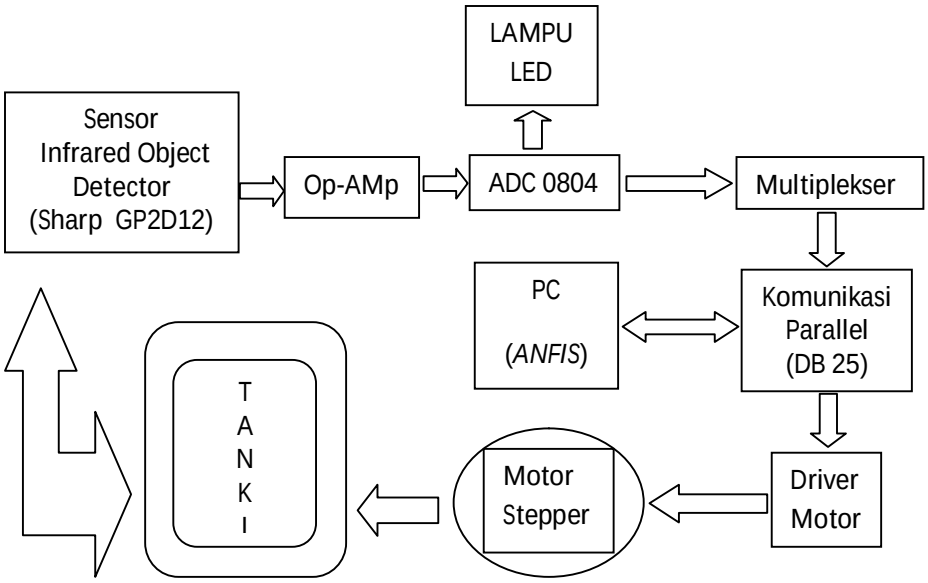
BAB III

PERANCANGAN SISTEM

3.1 Gambaran Umum Sistem

Gambaran umum dari sistem pengendalian level ketinggian air dapat dilihat dalam blok diagram di bawah ini :



Gambar 3.1 *Blok Diagram Sistem Level Tinggi Air*

Berdasarkan blok diagram keseluruhan sistem (gambar 3.1) dapat dijelaskan sebagai berikut :

a. Jarak obyek dengan sensor dapat dihitung dengan mengukur besarnya tegangan *output* pada sensor *InfraRed Object Detector*. *Output* dari sensor adalah berupa tegangan. Semakin dekat jarak obyek dengan sensor maka semakin tinggi pula tegangan yang dikeluarkan oleh *output* sensor.

Besarnya tegangan pada *output* sensor akan diperbarui secara terus-menerus kira-kira setiap 32ms sekali.

b. Data sensor tersebut kemudian masuk ke rangkaian buffer, agar menghindari tegangan balik dari ADC.
c. Kemudian tegangan *output* sensor tersebut diumpankan ke modul ADC 0804 agar data analog bisa diubah menjadi sinyal keluaran digital. Setelah *output* sensor sudah diumpankan ke ADC 0804 maka *output* tersebut dikirim ke multiplekser dengan tujuan agar keluaran menjadi satu keluaran.
d. Kemudian data dari multiplekser dikirim menggunakan komunikasi parallel (DB25) ke PC (*Personal Computer*) agar dapat diolah dan diproses menggunakan software LabVIEW.
e. Data yang sudah diolah oleh PC (*Personal Computer*) kemudian dikirim kembali menggunakan kabel DB 25 yang kemudian dipergunakan untuk mengontrol motor stepper (untuk mengontrol motor stepper harus membuat rangkaian *driver* terlebih dahulu).

