



**RANCANG BANGUN PROTOYPE SISTEM KONTROL PENGGUNAAN  
AIR PELANGGAN MENGGUNAKAN SENSOR FLOW METER DAN  
MOTOR SERVO BERBASIS MIKROKONTROLER**

**SKRIPSI**

Oleh

**Muh. Fajar Gunawan**

**NIM 131910201014**

**JURUSAN TEKNIK ELEKTRO STRATA 1  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2019**



**RANCANG BANGUN PROTOYPE SISTEM KONTROL PENGGUNAAN  
AIR PELANGGAN MENGGUNAKAN SENSOR FLOW METER DAN  
MOTOR SERVO BERBASIS MIKROKONTROLER**

**Skripsi**

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

Oleh

**Muh. Fajar Gunawan**

**NIM 131910201014**

**JURUSAN TEKNIK ELEKTRO STRATA 1  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2019**

## PERSEMBAHAN

Alhamdulillah, puji syukur kehadirat Allah SWT atas kasih setia-Nya yang telah melimpahkan segala rahmat yang tak ternilai, sehingga saya bisa 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 saya yaitu Bapak Muhammad dan Ibu Murtini serta kakak tercinta Raudatuljannah dan Munawaroh.
4. Teman-teman INTEL 2013 dan ELKA13 yang telah menjadikan kehidupan dimasa kuliah terasa sangat indah.
5. Ex-pengurus Robotika dan anggota UKM Robotika. Devry, Dika (Kambing), Riski (Bima), Rokhim, Rizky, Wandu (Bocil), Mas Wawan dan kawan-kawan terimakasih atas kebersamaan, kepedulian serta ilmu yang telah diberikan.
6. Seluruh asisten Laboratorium Teknik Elektro Universitas Jember. Terima kasih telah menjadi lawan diskusi yang menyenangkan.
7. Guru-guruku sejak TK hingga Perguruan Tinggi yang terhormat, terima kasih telah memberikan ilmu dan mendidik dengan penuh kesabaran.
8. Serta seluruh rekan-rekan yang penulis kenal dan rekan-rekan yang membaca skripsi ini. Terima kasih untuk segalanya.

**MOTTO**

**Aku adalah budak bagi orang yang mengajarku walaupun satu huruf  
(Sayidina Ali)**

**Segala sesuatu ada jalannya  
Jalan ke surga adalah ilmu pengetahuan  
(HR. Dailami)**



**PERNYATAAN**

Saya yang bertanda tangan dibawah ini:

Nama : Muh. Fajar Gunawan

NIM : 131910201014

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul ” Rancang Bangun Protoype Sistem Kontrol Penggunaan Air Pelanggan Menggunakan Sensor Flow Meter Dan Motor Servo Berbasis Mikrokontroler” 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, 31 Mei 2019

Yang menyatakan,

Muh. Fajar gunawan

NIM 131910201014

**SKRIPSI**

**RANCANG BANGUN PROTOYPE SISTEM KONTROL PENGGUNAAN  
AIR PELANGGAN MENGGUNAKAN SENSOR FLOW METER DAN  
MOTOR SERVO BERBASIS MIKROKONTROLER**

Oleh

Muh. Fajar Gunawan

NIM 131910201014

Pembimbing :

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

Dosen Pembimbing Anggota : Ali Rizal Chaidir, S.T., M.T.

**PENGESAHAN**

Skripsi berjudul “Rancang Bangun Protoype Sistem Kontrol Penggunaan Air Pelanggan Menggunakan Sensor Flow Meter Dan Motor Servo Berbasis Mikrokontroler” karya Muh. Fajar Gunawan telah diuji dan disahkan pada :

Hari : Jumat

Tanggal : 31 Mei 2019

Tempat : Fakultas Teknik Universitas Jember

Tim penguji,

Ketua,

Anggota I,

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

Ali Rizal Chaidir, S.T., M.T.  
NIP 760015754

Anggota II,

Anggota III,

S u m a r d I, S.T., M.T.  
NIP 196701131998021001

Catur Suko Sarwono ,S.T. Msi.  
NIP 196801191997021001

Mengesahkan  
Dekan,

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



## RINGKASAN

**Rancang Bangun Prototype Sistem Kontrol Penggunaan Air Pelanggan Menggunakan Sensor *Flowmeter* dan Motor Servo Berbasis Mikrokontroler;**  
Muh. Fajar Gunawan; 131910201014; 2019; 60 halaman; Jurusan Teknik Elektro  
Fakultas Teknik Universitas Jember.

Air merupakan kebutuhan pokok dalam kehidupan sehari-hari manusia dan mempunyai banyak kegunaan antara lain untuk minum, mandi, mencuci dan lain sebagainya. Di perkotaan, pelayanan jasa air bersih umumnya diselenggarakan oleh pemerintah. Air yang disalurkan ke rumah-rumah penduduk itu biasanya berasal dari air sungai yang ditampung terlebih dahulu di bak-bak penampungan (*reservoir*) yang kemudian diolah sehingga layak untuk digunakan dan memenuhi standar kualitas air. Pemerintah mempunyai cara untuk mengetahui jumlah air bersih yang digunakan warga/penduduk dengan memasang meteran pada pipa air yang masuk ke rumah-rumah. Selanjutnya, setiap bulan akan ada petugas yang mendatangi rumah penduduk dan mencatat volume air yang digunakan di masing-masing rumah penduduk. Setelah itu, akan dilakukan penghitungan terhadap penggunaan air selama sebulan tersebut dengan mengalikan harga setiap meter kubiknya menurut ketentuan yang telah ditetapkan berdasarkan tipe pengguna jasa air.

Namun hal ini menimbulkan permasalahan dalam pencatatan dengan cara seperti tersebut, bahkan seringkali terjadi kekeliruan. Seringkali data yang digunakan dalam perhitungan tidak sesuai dikarenakan petugas terkadang memperkirakan jumlah pemakaian air pelanggan rata-rata setiap bulannya. Akibatnya, konsumen merasa dirugikan dan hal ini dapat menurunkan tingkat kepercayaan konsumen dan menyebabkan perasaan curiga terhadap penyedia jasa air bersih.

Melihat permasalahan tersebut di atas dengan kemajuan ilmu pengetahuan dan teknologi, terutama dalam bidang elektronika dan instrumentasi dapat kita terapkan untuk mengatasi masalah tersebut, yaitu dengan mengontrol penggunaan



jumlah liter air. Jumlah liter air ini dapat diukur dengan menggunakan sensor laju aliran air, yang kemudian diproses oleh mikrokontroler dan ditampilkan dalam bentuk digital, berupa debit, arus dan biaya, sehingga pelanggan dapat lebih dimudahkan. Penelitian system penghitung penggunaan air ini memiliki tiga tahap yaitu. Pertama mendesain system perencanaan, merangkai system secara keseluruhan dengan mikrokontroler dan sensor aliran air. Tahap kedua yaitu inialisasi karakteristik sensor aliran air, dengan tujuan untuk mengetahui jumlah sinyal pulsa yang dimasukkan dan banyaknya liter air yang akan dikeluarkan. Tahap ketiga yaitu pengujian system yakni menguji kinerja motor servo ketika diberikan logika, lalu pengujian tampilan banyaknya jumlah liter air yang dikeluarkan pada LCD dan pengujian system secara keseluruhan.

Perancangan logika fuzzy dimulai dengan merancang fungsi keanggotaan (*membership function*) dari masukan dan keluaran, kemudian dirancang basis aturan (*Rule Base*) dan fungsi defuzzifikasi. Hasil rancang system di implementasikan dalam bentuk mikrokontroler menggunakan program Arduino. Alat pengering tembakau ini terintegrasi dengan sebuah webcam dengan interface USB yang dapat mendeteksi warna yang berkerja secara *real time*. Pemanfaat webcam ini dapat dikembangkan sehingga dapat digunakan sebagai sensor untuk memonitoring perubahan sebuah benda melalui beberapa tahap pengolahan citra.

Prinsip kerjanya adalah nilai *input* terdiri dari dua yaitu *error* dan *delta error* dari pembacaan sensor flowmeter yang akan dibandingkan dengan nilai set point, kemudian nilai tersebut akan diolah oleh FLC keluaran dari proses fuzzy logic controller berupa nilai derajat servo yang akan mengatur gerakan valve servo. *Membersip function* nilai error terdiri dari 5 himpunan, yaitu negatif besar (NB), negatif kecil (NK), Zero, positif kecil (PK), positif besar (PB). Nilai tersebut dibagi menjadi 3 rentang nilai yaitu negatif, nol, dan positif. *Membersip function* nilai delta error terdiri dari 5 himpunan, yaitu negatif besar (NB), negatif kecil (NK), Zero, positif kecil (PK), positif besar (PB)

## PRAKATA

Puji syukur kehadiran Allah SWT yang maha kuasa atas segalanya, karena dengan ridho, hidayah dan petunjukNya, penulis dapat menyelesaikan skripsi yang berjudul “Rancang Bangun Prototype Sistem Kontrol Penggunaan Air Pelanggan Menggunakan SensorFlowmeter dan Motor Servo Berbasis Mikrokontroler”. 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, Bapak Ali Rizal Chaidir, S.T., M.T. selaku dosen pembimbing yang telah membimbing menyelesaikan tugas akhir ini;
6. Bapak Sumardi S.T., M.T. , Bapak Catur Suko Sarwono, S.T., M.Si. selaku dosen penguji yang sudah memberikan saran untuk memperbaiki tugas akhir ini;
7. Keluarga besar INTEL’UJ 2013, terimakasih telah memberikan arti kekeluargaan 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, 31 Mei 2019

Penulis



DAFTAR GAMBAR

	Halaman
2.1 Board Arduino.....	6
2.2 Fisik dan skematik <i>flowmeter</i> .....	8
2.3 Contoh Gambar Bentuk <i>Keypad</i> .....	9
2.4 Contoh Gambar Bentuk <i>Buzzer</i> .....	10
2.5 Contoh Gambar Bentuk LCD .....	
11	
2.6 Contoh Gambar Bentuk Solenoid .....	
11	
2.7 Tahapan sistem infransi <i>fuzzy</i> .....	
20	

3.1	Diagram Alur Penelitian .....	21
3.2	Blok Diagram Perencanaan Sistem .....	23
3.3	Rancangan Sistem .....	24
3.4	Rancangan sistem servo valve .....	24
3.5	Blok Diagram <i>Fuzzy Logic</i> .....	25
3.6	Fungsi Keanggotaan Nilai Error .....	26
3.7	Fungsi Keanggotaan Nilai D_Error .....	27
3.8	Fungsi Keanggotaan Derajat Servo .....	27
3.9	Fungsi Keanggotaan <i>Output</i> Derajat Servo .....	28
3.10	Tabel <i>Rule Base Fuzzy</i> .....	28
3.11	Diagram Alir Perancangan .....	31

**DAFTAR TABEL**

Tabel 2.1	Proses <i>Flue-Curing</i> .....	9
Tabel 2.2	Spesifikasi sensor DHT22 .....	13
Tabel 2.3	Spesifikasi kamera Havit HV-V622 .....	15
Tabel 2.4	Parameter PID .....	20

