

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Studi Literatur

Kebutuhan pengendalian sistem untuk mendapatkan respon sistem yang baik sangat banyak diperlukan, maka dari itu peneliti dewasa ini telah banyak melakukan penelitian untuk mendapatkan pengendali yang kokoh, baik penelitian dengan menggunakan pengendali ON-OFF, maupun dengan pengendali konvensional dengan pengambilan keputusan lebih baik. Pengendali ON-OFF memiliki kelemahan, karena pengambilan keputusan tegas, sehingga tidak dapat mengendalikan pada *set point* tertentu. Pengendali ON-OFF telah diterapkan pada kendali suhu cairan berbasis Mikrokontroler AT90S8535 (Muchlas dkk, 2006), *set point* yang dikendalikan yaitu 27 °C – 99 °C, maka kendali ON-OFF akan mengambil keputusan jika suhu kurang dari 27 °C dan suhu lebih dari 99 °C sehingga jika ingin mengendalikan pada *set point* tertentu kendali ON-OFF tidak dapat digunakan.

Dengan kemajuan teknologi, maka peneliti dewasa telah beralih menggunakan pengendali yang dapat mengendalikan sistem lebih baik dan dapat menjaga nilai kestabilan *set point* tertentu, salah satu pengendali yang banyak digunakan saat ini diantaranya adalah PID. Pengendali PID ini telah banyak digunakan diantaranya adalah penelitian yang dilakukan Setyaningsih (2016), pengendali PID digunakan untuk mengendalikan suhu pada inkubator bayi dengan nilai *set point* 35 °C. Hasil penelitian terhadap pengendali PID dengan nilai parameter  $P = 0.00648020037858366$ ,  $I = 0.0129604007571673$ , dan  $D = 0$ , terbukti dapat mengendalikan *set point* dengan waktu 24 detik untuk mencapai kondisi *steady state* dan nilai presentasi *overshoot* pada sistem 8,08%.

Kemudian pengendali PID juga digunakan Yuliarmas dkk ( 2015), dengan mengimplementasikan pengendali PID pada mesin pengembang roti. parameter yang dikendalikan yaitu suhu dan kelembaban. Pengendali PID diaplikasikan pada dua Arduino Uno, yang pertama Arduino Uno berfungsi sebagai master tugasnya mengolah semua data yang masuk dan yang kedua Arduino Uno *slave* berfungsi mengirimkan data dari sensor ke Arduino Uno master. Untuk memenuhi kebutuhan suhu pada alat pengembang roti digunakan 4 buah lampu pijar 60 watt sebagai sumber panas dengan rangkaian AC *Dimmer* yang berfungsi untuk mengatur tingkat intensitas cahaya penerangan lampu pijar. Kemudian untuk mengurangi suhu ruang pada sistem digunakan kipas 12 VDC dan sensor yang digunakan adalah DHT 11 yang berfungsi sebagai deteksi suhu dan kelembaban. Dari

hasil pengujian pengendali PID didapatkan respon waktu tercepat dengan *set point* 35 °C pada detik 120, dengan nilai parameter  $K_p=20$  ,  $K_i = 0,01$  dan  $K_d = 1$ , dari hasil pengujian keseluruhan sistem, parameter kelembaban pada alat pengembang roti belum dapat terpenuhi.

Pengendali PID pada dasarnya sudah memiliki peformasi yang baik untuk menjaga kestabilan *set point*, tetapi pengendali PID masih memiliki beberapa kekurangan diantaranya presentasi *overshoot* besar dan karakteristiknya saling mempengaruhi antara P, I dan D sehingga kurang stabil apabila semakin banyak masukan data (Maerani dan bakhri,2013). Mengatasi permasalahan tersebut, maka diperlukan pengendali yang memiliki respon sistem yang stabil, sensitif terhadap gangguan dan tidak memerlukan permasalahan model matematika yang rumit, Pengendali yang memiliki karakteristik tersebut diantaranya adalah logika Fuzzy.

Penerapan kendali logika Fuzzy ini dibuktikan oleh Vionita dkk (2015), pengendali logika Fuzzy digunakan untuk mengendalikan inkubator fermentasi yoghurt, pengendali logika Fuzzy diaplikasikan pada Arduino Uno dengan *set point* 45 °C. Arduino Uno digunakan sebagai pengendali pemanas, kipas dan masukan data dari sensor DS18B20. Untuk menjaga agar sistem tetap pada *set point* digunakan perhitungan *error* dan *Derror* sesuai dengan hasil pembacaan sensor DS18B20, selanjutnya untuk respon keluaran sistem digunakan nilai *duty cycle* yang dihitung dengan menggunakan logika *fuzzy*, yang diaplikasikan pada respon keluaran *heater* dan kipas. Pengujian terhadap pengendali logika Fuzzy selama 7 jam, dengan pengambilan data selama 10 menit dengan *set point* 45 °C , hasilnya adalah untuk nilai *overshoot* 0,5 %, dan nilai *error steady state* 0,133% dengan nilai PWM 76 untuk mempertahankan suhu agar tetap stabil.

Rahman (2009), pada penelitiannya menggunakan pengendali logika Fuzzy yang diaplikasikan pada mikrokontroler ATMEGA 16 untuk mengendalikan kelembaban ruangan. Adapun parameter yang dikendalikan yaitu 80%, untuk mengetahui peformasi pengendali logika Fuzzy maka dilakukan pengujian respon sistem terhadap variasi *set point* kelembaban. Dari hasil pengujian respon sistem cukup baik untuk menjaga nilai *set point* baik dari respon keluaran kelembaban dan penambahan jumlah partikel. Hasil pengujian 1 didapatkan kestabilan sistem kendali 78 detik untuk mencapai *set point* 80% dan 68 detik untuk mencapai *set point* 88%, sedangkan kesamaan nilai sinyal keluaran rata-rata 99,58%.

Selain itu pengendali logika Fuzzy juga digunakan Afroni (2014), untuk mengendalikan inkubator telur, adapun parameter yang dikendalikan yaitu suhu dan kelembaban. Pengendali logika Fuzzy diaplikasikan pada mikrokontroler ATMEGA 8535 sebagai pengendali *heater* dan kipas, untuk *input* data suhu ruang inkubator menggunakan sensor SHT 11, *set point* yang akan dikendalikan yaitu untuk suhu pada *range* 36-40°C dan kelembaban 65-70%. Dari hasil pengujian kendali logika Fuzzy mampu mencapai *set point* yang diberikan dan dapat mempertahankan nilai kestabilan sistem.