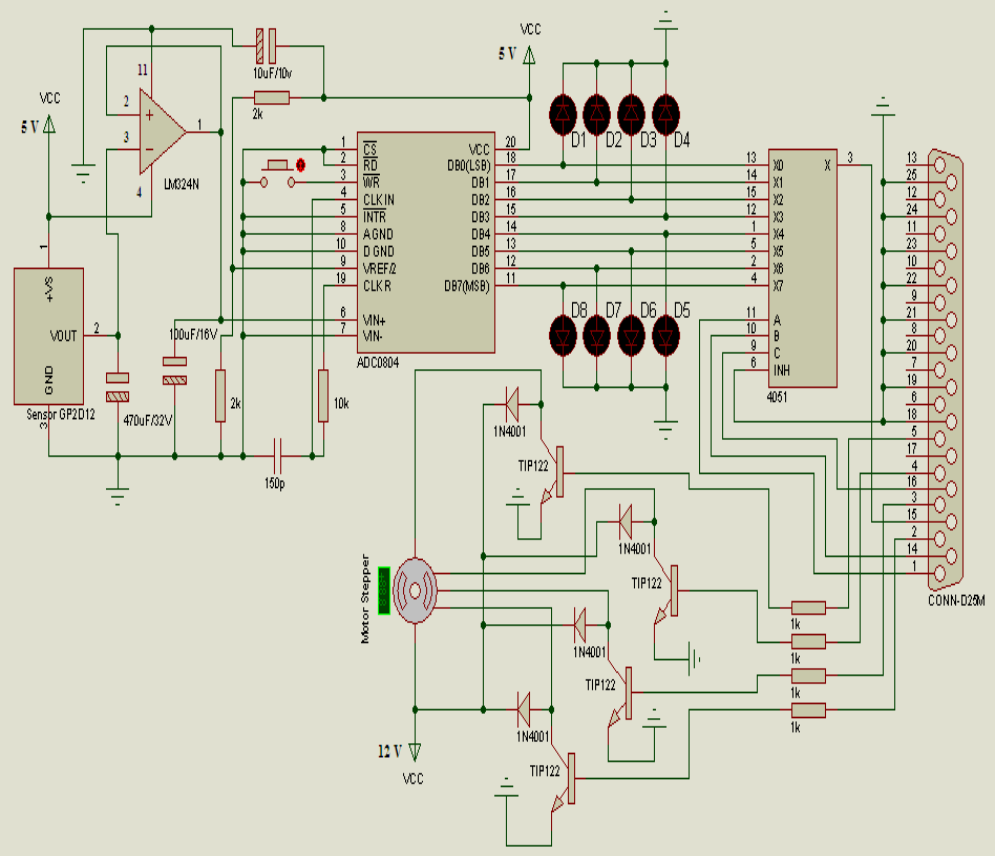
3.2 Perancangan Sistem

Agar tujuan dari perancangan sistem ini berjalan sesuai yang diharapkan yaitu mengendalikan level tinggi air menggunakan *Adaptif Neuro-Fuzzy Interference System* (*ANFIS*) maka ada dua hal yang harus direalisasikan yaitu :

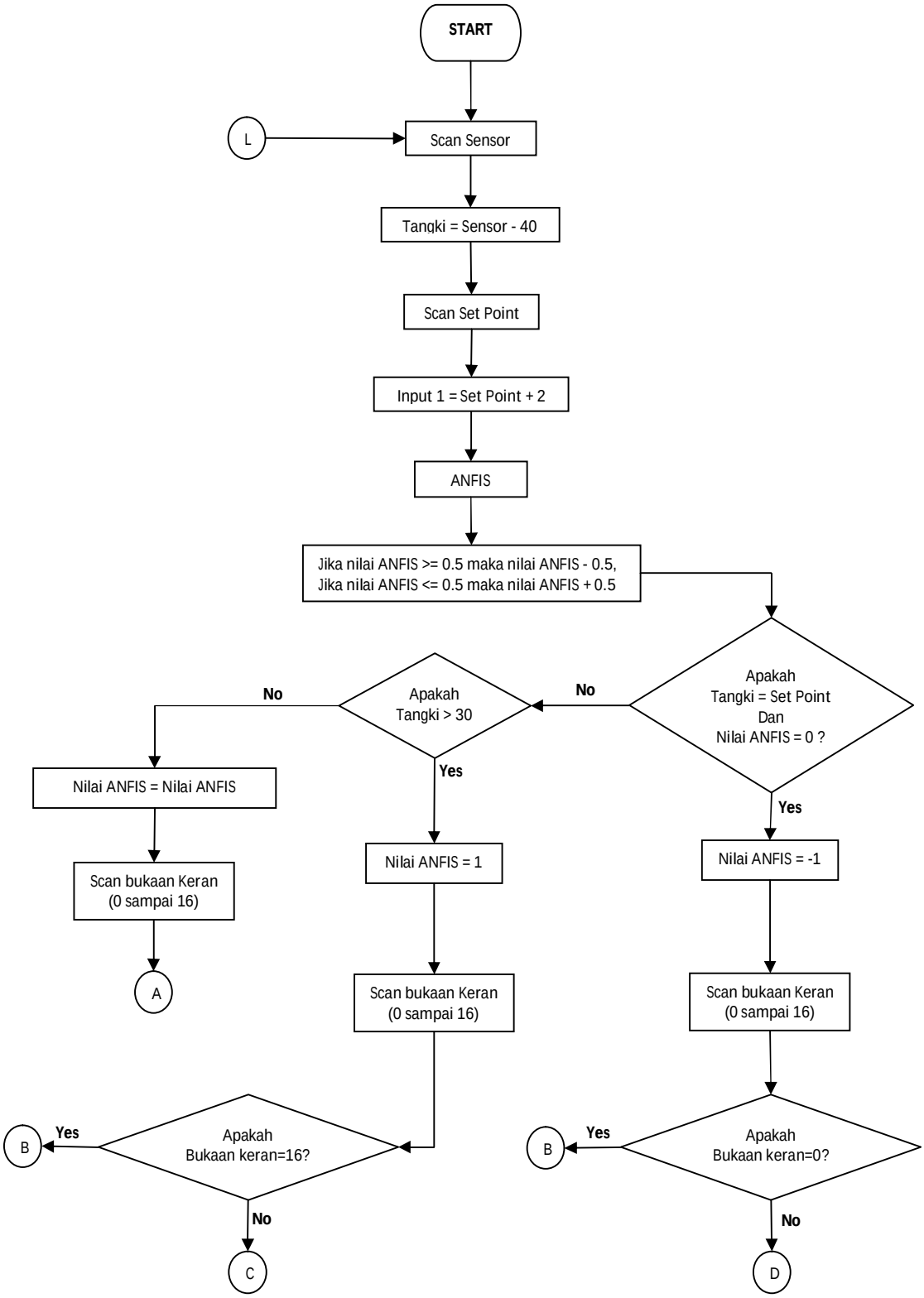
1. Perancangan perangkat keras yang mendukung terealisasikan pengendalian level tinggi air.

