

BAB II TINJAUAN PUSTAKA

2.1 Pengenalan Sensor

Sensor merupakan jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik. Sensor sering digunakan untuk pendeteksian pada saat melakukan pengukuran atau pengendalian. Beberapa jenis sensor yang banyak digunakan dalam rangkaian elektronik salah satunya yaitu sensor suhu (LM35), dan sensor kelembaban udara (DHT11).

Dalam melakukan analisa ini digunakan sensor tipe LM35 yang digunakan sebagai alat sensor suhu didalam rangkaian dengan menggunakan supply tegangan sebesar 0-5V dari *Arduino* yang merupakan *board* elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama yaitu sebuah chip mikrokontroler dengan jenis AVR dari perusahaan Atmel. Mikrokontroler itu sendiri adalah chip atau IC (*Integrated Circuit*) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca input, memproses input tersebut dan kemudian menghasilkan output sesuai yang diinginkan.

2.1.1 Sensor Suhu / *Temperature* (LM35)

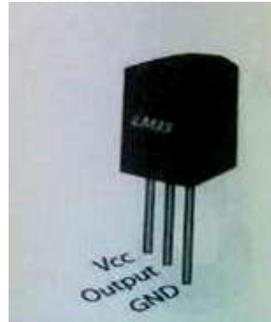
Karena komponen utama *Arduino* adalah *mikrokontroler*, maka *Arduino* pun dapat diprogram menggunakan komputer sesuai kebutuhan kita. Untuk dapat mengukur suhu ruangan dibutuhkan satu komponen saja yaitu LM35.

Selain modul mikrokontroler *Arduino*, LM35 adalah sensor suhu dari *National Semiconductor* yang mempunyai akurasi tinggi. Outputnya berupa tegangan analog dan memiliki jangkauan pengukuran -55°C hingga $+150^{\circ}\text{C}$ dengan akurasi $\pm 0.5^{\circ}\text{C}$. Tegangan output adalah $10\text{mV}/^{\circ}\text{C}$.

Output dapat langsung dihubungkan *port mikrokontroler* yang memiliki ADC atau dengan *Arduino*, karena *Arduino* memiliki port *ADC (analog input)*



sebanyak 6 buah. Kemudian rangkaian modul Arduino dengan sensor suhu seperti gambar di bawah ini ;



Gambar 2.1 Sensor Suhu/Temperature (LM35)

(Sumber: Arduino, Belajar Cepat dan Pemrograman/Informatika Bandung/2016)



Gambar 2.2 Tampak Bawah Sensor Suhu (LM35)

(Sumber: Arduino, Belajar Cepat dan Pemrograman/Informatika Bandung/2016)

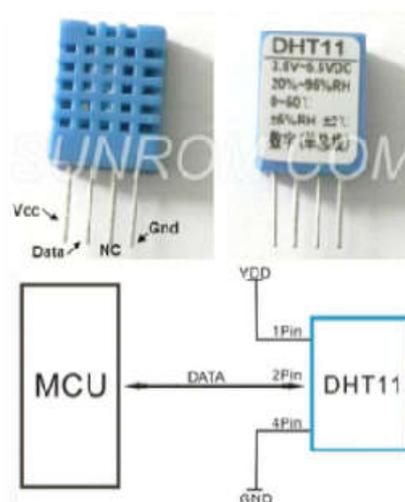
Dari penjelasan (gambar 2.1) dan seperti gambar (gambar 2.2) diatas bahwa struktur kaki-kaki yang merupakan bagian dari sensor suhu/temperature LM35 memiliki tiga buah kaki yaitu: pada bagian kaki ($+V_s$), dihubungkan ke bagian (V_{cc}) yg bernilai sebesar 5V, pada board arduino uno dan untuk bagian kaki GND dihubungkan ke *ground* (GND) pada *board* arduino uno, sedangkan pada bagian kaki (V_{Out}) yang merupakan keluaran (*Output*) dari hasil pengolahan data analog dari *sensor* LM35 yang dihubungkan ke bagian *analog input* (*pin A0*) pada *board* arduino uno.



2.1.2 Sensor Kelembaban Udara/*Humidity* (DHT11)

Kelembaban udara menggambarkan kandungan uap air di udara yang dapat dinyatakan sebagai kelembaban mutlak, kelembaban nisbi (relatif) maupun defisit tekanan uap air. Kelembaban nisbi adalah membandingkan antara kandungan/tekanan uap air aktual dengan keadaan jenuhnya atau pada kapasitas udara untuk menampung uap air.

Peralatan elektronik juga menjadi mudah berkarat jika udara disekitarnya memiliki kelembaban yang cukup tinggi. Oleh karena itu, informasi mengenai kelembaban udara pada suatu area tertentu menjadi sesuatu hal yang penting untuk diketahui karena menyangkut efek-efek yang ditimbulkannya.



Gambar 2.3 Sensor Kelembaban Udara/*Humidity* (DHT11)

(Sumber: <http://www.sunrom.com>, diakses pada tanggal 11 Mei 2017)

Informasi mengenai nilai kelembaban udara diperoleh dari proses pengukuran. Alat yang biasanya digunakan untuk mengukur kelembaban udara adalah higrometer. DHT11 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan Arduino. Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam OTP program memory, sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan



koefisien tersebut dalam kalkulasinya. DHT11 ini termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-*interference*. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter, dengan spesifikasi: *Supply Voltage*: +5 V, *Temperature range* : 0-50 °C *error of* ± 2 °C, *Humidity* : 20-90% RH $\pm 5\%$ RH *error*, dengan spesifikasi *digital interfacing system*. Produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu dan kelembaban.