Putra dkk (2013), mengimplementasikan pengndali logika Fuzzy untuk mengendalikan suhu pada tungku bakar. Pengendalian suhu pada tungku bakar menggunakan *valve* yang diputar menggunakan motor servo dengan mengatur sudut pembukaan agar sesuai dengan perintah mikrokontroler. Untuk mengetahui suhu ruang pada tungku bakar menggunakan sensor PT100. Pengendai logika Fuzzy diaplikasikan pada Arduino Uno dengan masukan *error* dan *derror*, dengan demikian suhu ruang pada tungku bakar dapat dikendalikan dengan baik, dari hasil pengujian respon yang dihasilkan adalah presentasi *error* secara keseluruhan sistem yaitu 1.147579%.

Berdasarkan kajian pustaka di atas, ternyata belum ada studi tentang mengimplementasikan logika Fuzzy pada mesin pengembang roti. Dengan menggunakan pengendali logika Fuzzy diharapkan mampu untuk mengendalikan suhu dan kelembaban dengan mendapatkan hasil respon sistem yang baik dan memperkecil nilai *error steady state* kecil selama proses berlangsung.

Pada penelitian ini penulis menggunakan kendali logika Fuzzy sebagai kendali suhu dan kelembaban pada ruang *proofing* roti. Pengendali Fuzzy diimplementasikan pada Arduino. Untuk kendali suhu menggunakan Arduino Mega 2560 dengan menggunakan 3 sensor yang diletakkan pada bagian bawah, tengah dan atas alat *proofing* roti sebagai *input* Arduino Mega 2560, untuk *output* yang berfungsi sebagai kendali *heater* elemen pemanas digunakan rangkaian AC *Dimmer* yang berfungsi sebagai pengendali tegangan masukan *heater* dengan mengendalikan PWM. Kemudian kendali kelembaban menggunakan Arduino Uno sensor yang digunakan yaitu DHT 11, untuk memenuhi kelembaban pada alat *proofing* roti digunakan penghasil kabut uap, kemudian kipas berfungsi untuk mengendalikan kelembaban pada ruang *proofing* roti putaran kipas dikendalikan dengan *driver* L293D supaya putaran kecepatan kipas dapat dikendalikan menggunakan PWM.

## 2.2. Mesin Pengembang Roti

Mesin *proofer* merupakan alat praktis yang dirancang khusus untuk membantu dalam melakukan proses *proofing* makanan, jenis-jenis makanan atau kue yang memerlukan proses *proofing* yaitu donat, bakpao dan lain sebagainya. Proses pengembangan adonan roti (*proofing*) diperlukan adanya panas dan kelembaban yang stabil untuk mendapatkan hasil roti yang seragam, dengan pengaturan temperatur dan kelembaban yang sesuai akan menghasilkan hasil roti yang seragam dan mempercepat proses *proofing*. Pembuatan alat *proofing* roti sangat membantu untuk menjaga kualitas produk dan mempercepat proses pembuatan roti (wiratech, 2016).

## 2.3. Komponen Terkait

### 2.3.1. LCD (*Liquid Cristal Display*)

LCD adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis *display* elektronika yang dibuat dengan teknologi CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari *back-lit*. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam karakter, huruf, angka maupun grafik (elektronika-dasar, 2017).

LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan *seven-segment* dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan *sadwich* memiliki polarizer cahaya vertikal depan dan polarizer cahaya horizontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segment yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan (elektronika-dasar, 2017).



Gambar 2.1. Contoh bentuk LCD ukuran 2 x 16

(Sumber : [www.lcd-modul.de](http://www.lcd-modul.de))

Dalam modul LCD (*Liquid Cristal Display*) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). Mikrokontroler pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan register. Memori yang digunakan mikrokontroler internal LCD adalah:

- a. DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan berada.
- b. CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.
- c. CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah dimana pola tersebut merupakan karakter dasar yang sudah ditemukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM (elektronika-dasar, 2017).

Register kontrol yang terdapat dalam suatu LCD diantaranya adalah :

- a. Register perintah yaitu yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) pada saat proses penulisan data atau tempat status dari panel LCD (*Liquid Cristal Display*) dapat dibaca pada saat pembacaan data.
- b. Register Data yaitu register untuk menuliskan atau membaca data dari atau ke DDRAM. Penulisan data pada register akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya (Rendy,2015).

Pin, kaki jalur *input* dan kontrol dalam suatu LCD (*Liquid Cristal Display*) diantaranya adalah :

- a. Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Cristal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit
- b. Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *LOW* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data.
- c. Pin R/W (*Read Write*) berfungsi sebagai instruksi pada modul jika *LOW* tulis data, sedangkan *high* baca data.
- d. Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.

- e. Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 K ohm, jika tidak digunakan dihubungkan ke *ground*, sedangkan tegangan catu daya ke LCD sebesar 5 Volt (elektronika-dasar, 2017).

### 2.3.2. Heater

*Electrical Heating Elemen* (elemen pemanas listrik) banyak dipakai dalam kehidupan sehari-hari ataupun peralatan mesin industri. Bentuk dan tipe dari *Electrical Heating Elemen* ini bermacam-macam disesuaikan dengan fungsi, tempat pemasangan dan media yang akan dipanaskan.

Panas yang dihasilkan oleh elemen pemanas listrik ini bersumber dari kawat ataupun pita bertahanan listrik tinggi (*Resistance Wire*) yang dialiri arus listrik pada kedua ujungnya dan dilapisi oleh isolator listrik yang mampu meneruskan panas dengan baik hingga aman jika digunakan.