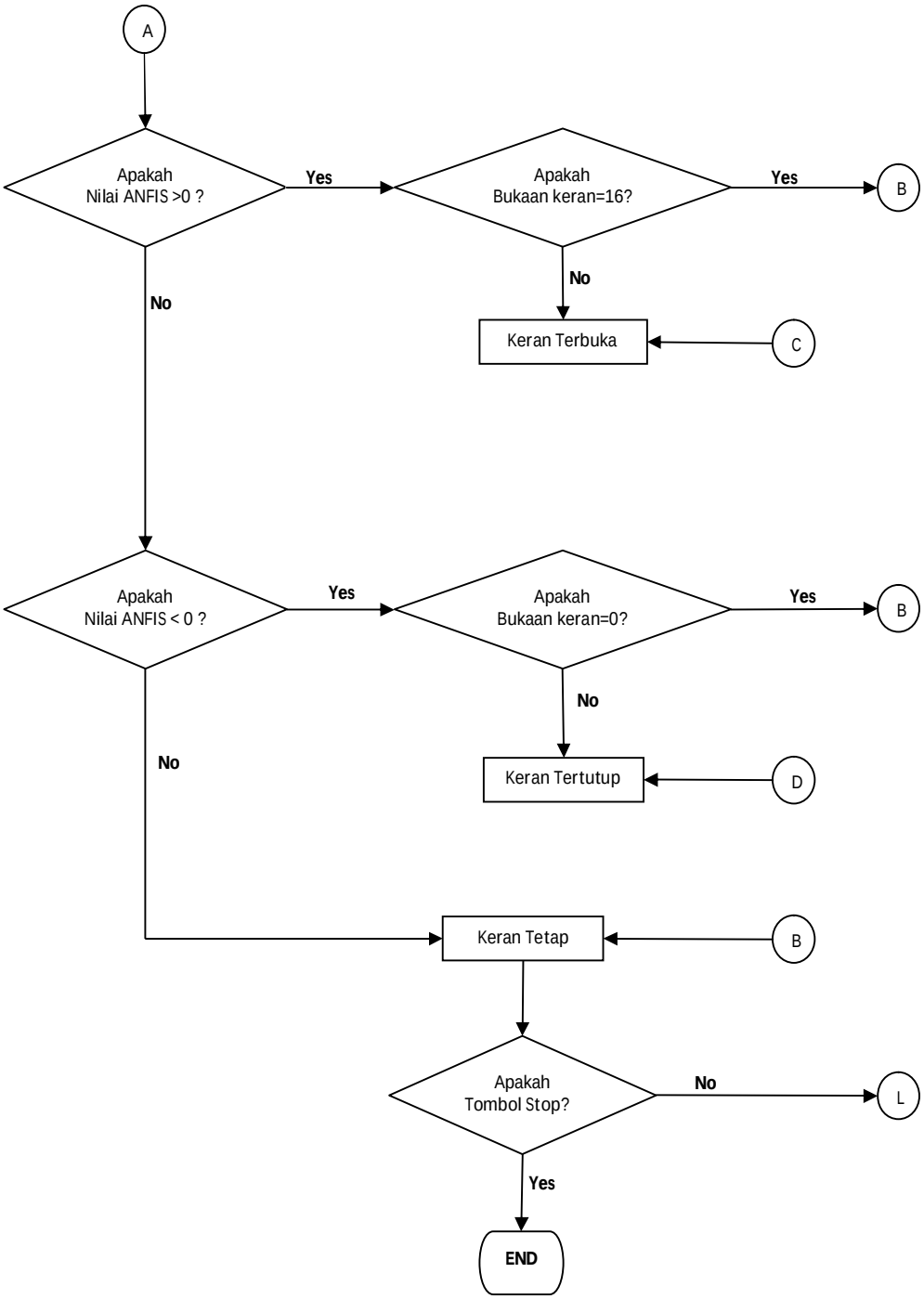
2. Perancangan perangkat lunak dengan memanfaatkan software LabVIEW. Sistem pengontrolan yang dilakukan adalah sistem *ANFIS*.

