8(:,i))+(c1*r1(:,i)*(pbest8(:,i)-
x8(:,i)))+(c2*r2(:,i)*(gbest8(:,i)-x8(:,i)));

x1(:,i+1)=x1(:,i)+v1(:,i+1);
x2(:,i+1)=x2(:,i)+v2(:,i+1);
x3(:,i+1)=x3(:,i)+v3(:,i+1);
x8(:,i+1)=x8(:,i)+v8(:,i+1);
```

```
[value(:,i),index(:,i)]=min(mse(:,i));
a(:,i+1)=a(:,i);
b(:,i+1)=b(:,1);
c(:,i+1)=pbest1(index(:,i));

d(:,i+1)=pbest2(index(:,i));
e(:,i+1)=e(:,1);
f(:,i+1)=pbest3(index(:,i));


o(:,i+1)=pbest8(index(:,i));
p(:,i+1)=p(:,1);
q(:,i+1)=q(:,1);

z(:,i)=min (mse(:,i));
end
```