Ada 2 macam jenis utama pada elemen pemanas listrik yaitu:

1. Elemen pemanas listrik bentuk dasar yaitu elemen pemanas dimana *Resistance Wire* hanya dilapisi oleh isolator listrik, macem-macam elemen pemanas bentuk ini adalah *Ceramik Heater, Quart Heater, Bank Channel Heater, Black Body Keramik Heater*.
2. Elemen pemanas listrik bentuk lanjut merupakan elemen pemanas dari bentuk dasar yang dilapisi oleh pipa atau lembaran plat logam untuk maksud sebagai penyesuaian terhadap pengguna dari elemen pemanas tersebut. Bahan logam yang bisa digunakan adalah : *mild stell, stainless stell*, tembaga dan kuningan. *Heater* yang termasuk dalam jenis ini adalah *Tubular Heater, Catridge Heater, Band, Nozzle* dan *stripe Heater*.

### 2.3.3. Evaporator

*Evaporator* adalah sebuah alat berfungsi mengubah sebagian atau keseluruhan sebuah pelarut dari sebuah larutan dari bentuk cair menjadi uap. *Evaporator* mempunyai dua prinsip dasar, untuk menukar panas dan untuk memisahkan uap yang terbentuk dari cairan. *Evaporator* umumnya terdiri dari tiga bagian, yaitu penukaran panas, bagian evaporasi dan pemisah untuk memisahkan uap dari cairan lalu dimasukkan ke dalam *kondensor*.

Fungsi dari *evaporator* adalah membuang panas yang tidak diinginkan dari benda melalui cairan pendingin. Cairan *refrigerant* yang terkandung dalam *evaporator* mendidih pada tekanan rendah (wikipedia,2017).

Tingkat tekanan ini ditentukan oleh dua faktor, diantaranya sebagai berikut:

- a. Tingkat dimana panas yang diserap dari benda ke cairan pendingin dievaporator.
- b. Tingkat dimana gas tekanan rendah akan dihisap dari *evaporator* ke kompresor.

### 2.3.4. Mikrokontroler

#### 2.3.4.1. Definisi Mikrokontroler

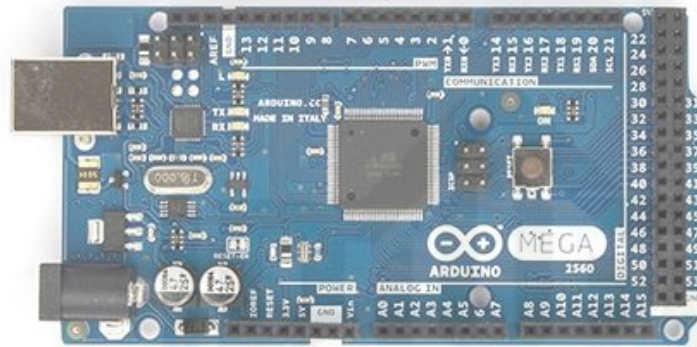
Mikrokontroler merupakan sebuah IC yang di dalamnya terdapat mikroprosesor dan memori program *Read only Memory* (ROM) serta memori *Random Acces Memory* (RAM), bahkan ada beberapa jenis mikrokontroler yang memiliki fasilitas ADC, TLL, EEPROM dalam satu kemasan. Ada perbedaan yang cukup penting antara Mikroprosesor dan Mikrokontroler. Jika Mikroprosesor merupakan CPU (*Central Processing Unit*) tanpa memori dan I/O pendukung dari sebuah komputer, maka Mikrokontroler umumnya terdiri dari CPU, Memori, I/O tertentu dan unit pendukung, misalnya *Analog to Digital Converter*(ADC) yang sudah terintegrasi di dalam mikrokontroler tersebut. Dengan kata lain, mikrokontroler adalah versi mini atau mikro dari sebuah komputer karena mikrokontroler sudah mengandung beberapa periferal yang langsung bisa dimanfaatkan, misalnya port paralel, port serial, komparator, konversi digital ke analog (DAC), konversi analog ke digital dan sebagainya hanya menggunakan sistem minimum yang tidak rumit atau kompleks. Mikrokontroler sering dimanfaatkan dalam kehidupan sehari-hari maupun dalam dunia industri sebagai otak dari sistem kontrol dan karena keunggulannya, antara lain (elektronika-dasar, 2017) :

- a. Ukurannya yang relatif kecil.
- b. Kecepatan pengoperasiannya tinggi.
- c. Handal.
- d. Kemampuan dan fleksibilitasnya lebih baik.

#### 2.3.4.2. Arduino Mega 2560

Arduino Mega 2560 adalah papan mikrokontroler berbasis ATmega2560). Arduino Mega2560 memiliki 54 pin digital *input/output*, dimana 15 pin dapat digunakan sebagai *output* PWM, 16 pin sebagai *input* analog, dan 4 pin sebagai UART (port serial hardware), 16 MHz kristal osilator, koneksi USB, *jack power*, header ICSP, dan tombol *reset*. Ini semua yang diperlukan untuk mendukung mikrokontroler. Cukup dengan

menghubungkannya ke komputer melalui kabel USB atau *power* dihubungkan dengan adaptor AC-DC atau baterai untuk mulai mengaktifkannya. Arduino Mega 2560 kompatibel dengan sebagian besar *shield* yang dirancang untuk Arduino *Duemilanove* atau Arduino *Diecimila*. Arduino Mega 2560 adalah versi terbaru yang menggantikan versi Arduino Mega (www.arduino.cc, 2017).



Gambar 2.2. Arduino Mega 2560  
(Sumber : www.arduino.cc)

### 2.3.4.3. Spesifikasi

Dibawah ini spesifikasi sederhana dari Arduino Mega 2560:

Tabel 2.1. Spesifikasi sederhana Arduino Mega 2560

Mikrokontroler	ATmega2560
Tegangan Operasi	5V
<i>Input Voltage</i> (disarankan)	7-12V
<i>Input Voltage</i> (limit)	6-20V
Pin Digital I/O	54 (yang 15 pin digunakan sebagai output PWM)
Pins <i>Input Analog</i>	16
Arus DC per pin I/O	40 Ma
Arus DC untuk pin 3.3V	50 Ma
Flash Memory	256 KB (8 KB digunakan untuk bootloader)
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

(Sumber : www.arduino.cc)



#### 2.3.4.4. Sumber Daya

Arduino Mega dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya eksternal (non-USB) dapat berasal baik dari adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan steker 2,1 mm yang bagian tengahnya terminal positif ke jack sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat langsung dihubungkan melalui header pin Gnd dan pin Vin dari konektor *power*.