Berdasarkan dua tujuan perancangan sistem di atas maka dibuat suatu rangkaian keseluruhan dari sistem tersebut dan *flowchart* sistem. Rangkaian dan *flowchart* sistem dapat dilihat pada gambar 3.2 dan 3.3.



Gambar 3.2 *Rangkaian Keseluruhan Sistem*


START
L
Scan Sensor
Tangki = Sensor - 40
Scan Set Point
Input 1 = Set Point + 2
ANFIS
Jika nilai ANFIS >= 0.5 maka nilai ANFIS - 0.5,
Jika nilai ANFIS <= 0.5 maka nilai ANFIS + 0.5
Apakah
Tangki = Set Point
Dan
Nilai ANFIS = 0 ?
No
Yes
Apakah
Tangki > 30
No
Yes
Nilai ANFIS = Nilai ANFIS
Nilai ANFIS = 1
Nilai ANFIS = -1
Scan bukaan Keran
(0 sampai 16)
Scan bukaan Keran
(0 sampai 16)
Scan bukaan Keran
(0 sampai 16)
A
B
Yes
Apakah
Bukaan keran=16?
No
C
B
Yes
Apakah
Bukaan keran=0?
No
D



Gambar 3.3 *Flowchart Program*

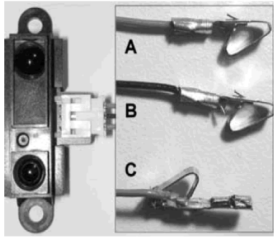
3.3 Perancangan Perangkat Keras *(Hardware)*

Dalam perancangan perangkat keras ini akan dibuat beberapa perangkat keras yang mendukung untuk pengendalian level ketinggian air, yaitu meliputi :

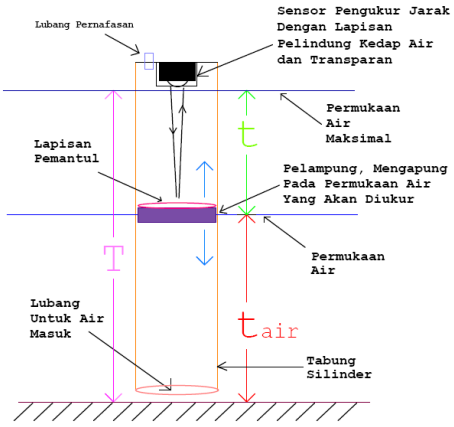
a. Sensor level ketinggian yang menggunakan *Infrared Object Detector* (Sharp GP2D12) .
b. Rangkaian Buffer yang digunnakan untuk menstabilkan keluaran sensor Sharp GP2D12.
c. Rangkaian ADC 0804 digunakan untuk mengubah data *analog* dari sensor menjadi data digital.
d. Multiplekser digunakan untuk mengubah banyak data atau sinyal menjadi satu data yang akan dikeluarkan pada satu *output*.
e. Rangkaian *driver* motor stepper.
f. Penggunaan *connector* DB 25 sebagai *interface* antara perangkat keras dengan komputer.
g. Rangakaian catu daya sebagai suplay tegangan.

3.3.1 *Infrared Object Detector* (Sharp GP2D12)

Jarak obyek dengan sensor dapat dihitung dengan mengukur besarnya tegangan *output* pada sensor *InfraRed Object Detector*.