Tabel 2.1 Tabel karakteristik sensor kelembaban udara/*Humidity*

| | |
|---------------------------|--|
| Model | DHT11 |
| Power supply | 3-5.5 V DC |
| Output signal | digital signal via single-bus |
| Measuring range | humidity 20-90% RH $\pm 5\%$ RH error temperature 0-50 °C error of ± 2 °C |
| Accuracy | humidity $\pm 4\%$ RH (Max $\pm 5\%$ RH); temperature ± 2.0 Celsius |
| Resolution or Sensitivity | humidity 1% RH; temperature 0.1 Celsius |
| Repeatability | humidity $\pm 1\%$ RH; temperature ± 1 Celsius |
| Humidity hysteresis | $\pm 1\%$ RH |
| Long-term Stability | $\pm 0.5\%$ RH/ year |
| Sensing period | Average: 2s |
| Interchangeability | fully interchangeable |
| Dimensions size | 12*15.5*5.5 mm |

(Sumber : <http://www.sunrom.com>, diakses pada tanggal 11 Mei 2017)

Dari penjelasan (Tabel 2.1) diatas bahwa struktur yang merupakan cara kerja dari sensor kelembaban udara/ *Humidity DHT11* memiliki empat buah kaki



yaitu: pada bagian kaki (V_{CC}), dihubungkan ke bagian V_{ss} yg bernilai sebesar 5V, pada board arduino uno dan untuk bagian kaki GND dihubungkan ke *ground* (GND) pada *board* arduino uno, sedangkan pada bagian kaki data yang merupakan keluaran (*Output*) dari hasil pengolahan data analog dari *sensor DHT11* yang dihubungkan ke bagian *analog input* (pin3), yaitu pada bagian pin *PWM* (*Pulse Width Modulation*) pada *board* arduino uno dan satu kaki tambahan yaitu kaki *NC* (*Not Connected*), yang tidak dihubungkan ke pin manapun. Sensor kelembaban lain yang banyak dikembangkan adalah jenis sensor serat optik yang menggunakan serat optik sebagai bahan sensor. Berbagai metode dan bahan untuk sensor telah dikembangkan pada sensor serat optik ini.

Metode pengukuran yang digunakan seperti misalnya; pengukuran serapan gelombang, pengukuran pelemahan gelombang, dan pengukuran intensitas. Material yang digunakan untuk sensor kebanyakan adalah bahan-bahan hidrogel seperti gelatin murni atau gelatin yang didoping, polimer yang didoping $CoCl_2+PVA$, polianilin dengan nano Co, dan agarosa. Pemanfaatan *POF* (*polymer optical fiber*) sebagai sensor kelembaban telah dilakukan oleh Shinzo dengan konfigurasi probe sensor berbentuk lurus, diperoleh rentang kelembaban yang dapat dideteksi antara 20-90%. Penelitian lain oleh Arreguidengan gel agarosa yang digunakan sebagai pengganti *cladding* dari probe, diperoleh hasil yang lebih baik. Rentang kelembaban yang mampu dideteksi 10-100% dengan waktu respon 90 detik.

2.2. Logika Fuzzy

2.2.1 Pengertian Logika Fuzzy

Logika *Fuzzy* merupakan salah satu komponen pembentuk *soft computing*. Logika *Fuzzy* pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh pada tahun 1965. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dari penalaran dengan logika *fuzzy* tersebut.



Dalam banyak hal, logika *fuzzy* digunakan sebagai suatu cara untuk memecahkan permasalahan dari *input* menuju ke *output* yang diharapkan. Beberapa contoh yang dapat diambil antara lain:

1. Manajer pergudangan mengatakan pada manajer produksi seberapa banyak persediaan barang pada akhir minggu ini, kemudian manajer produksi akan menetapkan jumlah barang yang harus diproduksi esok hari.
2. Seorang pegawai melakukan tugasnya dengan kinerja yang sangat baik, kemudian atasan akan memberikan *reward* yang sesuai dengan kinerja pegawai tersebut.

Logika *fuzzy* dapat dianggap sebagai kotak hitam yang menghubungkan antara *input* menuju ke ruang *output*. Kotak hitam tersebut berisi cara atau metode yang dapat digunakan untuk mengolah data *input* menjadi *output* dalam bentuk informasi yang baik.

2.2.2 Alasan Menggunakan Logika Fuzzy

Ada beberapa alasan mengapa orang menggunakan logika fuzzy, antara lain:

1. Konsep logika fuzzy mudah dimengerti. Karena logika fuzzy menggunakan dasar teori himpunan, maka konsep matematis yang mendasari penalaran fuzzy tersebut cukup mudah untuk dimengerti.
2. Logika fuzzy sangat fleksibel, artinya mampu beradaptasi dengan perubahan-perubahan, dan ketidakpastian yang menyertai permasalahan.
3. Logika fuzzy memiliki toleransi terhadap data yang tidak tepat. Jika diberikan sekelompok data yang cukup homogen, dan kemudian ada beberapa data yang “eksklusif”, maka logika fuzzy memiliki kemampuan untuk menangani data eksklusif tersebut.
4. Logika fuzzy mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.
5. Logika fuzzy dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.



Dalam hal ini, sering dikenal dengan nama *Fuzzy Expert Systems* menjadi bagian terpenting.

6. Logika fuzzy dapat bekerjasama dengan teknik-teknik kendali secara konvensional. Hal ini umumnya terjadi pada aplikasi di bidang teknik mesin maupun teknik elektro.

Logika fuzzy didasarkan pada bahasa alami. Logika fuzzy menggunakan bahasa sehari-hari sehingga mudah dimengerti.

2.2.3 Himpunan *Fuzzy*

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan, yaitu: satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan dan nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

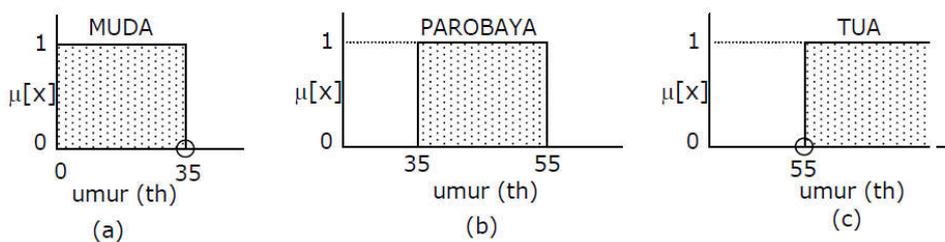
Misalkan variabel umur dibagi menjadi 3 kategori, yaitu :

MUDA umur < 35 tahun

PAROBAYA $35 \leq \text{umur} \leq 55$ tahun

TUA umur > 55 tahun

Nilai keanggotaan secara grafis, himpunan MUDA, PAROBAYA, dan TUA dapat dilihat pada gambar 2.4 di bawah ini.