Papan Arduino ATmega2560 dapat beroperasi dengan pasokan daya eksternal 6 Volt sampai 20 Volt. Jika diberi tegangan kurang dari 7 Volt, maka, pin 5 Volt mungkin akan menghasilkan tegangan kurang dari 5 Volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 Volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 Volt sampai 12 Volt (www.arduino.cc, 2017).

Pin tegangan yang tersedia pada papan Arduino adalah sebagai berikut:

- a. **VIN** : Adalah *input* tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal (sebagai 'saingan' tegangan 5 Volt dari koneksi USB atau sumber daya ter-regulator lainnya). Anda dapat memberikan tegangan melalui pin ini, atau jika memasok tegangan untuk papan melalui jack *power*, kita bisa mengakses/mengambil tegangan melalui pin ini.
- b. **5V** : Sebuah pin yang mengeluarkan tegangan ter-regulator 5 Volt, dari pin ini tegangan sudah diatur (ter-regulator) dari regulator yang tersedia (built-in) pada papan. Arduino dapat diaktifkan dengan sumber daya baik berasal dari jack *power* DC (7-12 Volt), konektor USB (5 Volt), atau pin VIN pada board (7-12 Volt). Memberikan tegangan melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator dapat merusak papan Arduino.
- c. **3V3** : Sebuah pin yang menghasilkan tegangan 3,3 Volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (on-board). Arus maksimum yang dihasilkan adalah 50 mA.
- d. **GND** : Pin Ground atau Massa.

#### 2.3.4.5. Memori

Arduino ATmega 2560 memiliki 256 KB flash *memory* untuk menyimpan kode (yang 8 KB digunakan untuk bootloader), 8 KB SRAM dan 4 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM) (www.arduino.cc, 2017).

### 2.3.4.6. Input Output

Masing-masing dari 54 digital pin pada Arduino Mega dapat digunakan sebagai *input* atau output, menggunakan fungsi `pinMode()` , `digitalWrite()` , dan `digitalRead()`. Arduino Mega beroperasi pada tegangan 5 volt. Setiap pin dapat memberikan atau menerima arus maksimum 40 mA dan memiliki resistor pull-up internal (yang terputus secara default) sebesar 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus, antara lain ([www.arduino.cc](http://www.arduino.cc), 2017) :

- **Serial** : 0 (RX) dan 1 (TX); **Serial 1** : 19 (RX) dan 18 (TX); **Serial 2** : 17 (RX) dan 16 (TX); **Serial 3** : 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data serial TTL. Pins 0 dan 1 juga terhubung ke pin chip ATmega16U2 Serial USB-to-TTL.
- **Eksternal Interupsi** : Pin 2 (*interrupt* 0), pin 3 (*interrupt* 1), pin 18 (*interrupt* 5), pin 19 (*interrupt* 4), pin 20 (*interrupt* 3), dan pin 21 (*interrupt* 2). Pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau perubah nilai.
- **SPI** : Pin 50 (MISO), pin 51 (MOSI), pin 52 (SCK), pin 53 (SS). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI. Pin SPI juga terhubung dengan header ICSP, yang secara fisik kompatibel dengan Arduino Uno, Arduino Duemilanove dan Arduino Diecimila.
- **LED** : Pin 13. Tersedia secara built-in pada papan Arduino Atmega 2560. LED terhubung ke pin digital 13. Ketika pin diset bernilai HIGH, maka LED menyala (ON), dan ketika pin diset bernilai *LOW*, maka LED padam (OFF).
- **TWI** : Pin 20 (SDA) dan pin 21 (SCL). Yang mendukung komunikasi TWI menggunakan perpustakaan *Wire*. Perhatikan bahwa pin ini tidak di lokasi yang sama dengan pin TWI pada Arduino Duemilanove atau Arduino Diecimila.

Arduino Mega 2560 memiliki 16 pin sebagai analog *input*, yang masing-masing menyediakan resolusi 10 bit (yaitu 1024 nilai yang berbeda). Secara *default* pin ini dapat diukur/diatur dari mulai Ground sampai dengan 5 Volt, juga memungkinkan untuk mengubah titik jangkauan tertinggi atau terendah mereka menggunakan pin AREF dan fungsi analog *Reference()*.

Ada beberapa pin lainnya yang tersedia, antara lain:

- **AREF** : Referensi tegangan untuk *input* analog. Digunakan dengan fungsi analog *Reference()*.

- **RESET** : Jalur *LOW* ini digunakan untuk me-*reset* (menghidupkan ulang) mikrokontroler. Jalur ini biasanya digunakan untuk menambahkan tombol *reset* pada *shield* yang menghalangi papan utama Arduino.

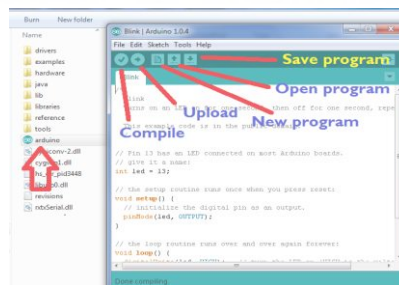
### 2.3.4.7. Komunikasi

Arduino Mega 2560 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, dengan Arduino lain, atau dengan mikrokontroler lainnya. Arduino ATmega328 menyediakan 4 hardware komunikasi serial UART TTL (5 Volt). Sebuah chip ATmega16U2 yang terdapat pada papan digunakan sebagai media komunikasi serial melalui USB dan muncul sebagai COM Port Virtual (pada *Device* komputer) untuk berkomunikasi dengan perangkat lunak pada komputer, untuk sistem operasi Windows masih tetap memerlukan file inf, tetapi untuk sistem operasi OS X dan Linux akan mengenali papan sebagai port COM secara otomatis. Perangkat lunak Arduino termasuk didalamnya serial monitor memungkinkan data tekstual sederhana dikirim ke dan dari papan Arduino. LED RX dan TX yang tersedia pada papan akan berkedip ketika data sedang dikirim atau diterima melalui chip USB-to-serial yang terhubung melalui USB komputer (tetapi tidak untuk komunikasi serial seperti pada pin 0 dan 1) .

