



**RANCANG BANGUN ALAT PENGERING IKAN OTOMATIS DENGAN
PENGENDALIAN SENSOR GAS MENGGUNAKAN KONTROL *FUZZY***

PROPOSAL SKRIPSI

Oleh

Aden Tia Zahrotur Riadh

NIM 141910201099

PROGRAM STUDI STRATA 1 TEKNIK ELEKTRO

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS JEMBER

2018



**RANCANG BANGUN ALAT PENGERING IKAN OTOMATIS DENGAN
PENGENDALIAN SENSOR GAS MENGGUNAKAN KONTROL *FUZZY***

Skripsi

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Elektro (S1)
dan mencapai gelar Sarjana Teknik

Oleh

Aden Tia Zahrotur Riadh

NIM 141910201099

**JURUSAN TEKNIK ELEKTRO STRATA 1
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2018**

PERSEMBAHAN

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan kasih sayangnya, sehingga saya dapat menyelesaikan penelitian ini. Akhirnya, saya persembahkan skripsi ini kepada:

1. Allah SWT yang Maha Pengasih lagi Maha Penyayang.
2. Nabi besar Muhammad SAW, yang menjadi suri tauladan bagi seluruh umat.
3. Ke-dua orang tua saya yaitu Bapak Agus Sutikno dan Ibu Ummul Khoiroh serta kakak, adik, dan keponakan tercinta Gandhi Tia Khoirimal, Rischia Diana Lestari, Hibban Tia Zahro Mujahid dan M. Arsen Al-Dimitri.
4. Keluarga PB SQUAD yaitu Tante Dr. NNP Martini G, SE, MM, Wahyu Eka Febrialiani, Shifa Adelia Septiduana, Richard Juliano, Ahmad Wahyu Tri Utama dan Putra Suci Bachtiar Syech Akbar. Terimakasih atas dukungan semangatnya.
5. Dulur KETEK UJ 2014 dan ELKA14 yang telah memberikan cerita indah dimasa-masa kuliah.
6. Ex-pengurus Robotika dan anggota UKM Robotika. Bos Wawan Oi CB, Fabian, Lukman, Faiq, Galih, Ghosal, Muti, Hanifatus, Faiqotul, Neni dan kawan-kawan anggota robotika unej lainnya, terimakasih atas kebersamaan, kepedulian, ilmu yang telah diberikan serta prestasi yang membanggakan.
7. Seluruh asisten Laboratorium Teknik Elektro Universitas Jember khususnya Laboratorium Elektronika dan Terapan. Terima kasih telah menjadi tempat berbagi ilmu.
8. Semua Guru-guruku mulai dari TK sampai kuliah, terima kasih telah memberikan ilmu dan mendidik dengan penuh kesabaran.
9. Serta seluruh rekan-rekan yang penulis kenal dan rekan-rekan yang membaca skripsi ini. Terima kasih untuk segalanya.

MOTTO

**Sebuah tantangan akan selalu menjadi beban, jika itu hanya dipikirkan.
Sebuah cita-cita juga adalah beban, jika itu hanya angan-angan.
(Putu Sutrisna)**

**Sabar bukan tentang berapa lama kau bisa menunggu
Melainkan tentang bagaimana perilakumu saat menunggu
(Anonim)**

**Barang siapa keluar rumah untuk menuntut ilmu maka ia dalam jihad fisabilah
hingga kembali (HR.Bukhari)**

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Aden Tia Zahrotur Riadh

NIM : 141910201099

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul ” Rancang Bangun Alat Pengering Ikan Otomatis Dengan Pengendalian Sensor Gas Menggunakan Kontrol *Fuzzy*” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun dan bukan karya jiplakan. Saya bertanggung jawab penuh atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, 4 Mei 2018

Yang menyatakan,

Aden Tia Zahrotur Riadh

NIM 141910201099

SKRIPSI

**RANCANG BANGUN ALAT PENDINGIN IKAN OTOMATIS DENGAN
PENGENDALIAN SENSOR GAS MENGGUNAKAN KONTROL *FUZZY***

Oleh

Aden Tia Zahrotur Riadh

NIM 141910201099

Pembimbing :

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

Dosen Pembimbing Anggota : Sumardi S.T., M.T.

PENGESAHAN

Skripsi berjudul “**Rancang Bangun Alat Pengering Ikan Otomatis Dengan Pengendalian Sensor Gas Menggunakan Kontrol *Fuzzy***” karya Aden Tia Zahrotur Riadh telah diuji dan disahkan pada :

Hari : Senin

Tanggal : 23 Juli 2018

Tempat : Fakultas Teknik Universitas Jember

Tim penguji,

Ketua,

Anggota I,

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

Sumardi S.T., M.T.
NIP 196701131998021001

Anggota II,

Anggota III,

Widya Cahyadi S.T., M.T.
NIP 198511102014041001

Guido Dias Kalandro S.ST., M.Eng.
NRP 760015734

Mengesahkan
Dekan,

Dr. Ir. Entin Hidayah, M.U.M.
NIP 196612151995032001

RINGKASAN

Rancang Bangun Alat Pengering Ikan Otomatis Dengan Pengendalian Sensor Gas Menggunakan Kontrol *Fuzzy*; Aden Tia Zahrotur Riadh; 141910201099; 2018; 98 halaman; Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Sumber daya alam di Indonesia sangatlah melimpah terutama disektor kelautan. Didalam laut pasti terdapat banyak ikan yang merupakan salah satu makanan favorit semua manusia. Selain memiliki kandungan protein yang tinggi, juga mengandung lemak yang lebih rendah daripada sumber protein hewani lainnya. Namun dalam sisi ekonomisnya ikan mudah busuk sehingga dapat merugikan nelayan. Dalam mengatasi hal tersebut, maka nelayan mempunyai cara alternatif supaya ikan tidak busuk dan terbuang sia-sia yaitu dengan cara dikeringkan.

Kebanyakan nelayan Indonesia dalam melakukan pengeringan ikan masih dengan cara tradisional. Padahal pengeringan ikan dengan cara tradisional ini kurang efektif dalam waktu pengeringan (tergantung oleh cuaca alam), memakan tempat yang luas, hasil pengeringan tidak merata, dan membutuhkan tenaga yang lebih.

Semakin pesatnya perkembangan teknologi di dunia membuat masyarakat harus berkembang, salah satunya yaitu bagaimana cara membuat alat yang bisa memudahkan pekerjaan dan mempercepat pekerjaan mereka. Pengeringan ikan dapat dibuat dengan cara meng-*oven* ikan tersebut mengganti tenaga matahari dengan pemanas. Pengeringan ikan dengan cara meng-*oven* adalah cara yang efektif dibandingkan melakukan pengeringan dengan cara tradisional. Hasil pengeringan ikan dengan cara ini akan lebih higienis, suhu pengeringan akan bisa diatur, waktu pengeringan lebih cepat, hemat tenaga dan hemat tempat pengeringan.

Alat pengering ikan ini didesain berbentuk balok kotak seperti lemari yang didalamnya terdapat 4 buah rak atau tempat untuk menaruh ikan yang akan dikeringkan. Alat pengering ikan ini dilengkapi dengan 2 buah pemanas yaitu *heater* dan kipas yang berguna untuk sirkulasi udara dalam alat pengering tersebut. Kipas didalam alat pengering ikan ini terbagi menjadi 2 ruangan yaitu ruangan pertama dibagian pemanas yang berguna untuk mengirim hawa panas dari ruang pemanas ke ruang ikan yang dikeringkan, sedangkan ruangan kipas yang kedua digunakan untuk mengeluarkan hawa panas tersebut keluar alat pengering ikan. Sensor gas MQ pada penelitian ini digunakan untuk mendeteksi aroma ikan yang dikeluarkan pada saat proses pengeringan. Sensor Gas MQ terdapat 3 buah yaitu MQ-5 (digunakan untuk

mendeteksi gas *methane*), MQ-7 (digunakan untuk mendeteksi gas *carbon monoxide*) dan MQ-8 (digunakan untuk mendeteksi gas *hydrogen*). Sensor DHT-22 digunakan untuk mendeteksi suhu pada ruangan alat pengering ikan.

Alat pengering ikan ini bekerja dengan cara mendeteksi aroma ikan yang sedang dikeringkan dengan sensor MQ-5, MQ-7 dan MQ-8. Alat pengering ikan ini menggunakan kontrol fuzzy, *input* dari kontrol ini adalah sensor gas dan *output* dari kontrol ini adalah berupa suhu. Saat proses pengeringan berlangsung aroma ikan yang dikeringkan dideteksi oleh sensor gas dan kemudian dikontrol oleh fuzzy logic dan mendapatkan *output* berupa suhu yang mengaktifkan kipas dan *heater* sesuai dengan keputusan dari pengontrol tersebut.

Dari penelitian ini menggunakan metode *Fuzzy Logic* sangat cocok untuk pengambilan keputusan dan dari data hasil pengujian kontrol *Fuzzy* didapatkan selisih suhu terbesar yaitu 0,14 °C. Penelitian ini dalam mengeringkan ikan membutuhkan waktu ± 11 jam dan nilai pembacaan sensor gas semakin keadaan ikan kering maka nilai sensor gas (ppm) semakin kecil. Dalam penelitian ini ikan yang digunakan adalah jenis ikan belok dan ikan layang. Untuk penelitian ini masih perlu adanya pengembangan dari segi penambahan sensor ataupun penggunaan metode.

PRAKATA

Puji syukur kehadirat Allah SWT yang maha kuasa atas segalanya, karena dengan ridho, hidayah dan petunjukNya, penulis dapat menyelesaikan skripsi yang berjudul “Rancang Bangun Alat Pengering Ikan Otomatis Dengan Pengendalian Sensor Gas Menggunakan Kontro *Fuzzy*”. Selama penyusunan skripsi ini penulis mendapat bantuan berbagai pihak yang turut memberikan bantuan berupa motivasi, inspirasi, bimbingan, doa, fasilitas dan dukungan lainnya yang membantu memperlancar pengerjaan skripsi ini.

Untuk itu penulis mengucapkan terimakasih kepada.

1. Allah SWT yang telah melimpahkan rezeki, rahmat, hidayah dan karunia serta kasih sayang-Nya sehingga penulis mampu menyelesaikan skripsi ini.
2. Nabi besar Muhammad SAW, yang telah menjadi suri tauladan bagi seluruh umat.
3. Ibu Dr. Ir. Entin Hidayah, M.U.M., selaku Dekan Fakultas Teknik Universitas Jember
4. Bapak Dr. Bambang Sri Kaloko, S.T., M.T., Selaku Ketua Jurusan Teknik Elektro Universitas Jember;
5. Bapak Khairul Anam, S.T., M.T.,Ph.D dan Bapak Sumardi S.T., M.T. selaku dosen pembimbing yang telah membimbing untuk menyelesaikan tugas akhir ini;
6. Bapak Widya Cahyadi S.T., M.T. dan Bapak Guido Dias Kalandro S.ST., M.Eng. selaku dosen penguji yang sudah bersedia memberikan kritik dan saran untuk memperbaiki tugas akhir ini;
7. Keluarga besar KETEK UJ 2014, terimakasih telah memberikan pengalaman yang luar biasa.
8. Keluarga besar Civitas Akademia Jurusan Teknik Elektro Universitas Jember. Serta semua pihak yang tidak dapat disebutkan satu per satu, yang telah mendukung dalam penyelesaian skripsi ini.

Semoga skripsi ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya untuk disiplin ilmu teknik elektro. Kritik dan saran yang membangun diharapkan terus mengalir untuk lebih menyempurnakan skripsi ini dan dapat dikembangkan untuk penelitian selanjutnya;

Jember, 4 Mei 2018

Penulis



DAFTAR ISI

HALAMAN Sampul	i
HALAMAN Judul	ii
PERSEMBAHAN	iii
MOTTO	iv
PERNYATAAN.....	v
HALAMAN Pembimbing.....	vi
PENGESAHAN	vii
RINGKASAN	viii
PRAKATA	x
DAFTAR ISI.....	xii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xix
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Masalah	3
1.4 Batasan Maslah	3
BAB 2. TINJAUAN PUSTAKA	5
2.1 Penelitian Terkait	5
2.2 Arduino Mega	6
2.3 <i>Heating Element</i>	7
2.4 Sensor DHT-22	8
2.5 Kipas / Fan	9
2.6 Sensor MQ-8	10
2.7 Sensor MQ-5	11
2.8 Sensor MQ-7	12

2.9 Kontrol Fuzzy Logic	13
2.9.1 Teori Sistem Kendali <i>Fuzzy Logic</i>	13
2.9.2 Fungsi Kanggotaan	14
2.9.3 Oprerasi Himpunan Fuzzy.....	14
2.9.4 <i>Fuzzification</i>	15
2.9.5 Sistem Inferensi Fuzzy	15
2.9.6 Defuzzyfikasi	15
BAB 3. METODOLOGI PENELITIAN	16
3.1 Waktu dan Tempat	16
3.1.1 Tempat	16
3.1.2 Waktu	16
3.2 Metodologi Penelitian	16
3.3 Perancangan Perangkat Keras	19
3.3.1 Alat dan Bahan	19
3.3.2 Desain Alat	20
3.3.3 Desain Elektronik	22
3.3.3.1 Rangkaian Sensor MQ	22
3.3.3.2 Rangkaian Sensor DHT-22	22
3.3.3.3 Rangkaian <i>Driver Motor Ventilasi</i>	24
3.3.3.4 Rangkaian <i>Driver Heater</i>	25
3.4 Perancangan Sistem Fuzzy Logic	26
3.4.1 Proses Fuzzifikasi	26
3.5 Flowchart	32
3.6 Pengujian dan Pengumpulan Data	33
3.7 Metode Analisa Data	33
BAB 4. HASIL DAN PEMBAHASAN	34
4.1 Kalibrasi Sensor Suhu	35
4.2 Kalibrasi Sensor Gas	37
4.3 Pengujian Sensor Suhu.....	41

4.4 Pengujian Sensor Gas	42
4.4.1 Pengujian Sensor Gas MQ-5.....	43
4.4.2 Pengujian Sensor Gas MQ-7.....	45
4.4.3 Pengujian Sensor Gas MQ-8.....	46
4.5 Sistem Kontrol <i>Fuzzy Logic</i>	47
4.5.1 Perhitungan <i>Fuzzy Logic</i>	47
4.5.1.1 Proses Fuzzifikasi.....	47
4.5.1.2 <i>Fuzzy Inference</i>	50
4.5.1.3 Defuzzifikasi	51
4.5.2 Pengujian <i>Fuzzy Logic</i>	51
4.6 Pengujian Keseluruhan	52
4.6.1 Pengujian Kapasitas Alat Pegering Ikan.....	53
4.6.1.1 Pengujian Kapasitas Penuh	53
4.6.1.2 Pengujian Kapasitas Setengah.....	55
4.6.2 Pengujian Tingkat Kekeringan Ikan	58
4.6.2.1 Pengujian Tingkat Kekeringan Ikan Dengan Kapasitas Penuh	58
4.6.2.2 Pengujian Tingkat Kekeringan Ikan Dengan Kapasitas Setengah.....	61
4.6.2.3 Pengujian Tingkat Kekeringan Ikan Dengan <i>Set Point</i> Suhu 70 °C	62
4.6.3 Pengujian Pengeringan Ikan Dengan Cara Konvensional .	64
4.6.3.1 Pengujian Waktu Pengeringan Ikan	64
4.6.3.2 Pengujian Tingkat Kekeringan Ikan.....	65
BAB 5. KESIMPULAN DAN SARAN	69
5.1 Kesimpulan	69
5.2 Saran	69
DAFTAR PUSTAKA	71
LAMPIRAN	73