Gambar 2.4 Himpunan : MUDA, PAROBAYA, dan TUA

Pada gambit 2.2 dapat dilihat bahwa :

- Apabila seseorang berusia 34 tahun, maka ia dikatakan MUDA ($\mu_{\text{MUDA}}[34]=1$);



- Apabila seseorang berusia 35 tahun, maka ia dikatakan TIDAK MUDA ($\mu_{MUDA} [35] = 0$);
- Apabila seseorang berusia 35 tahun kurang 1 hari, maka ia dikatakan TIDAK MUDA ($\mu_{MUDA} [35th - 1 hr] = 0$);
- Apabila seseorang dikatakan 35 tahun, maka ia dikatakan PAROBAYA ($\mu_{PAROBAYA} [35] = 1$);
- Apabila seseorang berusia 34 tahun, maka ia dikatakan TIDAK PAROBAYA ($\mu_{PAROBAYA} [34] = 0$);

Dari hal diatas dapat dikatakan bahwa pemakaian himpunan *crisp* untuk menyatakan umur tidak seimbang, adanya perubahan kecil pada suatu nilai mengakibatkan perbedaan kategori yang cukup signifikan [20]. Himpunan *fuzzy* digunakan untuk mengantisipasi hal tersebut.

Himpunan *fuzzy* memiliki dua atribut yaitu :

- a. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti : MUDA, PAROBAYA, TUA.
- b. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti : 40,30, 50, dsb.

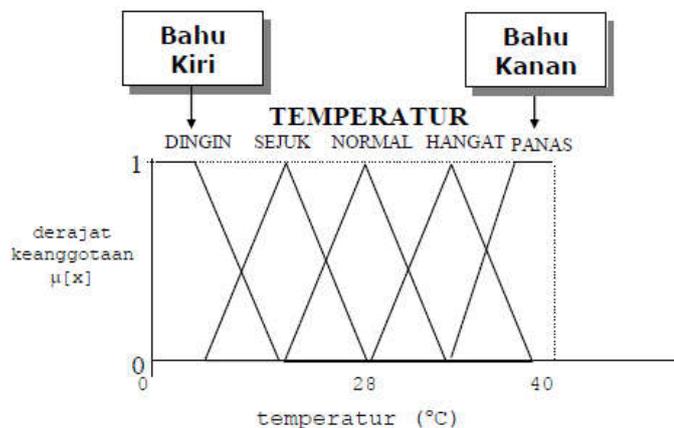
Ada beberapa hal yang harus diketahui dalam sistem himpunan *fuzzy*, yaitu :

- a. Variabel *fuzzy*

Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*, contoh: umur, temperatur, permintaan, jarak.

- b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Contoh variabel temperatur terbagi menjadi lima himpunan *fuzzy*, yaitu DINGIN, NORMAL, HANGAT, dan PANAS. Gambar 2.3 merupakan contoh himpunan fuzzy pada variabel temperatur.



Gambar 2.5 Himpunan *Fuzzy* Pada Variabel Temperatur

(Sumber : www.academia.edu/19471950/LOGIKA_FUZZY)

c. Semesta Pembicaraan

Semesta Pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta Pembicaraan merupakan himpunan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan[20]. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Contoh : semesta pembicaraan untuk variabel temperatur adalah [0 40].

d. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang akan diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan real yang senantiasa naik secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif atau negatif. Contoh : DINGIN = [0 20], SEJUK = [15 25], NORMAL = [20 30], HANGAT = [25 35] dan PANAS = [30 40].

2.2.4 Fungsi Keanggotaan

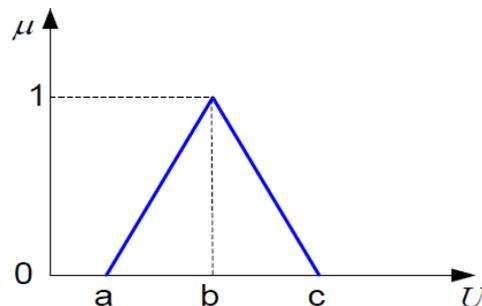
Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan). Fungsi keanggotaan memainkan



peranan yang sangat penting untuk merepresentasikan masalah dan menghasilkan keputusan akurat [19]. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi, antara lain dua bentuk fungsi yaitu segitiga dan trapesium yang umum digunakan adalah sebagai berikut :

a. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara dua garis (linear). Gambar 2.4 dibawah ini merupakan kurva segitiga pada *membership function*.



Gambar 2.6 Kurva Segitiga Pada *Membership Function*

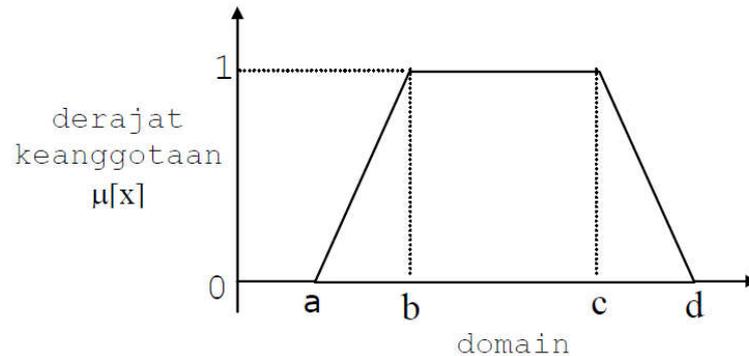
(Sumber : www.academia.edu/19471950/LOGIKA_FUZZY)

Fungsi keanggotaannya:

$$\mu(x) = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x-a)/(b-a); & a \leq x \leq b \\ (b-x)/(c-b); & b \leq x \leq c \end{cases} \dots\dots\dots 2.1$$

b. Representasi Kurva Trapesium

Kurva trapesium adalah kurva yang digunakan pada logika *fuzzy* robot *hexapod* pendeteksi kebocoran gas LPG. Kurva trapesium pada dasarnya sama seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan satu, yaitu $b \leq x \leq c$ dan parameternya berbeda yaitu a, b, c, dan d. Gambar 2.5 merupakan kurva trapesium *membership function*.