Sebuah perpustakaan *Software Serial* memungkinkan untuk komunikasi serial pada salah satu pin digital Mega2560. ATmega2560 juga mendukung komunikasi TWI dan SPI. Perangkat lunak Arduino termasuk perpustakaan Wire digunakan untuk menyederhanakan penggunaan bus TWI. Untuk komunikasi SPI, menggunakan perpustakaan SPI (www.arduino.cc, 2017).

### 2.3.4.8. Pemrograman

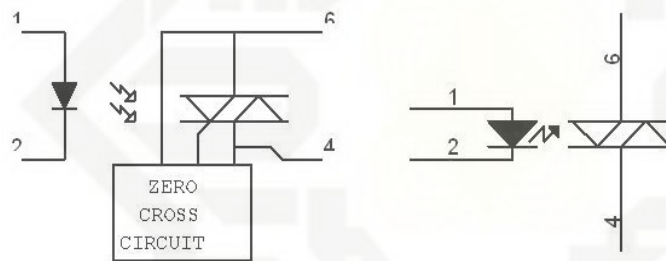
Arduino memiliki bahasa pemrograman tersendiri yaitu bahasa arduino, merupakan pengembangan dari bahasa C yang disederhanakan dan dipermudah dengan *libraries*. Untuk meng-*compile* dan meng-*upload* program ke board arduino dapat menggunakan *software* Arduino IDE( *Integrated Development Environment* ) (www.arduino.cc, 2017)).



Gambar 2.4. Contoh skema Software Arduino beserta contoh program (Sumber : www.arduino.cc)

### 2.3.5. OPTOTRIAC

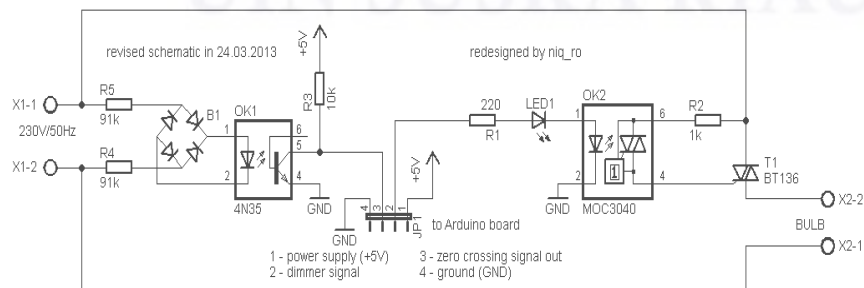
*Optoisolator* merupakan komponen semikonduktor yang tersusun atas LED infra merah dan sebuah *photo triac* yang digunakan sebagai pengendali *triac*. *Optoisolator* biasanya digunakan sebagai antar muka antara rangkaian pengendali dengan rangkaian daya (*triac*) dan juga sebagai pengaman rangkaian kendali, karena antara LED infra merah dan *photo triac* tidak terhubung secara elektrik, sehingga bila terjadi kerusakan pada rangkaian daya (*triac*) maka rangkaian pengendali tidak ikut rusak. *Optoisolator* biasanya terdiri dari dua macam yaitu *optoisolator* yang terintegrasi dengan rangkaian *zero crossing detector* dan *optoisolator* yang tidak memiliki rangkaian *zero crossing detector*. *Optoisolator* yang terintegrasi dengan *zero crossing detector* biasanya menggunakan *triac* sebagai solid state relay (SSR), sedangkan pada *optoisolator* yang tidak terintegrasi dengan *zero crossing detector* biasanya menggunakan *triac* untuk mengendalikan tegangan. Simbol dari *optoisolator* ini terlihat seperti pada gambar berikut (Elektronika Dasar, 2013).



Gambar 2.5 Simbol *Optoisolator*  
(Sumber : Elektronika Dasar 2013)

Hal-hal yang diperlukan dalam menggunakan *optoisolator* adalah besarnya arus pada diode infra merah untuk membuat *photo triac* terkunci (*latch*), juga besarnya arus maksimum yang mampu dilewati *photo triac* untuk mengalirkan arus *gate* pada *triac* daya.

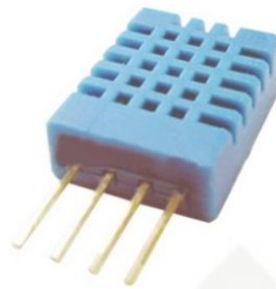
Di bawah ini adalah rangkaian Ac *Dimmer* dengan menggunakan Opto-*triac* atau Opto-*isolator* yang berfungsi sebagai pengendali tegangan.



Gambar 2.6 rangkaian Ac *Dimmer*  
(Sumber : www.arduinotehniq.com)

### 2.3.6. Sensor Suhu dan Kelembaban

#### 2.3.6.1. Sensor Kelembaban



Gambar 2.7. Sensor DHT 11

(Sumber : [www.droboticsonline.com](http://www.droboticsonline.com))

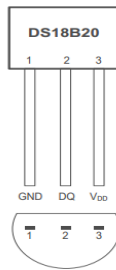
Sensor DHT 11 merupakan perangkat yang berperan dalam penginderaan derajat panas udara dan kadar air pada udara. sensor ini termasuk kategori *smart sensor* yang memiliki ADC dan mikrokontroler terintegrasi dalam kemasan sensor. Berdasarkan *datasheet*, sensor ini telah melalui kalibrasi di laboratorium termal. Pada *datasheet* juga disebutkan bahwa koefisien kalibrasi disimpan dalam OTP *program memory*, sehingga ketika sensor bekerja dan menghasilkan keluaran sinyal listrik, akan secara otomatis dikalkulasikan berapa derajat suhu dan berapa persen kelembaban relatif yang terukur.

Sensor ini memiliki kemasan berukuran 2x3,5 cm. Berdasarkan *datasheet*, hasil pengukuran dapat ditransmisikan melalui medium dengan panjang maksimum 20 meter. Spesifikasi lainnya adalah *supply voltage* : +5 V, *temperature range* : 0-50 °C dengan *error margin* ± 2°C, *humidity range* : 20-90% RH dengan *error margin* ± 5% RH. Gambar 2.9 menunjukkan tampilan sensor DHT 11 dan tabel 4 merupakan spesifikasi detail dari sensor DHT 11 (syam 2013,43).