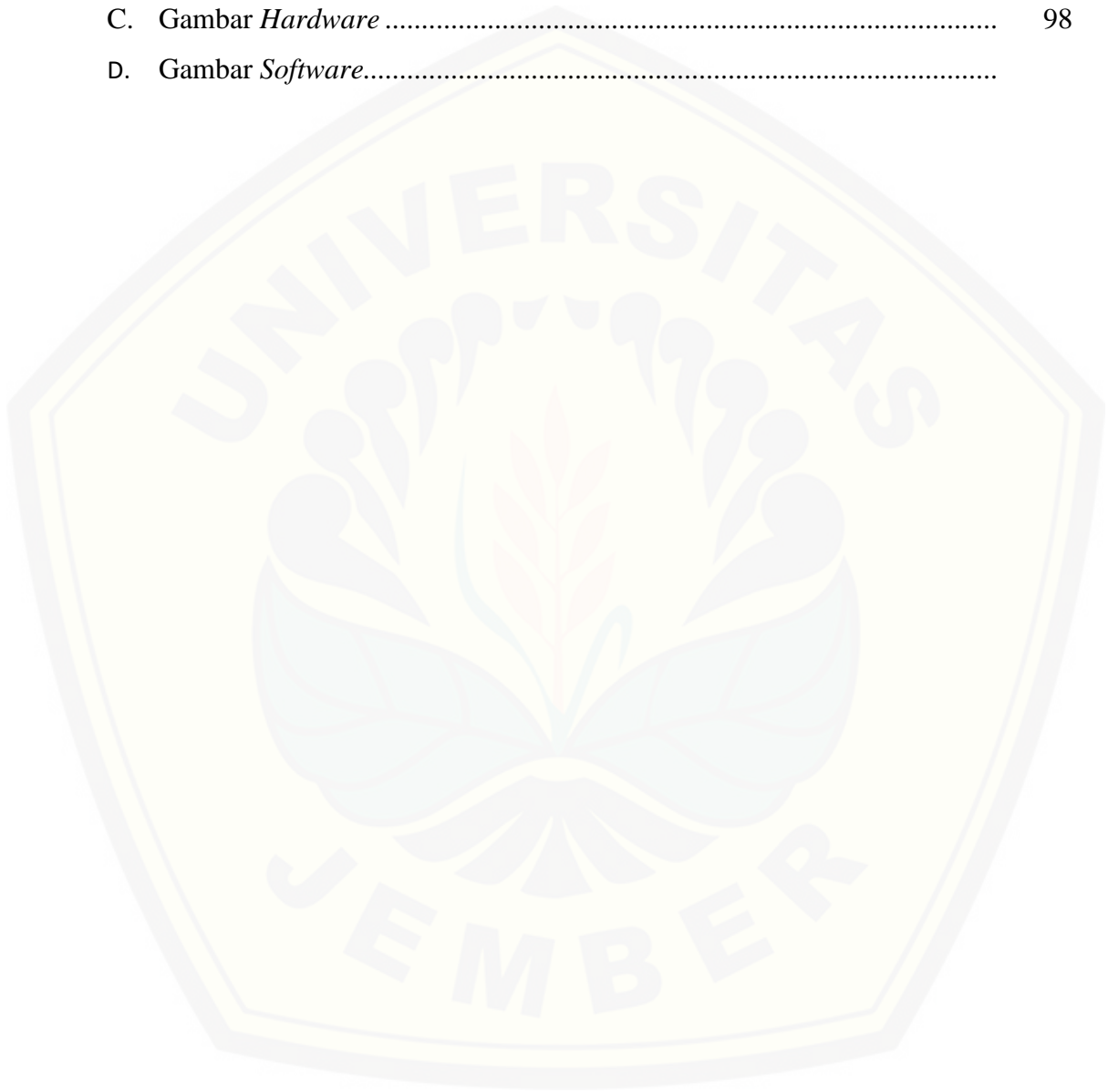
Tabel 2.5 Parameter PID untuk ZN tipe 1 .....	21
Tabel 2.6 Parameter PID Untuk ZN Tipe 2 .....	22
Tabel 3.1 <i>Timieline</i> Penelitian .....	26
Tabel 3.2 Daftar Peralatan yang Digunakan .....	29
Tabel 3.3 Indeks warna RGB daun tembakau.....	39
Tabel 3.4 <i>Rule output</i> Kp .....	53
Tabel 3.5 <i>Rule output</i> Ki .....	53
Tabel 3.6 <i>Rule output</i> Kd .....	53
Tabel 4.1 Perbandingan Sensor Suhu DHT22 dengan Digital Thermometer	60
Tabel 4.2 Perbandingan Hasil Kalibrasi Sensor Suhu Dht22 Dengan Digital Thermometer .....	61
Tabel 4.3 Nilai Aktual Terkoreksi Suhu Menggunakan Digital Termometer	62
Tabel 4.4 Nilai Aktual Terkoreksi Suhu Menggunakan Sensor DHT22 .....	62
Tabel 4.5 Perbandingan jumlah pixel saat mengalami perubahan tegangan sumber dengan tegangan awal 12,2volt dan jumlah pixel $\pm$ 47000 pixel ...	63
Tabel 4.6 Pengujian Citra Warna Kuning .....	64
Tabel 4.7 Pengujian Citra Warna Hijau .....	65
Tabel 4.8 Tanggapan Suhu dengan Kontrol P .....	67
Tabel 4.9 Tanggapan Suhu dengan Kontrol PI .....	68
Tabel 4.10 Tanggapan Suhu dengan Kontrol PD .....	69
Tabel 4.11 Tanggapan Suhu dengan Kontrol PID .....	70
Tabel 4.12 Perbandingan kontrol PID <i>tuning</i> manual dan <i>Ziegler-Nichols</i> ...	73
Tabel 4.13 Citra digital oven tembakau 1 tahap pengeringan .....	76
Tabel 4.14 Citra digital oven tembakau 2 tahap pengeringan .....	77
Tabel 4.15 Jumlah pixel warna kuning sebagai <i>setpoint</i> suhu .....	78
Tabel 4.16 Citra digital oven tembakau 4 tahap pengeringan dengan warna kuning sebagai <i>setpoint</i> .....	79
Tabel 4.17 Jumlah pixel warna hijau sebagai <i>setpoint</i> suhu .....	80
Tabel 4.18 Citra digital oven tembakau dengan 5 tahap pengeringan dengan warna kuning sebagai <i>setpoint</i> suhu .....	81





**DAFTAR LAMPIRAN**

A. <i>Listing</i> Program Arduino Mega 2560 .....	86
B. <i>Listing</i> Program VB .....	95
C. Gambar <i>Hardware</i> .....	98
D. Gambar <i>Software</i> .....	



## BAB 1. PENDAHULUAN

### 1) Latar Belakang

Air merupakan kebutuhan pokok dalam kehidupan sehari-hari manusia dan mempunyai banyak kegunaan antara lain untuk minum, mandi, mencuci dan lain sebagainya. Di perkotaan, pelayanan jasa air bersih umumnya diselenggarakan oleh pemerintah. Air yang disalurkan ke rumah-rumah penduduk itu biasanya berasal dari air sungai yang di tampung terlebih dahulu di bak-bak penampungan (*reservoir*) yang kemudian diolah sehingga layak untuk digunakan dan memenuhi standar kualitas air. Pemerintah mempunyai cara untuk mengetahui jumlah air bersih yang digunakan warga/penduduk dengan memasang meteran pada pipa air yang masuk ke rumah-rumah. Selanjutnya, setiap bulan akan ada petugas yang mendatangi rumah penduduk dan mencatat volume air yang digunakan di masing-masing rumah penduduk. Setelah itu, akan dilakukan penghitungan terhadap penggunaan air selama sebulan tersebut dengan mengalikan harga setiap meter kubiknya menurut ketentuan yang telah ditetapkan berdasarkan tipe pengguna jasa air.

Namun hal ini menimbulkan permasalahan dalam pencatatan dengan cara seperti tersebut, bahkan seringkali terjadi kekeliruan. Seringkali data yang digunakan dalam perhitungan tidak sesuai dikarenakan petugas terkadang memperkirakan jumlah pemakaian air pelanggan rata-rata setiap bulannya. Akibatnya, konsumen merasa dirugikan dan hal ini dapat menurunkan tingkat kepercayaan konsumen dan menyebabkan perasaan curiga terhadap penyedia jasa air bersih.

Melihat permasalahan tersebut diatas dengan kemajuan ilmu pengetahuan dan teknologi, terutama dalam bidang elektronika dan instrumentasi dapat kita terapkan untuk mengatasi masalah tersebut, yaitu dengan mengontrol penggunaan jumlah liter air. Jumlah liter air ini dapat diukur dengan menggunakan sensor laju aliran air, yang kemudian diproses oleh mikrokontroler dan ditampilkan dalam bentuk digital, berupa debit, arus dan biaya, sehingga pelanggan dapat lebih dimudahkan. Alat tersebut akan terrealisasikan dengan sebuah alat tugas proposal

yang diberi judul “Rancang Bangun Prototype Sistem Kontrol Penggunaan Air Pelanggan Menggunakan Sensor *Flowmeter* dan Motor Servo Berbasis Mikrokontroler”.

## 2) Rumusan Masalah

Berdasarkan latar belakang diatas maka rumusan masalah dari tugas proposal ini adalah

1. Bagaimana cara mengontrol jumlah dan biaya penggunaan liter air.
2. Bagaimana cara menerapkan logika fuzzy untuk mengontrol penggunaan air yang berlebih.

## 3) Tujuan Penelitian

Penelitian ini bertujuan untuk mengontrol jumlah liter, arus dan biaya penggunaan air pelanggan agar bisa lebih efektif, dengan menggunakan sensor *flowmeter* yang diaplikasikan ke dalam meteran air.

## 4) Manfaat Penelitian

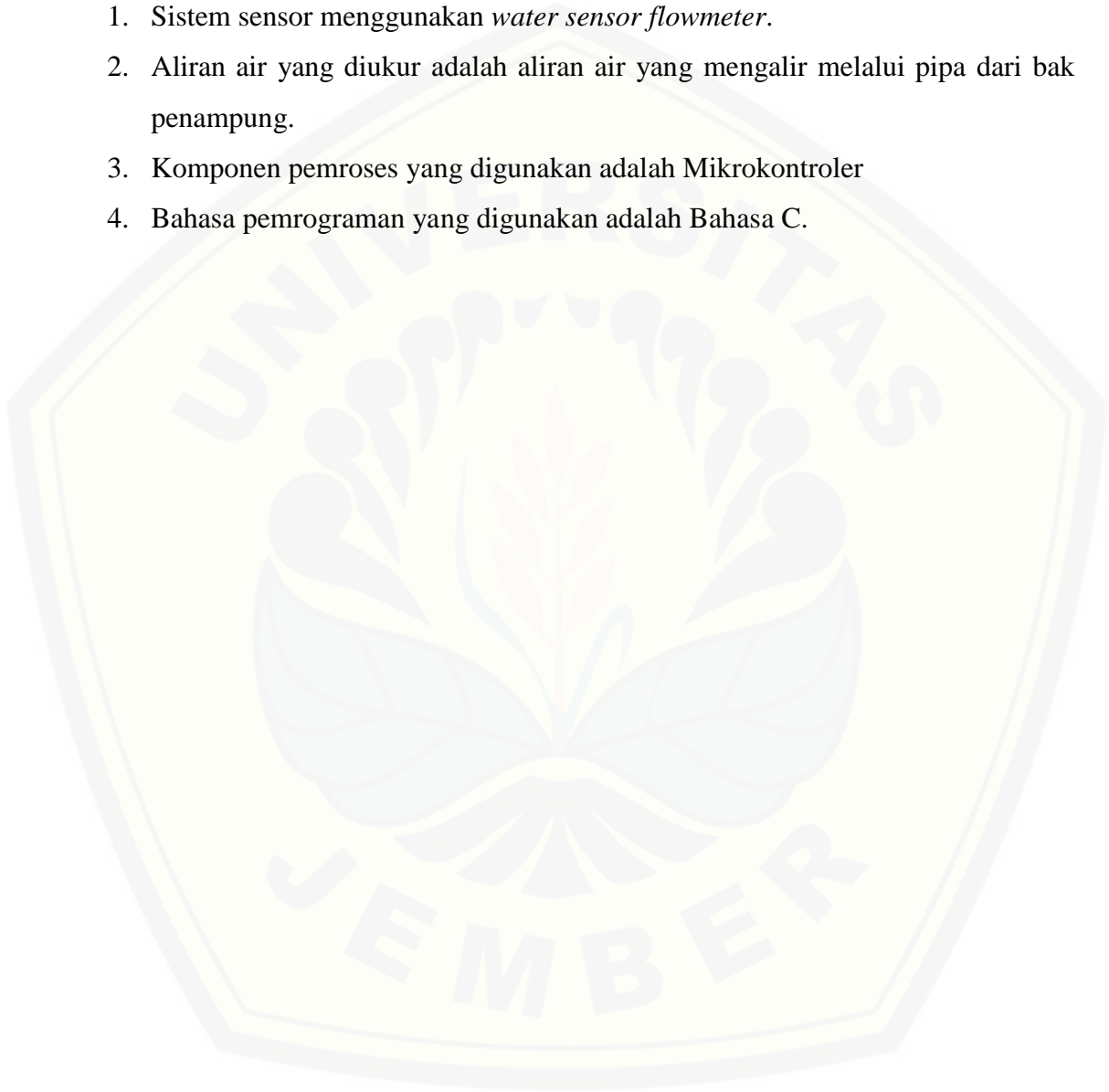
Penelitian ini diharapkan dapat bermanfaat untuk :

1. Memudahkan konsumen dalam penggunaan air yang lebih efektif.
2. Menghindari terjadinya kesalahan dalam pencatatan data penggunaan air oleh petugas penyedia jasa air.
3. Lebih menghemat biaya pengeluaran.
4. Mengontrol penggunaan air yang berlebih

### 5) Batasan Masalah

Agar penulisan dapat dikerjakan lebih fokus dan terarah, maka penelitian ini dibatasi oleh beberapa hal berikut :

1. Sistem sensor menggunakan *water sensor flowmeter*.
2. Aliran air yang diukur adalah aliran air yang mengalir melalui pipa dari bak penampung.
3. Komponen pemroses yang digunakan adalah Mikrokontroler
4. Bahasa pemrograman yang digunakan adalah Bahasa C.