DAFTAR GAMABAR

Gambar 2.1 Bentuk Fisik Arduino Mega 2560	7
Gambar 2.2 Bentuk-Bentuk <i>Tubular Heater</i>	7
Gambar 2.3 <i>Timing Diagram Protocol</i> Komunikasi Serial	9
Gambar 2.4 Sensor Gas DHT-22	9
Gambar 2.5 <i>Fan / Kipas</i>	10
Gambar 2.6 Sensor Gas MQ-8	11
Gambar 2.7 Sensor Gas MQ-5	12
Gambar 2.8 Sensor Gas MQ-7	13
Gambar 2.9 <i>Defuzzyfication</i>	15
Gambar 3.1 <i>Flowchart</i> Penelitian	16
Gambar 3.2 Desain Alat Pengering Ikan	20
Gambar 3.3 Blok Diagram Perangkat Keras	20
Gambar 3.4 Alat Pengering Ikan	21
Gambar 3.5 Rangkaian Sensor MQ	22
Gambar 3.6 Rangkaian Sensor DHT-22	23
Gambar 3.7 Rangkain <i>Driver</i> Motor	24
Gambar 3.8 Rangkain <i>Dimmer</i>	25
Gambar 3.9 <i>Membership Function Input MQ-5</i>	27
Gambar 3.10 <i>Membership Function Input MQ-7</i>	27
Gambar 3.11 <i>Membership Function Input MQ-8</i>	28
Gambar 3.12 <i>Membership Function Output</i>	29
Gambar 3.13 <i>Flowchart</i> Kendali <i>Fuzzy</i>	32
Gambar 4.1 Grafik Kalibrasi Sensor DHT-22	36
Gambar 4.2 Grafik Kalibrasi Sensor MQ-5	37
Gambar 4.3 Grafik Kalibrasi Sensor MQ-5	38
Gambar 4.4 Grafik Kalibrasi Sensor MQ-7	39

Gambar 4.5 Grafik Kalibrasi Sensor MQ-7	40
Gambar 4.6 Grafik Pengujian Sensor DHT-22	42
Gambar 4.7 Grafik Pengujian Sensor MQ-5.....	44
Gambar 4.8 Grafik Pengujian Sensor MQ-7.....	45
Gambar 4.9 Grafik Pengujian Sensor MQ-8.....	47
Gambar 4.10 Penempatan Ikan Dengan Kapasitas Penuh	53
Gambar 4.11 Grafik Pengujian Pengeringan Ikan Layang Dengan Kapasitas Penuh	54
Gambar 4.12 Grafik Pengujian Pengeringan Ikan Belok Dengan Kapasitas Penuh	55
Gambar 4.13 Penempatan Ikan Layang Dengan Kapasitas Setengah.....	56
Gambar 4.14 Grafik Pengujian Pengeringan Ikan Layang Dengan Kapasitas Setengah	57
Gambar 4.15 Grafik Pengujian Pengeringan Ikan Belok Dengan Kapasitas Setengah	58
Gambar 4.16 Ikan Hasil Pengeringan Menggunakan Alat Pengering Ikan	64
Gambar 4.17 Pengeringan Ikan Secara Konvensional.....	64
Gambar 4.18 Ikan Hasil Dari Pengeringan Secara Konvensional	66

DAFTAR TABEL

Tabel 2.1 Spesifikasi Arduino Mega 2560	6
Tabel 2.2 Spesifikasi Sensor DHT-22	8
Tabel 2.3 Spesifikasi <i>Fan</i> / Kipas	10
Tabel 2.4 Spesifikasi Sensor MQ-8	11
Tabel 2.5 Spesifikasi Sensor MQ-5	12
Tabel 2.6 Spesifikasi Sensor MQ-7	13
Tabel 3.1 Daftar Alat dan Bahan yang Digunakan	19
Tabel 3.2 <i>Rule Based Fuzzy Logic</i>	31
Tabel 4.1 Kalibrasi Sensor DHT-22	35
Tabel 4.2 Kalibrasi <i>Input</i> MQ-5.....	37
Tabel 4.3 Kalibrasi <i>Input</i> MQ-7.....	39
Tabel 4.4 Perbandingan Sensor Suhu DHT-22 Dengan Digital Thermometer .	42
Tabel 4.5 Perbandingan Sensor MQ-5 Dengan Metane-meter	43
Tabel 4.6 Perbandingan Sensor MQ-7 Dengan Karbon Monoksida-meter	45
Tabel 4.7 Perbandingan Sensor MQ-8	46
Tabel 4.8 Pengujian Kontrol Fuzzy	52
Tabel 4.9 Hasil Pengujian Pengeringan Ikan Layang Dengan Kapasitas Penuh	54
Tabel 4.10 Hasil Pengujian Pengeringan Ikan Belok Dengan Kapasitas Penuh	55
Tabel 4.11 Hasil Pengujian Pengeringan Ikan Layang Dengan Kapasitas Setengah.....	56
Tabel 4.12 Hasil Pengujian Pengeringan Ikan Belok Dengan Kapasitas Setengah	57
Tabel 4.13 Tingkat Kekeringan Ikan Layang Dengan Kapasitas Penuh	60
Tabel 4.14 Tingkat Kekeringan Ikan Layang Dengan Kapasitas Setengah	62
Tabel 4.15 tingkat Kekeringan Ikan Dengan <i>Set Point</i> Suhu 70 °C.....	63

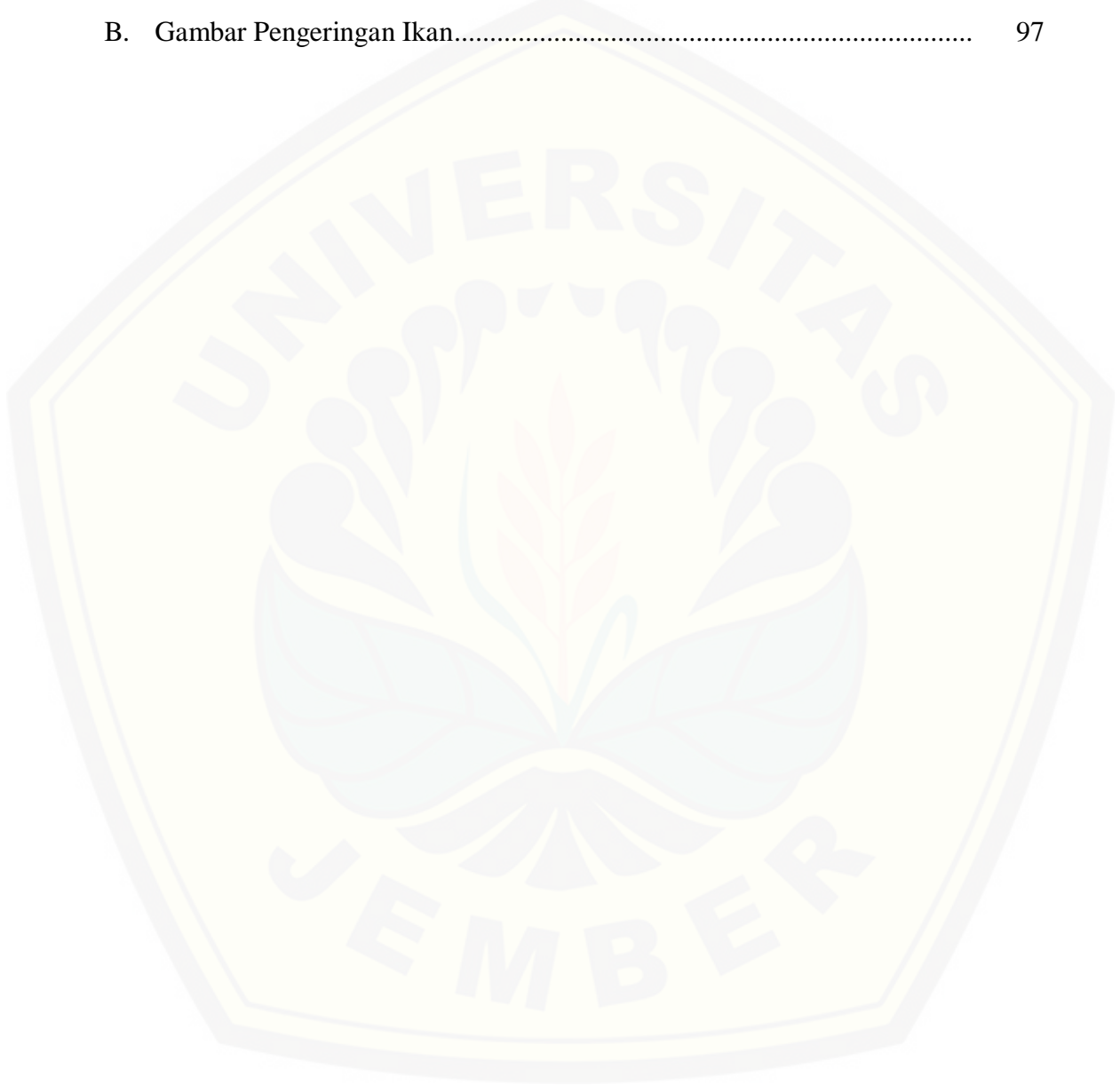
Tabel 4.16 Hasil Pengujian Pengeringan Ikan Secara Konvensional 65

Tabel 4.17 Data Hasil Tingkat Kekeringan Ikan 67



DAFTAR LAMPIRAN

A. <i>Listing</i> Program Arduino Mega 2560	73
B. Gambar Pengeringan Ikan.....	97



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Sumber daya alam di Indonesia sangatlah melimpah terutama disektor kelautan. Di dalam laut pasti terdapat banyak ikan yang merupakan salah satu makanan favorit masyarakat Indonesia maupun luar negeri. Selain memiliki kandungan protein yang tinggi, juga mengandung lemak yang lebih rendah daripada sumber protein hewani lainnya. Meskipun ikan berkhasiat bagi tubuh namun disisi ekonomisnya, ikan mudah busuk sehingga dapat merugikan nelayan jika ikan tidak habis terjual. Apalagi ikan yang dipanen tergantung musim, terkadang panen melimpah dan terkadang malah sebaliknya. Pada saat musim panen ikan melimpah, nelayan malah merugi dikarenakan sisa ikan yang tidak terjual menjadi busuk dan tidak bisa dikonsumsi lagi. Oleh sebab itu, untuk menunjang kesejahteraan para nelayan dilakukan proses pengawetan ikan. Proses pengawetan ikan dapat dilakukan dengan cara pengeringan, penggaraman, pemindangan, pengasapan, dan pendinginan.

Salah satu pengawetan yang sering dilakukan adalah dengan cara pengeringan. Cara melakukan pengeringan yaitu dengan penjemuran atau pemanasan ikan di bawah terik matahari. Cara tradisional ini sangat tidak efisien karena tergantung dengan keadaan cuaca, dan juga memerlukan waktu dan tenaga yang lebih untuk melakukan pengawetan ini.

Semakin pesatnya perkembangan teknologi di dunia membuat masyarakat harus berkembang salah satunya dengan bagaimana cara membuat alat yang bisa memudahkan pekerjaan dan mempercepat pekerjaan mereka. Pengeringan ikan dapat dibuat dengan cara meng-*oven* ikan tersebut dengan mengganti tenaga matahari untuk yang tradisional dengan pemanas. Ada beberapa penelitian yang menyangkut tentang sistem pengering yang memudahkan nelayan dalam melakukan pekerjaannya, salah satunya adalah Amalia (2015). Dia membuat penelitian tentang “Lemari Pengering dan Sterilisasi Pakaian Bayi Otomatis”. Alat ini digunakan untuk pengering baju bayi

otomatis dengan mengontrol suhu didalam ruangan lemari tersebut dengan sensor SHT 11 (Amalia, 2015). Kemudian terdapat penelitian dari Apriliyadi (2014). Dia membuat penelitian tentang “Perancangan Mesin Pengering Ikan Dengan Menggunakan Sistem Kendali Logika *Fuzzy*”. Alat ini digunakan untuk mengeringkan ikan dengan mengontrol suhu pada ruangan alat pengering dengan menggunakan sistem kendali *Fuzzy*. Alat ini didukung oleh sensor SHT-11 yang digunakan untuk mengetahui nilai suhu dan kelembapan pada ruangan alat pengering, dan juga didukung dengan sensor *Load Cell* (sensor berat) yang digunakan untuk mengetahui berat ikan selama proses pengeringan. Pada alat ini, proses pengeringan akan berhenti ketika kadar air pada ikan berkurang sebanyak 75% - 85% (Apriliyadi, 2014). Penelitian yang lain yaitu tentang “Model Simulasi Pengendalian Suhu Udara Pada Mesin Pengering Cabe Dengan Kontrol Logika *Fuzzy*”. Penelitian ini, mengambil data dengan simulasi mengontrol suhu udara dengan menggunakan logika *fuzzy*. Penelitian ini bertujuan untuk menurunkan kadar air pada cabe dengan mengatur *setpoint* suhu dan waktu yang telah ditempuh pada saat melakukan pengeringan (Dwi Argo, 2012). Pada penelitian Sofyan (2016) yang membuat tentang “Manipulasi Suhu Pada Pengeringan Ikan Teri Tenaga Surya Menggunakan Mikrokontroler ATmega 2560”. Pada penelitian ini, sumber pemanas berasal dari sinar matahari dan dikontrol suhunya dengan memanipulasi menggunakan cara memperpanjang laju lintasan udara. Untuk mengatur *setpoint* nilai suhu menggunakan mikrokontroler ATmega2560 dan suhu dijaga agar stabil menggunakan pemanas DC 12 Volt dan menggunakan sensor LM35 (Sofyan, 2016). Pada penelitian Darjat (2008) yang membuat tentang “Sistem Pengendalian Suhu Dan Kelembaban Pada Mesin Pengering Kertas”. Dia memaparkan bahwa dalam penelitiannya sensor suhu apabila diberi gangguan sesaat (dengan membuka pintu lemari pengering) akan menurunkan nilai suhunya dan untuk memulihkannya kembali membutuhkan waktu yang lama (Darjat, 2008).

Berdasarkan latar belakang yang disajikan diatas, rata-rata alat pengering tersebut menggunakan sensor suhu, sedangkan sensor suhu tersebut terdapat

kelemahannya. Oleh karena itu, perlu dikembangkan dari beberapa penelitian-penelitian yang terpapar diatas yaitu dengan menambahkan sensor bau yang digunakan untuk mendeteksi aroma objek pada saat matang. Dimana alat ini dikembangkan ke dalam bidang kelautan yaitu dengan membuat suatu “Rancang Bangun Alat Pengering Ikan Otomatis Dengan Pengendalian Sensor Gas Menggunakan Kontrol *Fuzzy*”.