Gambar 2.7 Kurva Trapesium Pada *Membership Function*

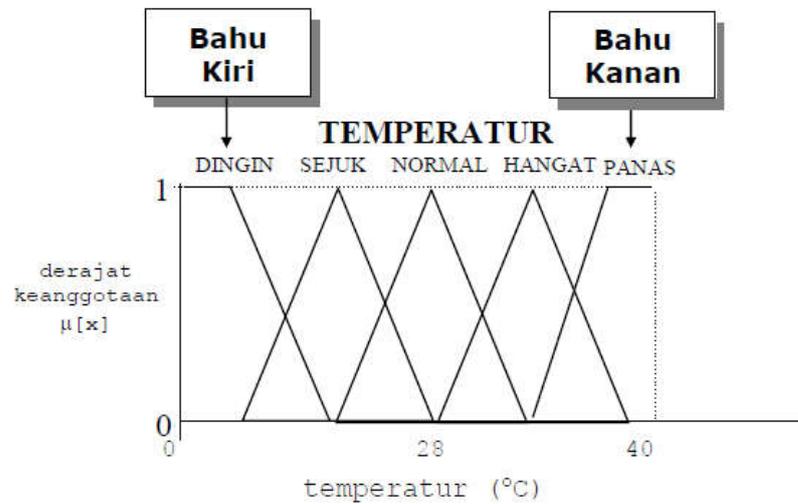
(Sumber : www.academia.edu/19471950/LOGIKA_FUZZY)

Fungsi keanggotaannya:

$$\mu(x) = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (b-x)/(c-b); & x \geq d \end{cases} \dots\dots\dots 2.2$$

c. Representasi Kurva Bentuk Bahu

Daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun (misalkan: DINGIN bergerak ke SEJUK bergerak ke HANGAT dan bergerak ke PANAS). Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Sebagai contoh, apabila telah mencapai kondisi PANAS, kenaikan temperatur akan tetap berada pada kondisi PANAS. Himpunan fuzzy ‘bahu’, bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah fuzzy. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar. Gambar 2.6 di bawah ini menunjukkan variabel TEMPERATUR dengan daerah bahunya.



Gambar 2.8 Kurva Bentuk Bahu

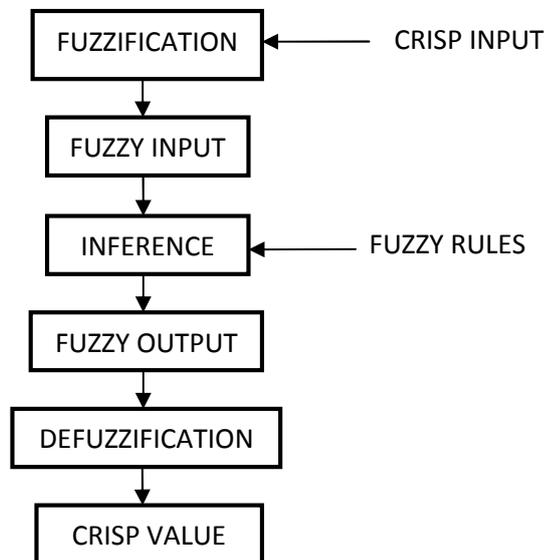
(Sumber : www.slideshare.net/ZaenalKhayat/contoh-peyelesaian-logika-fuzzy)

2.2.5 Sistem Inferensi *Fuzzy*

Inferensi adalah penarikan kesimpulan [21]. Sistem Inferensi *Fuzzy* (FIS) merupakan penarikan kesimpulan dari sekumpulan kaidah *fuzzy*. Sistem ini mempunyai kemampuan untuk mengembangkan sistem intelijen dalam lingkungan yang tidak pasti atau tidak tepat. Sistem ini menduga suatu fungsi dengan logika *fuzzy*. Sejumlah aturan dapat dibuat untuk pengendali logika samar. Contoh perlakuan pada pengendali sistem yang akan dibahas seperti, If (SensorKiri is Dekat) and (SensorTengah is Dekat) and (SensorKanan is Dekat) then (Kecepatan is Berhenti) and (Sudut_belok is Lurus). Di dalam FIS minimal harus ada dua buah kaidah *fuzzy*, yaitu input FIS (*crisp values*) dan output FIS (*crisp values*). FIS dapat dibangun dengan metode Mamdani atau Sugeno.



Skema tahapan inferensi *fuzzy*, terlihat pada gambar 2. di bawah ini.



Gambar 2.9 Skema Tahapan Inferensi *Fuzzy*

(Sumber : Rangkuti, Syahban. 2016. *Arduino & Proteus Simulasi dan Praktik*. Bandung : Informatika)

a. Fuzzifikasi

Fuzzifikasi adalah proses pemetaan dari input menjadi sekumpulan data samar dalam berbagai range output. Nilai input merupakan besaran analog yang diubah menjadi *fuzzy* input. Besaran analog atau crisp input dipetakan pada domain *membership function* yang sesuai dengan nilainya. Crisp input yang masuk ke domain label (*linguistic value*) akan menjadi *fuzzy* input. Pada tahap ini, masukan yang bersifat crisp (bernilai tunggal) dihitung derajat keanggotaannya terhadap setiap himpunan *fuzzy* input yang akan dikonversikan menjadi nilai *fuzzy*, sehingga dapat dikelompokkan dalam himpunan *fuzzy* tertentu [17].

b. Evaluasi aturan-aturan (rules)

Pada tahap ini, nilai-nilai input yang telah difuzzifikasikan diambil untuk diaplikasikan ke dalam aturan-aturan *fuzzy*.

c. *Rule Aggregation*

Setiap aturan memiliki nilai keluaran yang pada tahap ini akan diproses dan digabungkan sehingga mendapat sebuah kesimpulan.



d. Defuzzifikasi

Defuzzifikasi merupakan suatu proses pengubahan *fuzzy* output ke output yang bernilai tunggal (*crisp*).