#### 2.3.6.2. Sensor Suhu DS18B20

Banyak sensor suhu yang dipakai dalam implementasi sistem instrumentasi, salah satu contohnya adalah DS18B20. Sensor suhu DS18B20 ini telah memiliki keluaran digital meskipun bentuknya kecil (TO-92), cara untuk mengaksesnya adalah dengan metode sinyal 1 *wire*. Sensor ini sangat menghemat pin *port* mikrokontroler, karena 1 pin *port* mikrokontroler dapat digunakan untuk berkomunikasi dengan beberapa divansi lainnya. Sensor ini juga memiliki akurasi cukup tinggi, yaitu 0,5 °C pada rentang suhu -10 °C hingga +85 °C, sehingga banyak dipakai untuk aplikasi sistem monitoring suhu , aplikasi yang berhubungan dengan sensor sering kali membutuhkan ADC dan beberapa pin *port* mikrokontroler namun pada sensor DS18B20 ini tidak dibutuhkan ADC agar dapat

berkomunikasi dengan mikrokontroler. Bentuk fisik sensor DS18B20 ditunjukkan pada Gambar 2.6 (Ratih Pratiwi,2009).



Gambar 2.8 Sensor Suhu DS18B20  
(Sumber : [www.maximintegrated.com](http://www.maximintegrated.com))

Spesifikasi lain dari sensor DS18B20 adalah sebagai berikut:

- a. Memiliki kode serial 64-bit
- b. Dapat beroperasi tanpa *Power supply* dari luar
- c. *Power supply* 3-5,5 VDC. Dapat diperoleh dari aliran data
- d. Pengukuran temperatur dari -55°C- +125°C
- e. Resolusi ADC:9-bit
- f. Waktu konversi maks 750 ms

### 2.3.7. Logika Fuzzy

#### 2.3.7.1. Sistem Logika Fuzzy

Konsep Logika Fuzzy diperkenalkan oleh Prof. Lotfi Zadeh dari Universitas California di Berkeley pada 1965, yang dipresentasikan bukan sebagai suatu metodologi kontrol, tetapi sebagai suatu cara pemrosesan data dengan memperkenalkan penggunaan partial *set* membership dibanding *crisp set* membership atau non-membership (kusumadewi dan purnomo,2013).

Pengendali logika Fuzzy dikategorikan dalam kontrol cerdas (*intelligent control*). Unit logika Fuzzy memiliki kemampuan menyelesaikan masalah perilaku sistem yang kompleks, yang tidak dimiliki oleh pengendali konvensional. Secara umum kontroler logika Fuzzy memiliki kemampuan sebagai berikut:

1. Beroperasi tanpa campur tangan manusia secara langsung, tetapi memiliki efektifitas yang sama dengan kontrol manusia.
2. Mampu menangani sistem-sistem yang kompleks, non-linier dan tidak stationer.
3. Memenuhi spesifikasi operasional dan kriteria kinerja.
4. Strukturnya sederhana, kokoh dan beroperasi *real time*.

Ada beberapa hal yang perlu diketahui dalam memahami sistem logika Fuzzy, yaitu (Kusumadewi dan Purnomo, 2013):

a. Variabel Fuzzy

Variabel Fuzzy merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu sistem Fuzzy. Contoh : umur, temperatur, permintaan, dsb.

b. Himpunan logik Fuzzy

Himpunan Fuzzy merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel Fuzzy

c. semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel Fuzzy. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan.

d. Domain

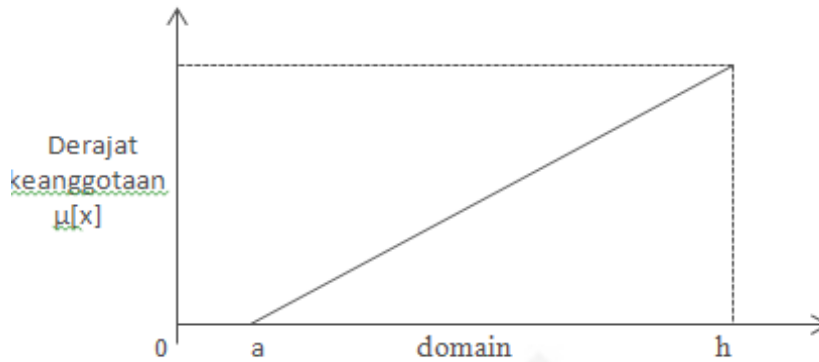
Domain himpunan Fuzzy adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan Fuzzy

**2.3.7.2. Fungsi Keanggotaan**

Fungsi keanggotaan (membership function) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki *interval* antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang bisa digunakan (Kusumadewi dan Purnomo, 2013).

A. Representasi Linier

Pada representasi linear, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Ada 2 keadaan himpunan Fuzzy yang linear. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi (gambar 2.7).



Gambar 2.9 Representasi Linier naik.

(sumber : Kusumadewi dan purnomo,2013)

Fungsi keanggotaan :

$$\mu_{[x]} = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & x \geq b \end{cases} \quad (2.1)$$

Kedua, merupakan kebalikan yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah (gambar 2.8).



Gambar 2.10 Representasi Linier Turun.

(Sumber : Kusumadewi dan purnomo,2013)

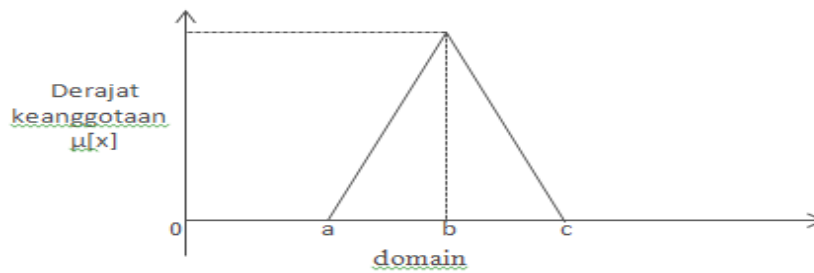
Fungsi Keanggotaan :

$$\mu_{[x]} = \begin{cases} 0; & x \geq b \\ \frac{b-x}{b-a}; & a \leq x \leq b \end{cases} \quad (2.2)$$

## B. Representasi Kurva Segitiga

Kurva Segitiga pada dasarnya merupakan gabungan antara 2 garis (linear) seperti terlihat pada gambar 2.9.