1.2 Rumusan Masalah

Adapun tujuan dari penelitian ini antara lain:

1. Bagaimana cara membuat rancang bangun alat pengering ikan dengan menggunakan sensor gas?
2. Bagaimana cara merancang kontrol *Fuzzy* pada alat pengering ikan untuk menentukan keputusan?
3. Bagaimana kinerja alat pengering ikan dengan menggunakan pendeteksian sensor gas?

1.3 Tujuan Masalah

Adapun tujuan dari penelitian ini antara lain:

1. Merancang alat pengering ikan otomatis.
2. Menerapkan kontrol *fuzzy logic* pada alat pengering ikan guna mendapatkan hasil yang optimal.
3. Menggunakan sensor gas pada alat pengering untuk mengetahui tingkat kematangan atau kekeringan dari ikan tersebut.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini diantaranya adalah:

1. Sistem kendali fuzzy digunakan untuk mengatur suhu pada alat pengering ikan.

2. Sensor gas digunakan untuk mendeteksi aroma ikan pada saat proses pengeringan ikan dan sebagai *input* pada kontrol fuzzy.
3. Pengeringan ikan dilakukan dengan jumlah ikan yang terbatas.



BAB 2. TINJAUAN PUSTAKA

Pada bab ini menjelaskan tentang referensi-referensi yang digunakan sebagai acuan pembuatan alat ini. Juga menjelaskan tentang kekurangan, kelebihan, dan mengangkat masalah-masalah yang terdapat pada materi-materi referensi, sehingga dapat mengoptimalkan alat yang akan dibuat. Dan juga menjelaskan tentang komponen-komponen yang digunakan pada alat ini. Dapat dijelaskan sebagai berikut, diantaranya adalah:

2.1 Penelitian Terkait

a. Lemari Pengering dan Sterilisasi Pakaian Bayi Otomatis

Penelitian terkait ini merupakan sebuah alat pengering yang memantu pekerjaan manusia yang dapat mengeringkan baju bayi secara otomatis. Alat ini menggunakan lampu *Ultraviolet* tipe-c yang berfungsi untuk membunuh bakteri yang terdapat pada pakaian bayi tersebut. Dan ditambahkan juga sensor SHT 11 yang berfungsi sebagai pendeteksi suhu dan kelembapan pada ruangan lemari pengering. Pada saat suhu $\leq 48^{\circ}\text{C}$ dan kelembapan $\geq 37\%$ alat ini akan bekerja komponen pengering dan pensterilisasi akan ON. (Rafika,2015).

b. Sistem Kendali Logika *fuzzy*

Logika *Fuzzy* adalah logika yang samar, dengan kata lain logika ini meniru sifat manusia yang samar dan tidak pasti. Logika *Fuzzy* berbeda dengan logika digital, jika logika digital nilainya selalu pasti (mengetahui dua keadaan) yaitu ON atau OFF, Ya atau Tidak, *High* atau *Low*. Sedangkan Logika *Fuzzy* nilainya samar yaitu 1,5 atau 0,5, jadi nilai objek dapat menjadi anggota dari semua himpunan dengan nilai derajat keanggotaan yang berbeda-beda. Pada penelitian yang terkait ini, logika *Fuzzy* digunakan untuk membantu penentuan penyakit dalam system pakar yang dilakukan melalui proses konsultasi antara sistem dengan pemakai. Pemakai akan menginput beberapa gejala, ciri, dan penyebab penyakit yang diderita pasien.(Malik dkk, 2007)

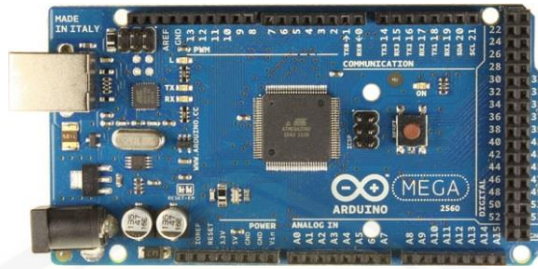
2.2 Arduino Mega

Arduino Mega 2560 adalah salah satu jenis dari produk arduino. Arduino ini menggunakan chip Atmega dan tidak lagi menggunakan chip FTDI untuk fungsi *USB to Serial Converter*. Berikut spesifikasi dari Arduino Mega 2560, antara lain:

Tabel 2.1 Spesifikasi Arduino Mega 2560

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

(<http://blog.famosastudio.com/2013/09/produk/arduino-mega-2560/531>)



Gambar 2.1 Bentuk Fisik Arduino Mega 2560
(<http://blog.famosastudio.com/2013/09/produk/arduino-mega-2560/531>)

2.3 Heating Element

Elemen pemanas listrik adalah komponen yang digunakan untuk pemanas dan mengakibatkan terjadinya perbedaan tekanan udara yang mengakibatkan percepatan proses pengeringan. Bentuknya seperti besi yang dilengkungkan dan terdapat kawat ataupun pita bertahanan listrik tinggi (*Resistance Wire*). Elemen pemanas listrik ini berfungsi pada saat tegangan 220 Vac 50Hz dan daya yang dibutuhkan pemanas ini antara 200-600 watt. Terdapat 2 macam jenis elemen pemanas listrik ini, antara lain:

1. Elemen Pemanas Listrik Bentuk Lanjut adalah elemen pemanas yang bentuk dasarnya dilapisi dengan pipa yang bertujuan untuk menyesuaikan penggunaan elemen pemanas tersebut.
2. Elemen Pemanas Listrik Bentuk Dasar adalah elemen pemanas yang mana *Resistance wire* hanya dilapisi dengan isolator listrik saja.



Gambar 2.2 Bentuk-Bentuk *Tubular Heater*
(<http://www.hotwatt.com/tastraight.htm>)

2.4 Sensor DHT-22

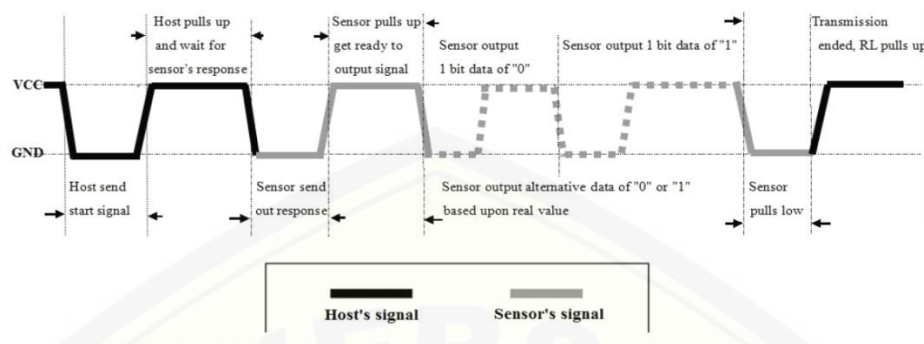
Sensor DHT-22 bisa membaca nilai kelembapan dengan range antara 0 sampai 100% dan bisa membaca nilai suhu dengan range antara -40°C sampai 125°C . Sensor ini pada saat bekerja dan menghasilkan keluaran sinyal listrik dengan otomatis akan dikalkulasi berapa persen kelembapan dan berapa derajat suhu yang terukur. Berikut adalah datasheet dari sensor DHT-22, yaitu:

Tabel 2.2 Spesifikasi sensor DHT-22

Model	DHT
Power Supply	3.3-6V DC
Output Signal	Digital signal via single-bus
Sensing Element	Polymer capacitor
Operatig range	Humidity 0-100%RH; temperature $-40\sim 80^{\circ}\text{C}$
Accuracy	Humidity $\pm 2\% \text{RH}$ (Max $\pm 5\% \text{RH}$); temperatur $< \pm 0.5^{\circ}\text{C}$
Resolution or sensitivity	Humidity 0.1%RH
Reoeaability	Humidity $\pm 1\% \text{RH}$
Humidity hysteresis	$\pm 0.3\% \text{RH}$
Long-term sability	$\pm 0.3\% \text{RH}/\text{year}$
Sensing period	Average: 2s
Interchangeability	Fully interchangeable
Dimensions	Small size 14x18x5.5mm; big size 22x28x5mm

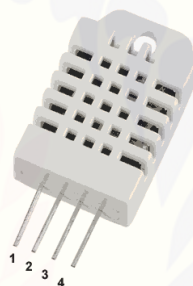
(Aosong Electronics Co.,Ltd)

Cara kerja sensor DHT-22 ini yaitu dengan mengirimkan *low signal* berturut-turut selama 18ms ke mikrokontroler dan akan dikembalikan dengan memberi signal *high* dan kemudian sensor DHT-22 ini akan mengirimkan data secara serial.



Gambar 2.3 *Timing Diagram Protocol Komunikasi Serial* (Aosong Electronics Co.,Ltd)

DHT22 pins	
1	VCC
2	DATA
3	NC
4	GND



Gambar 2.4 Sensor Gas DHT-22
(<https://www.electroschematics.com/11293/am2302-dht22-datasheet/>)

2.5 Kipas / Fan

Kipas adalah komponen elektronika yang berfungsi untuk menghasilkan aliran udara yang bertujuan untuk mempercepat proses pengeringan. Kipas terdiri dari baling-baling yang digunakan untuk menghasilkan aliran udara dan juga motor DC yang berfungsi untuk menggerakkan baling-baling sehingga menimbulkan aliran udara.

Tabel 2.3 Spesifikasi *Fan / Kipas*

Voltage-Rated	12VDC
Air Flow	37.0 CFM (1.04m ³ / min)
Static Pressure	0.150 in H2O (37.4 Pa)
Noise	30.0 dB(A)
Power (Watts)	1.40W
RPM	2900 RPM
Operating Temperature	14~158°F (-10 ~ 70 °C)
Current Rating	0.120A

(<https://www.digikey.com/product-detail/en/sunon-fans/EE80251S2-000U-999/259-1462-ND/2021090>)



Gambar 2.5 *Fan / Kipas*
(www.bhinneka.com)

2.6 Sensor MQ-8

Sensor ini merupakan sensor yang mendeteksi gas, sensor semikonduktor yang peka terhadap gas hidrogen yang respon pendekteksiannya cepat. Cara mengkalibrasi sensor ini yaitu dengan cara mengukurnya pada udara. Sensor ini terdiri dari sensor elektrokimia dan sebuah pemanas didalamnya. Sebagian dari sensor ini memerlukan tegangan 5V untuk memanaskan *heater* yang ada didalamnya secara terus menerus, sebagian lagi memerlukan tegangan 2V, dan sebagiannya memerlukan tegangan 5V dan 1,4V. arus tidak boleh lebih dari 40mA, jika lebih dari itu maka harus menggunakan mosfet atau transistor, jika dipaksakan maka akan kelebihan beban, dan bisa membuat arduino *overheating*. Berikut spesifikasi dari sensor MQ-8 antara lain:

Tabel 2.4 Spesifikasi Sensor MQ-8

Parameter name	Technical Condition
Circuit voltage	5V±0,1
Heating voltage	5V±0,1
Load resistance	10KΩ
Heater resistance	31±5%
Heating consumption	less than 800mW
Using Term	-10°C - 50°C
Storage Term	-20°C - 70°C
Related Humidity	less than 95% Rh
Sensing Resistance	10KΩ - 60KΩ (1000ppm H ₂)

(<http://www.datasheet4u.com/datasheet/M/Q/-/MQ-8- HANWEIELETRONICS.pdf.html>)



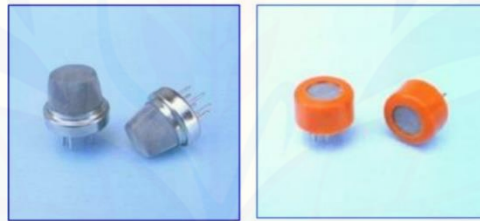
Gambar 2.6 Sensor Gas MQ-8
([arduinolearning.com](http://www.arduinolearning.com))

2.7 Sensor MQ-5

Sensor ini merupakan sensor yang dapat mendeteksi adanya gas LPG, natural gas dan *methane*. Sensor ini memerlukan tegangan pada *heater* sebesar 5V AC/DC dan sedangkan pada rangkaiannya memerlukan tegangan sebesar 5V DC. Batas minimal dan batas maksimal pembacaan sensor MQ-5 yaitu sebesar 200-10000 ppm. Suhu normal pada saat sensor ini mendeteksi adalah sebesar 20° C ±2° C dan kelembapannya sebesar 65% ± 5%. Disarankan pada saat menggunakan sensor MQ-5 harus dihubungkan dengan resistor sebesar 10 KΩ sampai 60 KΩ. Berikut spesifikasi sensor MQ-5, yaitu:

Tabel 2.5 Spesifikasi Sensor MQ-5

Parameter name	Technical Condition
Circuit voltage	5V±0,1
Heating voltage	5V±0,1
Load resistance	20KΩ
Heater resistance	31±10%
Heating consumption	less than 800mW
Using Term	-10°C - 50°C
Storage Term	-20°C - 70°C
Related Humidity	less than 95% Rh
Sensing Resistance	10KΩ - 60KΩ (5000ppm methane)

Gambar 2.7 Sensor Gas MQ-5
(Hanwei Electronics Co.,Ltd)

2.8 Sensor MQ-7

Sensor ini merupakan sensor gas yang sensitif terhadap jenis gas karbon monoksida, kemudian sensitif terhadap gas berjenis hidrogen. Adapun spesifikasi dari sensor ini yaitu:

Tabel 2.6 Spesifikasi Sensor MQ-7

Parameter Name	Condition
Concentration	300-10000ppm (Combustible gas)
Loop Voltage	$\leq 24V$ DC
Heater Voltage	$5.0V \pm 0.2V$ Ac or DC
Load Resistance	Adjustable
Heater Resistance	$31\Omega \pm 3\Omega$ (Room Temp)
Heater Consumption	$\leq 900mW$
Sensing Resistance	$2K\Omega - 20K\Omega$ (in 2000ppm C_3H_8)
Sensitivity	$R_s(\text{in air})/R_s(1000\text{ppm isobutane}) \geq$

(http://www.datasheet4u.com/datasheet/M/Q/-/MQ-7_Hanwei.pdf.html)



Gambar 2.8 Sensor Gas MQ-7

(<http://i.ebayimg.com/images/i/261534536013-0-1/s-11000.jpg>)

2.9 Kontrol Fuzzy Logic

2.9.1 Teori Sistem Kendali Fuzzy Logic

Logika *Fuzzy* adalah logika yang samar, dengan kata lain logika ini meniru sifat manusia yang samar dan tidak pasti. Logika *Fuzzy* berbeda dengan logika digital, jika logika digital nilainya selalu pasti (mengetahui dua keadaan) yaitu ON atau OFF, Ya atau Tidak, *High* atau *Low*. Sedangkan Logika *Fuzzy* nilainya samar yaitu 1,5 atau 0,5, jadi nilai objek dapat menjadi anggota dari semua himpunan dengan nilai derajat keanggotaan yang berbeda-beda. Pada penelitian yang terkait ini, logika *Fuzzy* digunakan untuk membantu penentuan penyakit dalam system pakar yang dilakukan melalui proses konsultasi antara sistem dengan pemakai. Pemakai akan

menginput beberapa gejala, ciri, dan penyebab penyakit yang diderita pasien.(Malik dkk, 2007).