## BAB 2. TINJAUAN PUSTAKA

### 2) Tinjauan Pustaka

Air merupakan sebuah kebutuhan pokok bagi semua orang. Umumnya Air yang disalurkan ke rumah-rumah penduduk itu biasanya berasal dari air sungai yang di tampung terlebih dahulu di bak-bak penampungan (*reservoir*) yang kemudian disalurkan menggunakan pipa menuju rumah warga. Pada daerah perkotaan untuk mengetahui jumlah liter air yang digunakan biasanya dipasang alat meteran pada pipa air. Selanjutnya, setiap bulan akan ada petugas yang mendatangi rumah penduduk dan mencatat volume air yang digunakan di masing-masing rumah penduduk. Setelah itu, akan melakukan penghitungan terhadap penggunaan air selama sebulan tersebut dengan mengalikan harga setiap meter kubiknya menurut ketentuan yang telah ditetapkan berdasarkan tipe pengguna jasa air. Namun hal ini menimbulkan permasalahan dalam pencatatan dengan cara seperti tersebut, bahkan seringkali terjadi kekeliruan. Seringkali data yang digunakan dalam perhitungan tidak sesuai dikarenakan petugas terkadang memperkirakan jumlah pemakaian air pelanggan rata-rata setiap bulannya. Akibatnya, pelanggan merasa dirugikan dan menyebabkan perasaan curiga terhadap penyedia jasa air bersih.

Melihat permasalahan tersebut diatas dengan kemajuan ilmu pengetahuan dan teknologi, terutama dalam bidang elektronika dan instrumentasi dapat kita terapkan untuk mengatasi masalah tersebut, yaitu dengan mengontrol penggunaan jumlah liter air. Jumlah liter air ini dapat diukur dengan menggunakan sensor laju aliran air, yang kemudian diproses oleh mikrokontroler dan ditampilkan dalam bentuk digital, berupa debit, arus dan biaya, sehingga pelanggan dapat lebih dimudahkan. Untuk itu kita perlu tau komponen dan teori apa saja yang akan digunakan.

#### 2.1) Sistem Token (prabayar)

Sistem prabayar biasa disebut dengan sistem pintar. Dimana pengguna dapat mengendalikan pemakaiannya sendiri. Pelanggan pasca bayar biasanya

terlebih dahulu menggunakan air yang digunakan, kemudian membayar tagihan rekening air dalam periode tertentu setiap bulan sesuai besarnya pemakaian. Dikatakan sebagai sistem Pra Bayar karena kita harus membayar terlebih dahulu alias membeli token pulsa sebelum kita menikmati fasilitas air yang diberikan oleh penyedia jasa. Jadi bayar/beli token dulu baru baru bisa menikmati fasilitas air. Sistemnya dibuat seperti membeli pulsa prabayar telepon genggam agar dapat memudahkan pengguna. Kelebihan lain dari sistem prabayar ini adalah pelanggan tidak perlu khawatir akan biaya keterlambatan yang menghantui, pemakaian air lebih terkendali, tanpa dikenakan denda keterlambatan, pembelian disesuaikan kemampuan, tidak ada masa aktif (aktif selama pulsa masih tersisa) dan terhindar dari kesalahan pencatatan meteran.

Terdapat tiga bagian utama dalam sistem rancangan ini yaitu :

1. Bagian *input* merupakan kode nilai pulsa yang didapatkan kemudian dimasukkan dalam kode *keypad*.
2. Bagian proses merupakan bagian yang ada di dalam arduino yakni pertama bagian deskripsi *input* pulsa untuk mendapatkan nilai set point prabayar, lalu hitung nilai prabayar untuk mendapatkan berapa liter air yang akan digunakan, lalu proses *counter* dilakukan setelah mendapatkan hasil dan perhitungan nilai *value*.
3. Bagian *output* terdiri dari tampilan LCD nilai pulsa dan aktuator berupa *solenoid valve* untuk membuka dan menutup aliran air.

## 2.2) Komponen Utama

### 2.2.1) Arduino Uno

Arduino adalah sebuah mikrokontroler yang mudah digunakan, karena menggunakan bahasa pemrograman basic yang menggunakan bahasa C. Arduino memiliki prosesor yang besar dan memori yang dapat menampung cukup banyak.

Arduino uno menggunakan *board* mikrokontroler yang didasarkan pada ATmega328, mempunyai 14 pin digital *input* dan *output* ( 6 diantaranya sebagai

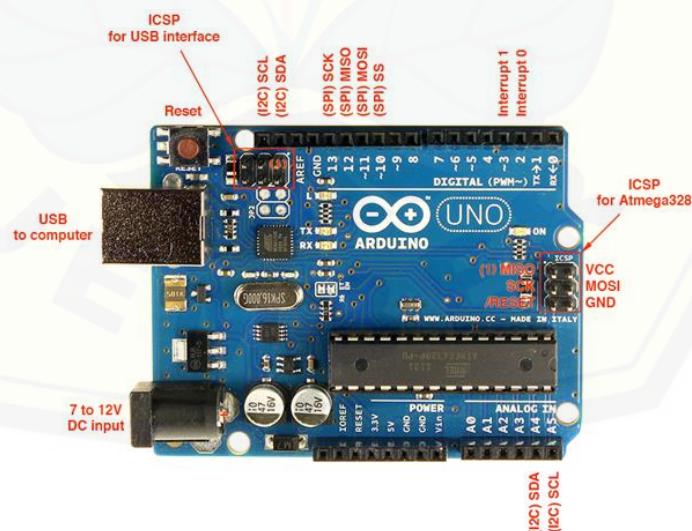


output PWM), 6 input analog yang merupakan osilator kristal 16Mhz, koneksi USB, power jack, ICSP header, dan tombol *reset*.(Dede Hendriono, 2011)

Arduinio uno dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal.Sumber daya dipilih secara otomatis.Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau *battery*. Adaptor dapat dihubungkan dengan mencolokkan sebuah *center-positive* plug yang panjangnya 2,1 mm ke power jack dari *board*. Kabel lead dari sebuah *battery* dapat dimasukkan dalam header/kepala pin *Ground* (Gnd) dan pin Vin dari konektor POWER.

Memory arduino, ATmega328 mempunyai 32 KB (dengan 0,5 KB digunakan untuk bootloader). ATmega 328 juga mempunyai 2 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis (RW/read and written) dengan EEPROM library).

Arduino UNO mempunyai sejumlah fasilitas untuk komunikasi dengan sebuah komputer, Arduino lainnya atau mikrokontroler lainnya.Atmega 328 menyediakan serial komunikasi UART TTL (5V), yang tersedia pada pin digital 0 (RX) dan 1 (TX).



Gambar 2.1 Board Arduino Uno R3(Dede Hendriono, 2011)

<http://arduino.cc/en/Main/arduinoBoardUno>



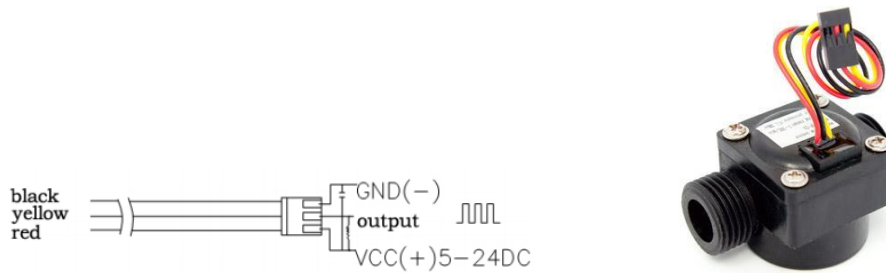
Deskripsi Arduino uno R3 :

Tabel 2.1 Deskripsi Arduino Uno

Mirkokontroler	ATmega328
Tegangan pengoperasian	5V
Tegangan input yang disarankan	7-12V
Batas tegangan input	6-20v
Jumlah pin I/O digital	14 (6 diantaranya output PWM)
Jumlah pin input analog	6
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Memori Flash	32 KB(ATmega328), 0.5kb bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	hz

### 2.2.2) *Flow meter*

*Water Flow sensor* terdiri dari tubuh katup plastik, rotor air, dan sensor hall efek. Ketika air mengalir melalui, gulungan rotor-rotor. Kecepatan perubahan dengan tingkat yang berbeda aliran. Sesuai sensor hall efek output sinyal pulsa. Kelebihan sensor ini adalah hanya membutuhkan 1 sinyal (SIG) selain jalur 5V dc dan Ground.



Gambar 2.2 Fisik dan skematik *flowmeter* (Johan, 2012 :66-67)

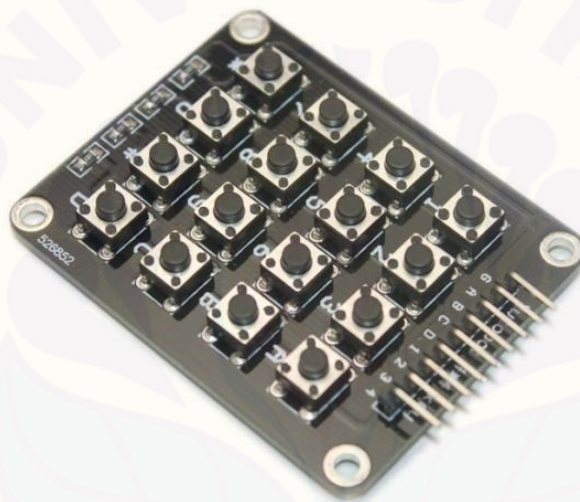
Spesifikasi sensor *Flowmeter* :

- a. Bekerja pada tegangan 5V DC-24VDC
- b. Arus Maksimum saat ini 15 mA (DC 5V)
- c. Berat sensor 43 g
- d. Tingkat Aliran rentang 0,5~ 60L / menit
- e. Suhu Pengoperasian 0°C~ 80°
- f. Operasi kelembaban 35%~ 90% RH
- g. Operasi tekanan bawah 1.75Mpa
- h. *Store temperature* -25°C~+80°
- i. *Store humidity* 25%~90%RH

*Water flow sensor* ini terdiri atas katup plastik, rotor air, dan sebuah sensor hall-effect. Prinsip kerja sensor ini adalah dengan memanfaatkan fenomena efek Hall. Efek Hall ini didasarkan pada efek medan magnetik terhadap partikel bermuatan yang bergerak. Ketika ada arus listrik yang mengalir pada divais efek Hall yang ditempatkan dalam medan magnet yang arahnya tegak lurus arus listrik, pergerakan pembawa muatan akan berbelok ke salah satu sisi dan menghasilkan medan listrik. Medan listrik terus membesar hingga gaya Lorentz yang bekerja pada partikel menjadi nol. Perbedaan potensial antara kedua sisi divais tersebut disebut potensial Hall. Potensial Hall ini sebanding dengan medan magnet dan arus listrik yang melalui divais. (Johan, 2012 )