2.2.6 Model *Fuzzy* Mamdani

Teknik inferensi *fuzzy* yang paling umum digunakan adalah metode mamdani. Model *Fuzzy* mamdani telah diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Metode ini memiliki beberapa kelebihan yang berdasarkan penalaran manusia, diterima oleh banyak pihak, lebih kompleks dan masukan berasal dari manusia. Metode mamdani juga disebut meto min max. Proses perhitungan *fuzzy* mamdani dibagi kedalam tahap :

1. Pembentukan himpunan *fuzzy*

Pada Metode Mamdani, baik variabel input maupun variabel output dibagi menjadi satu atau lebih himpunan *fuzzy*.

2. Aplikasi fungsi implikasi

Pada Metode Mamdani, fungsi implikasi yang digunakan adalah Min.

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 (*probor*).

a. Metode Max (Maximum)

Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah *fuzzy*, dan mengaplikasikannya ke output dengan menggunakan operator OR (*union*). Jika semua proposisi telah dievaluasi, maka output akan berisi suatu himpunan *fuzzy* yang merefleksikan kontribusi dari tiap-tiap proposisi. Secara umum dapat dituliskan:

$$\mu_{sf}[X_i] \leftarrow \max(\mu_{sf}[X_i], \mu_{kf}[X_i]) \dots\dots\dots 2.3$$



dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi fuzzy sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuen fuzzy aturan ke-i;

b. Metode Additive (Sum)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan bounded-sum terhadap semua output daerah fuzzy. Secara umum dituliskan:

$$\mu_{sf}[X_i] \leftarrow \min(1, \mu_{sf}[X_i] + \mu_{kf}[X_i]) \dots\dots\dots 2.4$$

dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi fuzzy sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuen fuzzy aturan ke-i;

c. Metode Probabilistik OR (probor)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan product terhadap semua output daerah fuzzy.

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. Ada beberapa metode defuzzifikasi pada komposisi aturan MAMDANI, yaitu:

a. Metode Centroid (Composite Moment)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil titik pusat (z^*) daerah fuzzy.

b. Metode Bisektor

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai pada domain fuzzy yang memiliki nilai keanggotaan separo dari jumlah total nilai keanggotaan pada daerah fuzzy.



c. Metode Mean of Maximum (MOM)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode Largest of Maximum (LOM)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. Metode Smallest of Maximum (SOM)

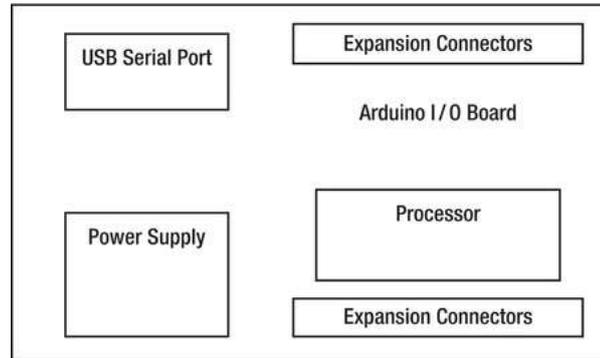
Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

2.3 Arduino Uno

Arduino adalah platform pembuatan prototipe elektronik yang bersifat *open-source hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Arduino ditujukan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau lingkungan yang interaktif.

Arduino pada awalnya dikembangkan di Ivrea, Italia. Nama Arduino adalah sebuah nama maskulin yang berarti teman yang kuat. Platform arduino terdiri dari arduino board, *shield*, bahasa pemrograman arduino, dan arduino *development environment*. Arduino board biasanya memiliki sebuah chip dasar mikrokontroler Atmel AVR ATmega8 berikut turunannya.

Blok diagram arduino board yang sudah disederhanakan dapat dilihat pada (gambar 2.2) *Shield* adalah sebuah papan yang dapat dipasang diatas arduino board untuk menambah kemampuan dari arduino board. Bahasa pemrograman arduino adalah bahasa pemrograman yang umum digunakan untuk membuat perangkat lunak yang ditanamkan pada arduino board. Bahasa pemrograman arduino mirip dengan bahasa pemrograman C++.



Gambar 2.10 Konfigurasi Pin ATmega328 Pada Board Arduino

(Sumber: <http://www.arduino.cc>, diakses tanggal 12 Mei 2017)

Arduino uno merupakan salah satu jenis rangkaian mikrokontroler yang menggunakan system *physical computing*. *Physical computing* adalah membuat sebuah sistem atau perangkat fisik dengan menggunakan software dan hardware yang sifatnya interaktif yaitu dapat menerima rangsangan dari lingkungan dan merespon balik. *Physical computing* adalah sebuah konsep untuk memahami hubungan yang manusiawi antara lingkungan yang sifat alaminya adalah analog dengan dunia digital. Pada prakteknya konsep ini diaplikasikan dalam desain-desain alat atau proyek-proyek yang menggunakan sensor dan *microcontroller* untuk menerjemahkan input analog ke dalam sistem software untuk mengontrol gerakan alat-alat elektro-mekanik seperti lampu, motor dan sebagainya.

Pembuatan *prototype* atau *prototyping* adalah kegiatan yang sangat penting di dalam prose *physical computing* karena pada tahap inilah seorang perancang melakukan eksperimen dan ujicoba dari berbagai jenis komponen, ukuran, parameter, program komputer dan sebagainya berulang-ulang kali sampai diperoleh kombinasi yang paling tepat.

Dalam hal ini perhitungan angka-angka dan rumus yang akurat bukanlah satu-satunya faktor yang menjadi kunci sukses di dalam mendesain sebuah alat karena ada banyak faktor eksternal yang turut berperan, sehingga proses mencoba dan menemukan/mengoreksi kesalahan perlu melibatkan hal-hal yang sifatnya non-eksakta.