2.9.2 Fungsi Keanggotaan

Untuk menghasilkan keputusan yang presisi dan mudah dalam mempresentasikan masalah maka perlu adanya peranan dari fungsi keanggotaan (*membership function*). Fungsi keanggotaan bekerja dengan cara memetakan masukan yang bernilai tegas dan tertentu ke dalam derajat keanggotaannya yang bernilai antara 0 sampai 1. Untuk mendapatkan nilai dari derajat keanggotaan perlu adanya pendekatan yaitu dengan cara mempresentasikannya kedalam beberapa fungsi diantaranya adalah:

1. Representasi Linear
2. Representasi Kurva Segitiga
3. Representasi Trapesium
4. Representasi Kurva Bentuk Bahu
5. Representasi Kurva Sigmoid
6. Representasi Kurva Phi

2.9.3 Operasi Himpunan Fuzzy

Setelah ditentukannya nilai dari fungsi keanggotaan fuzzy, maka perlu adanya modifikasi dan perpaduan dari himpunan fuzzy. Terdapat tiga jenis operator yang diciptakan oleh Zadeh, yaitu:

1. Operator *AND* (irisan)
$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$$
2. Operator *OR* (gabungan)
$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$
3. Operator *NOT*
$$\mu_{\bar{A}} = 1 - \mu_A[x]$$

2.9.4 *Fuzzification*

Fuzzifikasi adalah proses mengubah nilai masukan yang bersifat pasti kedalam bentuk variabel linguistik. Fuzzifikasi diharapkan dapat membantu dalam mempermudah komputasi untuk proses inferensinya.

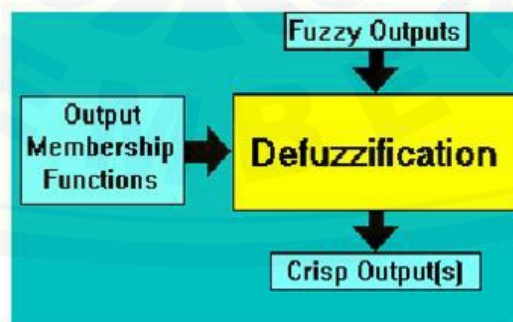
2.9.5 Sistem Inferensi Fuzzy

Fuzzy infernce adalah suatu penarikan kesimpulan dari proses penalaran *fuzzy input* dan *rule base* yang telah ditentukan dengan prosedur implikasi fuzzy. Terdapat dua model aturan fuzzy yaitu dengan model mamdani dan model sugeno.

2.9.6 *Defuzzyfikasi*

Defuzzifikasi adalah suatu proses penerjemahan dari nilai *fuzzy output* kedalam nilai yang tegas berdasarkan fungsi keanggotaan yang telah ditentukan. Terdapat beberapa metode defuzzifikasi untuk memodelkan sistem fuzzy, diantaranya adalah:

1. Metode Centroid
2. Metode Bisektor
3. Metode Mean of Maximum
4. Metode Smallestof
5. Metode Largest of Maximum



Gambar 2.9 *Defuzzyfication*
(fuzzylogic7.blogspot.co.id/2011/06/defuzzifikasi.html)

BAB 3. METODE PENELITIAN

3.1 Waktu dan Tempat

3.1.1 Tempat

Adapun tempat penelitian, pengujian dan analisis dilakukan secara umum dilakukan di:

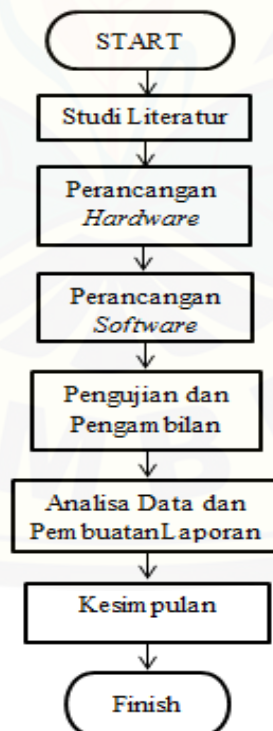
Tempat : Lab. Elektronika dan Terapan Fakultas Teknik Universitas Jember

Alamat : Jl. Slamet Riyadi No. 62, Patrang, Kabupaten Jember

3.1.2 Waktu

Adapun waktu penelitian, pengujian dan analisis dilakukan secara umum pada bulan Desember 2017 sampai dengan April 2018

3.2 Metodologi Penelitian



Gambar 3.1 *Flowchart* Penelitian

Alur penelitian “Rancang Bangun Alat Pengering Ikan Otomatis dengan Pengendalian Sensor Gas Menggunakan Kontrol *Fuzzy*” sebagai berikut:

1. Studi literatur

Alur penelitian yang pertama yaitu dengan melakukan pencarian literatur yang terkait dengan pengeringan ikan, sensor gas, dan kontrol *Fuzzy*. Tujuannya agar perancangan dan pelaksanaan penelitian dapat menghasilkan *Output* sesuai dengan yang diinginkan.

2. Perancangan *Hardware*

Alur penelitian yang kedua yaitu dengan melakukan proses perancangan *Hardware* yang terdiri dari desain alat pengering ikan, cara kerja dari masing-masing komponen elektronik yang saling bersinergi, alat dan bahan yang digunakan.

3. Perancangan *Software*

Alur penelitian yang ketiga yaitu dengan melakukan proses perancangan *plant* dan sistem kontrol *Fuzzy*, sensor suhu, sensor gas, program arduino, menentukan *Membership Function* dan menentukan *Rule Based* pada logika *Fuzzy* agar *Output* yang dihasilkan sesuai dengan yang diinginkan.

4. Pengujian dan Pengambilan Data

Alur penelitian yang keempat yaitu dengan melakukan pengujian terhadap komponen-komponen yang kritis seperti pengujian terhadap sensor gas dan suhu. Komponen-komponen ini sangatlah berpengaruh terhadap tingkat keberhasilan dari penelitian ini.

5. Analisa Data dan Pembuatan Laporan

Alur penelitian yang kelima yaitu dengan menganalisa data yang telah didapatkan dari penelitian sebelumnya. Data didapat dari hasil pengujian sensor gas dan suhu yang dikontrol menggunakan *Fuzzy*, sehingga dari analisa setiap percobaan diharapkan mendapatkan *Output* yang sesuai dengan yang diinginkan.

6. Penarikan Kesimpulan

Alur penelitian yang keenam yaitu dengan memberikan kesimpulan dari pengujian dan analisa data yang diperoleh.



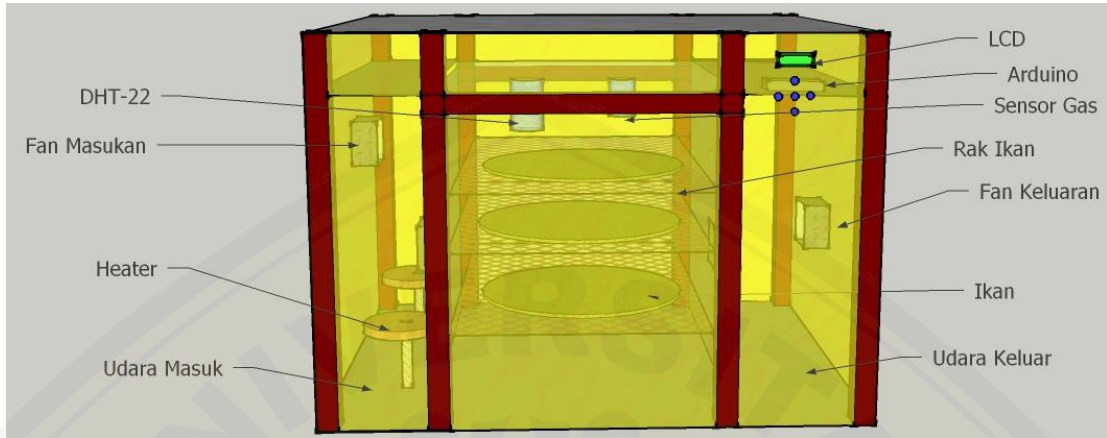
3.3 Perancangan Perangkat Keras

3.3.1 Alat dan Bahan

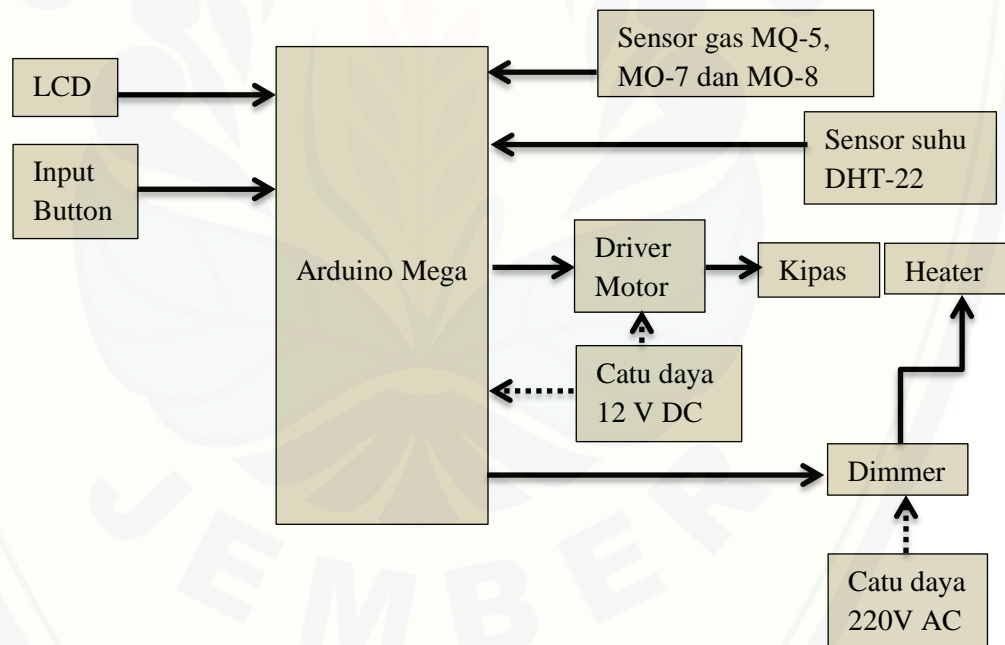
Tabel 3.1 Daftar Alat dan Bahan yang Digunakan

No	Nama Alat dan Bahan	Spesifikasi Teknis dan Penjelasan
1	<i>Power Supply</i>	<i>Input Voltage AC 85 -265V 50 / 60 Hz, Output Voltage 12V DC, 20A, dan 240W.</i>
2	Arduino	Arduino Mega 2560, Atmega 2560 <i>microcontroller</i> , <i>Input Voltage 7-12V, 5V Electric current : 500mA,</i> <i>3.3 Electric current : 50mA, 54 Digital I/O Pins</i> <i>(15 PWM Output), 16 Analog Inputs, 256 KB</i> <i>Flash Memory, 16 Clock Speed.</i>
3	DHT-22	<i>Input Voltage 3.3-6V, Digital signal via single-bus,</i> <i>Operating range temperature -40~80 °C.</i>
4	MQ-5	<i>Heating Voltage 5V±0,1, Sensing Resistance</i> <i>10KΩ-60KΩ (5000ppm methane), Storage Tem -</i> <i>20°C-70°C</i>
5	MQ-7	<i>Circuit voltage 5V±0,1, Detecting Range 20ppm-</i> <i>2000ppm Carbon monoxide,</i>
6	MQ-8	<i>Heating Voltage 5V±0,1, Storage Tem -20°C-70 °C,</i> <i>Detecting Concentration 100-10000ppm</i> <i>(Hydrogen)</i>
7	LCD	LCD 1602 + 12C, <i>Voltage 5V DC, Display 2-lines,</i> <i>16-characters.</i>
8	<i>Fan</i>	10 buah ukuran 5x5cm: <i>Electric current 0,09A</i> 2 buah ukuran 12x12cm: <i>Electric current 0,12A</i>
9	<i>Heater Element</i>	2 buah, <i>Power 190W, Voltage 220V, Frekuensi</i> <i>50Hz</i>
10	LED Strip	10 buah, <i>Voltage 12V</i>

3.3.2 Desain Alat



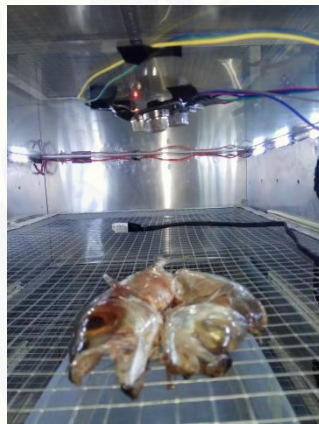
Gambar 3.2 Desain Alat Pengering Ikan



Gambar 3.3 Blok Diagram Perangkat Keras



Tampak Depan Alat Pengering

Rangkaian Driver kipas,
Kontrol dan dimmerRangkaian LCD
dan tombol

Penempatan Sensor Gas



Ruang Pemanas

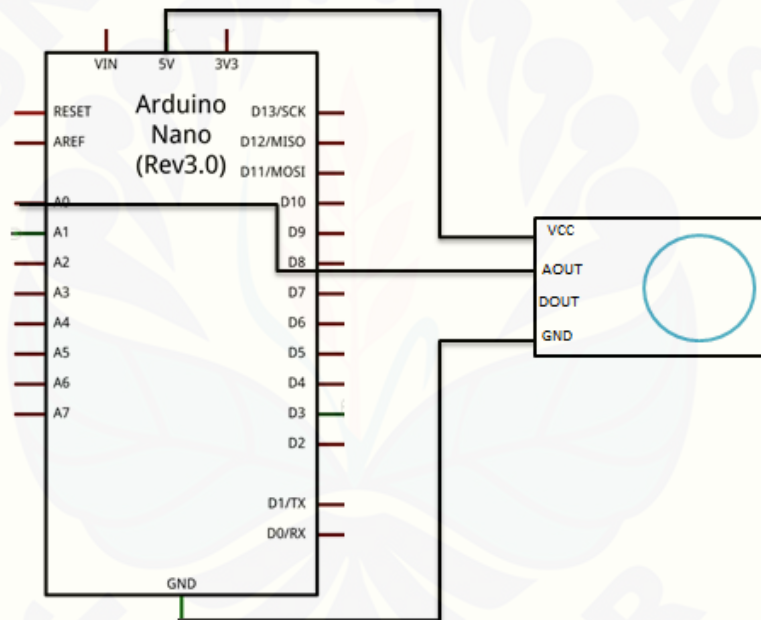
Gambar 3.4 Alat Pengering Ikan

Alat pengering ikan berbentuk seperti lemari yang terbuat dari aluminium, dilengkapi dengan *heating element* sebagai pemanas ruangan pengering ikan tersebut dan juga kipas sebagai pengatur sirkulasi udara pada ruangan pengering ikan tersebut. Adapun sensor yang dipakai yaitu sensor DHT-22 sebagai sensor suhu dan juga sensor gas yang terdiri dari MQ-5, MQ-7 dan MQ-8 digunakan untuk mendeteksi tingkat kekeringan ikan. Arduino Mega pada alat ini digunakan untuk sistem pengontrolan.