### 2.2.3) Keypad

*Keypad* adalah rangkaian tombol yang berfungsi untuk memberi sinyal pada suatu rangkaian dengan menghubungkan jalur-jalur tertentu. *Keypad* merupakan komponen elektronik yang digunakan sebagai masukan, disusun dari beberapa tombol/switch dengan teknik matrix. Berdasarkan penjelasan tersebut, bahwa sebenarnya *keypad* merupakan tombol-tombol yang dirangkai menjadi sebuah paket dengan teknik menghubungkan satu tombol dengan tombol yang lain dengan teknik matrix. Teknik matrix adalah bisa dikatakan *array*, memiliki kolom dan baris lebih dari satu. (Budiono Ardi, 2012)



Gambar 2.3 contoh gambar keypad (Budiono Ardi, 2012)

### 2.2.4) Buzzer

*Buzzer* adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loud speaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. *Buzzer* ini digunakan sebagai indikator (alarm). (Ali Muhammad, 2012)

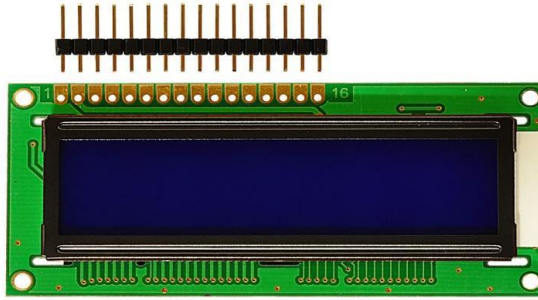
Gambar 2.4 contoh *Buzzer* (Ali Muhammad, 2012)

### 2.2.5) LCD (*Liquid Crystal Display*)

LCD (*Liquid Crystal Display*) adalah salah satu jenis *display* elektronik 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 Crystal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik. (Elektronika Dasar, 2015)

Tabel 2.2 Konfigurasi pin-pin LCD (*Liquid Crystal Display*)

No.Pin	Nama	Keterangan
1	GND	<i>Ground</i>
2	VCC	+5V
3	VEE	<i>Contras</i>
4	RS	<i>Register Select</i>
5	RW	<i>Read/write</i>
6	E	<i>Enable</i>
7-14	D0-D7	<i>Data bit 0-7</i>
15	A	<i>Anoda (back light)</i>
16	K	<i>Katoda (back light)</i>



Gambar 2.5 contoh LCD(Elektronika Dasar, 2015)

### 2.2.6) Motor Servo

Pengembangan dari DC Motor dan sudah memiliki Gear, tapi bedanya Motor Servo bisa diatur kecepatannya. Di dalamnya terdapat Potensio Meter dan Driver Motor, sehingga bisa diatur dengan derajat. Motor servo bisa digunakan untuk membuat robot berkaki, robot lengan, dan lain-lain. Batas maksimal pergerakan putar kanan hingga 180 derajat dan putar kiri hingga 180 derajat.



Gambar 2.6 schematic Motor Servo (Rochayati,2012)

Motor servo adalah sebuah perangkat atau aktuator putar (motor) yang dirancang dengan sistem kontrol umpan balik loop tertutup (servo), sehingga dapat di set-up atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. Motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer. Serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas putaran poros motor servo.



### 2.3) Pengendali Logika Fuzzy / Fuzzy Logic Controller

#### 2.3.1) Pengertian Fuzzy Logic

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* kedalam suatu ruang *output*. Titik awal dari konsep modern mengenai ketidakpastian adalah paper yang dibuat oleh Lofti A Zadeh (1972), dimana Zadeh memperkenalkan teori yang memiliki obyek-obyek dari *himpunan fuzzy* yang memiliki batasan yang tidak presisi dan keanggotaan dalam himpunan *fuzzy*, dan bukan dalam bentuk logika benar (*true*) atau salah (*false*), tapi dinyatakan dalam derajat (*degree*). Konsep seperti ini disebut dengan *Fuzziness* dan teorinya dinamakan *Fuzzy Set Theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri. Seringkali ditemui dalam pernyataan yang dibuat oleh seseorang, evaluasi dan suatu pengambilan keputusan.

*Fuzzy system* (sistem kabur) didasari atas konsep himpunan kabur yang memetakan domain *input* kedalam domain *output*. Perbedaan mendasar himpunan tegas dengan himpunan kabur adalah nilai keluarannya. Himpunan tegas hanya memiliki dua nilai *output* yaitu nol atau satu, sedangkan himpunan kabur memiliki banyak nilai keluaran yang dikenal dengan nilai derajat keanggotaannya.

Logika *fuzzy* adalah peningkatan dari logika *Boolean* yang berhadapan dengan konsep kebenaran sebagian. Dimana logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah *binary* (0 atau 1, hitam atau putih, ya atau tidak). Logika *fuzzy* menggantikan kebenaran *Boolean* dengan tingkat kebenaran. Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistic*, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat”. Logika ini diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Berkeley pada tahun 1965. Logika *fuzzy* telah digunakan pada bidang-bidang seperti taksonomi, topologi, linguistik, teori automata, teori pengendalian, psikologi, *pattern recognition*, pengobatan, hukum, *decision analysis*, *system theory and information retrieval*. Pendekatan *fuzzy* memiliki kelebihan pada hasil yang terkait dengan sifat kognitif manusia,

khususnya pada situasi yang melibatkan pembentukan konsep, pengenalan pola, dan pengambilan keputusan dalam lingkungan yang tidak pasti atau tidak jelas.

Ada beberapa alasan mengapa orang menggunakan logika *fuzzy* (Kusumadewi 2003) antara lain:

1. Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
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.
6. Logika *fuzzy* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

### 6.3.2) Konsep *FuzzyLogic*

Konsep penggunaan *FuzzyLogic* antara lain:

1. *FuzzyLogic* umumnya diterapkan pada masalah-masalah yang mengandung unsur ketidakpastian (*uncertainty*), ketidaktepatan (*imprecise*), *noisy*, dan sebagainya.
2. *FuzzyLogic* menjembatani bahasa mesin yang presisi dengan bahasa manusia yang menekankan pada makna atau arti (*significance*).
3. *FuzzyLogic* dikembangkan berdasarkan cara berfikir manusia

### 6.3.3) Himpunan *Fuzzy*

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item  $x$  dalam suatu himpunan  $A$ , yang sering ditulis dengan  $\mu_A[x]$ , memiliki 2 yaitu:

1. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau



2. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan

Terkadang kemiripan antara keanggotaan *fuzzy* dengan probabilitas menimbulkan kerancuan. Keduanya memiliki nilai pada interval  $[0,1]$ , namun interpretasi nilainya sangat berbeda antara kedua kasus tersebut. Keanggotaan *fuzzy* memberikan suatu ukuran terhadap pendapat atau keputusan, sedangkan probabilitas mengindikasikan proporsi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Misalnya, jika nilai keanggotaan bernilai suatu himpunan *fuzzy* USIA adalah 0,9 maka tidak perlu dipermasalahkan berapa seringnya nilai itu diulang secara individual untuk mengharapkan suatu hasil yang hampir pasti muda. Di lain pihak, nilai probabilitas 0,9 usia berarti 10% dari himpunan tersebut diharapkan tidak muda. Himpunan *fuzzy* memiliki 2 atribut, yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 40, 25, 50, dsb.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* (Aplikasi logika *fuzzy* untuk pendukung keputusan, Sri Kusumadewi, Hari Purnomo, Edisi kedua, Graha Ilmu, 2010), yaitu:

- a. Variable *fuzzy* Variable *fuzzy* merupakan variabel yang hendak dibahas dalam suatu system *fuzzy*. Contoh: umur, temperature, permintaan, dsb.
- b. Himpunan *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. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Ada kalanya nilai semesta pembicaraan ini tidak dibatasibatas atasnya.

Contoh:

a. Semesta pembicaraan untuk variabel mahasiswa:  $[0 \ 50]$

b. Semesta pembicaraan untuk variabel dosen:  $[0 \ 50]$

d. Domain

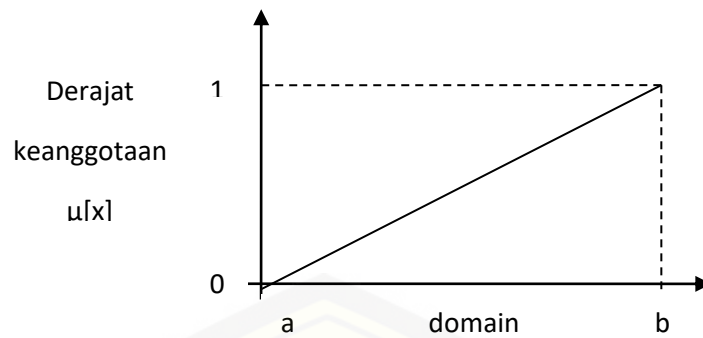
Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif dan bilangan negatif.

#### 6.3.4) Fungsi Keanggotaan

Fungsi keanggotaan 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. Apabila  $U$  menyatakan himpunan universal dan  $A$  adalah himpunan fungsi *fuzzy* dalam  $U$ ,  $Then A$  dapat dinyatakan sebagai pasangan terurut (Wang, 1997 yang dirujuk Wulandari, F. 2005). Ada beberapa fungsi yang bisa digunakan.

a. Representasi Linear

Pada representasi linear, pemetaan input ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas. 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 (Kusumadewi S, Purnomo H, 2010). Seperti terlihat pada gambar 2.16

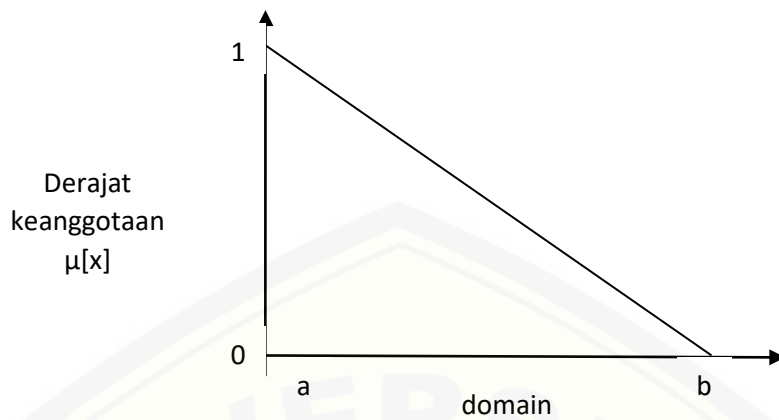


Gambar 6.5 Representasi linear naik (Kusumadewi S, Purnomo H, 2010)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x-a) / (b-a); & a < x < b \\ 1; & x \geq b \end{cases} \dots\dots\dots( 2.3 )$$

Kedua, merupakan kebalikan dari 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 pendek. Seperti terlihat pada gambar diatas.



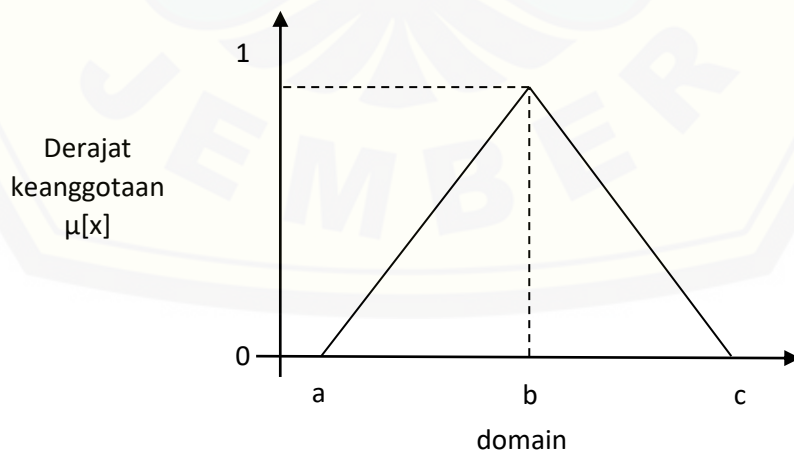
Gambar 6.6 Representasi linear turun (Kusumadewi S, Purnomo H, 2010)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \geq b \\ (b-x) / (b-a) & a < x < b \dots\dots\dots ( 2.4 \\ 1; & x \leq a \end{cases}$$

b. Representasi kurva segitiga

Kurva segitiga pad dasarnya merupakan gabungan antara 2 garis (*linear*).  
Seperti terlihat pada gambar 2.18.