Gambar 2.11 Representasi Kurva Segitiga .

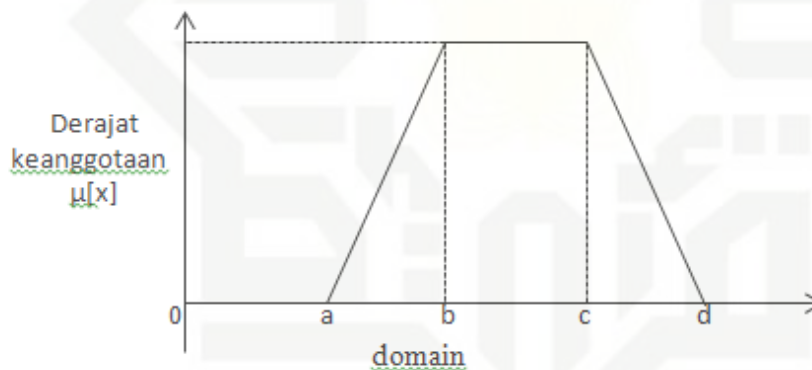
(Sumber : Kusumadewi dan purnomo,2013)

Fungsi Keanggotaan:

$$\mu_{[x]} = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ \frac{b-x}{c-b}; & b \leq x \leq c \end{cases} \quad (2.3)$$

### C. Representasi Kurva Trapesium

Kurva Segitiga pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2.12 Representasi Kurva Trapesium

(Sumber : Kusumadewi dan purnomo,2013)

Fungsi Keanggotaan :

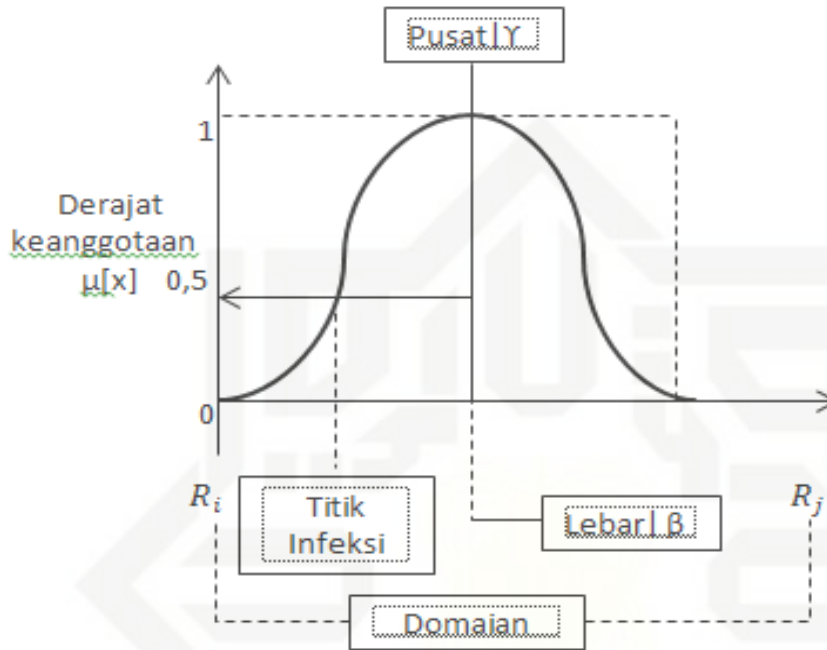
$$\mu_{[x]} = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{d-x}{d-c}; & x \geq c \end{cases} \quad (2.4)$$



1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:  
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.  
 b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.  
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

**F. Representasi Kurva Bentuk Lonceng (Bell Curve)**

Untuk merepresentasikan bilangan Fuzzy, biasanya digunakan kurva berbentuk lonceng. Kurva berbentuk lonceng ini terbagi atas 3 kelas, yaitu: himpunan Fuzzy PI, beta, dan Gauss. Perbedaan ketiga kurva ini terletak pada gradiennya.



Gambar 2.15 Representasi Kurva Bentuk Lonceng  
(Sumber : Kusumadewi dan purnomo,2013)

**G. Koordinat Keanggotaan**

Himpunan Fuzzy berisi urutan pasangan berurutan yang berisi nilai domain dan kebenaran nilai keanggotaannya dalam bentuk:

$$\text{Skalar}(i) / \text{Derajat}(i)$$

‘Skalar’ adalah suatu nilai yang digambar dari domain himpunan Fuzzy sedangkan ‘Derajat’ skalar merupakan derajat keanggotaan himpunan Fuzzy nya.

**2.3.7.3. Fuzzifikasi**

Proses fuzzifikasi merupakan proses untuk mengubah variable non Fuzzy (*variable numerik*) menjadi variable Fuzzy (variabel linguistik). Nilai masukan yang masih dalam bentuk *variable numeric* yang telah dikuantisasi sebelum diolah oleh pengendali Fuzzy harus diubah terlebih dahulu kedalam variabel Fuzzy. Melalui fungsi keanggotaan yang telah disusun maka nilai-nilai masukan tersebut menjadi informasi Fuzzy yang berguna nantinya untuk proses pengolahan secara Fuzzy pula. Proses ini disebut

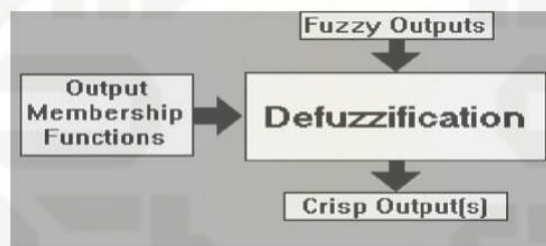
fuzzyfikasi, dengan kata lain fuzzyfikasi merupakan pemetaan titik-titik numerik (*crisp point*) (Kusumadewi dan Purnomo, 2013).