3.3.3 Desain Elektronik

3.3.3.1 Rangkaian Sensor MQ

Sensor gas yang saya gunakan diantaranya yaitu, MQ-5, MQ-7, dan MQ-8. Modul sensor MQ memiliki 4 kaki diantaranya yaitu, VCC, Aout, Dout, dan GND. Bentuk rangkaian dari sensor MQ ke arduino seperti pada gambar 3.3, jadi kaki VCC dari sensor MQ dihubungkan ke pin 5V pada arduino, kaki Aout pada sensor MQ dihubungkan ke pin A0 (*analog*) pada arduino dan kaki GND pada sensor MQ dihubungkan ke pin GND pada arduino, sedangkan kaki pin Dout pada sensor MQ tidak digunakan.



Gambar 3.5 Rangkaian Sensor MQ
(<http://ngopinih.blogspot.co.id/2017/11/m>)

3.3.3.2 Rangkaian Sensor DHT-22

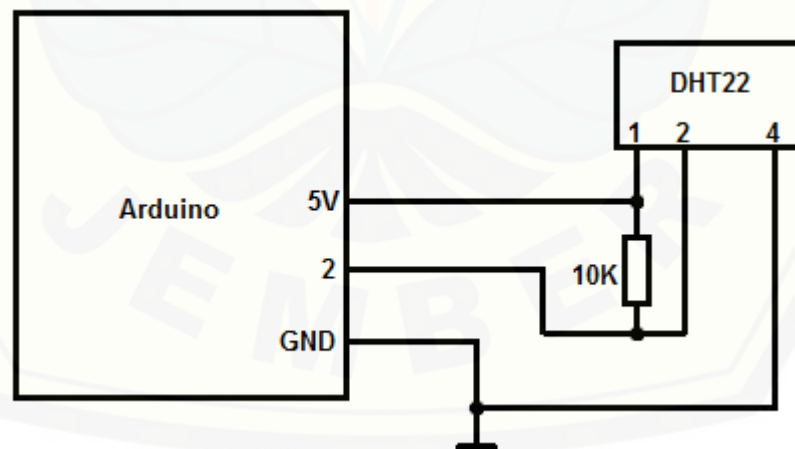
Sensor DHT-22 dihubungkan ke arduino yang sebagai perangkat akusisi data dengan cara menghubungkan pin catu daya ke pin 5V dan GND, sedangkan pin data dari sensor dihubungkan ke pin digital pada arduino. Komunikasi arduino dengan sensor DHT-22 menggunakan *single wire* atau *one wire two way serial interface*. Jadi, pin arduino digunakan untuk jalur input dari sensor, tidak menggunakan pin

serial UART (*universal asynchronous receiver/transmitter*), karena komunikasi tersebut membutuhkan dua jalur, yaitu pada pin *receiver* dan *transmitter*.

Listing program arduino:

```
1| #include <DHT.h>
2| #define DHTPIN 23
3| #define DHTTYPE DHT22
4| DHT dht(DHTPIN, DHTTYPE);
5| T = dht.readTemperature();
6| temp= (1.043*T) - 1.809;
```

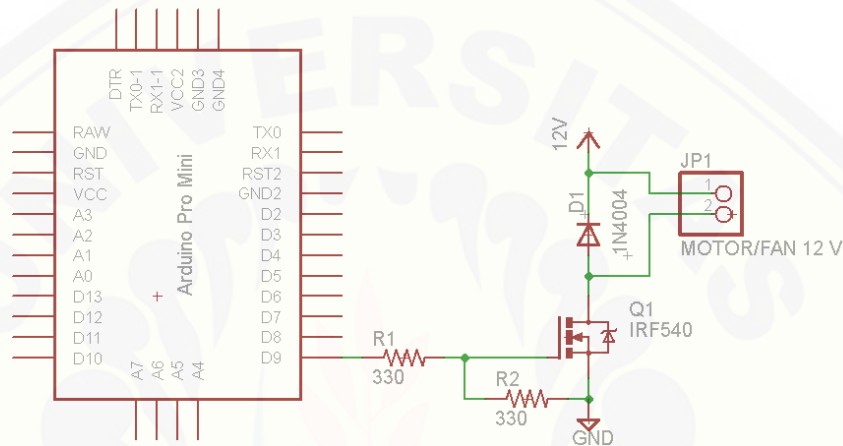
Arduino mempunyai *library* sendiri untuk sensor DHT. *Listing* program diatas pada baris pertama merupakan *library* dari DHT, pada baris ke-2 merupakan inisialisasi pin 23 pada sensor DHT, pada baris ke-3 merupakan inisialisasi dari tipe sensor DHT tersebut, pada baris ke-4 adalah penentuan tipe dan pin untuk sensor DHT, pada baris ke-5 membaca dan menyimpan nilai suhu dalam bentuk desimal, dan pada baris ke-6 merupakan rumus suhu yang sudah dilakukan pengkalibrasian.



Gambar 3.6 Rangkaian Sensor DHT-22
(<http://lang8088.blogspot.co.id/2015/07/tutorial-project-arduino-mengukur-suhu.html>)

3.3.3.3 Rangkaian Driver Motor Ventilasi

Rangkaian ini digunakan untuk mengontrol kecepatan motor DC yang berupa kipas. Kipas angin motor DC ini digunakan untuk mengalirkan udara dari lemari pengering ikan. Motor DC ini dikendalikan kecepatannya dengan input PWM dari arduino. Berikut adalah gambar rangkaian *driver* motor ventilasi.



Gambar 3.7 Rangkaian Driver Motor

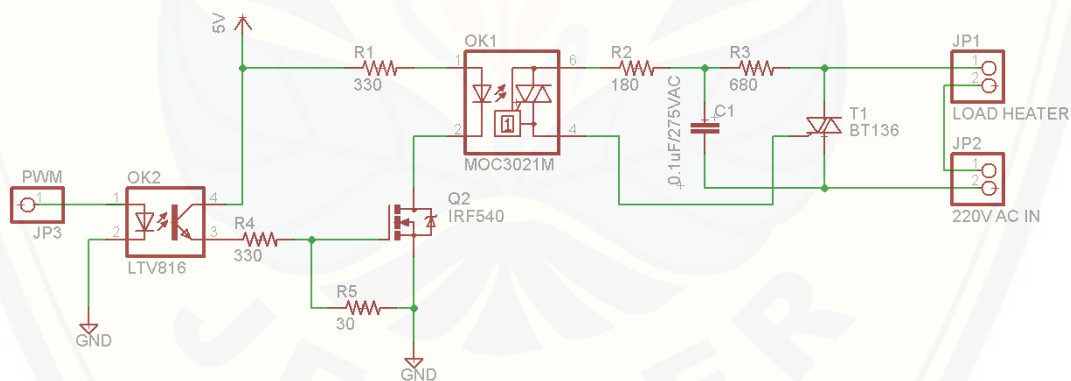
Dari rangkaian *driver* motor diatas menggunakan beberapa komponen elektronika yaitu MOSFET IRF540, dioda 1N4004 dan beberapa resistor. Pada rangkaian ini menggunakan IRF540 dikarenakan dalam penguatan arus pada rangkaian lebih kuat, *noise* rendah, waktu *switching* lebih cepat, impedansi *input* dan *output* lebih tinggi daripada menggunakan transistor BJT. Berikut adalah perhitungan dari rangkaian tersebut.

$$\begin{aligned}
 I_d &= 1090 \text{ mA} & V_g &= 5 \times R_1 / (R_1 + R_2) \\
 I_g &= I_d / h_{fe} & &= 5 \times R / (R + R) \\
 &= 1090 \text{ mA} / 100000 & &= 5 \times 1/2 \\
 &= 0,0109 \text{ mA} & V_g &= 2,5 \text{ volt} \\
 R_g &= (V_g - V_{gs}) / I_g \\
 R_g &= (2,5 \text{ volt} - 0,7 \text{ volt}) / 0,0109 \text{ mA} & R &= 165 \times 2 \\
 R_g &= 1,8 \text{ volt} / 0,0109 \text{ mA} & &= 330 \text{ ohm} \\
 R_g &= 165 \text{ ohm} & R_1 &= R_2
 \end{aligned}$$

3.3.3.4 Rangkaian *Driver Heater*

Alat pengering ikan ini menggunakan *Heater* yang digunakan untuk membuat udara panas dan kemudian dikompres udaranya oleh *fan* / kipas. *Element Heater* menggunakan daya sebesar 190 watt dan sedangkan kipas membutuhkan tegangan 12 V.

Heater element dapat diatur besar kecilnya intensitas pemanas itu dengan cara menghubungkannya dengan rangkaian kontrol *dimmer*. Rangkaian kontrol *dimmer* digunakan untuk mengendalikan pemanas dengan sistem PWM didalam mengendalikan tegangan AC. Komponen kontrol *dimmer* ini berupa TRIAC BT136, MOC 3021 dan ATmega328 yang berfungsi sebagai *device* pemberi sinyal PWM, lalu sinyal PWM diterima oleh MOC3021 yang akan mengaktifkan *led* inframerah *internal* dan mengaktifkan DIAC. Kemudian MOC3021 aktif dan mengendalikan TRIAC BT136 yang berfungsi untuk mengendalikan tingkat kepanasan dari pemanas *element* AC 220 volt, sebagai berikut gambar rangkaian *dimmer*.



Gambar 3.8 Rangkaian *Dimmer*

Dari gambar rangkaian *dimmer* diatas dapat dilihat bahwa rangkaian tersebut dapat masukan dari mikrokontroler berupa PWM, kemudian melalui *optocoupler* yang berfungsi untuk menyalurkan PWM ke kaki *gate* MOSFET IRF 540. Resistor yang terhubung langsung dengan vcc (R1) digunakan untuk *pull-up*. Jika pada saat PWM bernilai tinggi atau sama dengan kondisi *High* maka MOC3021M akan *On* dan menyebabkan triac BT136 juga dalam kondisi *On* dan beban akan aktif. Begitu

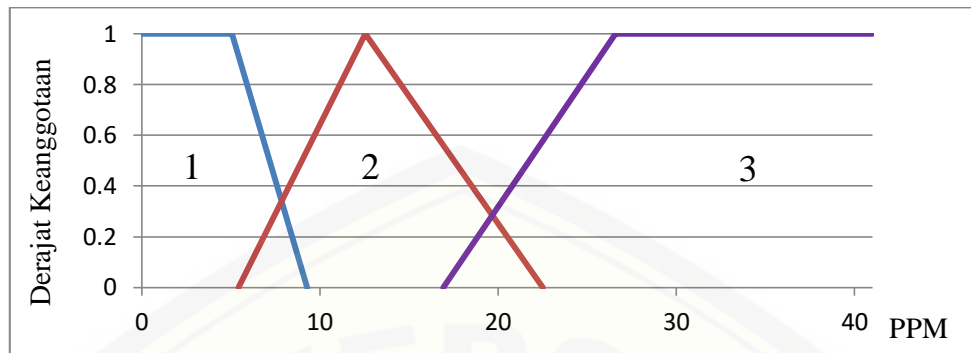
juga sebaliknya, jika pada saat PWM rendah atau sama dengan dalam keadaan *Low* maka arus dari VCC akan kembali ke pin mikrokontroler tidak masuk kedalam MOC3021M. Rangkaian *snubber* digunakan untuk meminimalisir terjadinya loncatan tegangan (*spike*). Rangkaian *snubber* ini terdiri dari beberapa komponen elektronik, yaitu berupa resistor dan kapasitor. Rangkaian *snubber* terdiri dari R2 dan C1 dan sedangkan R3 digunakan untuk membuang tegangan pada C1 pada saat BT136 *off*. Berikut perhitungan dari rangkaian *dimmer* diatas:

$$\begin{aligned}
 I_d &= 1090 \text{ mA} & V_g &= 5 \times R_1 / (R_1 + R_2) \\
 I_g &= I_d / h_{fe} & &= 5 \times R / (R + R) \\
 &= 1090 \text{ mA} / 100000 & &= 5 \times 1/2 \\
 &= 0,0109 \text{ mA} & V_g &= 2,5 \text{ volt} \\
 R_g &= (V_g - V_{gs}) / I_g & & \\
 R_g &= (2,5 \text{ volt} - 0,7 \text{ volt}) / 0,0109 \text{ mA} & R &= 165 \times 2 \\
 R_g &= 1,8 \text{ volt} / 0,0109 \text{ mA} & &= 330 \text{ ohm} \\
 R_g &= 165 \text{ ohm} & R_1 &= R_2
 \end{aligned}$$

3.4 Perancangan Sistem *Fuzzy Logic*

3.4.1 Proses Fuzzyfikasi

Pada sistem kontrol *fuzzy logic* terdapat tiga buah *input* yaitu berupa data pembacaan ketiga sensor gas. Dimana setiap sensor gas memiliki fungsi keanggotaan yang berbeda beda karena dalam proses pengeringan ikan kadar gas karbon monoksida, metana, dan hidrogen berbeda beda.

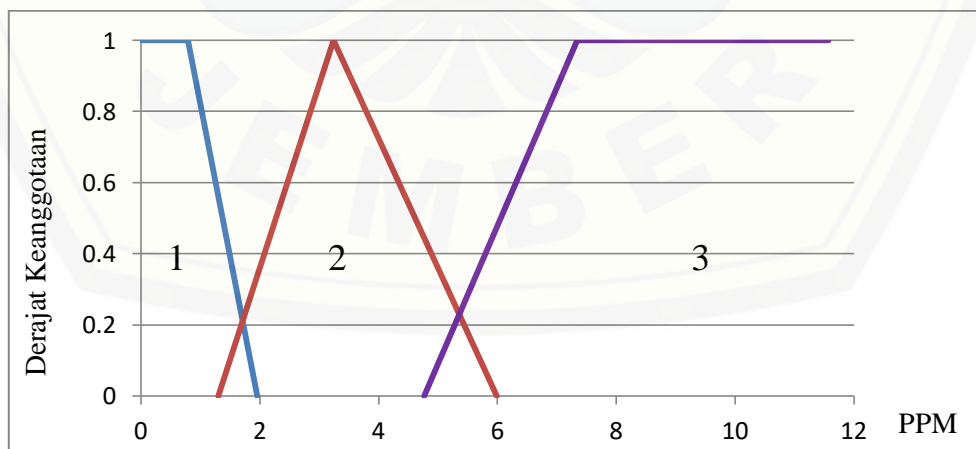


Gambar 3.9 Membership Function Input MQ-5

Keterangan:

1. Kecil = 0-9,28 (PPM)
2. Sedang = 5,44-22,5 (PPM)
3. Besar = 16,91-41 (PPM)

Data dari fungsi keanggotaan masih dalam bentuk ADC yang nantinya akan dikonversi ke dalam satuan ppm. Penentuan daerah batas *crisp* dilakukan dengan cara pembacaan sensor gas pada satu sampel yang kemudian akan didapatkan data pembacaan sensor gas sehingga dapat dimasukkan batasan *crisp* ke dalam fungsi keanggotaan. Pemberian label *membership function input MQ-5* yaitu pada nilai sebesar 0-9,28 ppm diberi label Kecil, pada nilai sebesar 5,44-22,5 ppm diberi label Sedang dan pada nilai sebesar 16,91-41 ppm diberi label Besar.