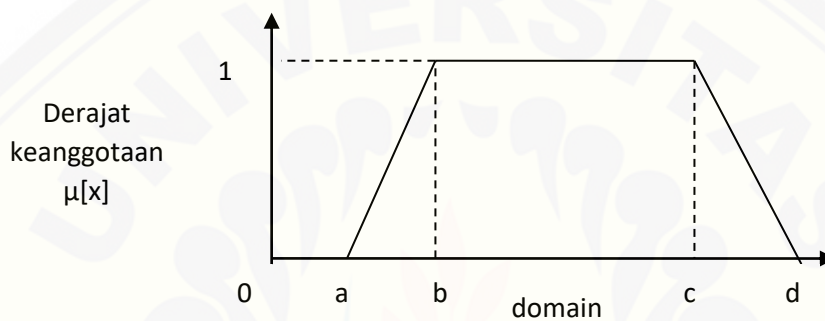
Gambar 6.7 kurva segitiga (Kusumadewi S, Purnomo H, 2010)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \geq c \text{ atau } x \leq a \\ (x-a) / (b-a) & a < x < b \\ (c-x) / (c-b) & b < x < c \end{cases} \dots\dots\dots (2.5)$$

c. Representase kurva trapezium (Kusumadewi S, Purnomo H, 2010)

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada titik yang memiliki nilai keanggotaan 1. Seperti terlihat pada gambar 2.19.



Gambar 2.19 Representasi kurva trapezium (Kusumadewi S, Purnomo H, 2010)

Fungsi keanggotaan:

$$\mu(x) = \begin{cases} 0; & x \geq d \text{ atau } x \leq a \\ \frac{(x-a)}{(b-a)}; & a < x < b \\ \frac{(d-x)}{(d-c)}; & c < x < d \\ 1; & b \leq x \leq c \end{cases} \dots\dots\dots(2.6)$$

**6.3.4) Sistem Inferensi Fuzzy**

Salah satu aplikasi logika fuzzy yang telah berkembang amat luas dewasa ini adalah sistem inferensi fuzzy (*Fuzzy Inference System / FIS*), merupakan suatu kerangka komputasi yang di dasarkan pada teori himpunan fuzzy, aturan fuzzy berbentuk IF – THEN, dan penalaran fuzzy. (Kusumadewi S, Hartati S, 2010).

System inferansi fuzzy menggunakan metode SUGENO, memiliki karakteristik yaitu konsekuen tidak merupakan himpunan fuzzy, namun merupakan suatu persamaan linier dengan variabel-variabel sesuai dengan variabel-variabel inputnya. (Kusumadewi S, Hartati S, 2010).

### 6.3.5) Penegasan (defuzzifikasi)

Input dari proses defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari suatu komposisi aturan – aturan *fuzzy*, sedangkan output yang dihasilkan merupakan suatu bilangan pada himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam range tertentu, maka harus dapat diambil suatu nilai crisp tertentu sebagai output.

Fuzzifikasi merupakan fase pertama dari perhitungan *fuzzy*, yaitu mengubah masukan - masukan yang nilai kebenarannya bersifat pasti ke dalam bentuk *fuzzy input* yang berupa tingkat keanggotaan / tingkat kebenaran. Fuzzifikasi adalah proses perubahan suatu nilai *crisp* ke dalam variabel *fuzzy* yang berupa variabel linguistik yang nantinya akan dikelompokkan menjadi himpunan *fuzzy*. Dengan demikian, tahap ini mengambil nilai-nilai *crisp* dan menentukan derajat di mana nilai-nilai tersebut menjadi anggota dari setiap himpunan *fuzzy* yang sesuai.

Ada beberapa metode defuzzifikasi pada komposisi aturan Mamdani, antara lain:

- a. Metode Centroid (Composite Moment)
- b. Metode Largest of Maximum (LOM)
- c. Metode Mean of Maksimum (MOM)
- d. Metode Bisektor

Pada tahap defuzzifikasi penulis menggunakan salah satu metode defuzzifikasi yaitu Metode Bisektor. Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai pada domain *fuzzy* yang memiliki nilai keanggotaan setengah dari jumlah total nilai keanggotaan pada daerah *fuzzy*. Secara umum dituliskan:

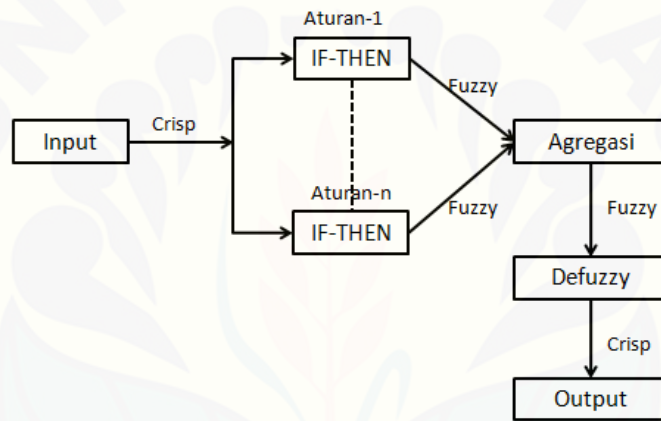


$$\mu_{(d)} = \frac{1}{2} \sum_{i=1}^n U_{A_i}(d_i) \dots\dots\dots(2.7)$$

Keterangan:

- d = nilai hasil penegasan (defuzzifikasi),
- $d_i$  = nilai keluaran pada aturan ke-i,
- $\mu_{A_i}(d_i)$  = derajat keanggotaan nilai keluaran pada aturan ke-i,
- n = banyak aturan yang digunakan

Secara garis besar sistem inferensi *fuzzy* dapat dilihat pada gambar 2.20.



Gambar 2.7 Tahapan sistem infransi *fuzzy*(Kusumadewi, 2006)



### BAB 3. METODE PENELITIAN

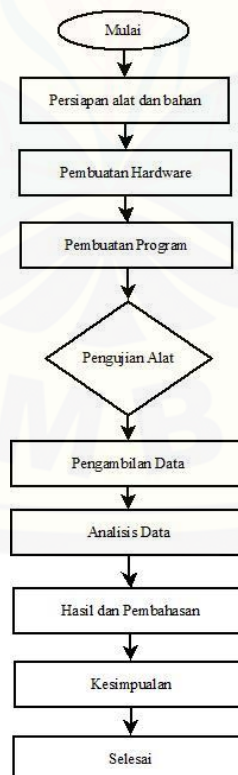
#### 3) Metode penelitian

##### 3.1) Rancangan Penelitian

Penelitian system penghitung penggunaan air ini memiliki tiga tahap yaitu. Pertama mendesain system perencanaan, merangkai system secara keseluruhan dengan mikrokontroler dan sensor aliran air. Tahap kedua yaitu inisialisasi karakteristik sensor aliran air, dengan tujuan untuk mengetahui jumlah sinyal pulsa yang dimasukkan dan banyaknya liter air yang akan dikeluarkan. Tahap ketiga yaitu pengujian system yakni menguji kinerja motor servo ketika diberikan logika, lalu pengujian tampilan banyaknya jumlah liter air yang dikeluarkan pada LCD dan pengujian system secara keseluruhan.

##### 3.2) Alur Penelitian

Proses penelitian dilakukan dengan melalui beberapa tahap seperti berikut ini:



Gambar 3.1 Diagram Alur Penelitian

### 3.3) Tempat Dan Waktu

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

Tempat : -Lab. Elektronika Terapan Fakultas Teknik Universitas Jember

- Kontrakan Kaliurang

Alamat : - Jl.SlametRiyadi No. 62, Patrang, KabupatenJember

- Jl.Kaliurang, Perumahan Kaliurang Puncak

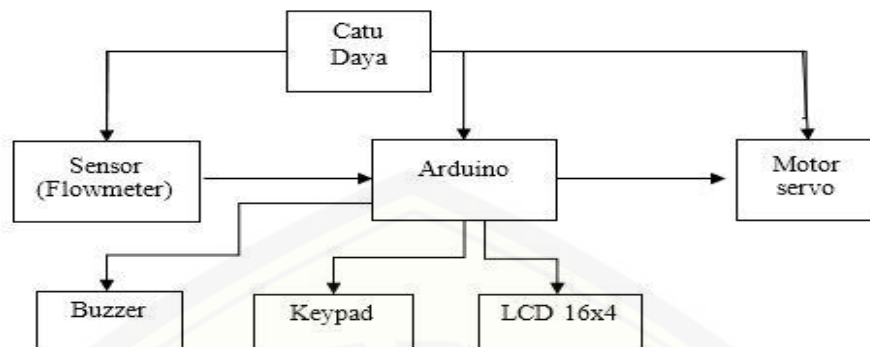
Waktu : Januari 2019 – April 2019

#### Rencana Jadwal Pelaksanaan Penelitian

No	Kegiatan	Bulan															
		Januari				Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Persiapan	■	■														
2	Studi Literatur		■	■	■												
3	Pembuatan Desain				■	■	■	■	■								
4	Pembelian Alat dan Bahan		■	■	■	■	■										
5	Pembuatan/ Perakitan						■	■	■	■	■	■	■				
6	Melakukan Pengujian Alat												■	■	■		
7	Analisa Data												■	■	■	■	
8	Penulisan Laporan dan Kesimpulan													■	■	■	■

Tabel 7.2 Rencana Jadwal Pelaksanaan Penelitian

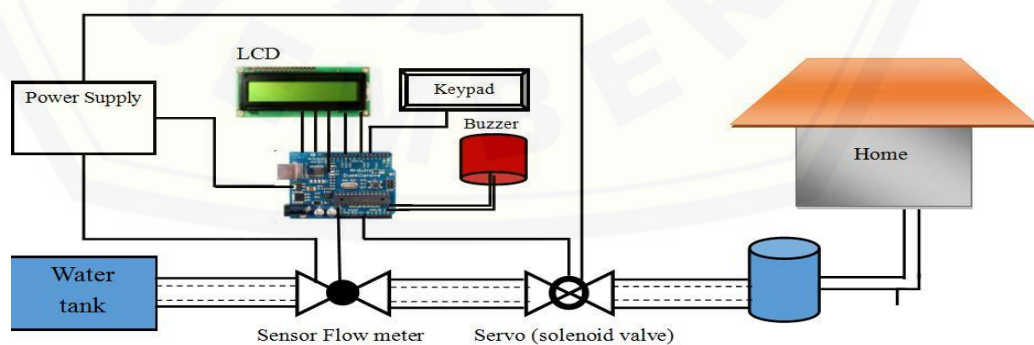
### 3.4) Desain Penelitian dan Rancangan Penelitian



Gambar 3.2 Blok Diagram Perencanaan Sistem

Dari gambar blok diagram di atas terdapat beberapa tahap yaitu, air yang mengalir akan melewati sensor (*flowmeter*) pembacaan *flowmeter* sensor, berfungsi untuk mengetahui jumlah pulsa atau liter air yang akan digunakan sehingga banyaknya jumlah air akan terkontrol. Keluaran dari pembacaan sensor air tersebut akan di kirim menuju Arduino untuk diproses yang kemudian akan ditampilkan oleh LCD 16x4 berupa jumlah liter air yang digunakan dan menampilkan batas pemakaian, sehingga apa bila jumlah liter air yang digunakan sudah mencapai batas maka aliran air akan di putus dan ditutup oleh solenoid valve dan *buzzer* pun akan berbunyi.