Prototyping adalah gabungan antara akurasi perhitungan dan seni. Proses prototyping bisa menjadi sebuah kegiatan yang menyenangkan atau menyebalkan,



itu tergantung bagaimana kita melakukannya. Misalnya jika untuk mengganti sebuah komponen, merubah ukurannya atau merombak kerja sebuah prototype dibutuhkan usaha yang besar dan waktu yang lama, mungkin prototyping akan sangat melelahkan karena pekerjaan ini dapat dilakukan berulang-ulang sampai puluhan kali – bayangkan betapa frustasinya perancang yang harus melakukan itu.

Idealnya sebuah prototype adalah sebuah sistem yang fleksibel dimana perancang bisa dengan mudah dan cepat melakukan perubahan-perubahan dan mencobanyalagi sehingga tenaga dan waktu tidak menjadi kendala berarti. Dengan demikian harus ada sebuah alat pengembangan yang membuat proses prototyping menjadi mudah. Pada masa lalu (dan masih terjadi hingga hari ini) bekerja dengan hardware berarti membuat rangkaian menggunakan berbagai komponen elektronik seperti resistor, kapasitor, transistor dan sebagainya. Setiap komponen disambungkan secara fisik dengan kabel atau jalur tembaga yang disebut dengan istilah “*hard wired*” sehingga untuk merubah rangkaian maka sambungan-sambungan itu harus diputuskan dan disambung kembali.

Dengan hadirnya teknologi digital dan microprocessor fungsi yang sebelumnya dilakukan dengan hired wired digantikan dengan program-program software. Ini adalah sebuah revolusi di dalam proses prototyping. Di antara sekian banyak alat pengembangan prototype, Arduino adalah salah satunya yang paling banyak digunakan. **Arduino** dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*. Pertama-tama perlu dipahami bahwa kata “platform” di sini adalah sebuah pilihan kata yang tepat.

Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari hardware, bahasa pemrograman dan Integrated Development Environment (IDE) yang canggih. IDE adalah sebuah software yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory* microcontroller. Ada banyak projek dan alat-alat dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino.



Arduino berevolusi menjadi sebuah platform karena ia menjadi pilihan dan acuan bagi banyak praktisi. Software lebih mudah diubah dibandingkan hardware, dengan beberapa penekanan tombol kita dapat merubah logika alat secara radikal dan mencoba versi ke-dua, ke-tiga dan seterusnya dengan cepat tanpa harus mengubah pengkabelan dari rangkaian. Salah satu yang membuat Arduino memikat hati banyak orang adalah karena sifatnya yang open source, baik untuk hardware maupun software-nya.

Arduino Uno adalah arduino board yang menggunakan mikrokontroler ATmega328. Arduino Uno memiliki 14 pin digital (6 pin dapat digunakan sebagai output PWM), 6 input analog, sebuah 16 MHz osilator kristal, sebuah koneksi USB, sebuah konektor sumber tegangan, sebuah header ICSP, dan sebuah tombol reset. Arduino Uno memiliki area cakupan yang luas untuk segala hal yang dibutuhkan untuk mendukung sebuah aplikasi yang berbasis mikrokontroler. Hanya dengan menghubungkannya ke sebuah komputer melalui USB atau memberikan tegangan DC dari baterai atau adaptor AC ke DC sudah dapat membuat aplikasinya bekerja. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai *USB-to-serial converter* untuk komunikasi serial ke komputer melalui port USB.

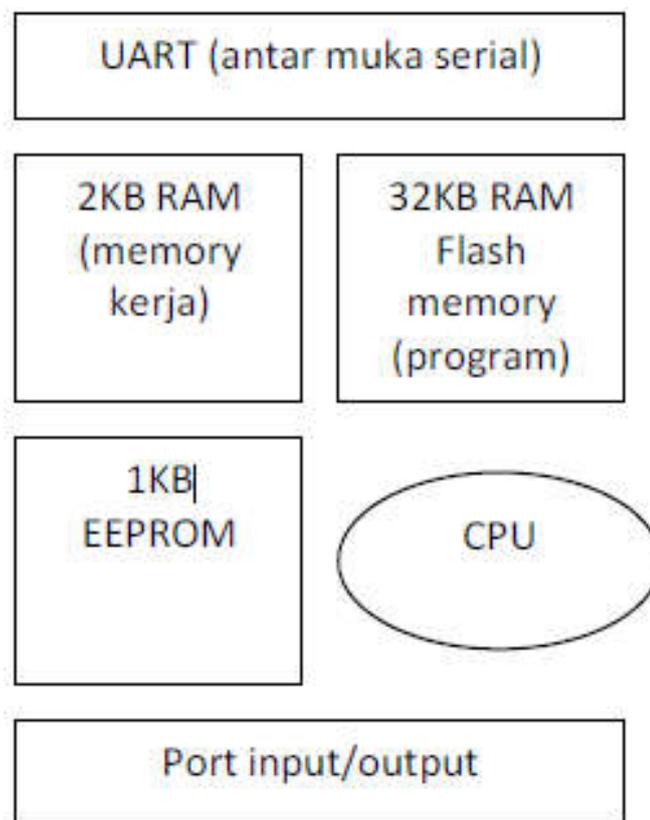
Adapun data teknis yang terdapat board Arduino UNO R3 adalah sebagai berikut:

- Mikrokontroler: ATmega328
- Tegangan Operasi: 5V
- Tegangan Input (recommended): 7 - 12 V
- Tegangan Input (limit): 6-20 V
- Pin digital I/O: 14 (6 diantaranya pin PWM)
- Pin Analog input: 6 input pin
- Arus DC per pin I/O: 40 mA
- Arus DC untuk pin 3.3 V: 150 mA
- Flash Memory: 32 KB dengan 0.5 KB digunakan sebagai *bootloader*
- SRAM: 2 KB
- EEPROM: 1 KB



- Kecepatan besaran waktu sebesar: 16 Mhz sebagai komponen untuk
(*Crystall oscillator*)