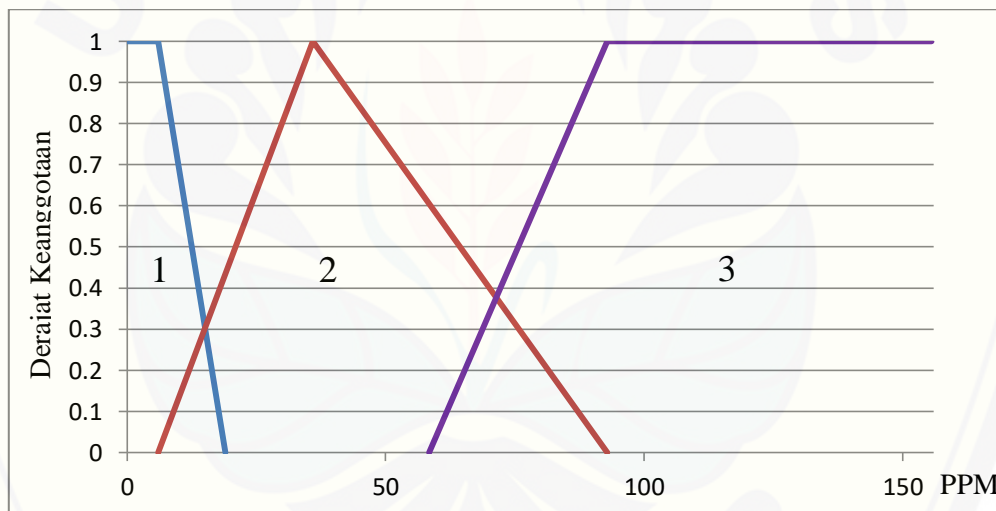


Gambar 3.10 Membership Function Input MQ-7

Keterangan:

1. Kecil = 0-1,96 (PPM)
2. Sedang = 1,3-5,99 (PPM)
3. Besar = 4,77-11,57 (PPM)

Data dari fungsi keanggotaan masih dalam bentuk ADC yang nantinya akan dikonversi ke dalam satuan ppm. Penentuan daerah batas *crisp* dilakukan dengan pembacaan sensor gas sehingga dapat dimasukan batasan *crisp* ke cara pembacaan sensor gas pada satu sampel yang kemudian akan didapatkan data dalam fungsi keanggotaan. Pemberian *label membership function input MQ-7* yaitu pada nilai sebesar 0-1,96 ppm diberi label Kecil, pada nilai sebesar 1,3-5,99 ppm diberi label Sedang dan pada nilai sebesar 4,77-11,57 ppm diberi label Besar.



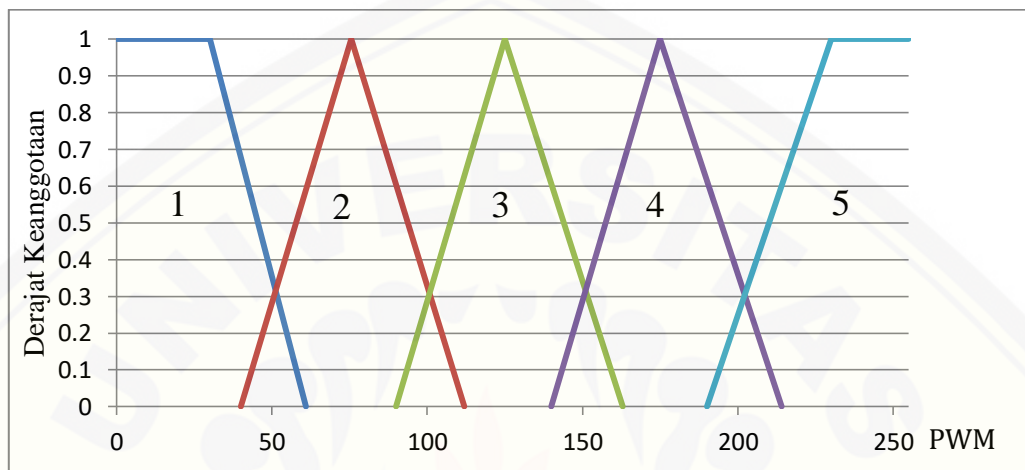
Gambar 3.11 Membership Function Input MQ-8

Keterangan:

1. Kecil = 0-19,04 (PPM)
2. Sedang = 6,05-92,86 (PPM)
3. Besar = 58,41-155,64 (PPM)

Data dari fungsi keanggotaan masih dalam bentuk ADC yang nantinya akan dikonversi ke dalam satuan ppm. Penentuan daerah batas *crisp* dilakukan dengan cara pembacaan sensor gas pada satu sampel yang kemudian akan didapatkan data pembacaan sensor gas sehingga dapat dimasukan batasan *crisp* ke dalam fungsi

keanggotaan. Pemberian *label membership function input MQ-8* yaitu pada nilai sebesar 0-19,04 ppm diberi label Kecil, pada nilai sebesar 6,05-92,86 ppm diberi label Sedang dan pada nilai sebesar 58,41-155,64 ppm diberi label Besar.



Gambar 3.12 *Membership Function Output*

Keterangan:

1. Kecil = 0-61,02
2. Agak Kecil = 40-112
3. Sedang = 89,98-163
4. Agak Besar = 140-214
5. Besar = 190-255

Output dari sistem kendali *fuzzy* pada alat pengering ikan ini yaitu berupa PWM dari *heater* dan kipas yang digunakan untuk mengatur suhu pada ruangan alat pengering ikan. Suhu ruangan pengering ikan diatur sesuai dengan *set point* yang sudah ditentukan, sedangkan *set point* suhu ruangan tersebut diperoleh dari *output* kontrol *fuzzy*. Sebelum mendapatkan *output* dari kontrol *fuzzy*, harus melalui beberapa tahap proses yaitu dengan menentukan *membership function input* dan juga menentukan *rule base*. Dalam menentukan *rule base* harus melakukan beberapa kali percobaan perhitungan agar mendapatkan hasil *output* dari kontrol *fuzzy* yang sesuai dengan keinginan. Hal ini terjadi karena apabila pada saat pembuatan peraturan dari setiap kondisi proses kontrol tidak sesuai dengan kondisi yang terjadi, maka akan

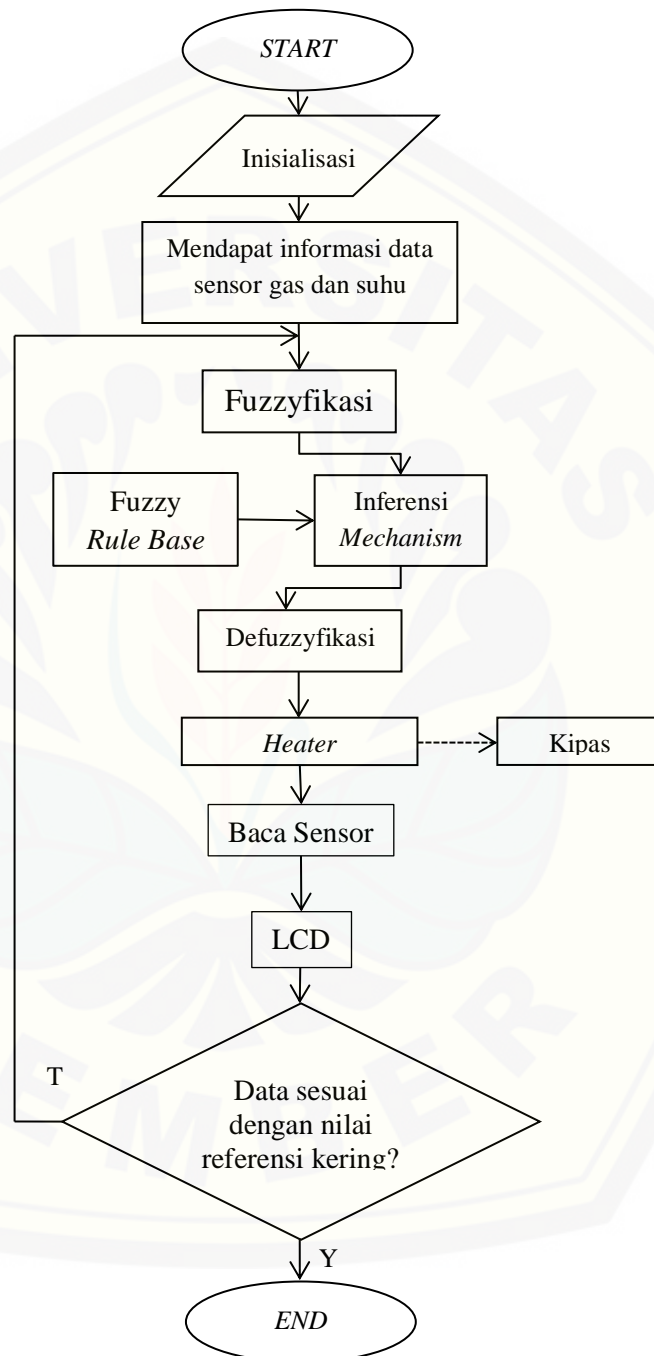
mempengaruhi *output* dari kontrol *fuzzy* tersebut dan membuat *output* tidak sesuai dengan target yang diinginkan. Berikut adalah tabel *rule base* dari sistem kontrol *fuzzy* yang sesuai dengan target yang diinginkan.



Tabel 3.2 *Rule Based Fuzzy Logic*

No	MQ-5	MQ-7	MQ-8	OUTPUT
1	Kecil	Kecil	Kecil	Kecil
2	Kecil	Kecil	Sedang	Agak Kecil
3	Kecil	Kecil	Besar	Sedang
4	Kecil	Sedang	Kecil	Sedang
5	Kecil	Sedang	Sedang	Agak Besar
6	Kecil	Sedang	Besar	Agak Besar
7	Kecil	Besar	Kecil	Sedang
8	Kecil	Besar	Sedang	Agak Besar
9	Kecil	Besar	Besar	Besar
10	Sedang	Kecil	Kecil	Agak Kecil
11	Sedang	Kecil	Sedang	Sedang
12	Sedang	Kecil	Besar	Agak Besar
13	Sedang	Sedang	Kecil	Sedang
14	Sedang	Sedang	Sedang	Sedang
15	Sedang	Sedang	Besar	Agak Besar
16	Sedang	Besar	Kecil	Agak Besar
17	Sedang	Besar	Sedang	Agak Besar
18	Sedang	Besar	Besar	Besar
19	Besar	Kecil	Kecil	Agak Kecil
20	Besar	Kecil	Sedang	Sedang
21	Besar	Kecil	Besar	Agak Besar
22	Besar	Sedang	Kecil	Sedang
23	Besar	Sedang	Sedang	Agak Besar
24	Besar	Sedang	Besar	Besar
25	Besar	Besar	Kecil	Agak Besar
26	Besar	Besar	Sedang	Agak Besar
27	Besar	Besar	Besar	Besar

3.5 Flowchart



Gambar 3.13 Flowchart Kendali Fuzzy

3.6 Pengujian dan Pengumpulan Data

Pengujian alat dilakukan dengan cara menggunakan *heater*, kipas, sensor DHT-22, arduino dan sensor gas yang terdiri dari sensor gas MQ-5, MQ-7 dan MQ-8. Pengujian dilakukan dengan cara *heater* bekerja dengan tujuan untuk mengeringkan objek dan kipas bekerja bersamaan dengan *heater* bertujuan untuk mengatur sirkulasi udara dalam lemari pengering ikan. Sensor DHT-22 melakukan pembacaan terus-menerus untuk memantau suhu dan kelembapan dalam ruangan dan sensor gas MQ-5, MQ-7, MQ-8 juga melakukan pembacaan terus-menerus untuk memantau bau yang dikeluarkan ikan selama proses pengeringan. Kemudian semua sensor mengirimkan informasi tersebut ke arduino, lalu arduino melakukan pengontrolan terhadap *heater* dan kipas dengan menggunakan sistem *Fuzzy Logic*.

3.7 Metode Analisa Data

Pengujian dilakukan dengan cara mengkombinasi antara sensor gas dan sensor suhu. Pada saat proses pengeringan ikan berlangsung suhu ruangan pada alat pengering ikan ini dipengaruhi oleh sensor gas. Pada saat mentah, sedang dan kering aroma yang dikeluarkan dari ikan tersebut berbeda sehingga membuat suhu dikontrol sesuai perintah yang sudah ditentukan. Selain itu, pada saat aroma yang dikeluarkan mulai berbeda dari aroma sebelumnya, panelis mengontrol keadaan ikan yang dikeringkan masih dalam keadaan basah atau kering. Hal ini bertujuan untuk mendapatkan nilai referensi aroma pada saat dalam keadaan basah, sedang dan kering. Sedangkan untuk indeks keberhasilan dari sistem ini yaitu dengan cara membandingkan hasil dari pengujian alat tanpa menggunakan sensor gas (mendeteksi aroma) dengan pengujian alat menggunakan sensor gas (mendeteksi aroma).

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil yang telah diperoleh dalam penelitian yang telah dilakukan, dapat diambil beberapa kesimpulan sebagai berikut:

1. Aroma ikan pada saat dikeringkan selalu berubah pada setiap waktu. Semakin kecil aroma ikan yang dideteksi oleh sensor gas maka ikan tersebut semakin kering. Suhu yang digunakan pada pengeringan ikan ini sebaiknya diantara 20 °C sampai dengan 70 °C.
2. Alat pengering ikan menggunakan sensor gas dengan kendali fuzzy *logic* ini bekerja dengan baik. Kontrol fuzzy dalam menentukan keputusan tingkat kekeringan ikan sangat bagus dan sesuai dengan yang diharapkan. *Output* dari kendali fuzzy ini adalah suhu ruangan alat pengering ikan
3. Pengeringan ikan menggunakan alat ini dengan kapasitas penuh membutuhkan waktu ± 12 jam dan rata-rata kadar air ikan sebesar 39,5%, sedangkan pengeringan ikan dengan kapasitas setengah membutuhkan waktu ± 11 jam dan rata-rata kadar air ikan sebesar 40,07%. Proses pengeringan ikan dengan *set point* suhu sebesar 70 °C membutuhkan waktu selama $\pm 9,5$ jam dan rata-rata kadar air ikan sebesar 40,82%. Pengeringan ikan dengan cara konvensional (dijemur dibawah terik matahari) membutuhkan waktu selama ± 3 hari, apabila selama proses pengeringan ikan cuaca tidak mendukung maka ikan akan terjadi pembusukan dan produk yang dihasilkan tidak higienis karena terkontaminasi dengan bakteri dan binatang. Sehingga cara pengeringan ikan dengan menggunakan alat pengering ikan ini sangat efektif dibandingkan dengan menggunakan cara konvensional.