#### 3.5.1) Rancangan Sistem

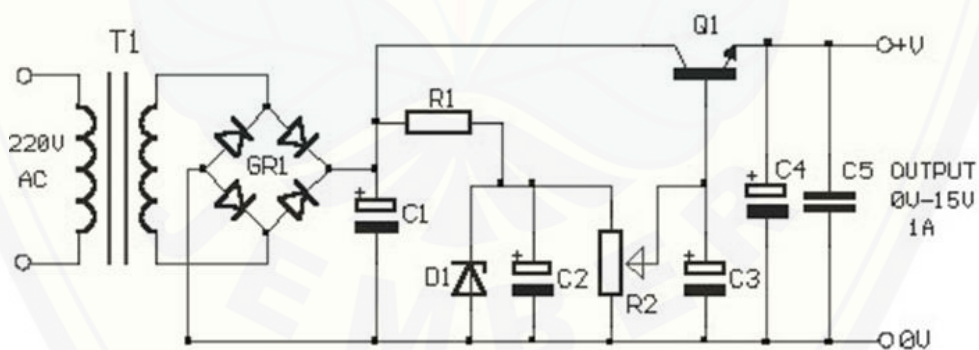


Gambar 3.3 Rancangan Sistem



Gambar.3.4 gambar alat secara keseluruhan

### 3.5) Desain Rangkain Alat

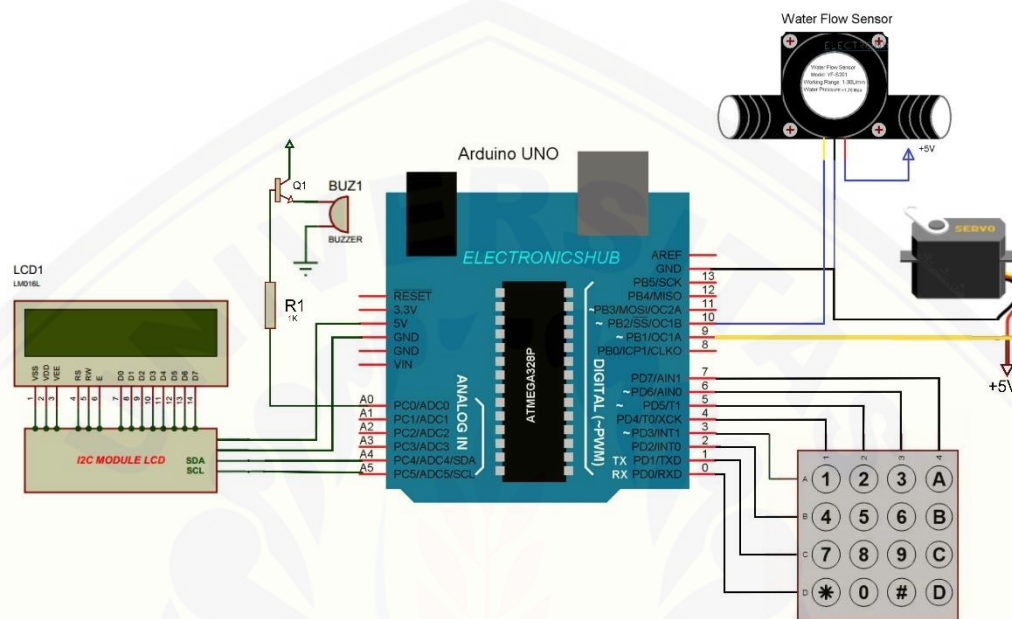


Gambar 3.5 Gambar Rangkaian Catu daya (power supply)

Power supply alias yang juga dikenal dengan sebutan catu daya merupakan suatu alat yang dipakai untuk menyediakan energi listrik untuk perangkat-perangkat elektronika.. Prinsip kerja dari power supply ini yakni input yang diterima oleh rangkaian power supply berupa arus AC yang kemudian diturunkan tegangannya melewati komponen transformator. Sesuai arus AC

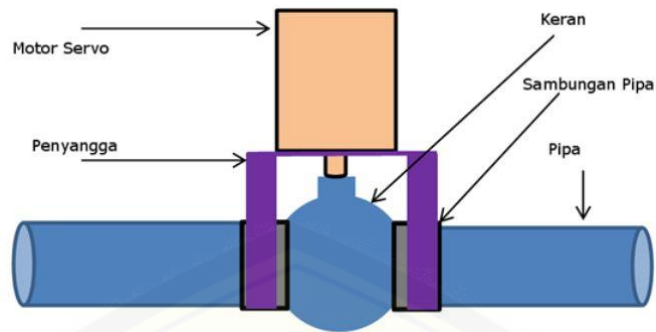


tersebut turun menjadi kualitas tertentu, komponen dioda yang bertugas menyearahkan arus AC menjadi DC. Jadi, arus DC tersebutlah yang menjadi keluaran dari power supply yang nantinya dialirkan langsung menuju prangkat-perangkat elektronika jadi bisa dimanfaatkan sebagaimana mustinya.



Gambar 3.6 Gambar Rangkaian Alat dengan Arduino

Dari gambar rangkaian diatas dapat kita lihat bahwa komponen-komponen terhubung dengan pin pada arduino dimana keypad terhubung dengan 8 pin digital arduino yakni diantaranya adalah pin 0 hingga pin 7 kemudian pada sesnsor flow meter terdiri dari 3 keluaran yakni Vcc, ground dan data dihubungkan pada digital pin 10 pada arduino, lalu aktuator servo memiliki 3 keluaran diantaranya yakni kabel merah terhubung pada Vcc, kabel hitam pada ground dan kabel kuning pada pin digital 9 pada arduino, kemudian komponen LCD dihubungkan menggunakan modul I2C dimana keluarannya terdiri dari 4 output yakni VCC, ground, SCL dan SDA diman SDA dan SCL dihubungkan dengan pin analog A5 dan A0 pada arduino dan kemudian komponen buzzer memiliki 2 keluaran yakni ground dan data yang dihubungkan pada pin A0 pada arduino.



Gambar 3.6 Rancangan sistem servo valve

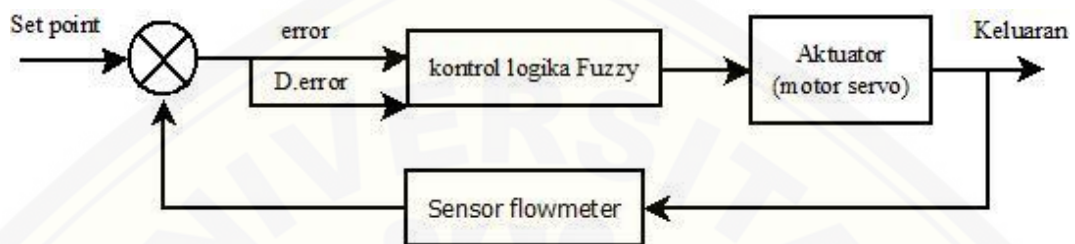
Dari gambar 3.3 desain diatas dapat kita jelaskan bahwa sistem bekerja dengan sistem alur sebagai berikut, air yang dialiri oleh kran akan mengalir melewati sensor (*flowmeter*) pembacaan *flowmeter* sensor, berfungsi untuk mengetahui jumlah liter air yang akan digunakan sehingga banyaknya jumlah air akan terkontrol. Dari pembacaan sensor akan di proses oleh Arduino sehingga pembacaan jumlah air dan arus yang keluar dapat terkontrol dari *input* yang di masukkan melalui *keypad* kita dapat mengatur berapa jumlah dan besar arus yang akan kita gunakan sesuai dengan kebutuhan yang diinginkan yang kemudian akan dikirim menuju Arduino untuk diproses yang kemudian akan ditampilkan oleh LCD 16x4 berupa jumlah liter air dan besar arus yang digunakan dan menampilkan batas pemakaian, setiap aliran air yang keluar akan di baca oleh sensor yang nantinya keluaran akan menentukan system kerja dari servo, sehingga apabila jumlah liter air yang digunakan sudah mencapai batas maka aliran air akan diputus dan ditutup oleh servo(solenoid valve) dan buzzer pun akan berbunyi. Sudut servo yang digunakan yakni dirange antara 90° hingga 180° dengan kondisi 90° dalam keadaan kran terbuka dan kondisi 180° dalam keadaan kran tertutup.



### 3.6) Metode Analisis Data dan Pengujian Hipotesis

#### 7.6.1) Perancangan Sistem Logika Fuzzy

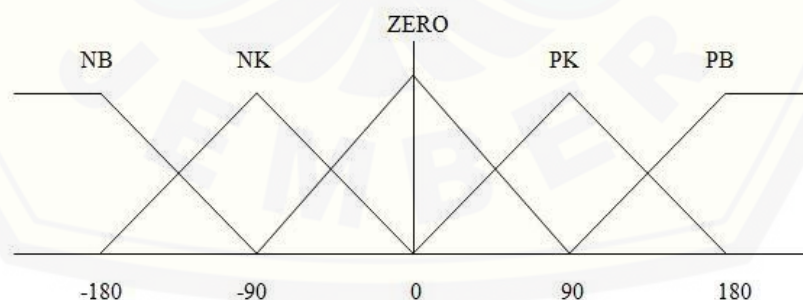
Perancangan logika fuzzy dimulai dengan merancang fungsi keanggotaan (*membership function*) dari masukan dan keluaran, kemudian dirancang basis aturan (*Rule Base*) dan fungsi defuzzifikasi. Hasil rancang system di implementasikan dalam bentuk mikrokontroler menggunakan program Arduino.



Gambar 3.5 Blok Diagram *Fuzzy Logic*

Prinsip kerjanya adalah nilai *input* terdiri dari dua yaitu *error* dan *delta error* dari pembacaan sensor flowmeter yang akan dibandingkan dengan nilai set point, kemudian nilai tersebut akan diolah oleh FLC keluaran dari proses fuzzy logic controller berupa nilai derajat servo yang akan mengatur gerakan valve servo.

*Membership function* nilai error terdiri dari 5 himpunan, yaitu negatif besar (NB), negatif kecil (NK), Zero, positif kecil (PK), positif besar (PB). Nilai tersebut dibagi menjadi 3 rentang nilai yaitu negatif, nol, dan positif



Gambar 3.6 fungsi keanggotaan nilai error

Secara matematis, fungsi keanggotaan nilai *input* error dapat ditulis sebagai berikut :

persamaan fungsi keanggotaan

$$f(x, a, b, c) = \begin{cases} 0; & x \geq a \text{ atau } x \leq b \\ \frac{(x-a)}{(b-a)}; & a < x < b \\ \frac{(c-x)}{(c-b)}; & b < x < c \\ 1; & 0, x > c \end{cases} \dots\dots\dots(7.1)$$

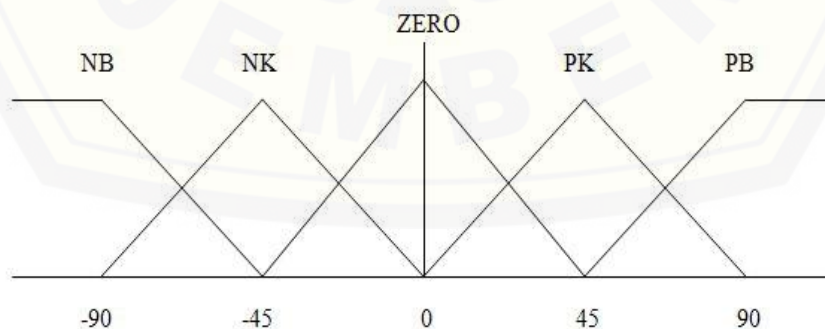
Fungsi keanggotaan negatif besar (NB)

$$NB(x; a, b) = \begin{cases} 1; & x \leq a \\ \frac{(b-x)}{(b-a)}; & a \leq x \leq b \\ 0; & x \geq b \end{cases} \dots\dots\dots(7.2)$$

Fungsi keanggotaan negatif kecil(NK)