#### 2.3.7.4. Penurunan Rules Base

*Rule base* adalah sekelompok aturan Fuzzy dalam berhubungan dengan keadaan sinyal masukan dan sinyal keluaran. *Rule base* merupakan dasar dari pengambilan keputusan atau *inference* proses untuk mendapatkan aksi keluaran sinyal kontrol dari suatu kondisi masukan yaitu *error* dan *delta error* dengan berdasarkan *rule-rule* yang telah ditetapkan. Proses *rule base* berfungsi untuk mencari suatu nilai Fuzzy *output* dari Fuzzy *input* (Kusumadewi dan Purnomo, 2013).

#### 2.3.7.5. Defuzzyfikasi

Defuzzyfikasi adalah proses pemetaan dari hasil aksi kontrol inferensi Fuzzy ke aksi kendali non Fuzzy.



Gambar 2.16 Defuzzyfikasi

(Sumber : Kusumadewi dan Purnomo, 2013)

#### 2.3.7.6. Sistem Inferensi Fuzzy

Dalam membangun sebuah sistem Fuzzy dikenal beberapa metode penalaran yaitu metode Tsukamoto, metode Mamdani (*max – min*) dan metode Takagi Sugeno.

#### 2.3.7.7. Metode Tsukamoto

Metode Tsukamoto merupakan perluasan dari penalaran monoton, pada metode Tsukamoto, setiap konsekuen pada aturan yang berbentuk *If-Then* harus direpresentasikan dengan suatu himpunan Fuzzy dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil penarikan kesimpulan (*inference*) dari tiap-tiap aturan diberikan secara tegas (*crisp*) berdasarkan  $\alpha$ -predikat (*fire strength*). Hasil akhir diperoleh dengan menggunakan rata-rata berbobot (*weight average*) (Kusumadewi dan Purnomo, 2013).

### 2.3.7.8. Metode Mamdani (*Max-Min*)

Metode Mamdani sering juga dikenal dengan nama Metode *Max-Min*. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975 (kusumadewi dan purnomo,2013).. Untuk mendapatkan *output*, diperlukan 4 tahapan :

#### 1. Pembentukan himpunan Fuzzy.

Pada proses fuzzifikasi langkah pertama adalah menentukan *variable* Fuzzy dan himpunan Fuzzy. Kemudian tentukan derajat kesepadanan (*degree of match*) antara data masukan Fuzzy dengan himpunan Fuzzy yang didefinisikan untuk setiap variabel masukan sistem dari setiap aturan Fuzzy. Pada metode mamdani, baik variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan Fuzzy (kusumadewi dan purnomo,2013).

#### 2. Aplikasi fungsi implikasi

Pada metode mamdani, fungsi implikasi yang digunakan adalah *Min*. Lakukan implikasi Fuzzy berdasar pada kuat penyulutan dan himpunan Fuzzy terdefinisi untuk setiap variabel keluaran di dalam bagian konsekuensi dari setiap aturan. Hasil implikasi Fuzzy dari setiap aturan ini kemudian digabungkan untuk menghasilkan keluaran *inferensi* Fuzzy (kusumadewi dan purnomo,2013).

#### 3. Komposisi Aturan

Tidak seperti penalaran monoton, apabila sistem terdiri dari beberapa aturan, maka inferensi diperoleh dari kumpulan dan korelasi antar aturan. Ada 3 metode yang digunakan dalam melakukan inferensi sistem Fuzzy, yaitu: *max*, *additive* dan probabilistik OR (kusumadewi dan purnomo,2013).

#### 4. Penegasan (*defuzzy*)

*Input* dari proses defuzzifikasi adalah suatu himpunan Fuzzy yang diperoleh dari komposisi aturan-aturan Fuzzy, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan Fuzzy tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output* (kusumadewi dan purnomo,2013).

### 2.3.7.9. Metode Sugeno

Penalaran dengan metode Sugeno hampir sama dengan penalaran Mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan Fuzzy, melainkan berupa konstanta atau persamaan linear. Metode ini diperkenalkan oleh Takagi Sugeno Kang pada tahun

1985, sehingga metode ini sering juga dinamakan dengan metode TSK (kusumadewi dan purnomo,2013).

### 2.3.7.10. Analisa Time Respons

Analisa *time respons* sistem kendali terdiri dari respon *transient* dan *steady state*. Respon *transient* adalah respon sistem yang berlangsung dari keadaan awal sampai keadaan akhir, sedangkan respon *steady state* adalah kondisi keluaran sesudah habis respon *transient*. Parameter – parameter yang dapat dianalisa adalah :

1. Konstanta waktu

Ketika waktu  $t = \tau$ , maka  $y = k(1 - e^{-1}) = 0,63k$ . Jadi, konstanta waktu  $\tau$  untuk sistem orde pertama ketika diberikan sinyal masukan berbentuk tangga adalah waktu yang dibutuhkan oleh keluaran sistem untuk mencapai 0,63 dari nilai keadaan tunaknya (Rachman,2012).

2. Waktu Tunda (*Delay Time*)

Waktu tunda ( $t_d$ ) adalah waktu yang dibutuhkan oleh keluaran sistem untuk mencapai nilai 50% dari nilai keadaan tunaknya (Rachman,2012)..

3. Waktu Naik (*Rise Time*)

Waktu naik ( $t_r$ ) adalah waktu yang dibutuhkan oleh keluaran sistem untuk naik dari 10% nilai keadaan tunaknya hingga mencapai nilai 90% nilai keadaan tunaknya (Rachman,2012)..

4. Waktu Pemantapan (*Settling Time*)

Waktu pemantapan ( $t_s$ ) merupakan ukuran dari waktu yang diperlukan oleh osilasi respon sistem untuk menghilang. Lebih spesifiknya, waktu pemantapan adalah waktu yang diperlukan oleh respon sistem untuk jatuh menuju suatu nilai tertentu dan kemudian tetap berada pada nilai ini, yang biasanya dinyatakan sebagai persentase terhadap nilai *keadaantunaknya* (Rachman,2012).

$$Overshoot = \frac{|T_{puncak} - T_{steady}|}{T_{Steady}} \times 100\% \quad (2.5)$$

5. Kesalahan Tunak (*Error Steady State = ess*)

Yaitu kesalahan yang merupakan selisih antara keluaran yang sebenarnya dengan keluaran yang diinginkan.

$$error\ steady\ state = \frac{|T_{steady} - T_{setpoint}|}{T_{setpoint}} \times 100\% \quad (2.6)$$