5.2 Saran

Dengan harapan penelitian ini dapat dikembangkan dan disempurnakan dikemudian hari, penulis memberikan saran sebagai berikut:

1. Alat pengering ikan ini masih perlu adanya perbaikan dalam segi kapasitas penampungan ikan. Pada alat ini masih terbatas kapasitas ikan yang

dikeringkan. Semakin besar kapasitas penampungan semakin sedikit biaya yang harus dikeluarkan. Pada saat pengeringan ikan dengan kapasitas penuh membutuhkan waktu ± 12 jam dengan jumlah ikan sebanyak 30 ekor ikan. Sedangkan pada saat pengeringan ikan dengan kapasitas setengah membutuhkan waktu ± 11 jam dengan jumlah ikan sebanyak 15 ekor ikan. Dari data tersebut dapat dilihat bahwa semakin banyak ikan yang dikeringkan atau kapasitas penampungan semakin besar, maka proses pengeringan ikan akan semakin ekonomis

2. Agar tingkat keakuratan dalam proses pengeringan ikan bertambah, maka perlu adanya penambahan sensor berat guna menambah indikator pembacaan sensor dalam proses pengeringan ikan.

DAFTAR PUSTAKA

- Adawyah, 2007. Pengolahan dan Pengawetan Ikan. Bumi Aksara, Jakarta.
- Aosong Electronics Co, (2012), *Temperature and Humidity Module. DHT22.Product Manual*,Guangzhou, China.
- Apriliyadi. 2014. Perancangan mesin pengering ikan dengan menggunakan sistem kendali logika fuzzy. (910017111013):1–7.
- Darjat. 2008. Sistem pengendalian suhu dan kelembaban pada mesin pengering kertas. *Jurnal Teknik Elektro*. 10(2):82–88.
- Dwi Argo, D. 2012. Model simulasi pengendalian suhu udara pada mesin pengering cabe dengan kontrol logika fuzzy abstrak pendahuluan cara pengawetan cabe yang sering dilakukan adalah melalui pengeringan , baik pengeringan secara alami dengan sinar matahari ataupun pengeringa. 5(3):156–172.
- Malik dkk, 2007. Penerapan Logika *Fuzzy* Pada Aplikasi Sistem Pakar Untuk Mendiagnosa Penyakit Neurologi Pada Manusia Berbasis WEB
- Rafika Amalia, S. 2015. Jurnal teknologi elektro dan kejuruan. 23
- Sofyan. 2016. Manipulasi suhu pada pengeringan ikan teri tenaga surya menggunakan mikrokontroler atmega 2560 (skripsi) oleh sofyan abstract manipulation temperature on fish teri drying solar power using microcontroller atmega 2560 by sofyan. 2560
- Wolok, E. 2012. Perancangan dan Analisis *Performance* Alat Pengering Mekanik Multi Komoditi Tipe Udara Hembus
2011. Defuzzifikasi
fuzzylogic7.blogspot.co.id/2011/06/defuzzifikasi.html
2013. Arduino Mega 2560
<http://blog.famosastudio.com/2013/09/produk/arduino-mega-2560/531>
2013. MQ-7 Sensor Gas CO
<http://baskarapunya.blogspot.co.id/2013/05/mq-7-sensor-gas-co.html?m=1>

2014. Definisi Logika *Fuzzy*.

<http://kasminarif.blogspot.co.id/2013/05/definisi-logika-fuzzy.html>

2015. Bentuk dan Karakteristik *Heater*.

<http://www.hotwatt.com/tastraight.htm>

2015. Tutorial *Project* Arduino Mengukur Suhu

<http://lang8088.blogspot.co.id/2015/07/tutorial-project-arduino-mengukur-suhu.html>

2016. *Heater Element*.