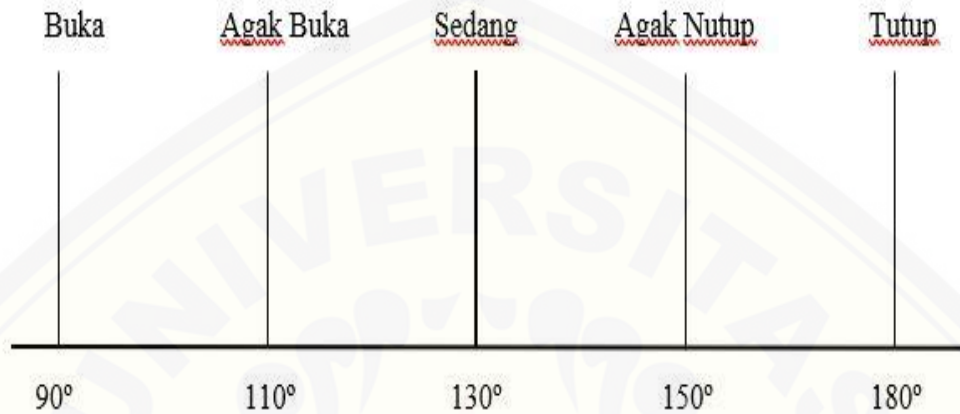
$$NK(x; a, b, c) = \begin{cases} 0; & x \leq a \\ \frac{(b-x)}{(b-a)}; & a \leq x \leq b \\ 1 & x = c \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0; & x \geq b \end{cases} \dots\dots\dots(7.3)$$

Membership function nilai delta error terdiri dari 5 himpunan ,yaitu negatif besar (NB), negatif kecil (NK), Zero, positif kecil(PK), positif besar(PB)

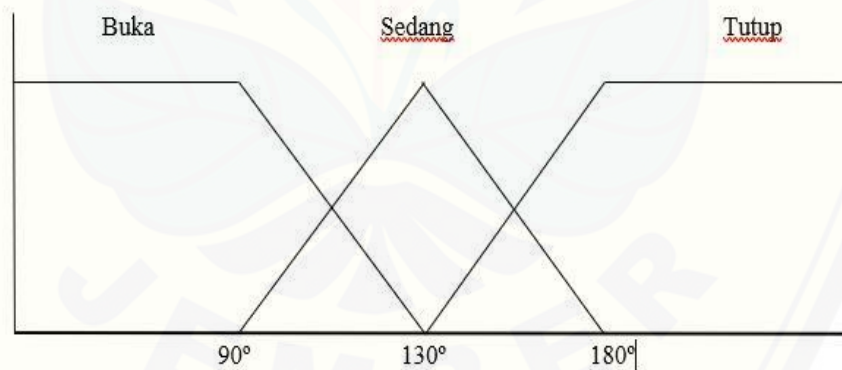


Gambar 3.7 fungsi keanggotaan nilai d\_error

Fungsi keanggotaan *output* nilai derajat servo terdiri dari 5 himpunan yaitu tutup, agak nutup, sedang, agak buka dan buka.



Gambar 3.9 fungsi keanggotaan *output* derajat servo



Gambar 3.10 fungsi keanggotaan *output* derajat servo

Setpoint yang dimasukkan tergantung dari jumlah *input* meter kubik liter air yang dimasukkan jika *input* awal kurang dari setpoint maka servo off (kondisi buka), jika *input* mulai mendekati setpoint maka servo on (kondisi agak nutup) dan jika *input* telah melebihi batas setpoint maka servo on(kondisi menutup).

### 3.6.2) Rule Base Fuzzy

Menyusun basis aturan, yaitu aturan-aturan berupa implikasi-implikasi *fuzzy* yang menyatakan relasi antara variabel *input* dan variabel *output*. Pada penelitian ini fungsi implikasi digunakan dengan bentuk sebagai berikut :

E Δ E	SERVO DOWN		OFF	SERVO UP	
	NB	NK	ZERO	PK	PB
NB	TUTUP	AGAK NUTUP	SEDANG	AGAK BUKA	BUKA
NK	TUTUP	AGAK NUTUP	SEDANG	AGAK BUKA	BUKA
Z	TUTUP	AGAK NUTUP	SEDANG	AGAK BUKA	BUKA
PK	TUTUP	AGAK NUTUP	SEDANG	AGAK BUKA	BUKA
PB	TUTUP	AGAK NUTUP	SEDANG	AGAK BUKA	BUKA

Gambar 3.10 Tabel *Rule Base Fuzzy*

Operasi logika *fuzzy* yang digunakan adalah operator OR. Dalam bentuk IF-THEN atau Jika-Maka, dapat dicontohkan sebagai berikut :

Jika *Error is NB* and *Delta Error is NB* Maka *Servo is TUTUP*

Jika *Error is NB* and *Delta Error is NK* Maka *Servo is AGAK NUTUP*

Jika *Error is NB* and *Delta Error is ZERO* Maka *Servo is SEDANG*

Jika *Error is NB* and *Delta Error is PK* Maka *Servo is AGAK BUKA*

Jika *Error is NB* and *Delta Error is PB* Maka *Servo is BUKA*

Jika *Error is NK* and *Delta Error is NB* Maka *Servo is TUTUP*

Jika *Error is NK* and *Delta Error is NK* Maka *Servo is AGAK NUTUP*

Jika *Error is NK* and *Delta Error is ZERO* Maka *Servo is SEDANG*

Jika *Error is NK* and *Delta Error is PK* Maka *Servo is AGAK BUKA*

Jika *Error is NK* and *Delta Error is PB* Maka *Servo is BUKA*

Jika *Error is Z* and *Delta Error is NB* Maka *Servo is TUTUP*

*Jika Error is Z and Delta Error is NK Maka Servo is AGAK NUTUP*

*Jika Error is Z and Delta Error is ZERO Maka Servo is SEDANG*

*Jika Error is Z and Delta Error is PK Maka Servo is AGAK BUKA*

*Jika Error is Z and Delta Error is PB Maka Servo is BUKA*

*Jika Error is PK and Delta Error is NB Maka Servo is TUTUP*

*Jika Error is PK and Delta Error is NK Maka Servo is AGAK NUTUP*

*Jika Error is PK and Delta Error is ZERO Maka Servo is SEDANG*

*Jika Error is PK and Delta Error is PK Maka Servo is AGAK BUKA*

*Jika Error is PK and Delta Error is PB Maka Servo is BUKA*

*Jika Error is PB and Delta Error is NB Maka Servo is TUTUP*

*Jika Error is PB and Delta Error is NK Maka Servo is AGAK NUTUP*

*Jika Error is PB and Delta Error is ZERO Maka Servo is SEDANG*

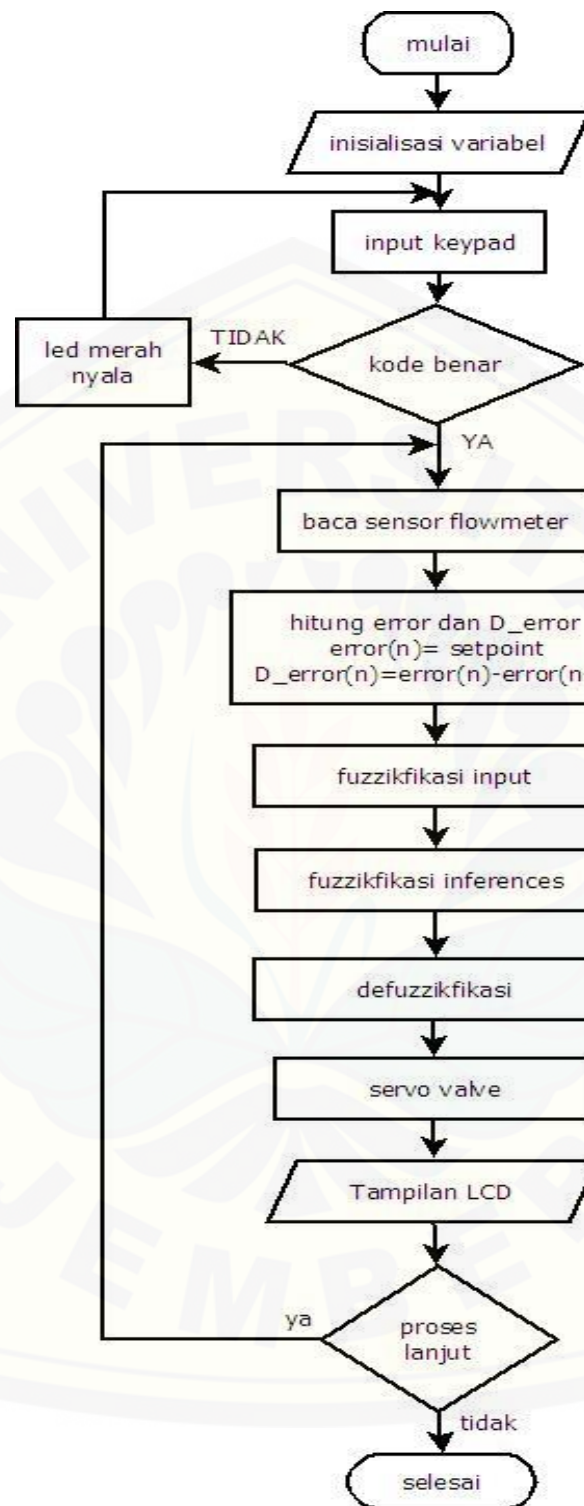
*Jika Error is PB and Delta Error is PK Maka Servo is AGAK BUKA*

*Jika Error is PB and Delta Error is PB Maka Servo is BUKA*

### 3.6) Diagram Alir

Prinsip kerjanya yaitu pertama menginisialisasi variabel sensor *flowmeter*, arduino, LCD, *Keypad* dan servo valve .Kemudian masukkan *input keypad* apabila kode yang dimasukkan salah maka led merah akan menyala dan system akan kembali keawal dan jika benar maka sistem lanjutke proses membaca data yang terdeteksi oleh sensor *flowmeter*. Setelah data diperoleh, kemudian masuk ke dalam sistem pemrosesan program berupasinyal data sebagai *setpoint* yang nantinya akan dikonversikan menjadi jumlah liter dan arus air sehingga diperoleh nilai berupa Batasan aliran air dan besar arus yang, *inputfuzzylogic* terdiri dari error dan delta error yang kemudian diproses dengan nilai  $error(n)=setpoint$ ,  $delta\ error(n)= error(n)-D\_error(n-1)$  kemudian didapatkan *defuzzifikasi* sesuai dengan *rule base* yang telah ditentukan dimana saat awal proess berjalan dan *setpoint* sudah dtentukan maka servo dalam kondisi terbuka kemudian sensor akan terus membaca jumlah air yang telah digunakan jika aliran air mulai mendekati set point maka servo akan aktif dengan control agak nutup dengan arus air mulai melambat dan saat air sudah melebihi set point maka servo akan aktif dengan kontrol servo menutup aliran air dan *buzzer* akan menyala sebagai *indikasi* bahwa jumlah air yang digunakan sudah melebihi batas, sehingga air tidakakan mengalir, apabila ingin air tetap mengalir masukkan *input* pulsa melalui *keypad* jika tidak maka proses akan berhenti atau selesai.





Gambar 3.11. Diagram Alir Perancangan

## 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. Telah berhasil dibuat sebuah alat system untuk mengontrol jumlah air pelanggan berbasis mikrokontroler.
2. Penerapan system prabayar berjalan sesuai dengan apa yang diharapkan sehingga pelanggan dapat membatasi jumlah penggunaan air.
3. Hasil pengujian terhadap sensor menunjukkan bahwa sensor flow meter memiliki nilai error sensor yang signifikan hingga mencapai 6,5%

### 5.2 Saran

Berdasarkan hasil penelitian yang telah diperoleh, disadari bahwa rancangan alat yang telah dibuat ini masih memiliki banyak kekurangan, penulis memberikan saran sebagai berikut:

1. Sebagai pengembang kedepan penampilan monitoring bias diganti dengan monitoring melalui web ataupun gadget.
2. Mengimplementasikan pengembangan dengan menggunakan sensor yang memiliki tingkat keakuratan dan nilai kapasitas yang lebih besar.