Untuk memberikan gambaran mengenai apa saja yang terdapat di dalam sebuah microcontroller, pada gambar berikut ini diperlihatkan contoh diagram blok sederhana dari microcontroller ATmega328 (dipakai pada Arduino Uno) seperti gambar blok diagram sederhana dibawah ini:



Gambar 2.11 Konfigurasi Pin ATMega328

(Sumber: <http://www.arduino.cc>, diakses tanggal 12 Mei 2017)

Blok-blok di atas dijelaskan sebagai berikut:

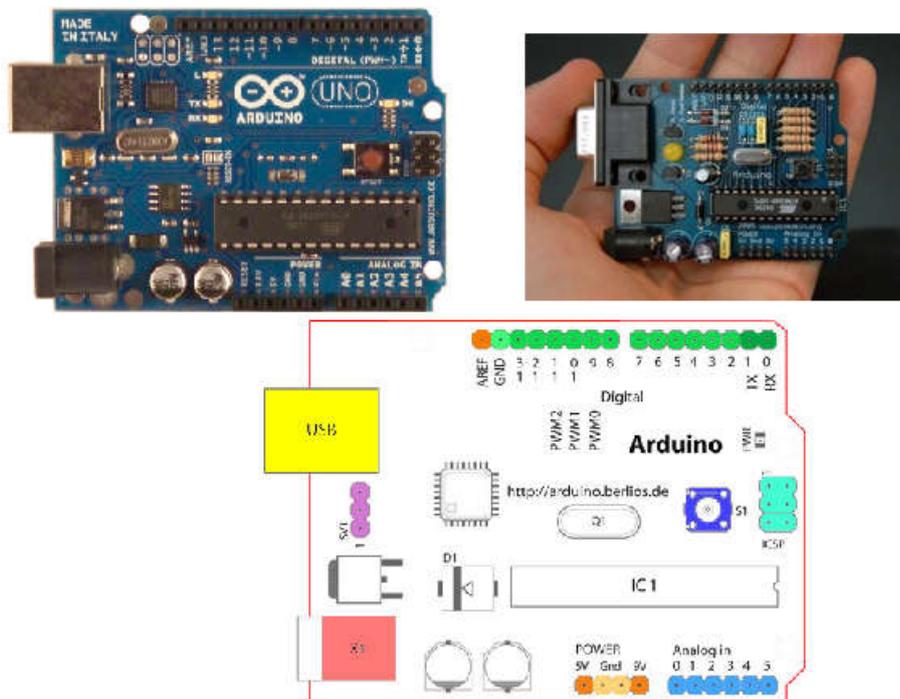
- *Universal Asynchronous Receiver/Transmitter (UART)* adalah antar muka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-422 dan RS-485.



- 2KB RAM pada memory kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variable-variabel di dalam program.
- 32KB RAM flash memory bersifat *non-volatile*, digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, flash memory juga menyimpan *bootloader*. **Bootloader** ini yang menjembatani antara **software compiler arduino** dengan mikrokontroler. Dan ketika pengguna papan mikrokontroler arduino menulis program tidak perlu banyak menuliskan sintak bahasa C, cukup melakukan pemanggilan fungsi program, hemat waktu dan pikiran.
- 1KB EEPROM bersifat non-volatile, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada papan Arduino.
- *Central Processing Unit (CPU)*, bagian dari microcontroller untuk menjalankan setiap instruksi dari program.
- Port input/output, pin-pin untuk menerima data (input) digital atau analog, dan mengeluarkan data (output) digital atau analog.

2.3.1 Konfigurasi Pin Arduino Uno

Setelah mengenal bagian-bagian utama dari microcontroller ATmega sebagai komponen utama, selanjutnya mengenal bagian-bagian dari papan Arduino itu sendiri. Dengan mengambil contoh sebuah papan Arduino tipe USB, bagian-bagiannya dapat dijelaskan sebagai berikut seperti pada gambar dibawah ini:



Gambar 2.12 Konfigurasi Pin Atmega328 Arduino Uno

(Sumber: <http://arduino.cc/en/main/hardware>, diakses pada tanggal 12 Mei 2017)

Tabel 2.2 Tabel karakteristik rangkaian pada board arduino uno

14 pin input/output digital (0-13)

Berfungsi sebagai input atau output, dapat diatur oleh program. Khusus untuk 6 buah pin 3, 5, 6, 9, 10 dan 11, dapat juga berfungsi sebagai pin analog output dimana tegangan output-nya dapat diatur. Nilai sebuah pin output analog dapat diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5V

USB

Berfungsi untuk:

- Memuat program dari komputer ke dalam papan
- Komunikasi serial antara papan dan komputer
- Memberi daya listrik kepada papan

Sambungan SV1



Sambungan atau *jumper* untuk memilih sumber daya papan, apakah dari sumber eksternal atau menggunakan USB.

Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis

Q1 – Kristal (*quartz crystal oscillator*)

Jika microcontroller dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada microcontroller agar melakukan sebuah operasi untuk setiap detak-nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).

Tombol Reset S1

Untuk me-reset papan sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol reset ini bukan untuk menghapus program atau mengosongkan microcontroller.

In-Circuit Serial Programming (ICSP)

Port ICSP memungkinkan pengguna untuk memprogram microcontroller secara langsung, tanpa melalui bootloader. Umumnya pengguna Arduino tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.

IC 1 – Microcontroller Atmega

Komponen utama dari papan Arduino, di dalamnya terdapat CPU, ROM dan RAM.

sum X1 – sumber

Jika hendak disuplai dengan sumber daya eksternal, papan Arduino dapat diberikan tegangan DC antara 9-12V.

6 pin input analog (0-5)

Pin ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca



nilai sebuah pin input antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

Tanpa melakukan konfigurasi apapun, begitu sebuah papan Arduino dikeluarkan dari kotak pembungkusnya ia dapat langsung disambungkan ke sebuah komputer melalui kabel USB. Selain berfungsi sebagai penghubung untuk pertukaran data, kabel USB ini juga akan mengalirkan arus DC sebesar: 5 Volt kepada papan Arduino sehingga praktis tidak diperlukan sumber daya dari luar. Saat mendapat suplai daya, lampu LED indikator daya pada papan Arduino akan menyala menandakan bahwa ia siap bekerja.