<https://sukeslajujaya.wordpress.com/2016/03/11/heater-elements/>

2016. Macam-macam kipas (*Fan*).

<http://artikel-teknologi.com/macam-macam-kipas-fan/>

2017. Pengertian Sensor MQ-8.

<http://www.vcc2gnd.com/m/p.php?sku=MQ8>

LAMPIRAN

LAMPIRAN A. *Listing* Program Arduino

```
#include <Servo.h>

#include <DHT.h>

#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x3F ,2,1,0,4,5,6,7,3, POSITIVE);

#include <Wire.h>

Servo myservo;

#include "fis_header.h"

char derajat = 0xDF;

#define DHTPIN 23

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

#include <SD.h>

#include <SPI.h>

#include "RTClib.h"

RTC_DS1307 rtc;

File myFile;

int pinCS = 53;

//-----Tombol-----

const int tblAtas = 25, tblKi = 31, tblKa = 29, tblTgh = 35,
tblBawah = 27;

int pilihan = 1;

//-----Temp-----

int chk;
```

```
float temp;

float T=0;

int kode=0;

int hari,bulan,tahun,jam,menit,detik;

//-----

int lastSetMenit;

float rataTemp,rataOutPID,rataError,rataDError;

//-----MQ-----

int sampel=0;

const int tSampel =20;

int adc1[tSampel],adc2[tSampel],adc3[tSampel];

long tAdc1,tAdc2,tAdc3;

float RS = 0,
      RL = 10,
      RO = 75,
      ratio = 0,
      ppm1 = 1000,
      ppm2 = 1000,
      ppm3 = 1000,
      nilaiVL = 0;

double nilaiTotal=0,rAdc1,rAdc2,rAdc3;

float ratappm;

//-----pwm-----

int vKipas1=0,vKipas2=0;

float pwm_IN, pwm_OUT;
```

```
int vServo=0;;

float pwm;

int pwm1,pwm2;

//-----PID-----

double error = 0,

    kp    = 0.1,

    ki    = 0.1,

    kd    = 0.1,

    out   = 0.1,

    lError = 0,

    I     = 0;

//-----kipas-----

const int K2 = 4;

const int K1 = 3;

const int heat=43;

//-----fuzzy-----

double tSuhu = 30;

const int fis_gcI = 3; //input

const int fis_gcO = 1; //output

const int fis_gcR = 27; //rules

FIS_TYPE g_fisInput[fis_gcI];

FIS_TYPE g_fisOutput[fis_gcO];

//-----

void bacaMq1() {

    tAdc1      -= adc1[sampel];

    adc1[sampel] = nilaiAdc1;//map(analogRead(A8),0,1023,0,200);
```

```
tAdc1      += adc1[sampel];

rAdc1= tAdc1/tSampel;

nilaiVL = (rAdc1*5)/1023;
RS      = (5*RL/nilaiVL)-RL;
ratio   = RS/RO;
ppm1    = 10-(13*(2.718^(0.0422*nilaiVL)));
ppm1b   = 10796*(nilaiVL^2)-(20+(7*nilaiVL)+20);
}

void bacaMq2() {
    tAdc2      -= adc2[sampel];
    adc2[sampel]= nilaiAdc2;//map(analogRead(A9),0,1023,0,130);
    tAdc2      += adc2[sampel];
    rAdc2      = tAdc2/tSampel;
    nilaiVL    = (rAdc2*5)/1023;
    RS        = (5*RL/nilaiVL)-RL;
    ratio      = RS/RO;
    ppm2      = 0.0053*(nilaiVL^2.0396);
}

void bacaMq3() {
    tAdc3      -= adc3[sampel];
    adc3[sampel] = nilaiAdc3;//map(analogRead(A10),0,1023,0,230);
    tAdc3      += adc3[sampel];
    rAdc3      = tAdc3/tSampel;
    nilaiVL    = (rAdc3*5)/1023;
```

```
RS      = (5*RL/nilaiVL)-RL;
ratio   = RS/RO;
ppm3    = 45.504*(2.718^(0.0058*nilaiVL));
ppm3b   = 0.6742*(nilaiVL^2)-(1086.2*nilaiVL+442863);
}

void bacaDht(){
    T = dht.readTemperature();
    temp= (1.043*T) - 1.809;
}

void cekTombol(){
    if      (digitalRead(tblAtas  ) == HIGH) pilihan = 0;
    else if (digitalRead(tblKa    ) == HIGH) pilihan = 1;
    else if (digitalRead(tblTgh   ) == HIGH) pilihan = 2;
    else if (digitalRead(tblKi    ) == HIGH) pilihan = 3;
    else if (digitalRead(tblBawah ) == HIGH) pilihan = 4;
}

void tampilLCD(){
    lcd.setCursor(0,0);lcd.print("                ");
    lcd.setCursor(0,1);lcd.print("                ");

    if      (pilihan == 0){
        lcd.setCursor(0,0);lcd.print(" [MQ-5]  [MQ-8] ");
        lcdMQ-5();
        lcd.setCursor(9,1);lcd.print(ppm2,2);
    }

    else if (pilihan==1){
```

```
lcd.setCursor(0,0);lcd.print("T:");
lcd.setCursor(3,0);lcd.print(temp,1);
lcd.setCursor(8,0);lcd.print("Ts:");
lcd.setCursor(12,0);lcd.print(tSuhu,1);
lcd.setCursor(0,1);lcd.print("[K1]      [K2]   ");

if (out<0) {
  lcd.setCursor(5,1);lcd.print(abs(out),0);
  lcd.setCursor(13,1);lcd.print("0");}

else {
  lcd.setCursor(5,1);lcd.print("0");
  lcd.setCursor(13,1);lcd.print(out,0);}
}

else if (pilihan == 2){

  lcd.setCursor(0,0);lcd.print("JAM :
"+String(jam)+"."+String(menit)+"."+String(detik));

  lcd.setCursor(0,1);lcd.print("TGL : "+String(hari)+"-
"+String(bulan)+"-"+String(tahun));

}

else if (pilihan==3){

  lcdFuzzy();

  lcd.setCursor(0,0);lcd.print(" KONDISI OUTPUT ");

}

else if (pilihan==4){

  lcdMQ-7();

  lcd.setCursor(0,0);lcd.print("      [MQ-7]      ");

}

}

void lcdMQ-5(){

  if ((g_fisInput[0]>=0)&&(g_fisInput[0]<=1010.64)){

    lcd.setCursor(1,1);
```

```
        lcd.print(ppm1,2);
    }
    else if ((g_fisInput[0]>=1012.8)&&(g_fisInput[0]<=1016.8)){
        lcd.setCursor(1,1);
        lcd.print(ppm1b,2);
    }
}
void lcdMQ-7(){
    if ((g_fisInput[2]>=0)&&(g_fisInput[2]<=719.56)){
        lcd.setCursor(5,1);
        lcd.print(ppm3,2);
    }
    else if ((g_fisInput[2]>=813.71)&&( g_fisInput[2]<=933.13)){
        lcd.setCursor(5,1);
        lcd.print(ppm3b,2);
    }
}
void lcdFuzzy(){
    if ((g_fisOutput[0]>=0)&&(g_fisOutput[0]<=33.2)){
        lcd.setCursor(0,1);
        lcd.print("    KERING    ");
    }
    else if ((g_fisOutput[0]>=33.3)&&(g_fisOutput[0]<=40.7)){
        lcd.setCursor(0,1);
        lcd.print("  AGAK  KERING  ");
    }
}
```

```
else if ((g_fisOutput[0]>=40.8)&&(g_fisOutput[0]<=47.2)){
    lcd.setCursor(0,1);
    lcd.print("    SEDANG    ");
}
else if ((g_fisOutput[0]>=47.3) &&(g_fisOutput[0]<=56.8)){
    lcd.setCursor(0,1);
    lcd.print("    AGAK BASAH    ");
}
else if ((g_fisOutput[0]>=56.9) &&(g_fisOutput[0]<=70)) {
    lcd.setCursor(0,1);
    lcd.print("    BASAH.    ");
}
}
}
void sServo(){
    vServo = map((int)pwm, 0, 255, 0, 180);
    myservo.write(vServo);
    delay(15);
    if (vServo>25){digitalWrite (heat,HIGH);}
    else {digitalWrite (heat,LOW);}
}
void output(){
    if (error <= 0 ){digitalWrite(heat,HIGH);
    analogWrite(K1,abs(out));analogWrite(K2,0);}

    else if( error > 2 ) {digitalWrite(heat,LOW);
    analogWrite(K1,0);analogWrite(K2,out);}

    else {digitalWrite(heat,LOW);
    analogWrite(K1,0);analogWrite(K2,0);}
```



```
}  
  
void pid(){  
    error = temp - tSuhu;  
    if (error!=0){I=I+error;}  
    else{I=0;}  
  
    out=(int) ((out+(error*kp))+((error-lError)*kd)+((ki*I)/35));  
    lError=error;  
  
    if(out<-255)out=-255;  
    if(out>255)out=255;  
}  
  
void Fuzzy(){  
    g_fisInput[0]   = rAdc1;  
    g_fisInput[1]   = rAdc2;  
    g_fisInput[2]   = rAdc3;  
    g_fisOutput[0]  = 0;  
    fis_evaluate();  
    tSuhu           = g_fisOutput[0];  
}  
  
void bacaSensor(){  
    bacaMq1();  
    bacaMq2();  
    bacaMq3();  
    bacaDht();
```

```
sampel++;

if (sampel >= tSampel) sampel = 0;

}

void dataLogPenting() {

    DateTime now = rtc.now();

    hari = now.day();
    bulan = now.month();
    tahun = now.year();
    jam = now.hour();
    menit = now.minute();
    detik = now.second();

    if (menit % 5 == 0) {

        if (kode == 0) {

            Serial.println("SIMPAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAn");

            bacaDht();

            myFile = SD.open("Log31.txt", FILE_WRITE);

            if (myFile) {

                myFile.print( g_fisInput[0]);
                myFile.print("\t");
                myFile.print( g_fisInput[1]);
                myFile.print("\t");
                myFile.print( g_fisInput[2]);
                myFile.print("\t");
                myFile.print( g_fisOutput[0]);
```

```
myFile.print("\t");
myFile.print(temp);
myFile.print("\t");
myFile.print(tSuhu);
myFile.print("\t");
myFile.print(String(hari)+"-"+String(bulan)+"-"+String(tahun));
myFile.print("\t");
myFile.print(String(jam)+"."+String(menit)+"."+String(detik));
myFile.print("\t");
myFile.println(error);
myFile.close();
kode=1;
}
else {Serial.println("error buka SD Chard Log1");}
}
}
else{
kode=0;
}
}

void setup() {
Serial.begin(9600);
rtc.begin();
if (! rtc.begin()) { Serial.println("Couldn't find RTC"); while
(1);}
```

```
if (! rtc.isrunning()) {
    Serial.println("RTC lost power, lets set the time!");
    rtc.adjust(DateTime(F(__DATE__), 0,0,0));
}

pinMode(pinCS, OUTPUT);
if (SD.begin()){Serial.println("SD card bisan digunakan");}
else{ Serial.println("SD card initialization failed"); return;}
myFile = SD.open("Log31.txt", FILE_WRITE);
myFile.println(" //////////baru////////////////////////////////");
myFile.close();
Wire.begin();
dht.begin();
lcd.begin(16,2);
pinMode(heat,OUTPUT); pinMode(K1,OUTPUT);   pinMode(K2,OUTPUT);
pinMode(A8 , INPUT);  pinMode(A9 , INPUT);  pinMode(A10 , INPUT);
pinMode(tblAtas,INPUT_PULLUP);pinMode(tblBawah,INPUT_PULLUP);
pinMode(tblKi,INPUT_PULLUP);pinMode(tblKa,INPUT_PULLUP);
pinMode(tblTgh,INPUT_PULLUP);
}
void loop() {
    bacaSensor();
    if(sampel==tSampel-1){
        Serial.println("-----");
        Fuzzy();
    }
}
```

```
pid();

output();

cekTombol();

tampilLCD();

dataLogPenting();

Serial.println(String(rAdc1)+" adc, "+String(rAdc2)+" adc,
"+String(rAdc3)+" adc, "+String(temp)+" C, "+String(tSuhu)+" C,
"+String(error)+" error "+String(out));

delay(100);
}

//Fuzzy

FIS_TYPE fis_trapmf(FIS_TYPE x, FIS_TYPE* p){
    FIS_TYPE a = p[0], b = p[1], c = p[2], d = p[3];

    FIS_TYPE t1 = ((x <= c) ? 1 : ((d < x) ? 0 : ((c != d) ? ((d -
x) / (d - c)) : 0)));

    FIS_TYPE t2 = ((b <= x) ? 1 : ((x < a) ? 0 : ((a != b) ? ((x -
a) / (b - a)) : 0)));

    return (FIS_TYPE) min(t1, t2);
}

FIS_TYPE fis_trimf(FIS_TYPE x, FIS_TYPE* p){
    FIS_TYPE a = p[0], b = p[1], c = p[2];

    FIS_TYPE t1 = (x - a) / (b - a);

    FIS_TYPE t2 = (c - x) / (c - b);

    if ((a == b) && (b == c)) return (FIS_TYPE) (x == a);

    if (a == b) return (FIS_TYPE) (t2*(b <= x)*(x <= c));

    if (b == c) return (FIS_TYPE) (t1*(a <= x)*(x <= b));

    t1 = min(t1, t2);
```

```
    return (FIS_TYPE) max(t1, 0);
}

FIS_TYPE fis_min(FIS_TYPE a, FIS_TYPE b){
    return min(a, b);
}

FIS_TYPE fis_max(FIS_TYPE a, FIS_TYPE b){
    return max(a, b);
}

FIS_TYPE fis_array_operation(FIS_TYPE *array, int size, _FIS_ARR_OP
pfnOp){
    int i;
    FIS_TYPE ret = 0;

    if (size == 0) return ret;
    if (size == 1) return array[0];

    ret = array[0];
    for (i = 1; i < size; i++)
    {
        ret = (*pfnOp)(ret, array[i]);
    }

    return ret;
}

//*****
*****

_FIS_MF fis_gMF[] ={
```

```
    fis_trapmf, fis_trimf
};

int fis_gIMFCount[] = { 3, 3, 3 };
int fis_gOMFCount[] = { 5 };
FIS_TYPE fis_gMFI0Coeff1[] = { 0, 0, 29, 39 };
FIS_TYPE fis_gMFI0Coeff2[] = { 30, 45, 60 };
FIS_TYPE fis_gMFI0Coeff3[] = { 52, 65, 80, 80 };
FIS_TYPE* fis_gMFI0Coeff[] = { fis_gMFI0Coeff1, fis_gMFI0Coeff2,
    fis_gMFI0Coeff3 };
FIS_TYPE fis_gMFI1Coeff1[] = { 0, 0, 21.54, 33.85 };
FIS_TYPE fis_gMFI1Coeff2[] = { 27.69, 43.08, 58.46 };
FIS_TYPE fis_gMFI1Coeff3[] = { 52.31, 64.62, 80, 80 };
FIS_TYPE* fis_gMFI1Coeff[] = { fis_gMFI1Coeff1, fis_gMFI1Coeff2,
    fis_gMFI1Coeff3 };
FIS_TYPE fis_gMFI2Coeff1[] = { 0, 0, 31.58, 55.26 };
FIS_TYPE fis_gMFI2Coeff2[] = { 31.58, 75, 118.4 };
FIS_TYPE fis_gMFI2Coeff3[] = { 94.74, 118.4, 150, 150 };
FIS_TYPE* fis_gMFI2Coeff[] = { fis_gMFI2Coeff1, fis_gMFI2Coeff2,
    fis_gMFI2Coeff3 };
FIS_TYPE** fis_gMFICoeff[] = { fis_gMFI0Coeff, fis_gMFI1Coeff,
    fis_gMFI2Coeff };

// Coefficients for the Output Member Functions
FIS_TYPE fis_gMFO0Coeff1[] = { 20, 20, 27.5, 35 };
FIS_TYPE fis_gMFO0Coeff2[] = { 42.45, 48.63, 55.51 };
FIS_TYPE fis_gMFO0Coeff3[] = { 50, 57.5, 65, 65 };
FIS_TYPE fis_gMFO0Coeff4[] = { 28.19, 34.5, 40.96 };
FIS_TYPE fis_gMFO0Coeff5[] = { 35.04, 41.23, 47.91 };
```

```
FIS_TYPE* fis_gMFO0Coeff[] = { fis_gMFO0Coeff1, fis_gMFO0Coeff2,
fis_gMFO0Coeff3, fis_gMFO0Coeff4, fis_gMFO0Coeff5 };

FIS_TYPE** fis_gMFOCoeff[] = { fis_gMFO0Coeff };

// Input membership function set

int fis_gMFI0[] = { 0, 1, 0 };
int fis_gMFI1[] = { 0, 1, 0 };
int fis_gMFI2[] = { 0, 1, 0 };
int* fis_gMFI[] = { fis_gMFI0, fis_gMFI1, fis_gMFI2};

// Output membership function set

int fis_gMFO0[] = { 0, 1, 0, 1, 1 };
int* fis_gMFO[] = { fis_gMFO0};

// Rule Weights

FIS_TYPE fis_grWeight[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 };

// Rule Type

int fis_gRType[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1 };

// Rule Inputs

int fis_gRI0[] = { 1, 1, 1 };
int fis_gRI1[] = { 1, 1, 2 };
int fis_gRI2[] = { 1, 1, 3 };
int fis_gRI3[] = { 1, 2, 1 };
```



```
int fis_gRI4[] = { 1, 2, 2 };
int fis_gRI5[] = { 1, 2, 3 };
int fis_gRI6[] = { 1, 3, 1 };
int fis_gRI7[] = { 1, 3, 2 };
int fis_gRI8[] = { 1, 3, 3 };
int fis_gRI9[] = { 2, 1, 1 };
int fis_gRI10[] = { 2, 1, 2 };
int fis_gRI11[] = { 2, 1, 3 };
int fis_gRI12[] = { 2, 2, 1 };
int fis_gRI13[] = { 2, 2, 2 };
int fis_gRI14[] = { 2, 2, 3 };
int fis_gRI15[] = { 2, 3, 1 };
int fis_gRI16[] = { 2, 3, 2 };
int fis_gRI17[] = { 2, 3, 3 };
int fis_gRI18[] = { 3, 1, 1 };
int fis_gRI19[] = { 3, 1, 2 };
int fis_gRI20[] = { 3, 1, 3 };
int fis_gRI21[] = { 3, 2, 1 };
int fis_gRI22[] = { 3, 2, 2 };
int fis_gRI23[] = { 3, 2, 3 };
int fis_gRI24[] = { 3, 3, 1 };
int fis_gRI25[] = { 3, 3, 2 };
int fis_gRI26[] = { 3, 3, 3 };

int* fis_gRI[] = { fis_gRI0, fis_gRI1, fis_gRI2, fis_gRI3, fis_gRI4,
fis_gRI5, fis_gRI6, fis_gRI7, fis_gRI8, fis_gRI9, fis_gRI10,
fis_gRI11, fis_gRI12, fis_gRI13, fis_gRI14, fis_gRI15, fis_gRI16,
fis_gRI17, fis_gRI18, fis_gRI19, fis_gRI20, fis_gRI21, fis_gRI22,
fis_gRI23, fis_gRI24, fis_gRI25, fis_gRI26 };
```

```
// Rule Outputs  
int fis_gRO0[] = { 1 };  
int fis_gRO1[] = { 4 };  
int fis_gRO2[] = { 5 };  
int fis_gRO3[] = { 5 };  
int fis_gRO4[] = { 2 };  
int fis_gRO5[] = { 2 };  
int fis_gRO6[] = { 5 };  
int fis_gRO7[] = { 2 };  
int fis_gRO8[] = { 3 };  
int fis_gRO9[] = { 4 };  
int fis_gRO10[] = { 5 };  
int fis_gRO11[] = { 2 };  
int fis_gRO12[] = { 5 };  
int fis_gRO13[] = { 5 };  
int fis_gRO14[] = { 2 };  
int fis_gRO15[] = { 2 };  
int fis_gRO16[] = { 2 };  
int fis_gRO17[] = { 3 };  
int fis_gRO18[] = { 4 };  
int fis_gRO19[] = { 5 };  
int fis_gRO20[] = { 2 };  
int fis_gRO21[] = { 5 };  
int fis_gRO22[] = { 2 };  
int fis_gRO23[] = { 3 };
```

```
int fis_gRO24[] = { 2 };

int fis_gRO25[] = { 2 };

int fis_gRO26[] = { 3 };

int* fis_gRO[] = { fis_gRO0, fis_gRO1, fis_gRO2, fis_gRO3, fis_gRO4,
fis_gRO5, fis_gRO6, fis_gRO7, fis_gRO8, fis_gRO9, fis_gRO10,
fis_gRO11, fis_gRO12, fis_gRO13, fis_gRO14, fis_gRO15, fis_gRO16,
fis_gRO17, fis_gRO18, fis_gRO19, fis_gRO20, fis_gRO21, fis_gRO22,
fis_gRO23, fis_gRO24, fis_gRO25, fis_gRO26 };

// Input range Min
FIS_TYPE fis_gIMin[] = { 0, 0, 0 };

// Input range Max
FIS_TYPE fis_gIMax[] = { 80, 80, 150 };

// Output range Min
FIS_TYPE fis_gOMin[] = { 0 };

// Output range Max
FIS_TYPE fis_gOMax[] = { 65 };

//*****
*****

FIS_TYPE fis_MF_out(FIS_TYPE** fuzzyRuleSet, FIS_TYPE x, int o){
    FIS_TYPE mfOut;
    int r;

    for (r = 0; r < fis_gCR; ++r)
    {
        int index = fis_gRO[r][o];
        if (index > 0)
        {
```

```
        index = index - 1;

        mfOut = (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOcoeff[o][index]);
    }
    else if (index < 0)
    {
        index = -index - 1;
        mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOcoeff[o][index]);
    }
    else
    {
        mfOut = 0;
    }

    fuzzyRuleSet[0][r] = fis_min(mfOut, fuzzyRuleSet[1][r]);
}
return fis_array_operation(fuzzyRuleSet[0], fis_gcR, fis_max);
}

FIS_TYPE fis_defuzz_centroid(FIS_TYPE** fuzzyRuleSet, int o){
    FIS_TYPE step = (fis_gOMax[o] - fis_gOMin[o]) / (FIS_RESOLUTION
- 1);
    FIS_TYPE area = 0;
    FIS_TYPE momentum = 0;
    FIS_TYPE dist, slice;

    int i;
```

```
// calculate the area under the curve formed by the MF outputs
for (i = 0; i < FIS_RESOLUTION; ++i){
    dist = fis_gOMin[o] + (step * i);
    slice = step * fis_MF_out(fuzzyRuleSet, dist, o);
    area += slice;
    momentum += slice*dist;
}

return ((area == 0) ? ((fis_gOMax[o] + fis_gOMin[o]) / 2) :
(momentum / area));
}

//*****
*****

void fis_evaluate(){
    FIS_TYPE fuzzyInput0[] = { 0, 0, 0 };
    FIS_TYPE fuzzyInput1[] = { 0, 0, 0 };
    FIS_TYPE fuzzyInput2[] = { 0, 0, 0 };

    FIS_TYPE* fuzzyInput[fis_gcI] = { fuzzyInput0, fuzzyInput1,
fuzzyInput2, };

    FIS_TYPE fuzzyOutput0[] = { 0, 0, 0, 0, 0 };
    FIS_TYPE* fuzzyOutput[fis_gcO] = { fuzzyOutput0, };
    FIS_TYPE fuzzyRules[fis_gcR] = { 0 };
    FIS_TYPE fuzzyFires[fis_gcR] = { 0 };
    FIS_TYPE* fuzzyRuleSet[] = { fuzzyRules, fuzzyFires };
    FIS_TYPE sW = 0;

    // Transforming input to fuzzy Input
```

```
int i, j, r, o;
for (i = 0; i < fis_gcI; ++i)
{
    for (j = 0; j < fis_gIMFCount[i]; ++j)
    {
        fuzzyInput[i][j] =
            (fis_gMF[fis_gMFI[i][j]])(g_fisInput[i],
fis_gMFICoeff[i][j]);
    }
}


int index = 0;
for (r = 0; r < fis_gcR; ++r)
{
    if (fis_gRType[r] == 1)
    {
        fuzzyFires[r] = FIS_MAX;
        for (i = 0; i < fis_gcI; ++i)
        {
            index = fis_gRI[r][i];
            if (index > 0)
                fuzzyFires[r] = fis_min(fuzzyFires[r],
fuzzyInput[i][index - 1]);
            else if (index < 0)
                fuzzyFires[r] = fis_min(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
            else
```

```
        fuzzyFires[r] = fis_min(fuzzyFires[r], 1);
    }
}
else
{
    fuzzyFires[r] = FIS_MIN;
    for (i = 0; i < fis_gcI; ++i)
    {
        index = fis_gRI[r][i];
        if (index > 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r],
fuzzyInput[i][index - 1]);
        else if (index < 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
        else
            fuzzyFires[r] = fis_max(fuzzyFires[r], 0);
    }
}

fuzzyFires[r] = fis_gRWeight[r] * fuzzyFires[r];
sW += fuzzyFires[r];
}

if (sW == 0)
{
    for (o = 0; o < fis_gcO; ++o)
```

```
    {  
        g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) / 2);  
    }  
}  
else  
{  
    for (o = 0; o < fis_gc0; ++o)  
    {  
        g_fisOutput[o] = fis_defuzz_centroid(fuzzyRuleSet, o);  
    }  
}  
}
```

A large, semi-transparent watermark of the Universitas Jember logo is centered on the page. The logo is a shield-shaped emblem with a yellow background and a grey border. It features a stylized green and red plant in the center, with the words 'UNIVERSITAS' and 'JEMBER' written in grey capital letters above and below the plant, respectively.

LAMPIRAN B. Gambar Pengeringan Ikan



Proses pengeringan ikan dengan menggunakan alat pengering ikan



Proses pengeringan ikan dengan cara konvensional



Jumlah berat ikan pada saat basah



Jumlah berat ikan pada saat kering



Hasil pengeringan ikan dengan menggunakan alat pengering ikan



Hasil pengeringan ikan dengan cara konvensional