LAMPIRAN

liter (mL)	Percobaan(mL)									
	1	2	3	4	5	6	7	8	9	10
1000	928	850	780	884	997	939	956	973	991	1007
2000	1892	1759	1880	1786	1875	1836	1835	1834	1834	1832
3000	2922	2877	2560	2879	3022	2912	2932,8	2953	2973	2993
4000	3798	3823	3789	3670	3370	3387	3286	3185	3084	2983
5000	4739	4870	4489	4756	4689	4644	4623	4601	4580	4558
6000	5889	5680	5874	5763	6145	6048	6108	6167	6227	6286
7000	6790	6873	6687	6782	7211	7093	7169	7244	7319	7394
8000	7860	7700	7849	7932	7879	7925	7952	7979	8006	8033
9000	8365	8529	8736	8215	8832	8721	8783	8845	8907	8968
10000	9570	9478	9513	9768	9756	9815	9881	9948	10014	10080

# Digital Repository Universitas Jember

liter (ml)	Percobaan (mL)										Rata-Rata (mL)
	1	2	3	4	5	6	7	8	9	10	
1000	928	850	780	884	997	939	956	973	991	1007	930
2000	1892	1759	1880	1786	1875	1836	1835	1834	1834	1832	1836
3000	2922	2877	2560	2879	3022	2912	2932,8	2953	2973	2993	2902
4000	3798	3823	3789	3670	3370	3387	3286	3185	3084	2983	3437
5000	4795	4836	4679	4667	4382	4373	4274	4175	4075	3976	4423
6000	5759	5840	5649	5612	5209	5215	5083	4950	4816	4684	5282
7000	6723	6843	6620	6557	6035	6057	5891	5726	5558	5393	6140
8000	7687	7847	7591	7502	6862	6899	6700	6501	6300	6102	69999
9000	8651	8851	8561	8447	7688	7741	7509	7277	7042	6811	7858
10000	9615	9855	9532	9393	8515	8583	8318	8052	7784	7520	8716

Data kalibrasi sensor flow

liter (ml)	Percobaan(mL)										Rata-Rata (mL)
	1	2	3	4	5	6	7	8	9	10	
1000	928	850	780	884	997	939	956	973	991	1007	930,5
2000	1892	1759	1880	1786	1875	1836	1835	1834	1834	1832	1836,3
3000	2922	2877	2560	2879	3022	2912	2932,8	2953	2973	2993	2902,3
4000	3798	3823	3789	3670	3370	3387	3286	3185	3084	2983	3437,5
5000	4795	4836	4679	4667	4382	4373	4274	4175	4075	3976	4423,4
6000	5759	5840	5649	5612	5209	5215	5083	4950	4816	4684	5282,1
7000	6723	6843	6620	6557	6035	6057	5891	5726	5558	5393	6140,8
8000	7687	7847	7591	7502	6862	6899	6700	6501	6300	6102	69999,5
9000	8651	8851	8561	8447	7688	7741	7509	7277	7042	6811	7858,2
10000	9615	9855	9532	9393	8515	8583	8318	8052	7784	7520	8716,9



## LISTING PROGRAM

```
/*
Liquid flow rate sensor -DIYhacking.com Arvind Sanjeev

Measure the liquid/water flow rate using this code.
Connect Vcc and Gnd of sensor to arduino, and the
signal line to arduino digital pin 2.

*/
#include <Servo.h>
#include <Keypad.h>
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
LiquidCrystal_I2C lcd(0x3F,16,2);

Servo myservo;

char customKey;
const byte ROWS = 4;
const byte COLS = 4;

char keys[ROWS][COLS] = {
  {'D', 'C', 'B', 'A'},
  {'#', '9', '6', '3'},
  {'0', '8', '5', '2'},
  {'*', '7', '4', '1'}
};

byte rowPins[ROWS] = {6,5,4,3};
byte colPins[COLS] = {10,9,8,7};

int x = 0;

Keypad customKeypad = Keypad( makeKeymap(keys),
rowPins, colPins, ROWS, COLS);

byte statusLed = 13;

byte sensorInterrupt = 0; // 0 = digital pin 2
byte sensorPin = 2;
```

```
// The hall-effect flow sensor outputs approximately 4.5
// pulses per second per
// litre/minute of flow.
float calibrationFactor = 4.5;

volatile byte pulseCount;

float flowRate;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;

unsigned long oldTime;

void setup()
{
  myservo.attach(11);
  myservo.write(175);
  delay(100);

  // Initialize a serial connection for reporting values to the
  // host
  Serial.begin(9600);

  // Set up the status LED line as an output
  pinMode(statusLed, OUTPUT);
  digitalWrite(statusLed, HIGH); // We have an active-low
  LED attached

  pinMode(sensorPin, INPUT);
  digitalWrite(sensorPin, HIGH);

  pulseCount    = 0;
  flowRate      = 0.0;
  flowMilliLitres = 0;
  totalMilliLitres = 0;
  oldTime       = 0;

  // The Hall-effect sensor is connected to pin 2 which uses
  interrupt 0.
```

```
// Configured to trigger on a FALLING state change
// (transition from HIGH
// state to LOW state)
attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
}

/**
 * Main program loop
 */
void loop()
{
  customKey = customKeypad.getKey();

  if ( customKey == 'A')
  {
    myservo.write(90);
    delay(100);
  }
  if (customKey == 'B')
  {
    myservo.write(120);
    delay(100);
  }
  if ( customKey == 'C')
  {
    myservo.write(140);
    delay(100);
  }
  if (customKey == 'D')
  {
    myservo.write(160);
    delay(100);
  }
  if (customKey == '*')
  {
    myservo.write(175);
    delay(100);
  }
}
```

```
if (totalMilliLitres <= 1258) // based on the number of pulses per second per units of
{ // measure (litres/minute in
  myservo.write(180); // this case) coming from the sensor.
  delay(50); // flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount)
} // calibrationFactor;

if((millis() - oldTime) > 1000) // Only process counters // Note the time this processing pass was executed. Note
once per second // that because we've
{ // disabled interrupts the millis() function won't actually
  // Disable the interrupt while calculating flow rate and // be incrementing right
  // sending the value to // at this point, but it will still return the value it was set to
  // the host // just before
  detachInterrupt(sensorInterrupt); // interrupts went away.
  // oldTime = millis();

  // Because this loop may not complete in exactly 1 second // Divide the flow rate in litres/minute by 60 to determine
intervals we calculate // how many litres have
  // the number of milliseconds that have passed since the // passed through the sensor in this 1 second interval, then
last execution and use // multiply by 1000 to
  // that to scale the output. We also apply the // convert to millilitres.
  calibrationFactor to scale the output
```

```
flowMilliLitres = (flowRate / 60) * 1000;
Serial.print(totalMilliLitres/1000);

// Add the millilitres passed in this second to the
cumulative total
Serial.print("L");

totalMilliLitres += flowMilliLitres;
// Reset the pulse counter so we can start incrementing
again
unsigned int frac;
pulseCount = 0;

// Print the flow rate for this second in litres / minute
Serial.print("Flow rate: ");
// Enable the interrupt again now that we've finished
sending output
Serial.print(int(flowRate)); // Print the integer part of the
variable
attachInterrupt(sensorInterrupt, pulseCounter,
FALLING);
Serial.print("L/min");
}
Serial.print("\t"); // Print tab space
}

// Print the cumulative total of litres flowed since starting
/*
Serial.print("Output Liquid Quantity: ");
Insterrupt Service Routine
Serial.print(totalMilliLitres);
*/
Serial.println("mL");
void pulseCounter()
Serial.print("\t"); // Print tab space
{
```

```
// Increment the pulse counter
pulseCount++;
}int x = 0;
int buzzer = 12;
float harga;
Keypad customKeypad = Keypad( makeKeymap(keys),
rowPins, colPins, ROWS, COLS);

byte sensorInterrupt = 0; // 0 = digital pin 2
byte sensorPin      = 2;

// The hall-effect flow sensor outputs approximately 4.5
pulses per second per
// litre/minute of flow.
//hitung pulse counter dulu
//lalu bikin variabel,isikan 1000/pulse counter
//calibration factor = 1000/pulse counter
float calibrationFactor = 1;

volatile byte pulseCount;

float flowRate;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;

unsigned long oldTime;

void setup(){
myservo.attach(11);
lcd.init();
lcd.backlight();

pinMode(sensorPin, INPUT);
digitalWrite(sensorPin, HIGH);

pulseCount      = 0;
flowRate        = 0.0;
flowMilliLitres = 0;
totalMilliLitres = 0;
oldTime         = 0;

pinMode(buzzer,OUTPUT);
```



```
    }  
  
    // The Hall-effect sensor is connected to pin 2 which uses  
interrupt 0.  
    // Configured to trigger on a FALLING state change  
(transition from HIGH  
    // state to LOW state)  
    attachInterrupt(sensorInterrupt, pulseCounter, FALLING);  
}  
  
void loop(){  
    customKey = customKeypad.getKey();  
  
    if(x == 0){  
        lcd.setCursor(0,0);  
        lcd.print("1.SET HARGA ");  
    }  
  
    if(x == 1){  
        lcd.setCursor(0,0);  
        lcd.print("2.MULAI ");  
  
        switch(customKey)  
        {  
            case '0' ... '9':  
                break;  
  
            case '#':  
                break;  
  
            case '*':  
                break;  
  
            case 'A':
```

```
x++;  
    break;  
  
case 'B':  
x--;  
    break;  
  
case 'C':  
    break;  
  
case 'D':  
if(x == 0){  
    lcd.clear();  
    setharga();  
    //bacaSensor();  
}  
if(x == 1){  
    lcd.clear();  
    mulai();  
}  
  
    if(x == 2){  
        lcd.clear();  
        bacaSensor();  
    }  
    break;  
}  
  
    if(x > 2){  
        x = 0;  
    }  
  
    if(x < 0){  
        x = 2;  
    }  
}  
  
void setharga(){  
    lcd.setCursor(0,0);  
    lcd.print("SET HARGA  ");  
}
```

```
customKey = customKeypad.getKey();  
  
if(customKey >= '0' && customKey <= '9')  
{  
    harga = harga * 10 + (customKey - '0');  
    lcd.setCursor(0,1);  
    lcd.print(harga);  
}  
  
if(customKey == 'A'){  
    lcd.clear();  
    delay(1000);  
    harga = 1000;  
    return;  
}  
  
if(customKey == 'B'){  
    lcd.clear();  
    delay(1000);  
  
    harga = 2000;  
    return;  
}  
  
if(customKey == '*'){  
    lcd.clear();  
    delay(1000);  
    harga = harga;  
    return;  
}  
  
setharga();  
  
void mulai(){  
    lcd.setCursor(0,0);  
    lcd.print("MULAI");  
  
    digitalWrite(buzzer,HIGH);  
    delay(1000);
```

```
digitalWrite(buzzer,LOW);
myservo.write(90);
delay(100);
bacasensor();
myservo.write(0);
delay(50);

customKey = customKeypad.getKey();

if(customKey == '*'){
  lcd.clear();
  harga=0;
  return;
}
mulai();
}

void bacasensor(){
    if((millis() - oldTime) > 1000) // Only process counters
    once per second
    {
        detachInterrupt(sensorInterrupt);
        flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount)
        / calibrationFactor;
        oldTime = millis();
        flowMilliLitres = (flowRate / 60) * 1000;
        // Add the millilitres passed in this second to the
        cumulative total
        totalMilliLitres += flowMilliLitres;
        unsigned int frac;
        // Print the flow rate for this second in litres / minute
        lcd.setCursor(0,0);
        lcd.print(" Flow Rate: ");
        lcd.setCursor(1,1);
        lcd.print(totalMilliLitres);
        lcd.print("ml");
        lcd.setCursor(8,1);
        lcd.print(totalMilliLitres/1000);
    }
}
```

```
lcd.print("L");

// Reset the pulse counter so we can start incrementing again
pulseCount = 0;

// Enable the interrupt again now that we've finished
sending output
attachInterrupt(sensorInterrupt, pulseCounter,
FALLING);
}
}

void pulseCounter()
{
// Increment the pulse counter
pulseCount++;
}
```