Gambar 2.12 Komponen LED Yang Terdapat Pada Board Arduino Uno

(Sumber: <http://arduino.cc/en/main/hardware>, diakses pada tanggal 12 Mei 2017)

Pada papan Arduino Uno terdapat sebuah LED kecil yang terhubung ke pin digital 13. LED ini dapat digunakan sebagai output saat seorang pengguna membuat sebuah program dan ia membutuhkan sebuah penanda dari jalannya program tersebut. Ini adalah cara yang praktis saat pengguna melakukan uji coba.

Umumnya microcontroller pada papan Arduino telah memuat sebuah program kecil yang akan menyalakan LED tersebut berkedip-kedip dalam jeda satu detik. Jadi sangat mudah untuk menguji apakah sebuah papan Arduino baru



dalam kondisi baik atau tidak, cukup sambungkan papan itu dengan sebuah komputer dan perhatikan apakah LED indikator daya menyala konstan dan LED dengan pin-13 itu menyala berkedip-kedip.

2.4 Faktor - Faktor Yang Mempengaruhi Kelembaban Udara

Kelembaban adalah konsentrasi uap air di udara. Angka konsentrasi ini dapat diekspresikan dalam kelembaban absolut, kelembaban spesifik atau kelembaban relatif. Alat untuk mengukur kelembaban disebut higrometer. Sebuah humidistat digunakan untuk mengatur tingkat kelembaban udara dalam sebuah bangunan dengan sebuah pengawalembap (dehumidifier). Dapat dianalogikan dengan sebuah termometer dan termostat untuk suhu udara. Perubahan tekanan sebagian uap air di udara berhubungan dengan perubahan suhu. Konsentrasi air di udara pada tingkat permukaan laut dapat mencapai 3% pada 30 °C (86 °F), dan tidak melebihi 0,5% pada 0 °C.

Kelembaban udara menggambarkan kandungan uap air di udara yang dapat dinyatakan sebagai kelembaban mutlak, kelembaban nisbi (relatif) maupun defisit tekanan uap air. Kelembaban mutlak adalah kandungan uap air (dapat dinyatakan dengan massa uap air atau tekanannya) per satuan volum. Kelembaban nisbi membandingkan antara kandungan/tekanan uap air aktual dengan keadaan jenuhnya atau pada kapasitas udara untuk menampung uap air. Kapasitas udara untuk menampung uap air tersebut (pada keadaan jenuh) ditentukan oleh suhu udara. Sedangkan defisit tekanan uap air adalah selisih antara tekanan uap jenuh dan tekanan uap aktual. Masing-masing pernyataan kelembaban udara tersebut mempunyai arti dan fungsi tertentu dikaitkan dengan masalah yang dibahas.

Semua uap air yang ada di dalam udara berasal dari penguapan. Penguapan adalah perubahan air dari keadaan cair ke keadaan gas. Pada proses penguapan diperlukan atau dipakai panas, sedangkan pada pengembunan dilepaskan panas. Seperti diketahui, penguapan tidak hanya terjadi pada permukaan air yang terbuka saja, tetapi dapat juga terjadi langsung dari tanah dan lebih-lebih dari tumbuh-tumbuhan. Penguapan dari tiga tempat itu disebut dengan Evaporasi (Karim, 1985).



Kelembaban udara dalam ruang tertutup dapat diatur sesuai dengan keinginan. Pengaturan kelembaban udara ini didasarkan atas prinsip kesetaraan potensi air antara udara dengan larutan atau dengan bahan padat tertentu. Jika ke dalam suatu ruang tertutup dimasukkan larutan, maka air dari larutan tersebut akan menguap sampai terjadi keseimbangan antara potensi air pada udara dengan potensi air larutan. Demikian pula halnya jika hidrat kristal garam-garam (salt crystal hydrate) tertentu dimasukkan dalam ruang tertutup maka air dari hidrat kristal garam akan menguap sampai terjadi keseimbangan potensi air.

Kelembaban relatif adalah istilah yang digunakan untuk menggambarkan jumlah uap air yang terkandung di dalam campuran air-udara dalam fasa gas.

Kelembaban relatif dari suatu campuran udara-air didefinisikan sebagai rasio dari tekanan parsial uap air dalam campuran terhadap tekanan uap jenuh air pada temperatur tersebut. Perhitungan kelembaban relatif ini merupakan salah satu data yang dibutuhkan (selain suhu, curah hujan, dan observasi visual terhadap vegetasi) untuk melihat seberapa kering areal perkebunan sehingga nantinya dapat ditentukan tingkat potensi kebakaran lahan.

Cara yang lebih praktis yaitu dengan menggunakan 2 termometer, yang basah dan kering. Prinsipnya semakin kering udara, maka air semakin mudah menguap. Karena penguapan butuh kalor maka akan menurunkan suhu pada termometer basah. Sedangkan termometer kering mengukur suhu aktual udara. Akibatnya jika perbedaan suhu antara keduanya semakin besar, maka artinya kelembaban relatif udara semakin rendah. Sebaliknya jika suhu termometer basah dan termometer kering sama, artinya udara berada pada kondisi lembab jenuh.

Tinggi rendahnya kelembaban udara di suatu tempat sangat bergantung pada beberapa faktor sebagai berikut:

- a. Suhu.
- b. Tekanan udara.
- c. Pergerakan angin.
- d. Kuantitas dan kualitas penyinaran.
- e. Vegetasi dsb.



f. Ketersediaan air di suatu tempat (air, tanah, perairan).

Suhu menunjukkan derajat panas benda. Mudahnya, semakin tinggi suhu suatu benda, semakin panas benda tersebut. Secara mikroskopis, suhu menunjukkan energi yang dimiliki oleh suatu benda. Setiap atom dalam suatu benda masing-masing bergerak, baik itu dalam bentuk perpindahan maupun gerakan di tempat berupa getaran. Makin tingginya energi atom-atom penyusun benda, makin tinggi suhu benda tersebut.