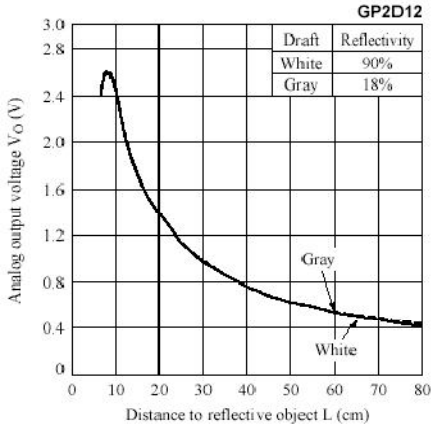


Gambar 3.4 *Sensor Sharp GP2D12 dan Kabel* Pin

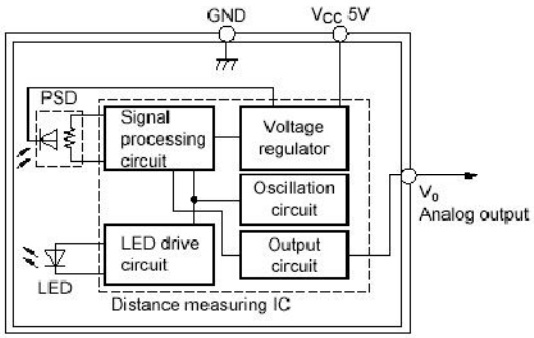


Gambar 3.5 *Desain Sensor Tinggi Level Air*

Output dari sensor adalah berupa tegangan. Semakin dekat jarak obyek dengan sensor maka semakin tinggi pula tegangan yang dikeluarkan oleh *output* sensor. Besarnya tegangan pada *output* sensor akan diperbarui secara terus-menerus kira-kira setiap 32ms sekali. Perubahan tegangan *output* sensor terhadap perubahan jarak obyek adalah tidak linier, seperti yang terdapat pada gambar 3.6, yaitu grafik respon sensor, yaitu grafik yang menunjukkan besarnya tegangan *output* sensor sesuai dengan jarak obyek yang terukur. Sedangkan blok diagram *internal* sensor terdapat pada gambar 3.7.



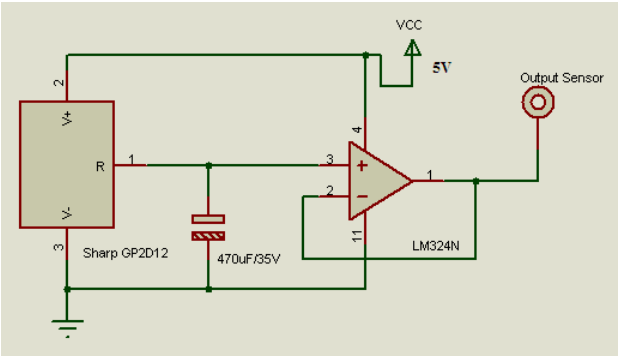
Gambar 3.6 *Grafik Respon Sensor*



Gambar 3.7 *Blok Diagram Internal Sensor*

3.3.2 Rangkaian Buffer

Rangkaian buffer adalah rangkaian yang *input*nya sama dengan hasil *output*nya. Op-amp yang dipakai adalah LM324N dan rangkaiannya seperti pada gambar berikut ini



Gambar 3.8 *Rangkaian Buffer*

3.3.3 *Analog to Digital Converter* (IC ADC 0804)

Suatu tegangan *analog* dengan ordo yang sangat kecil akan sulit dideteksi, agar tegangan *analog* ini mudah dimengerti maka harus diubah kesuatu keluaran

biner. Untuk menghasilkan keluaran biner ini diperlukan suatu *converter* dalam hal ini ADC 0804 mampu melakukannya.

Dalam fungsinya ada beberapa jenis ADC, yang masing-masing mempunyai kelebihan, berdasarkan pada metode pengubahan isyarat *analog* ke digital ADC dibedakan menjadi :

1. Metode Pencacah (*Counting*)
2. Metode *Dual Slope* atau *ratiometrik*
3. Metode pendekatan berurutan (*Successive Approximation* / SAC)
4. Metode Pendekatan paralel (*Paralel-Comparator*)

Untuk menentukan ADC yang digunakan dalam sistem akuisisi data ada beberapa hal yang perlu diperhatikan yaitu :

1. Kecepatan konversi
2. Resolusi
3. Rentang masukan *analog* maksimum
4. Jumlah kanal masukan

Pemilihan ADC umumnya ditentukan oleh metode yang digunakan untuk konversi data, sedangkan rentang tegangan masukan *analog* maksimum adalah watak untai ADC yang digunakan sehingga masukan *analog* yang akan dimasukkan ke ADC tersebut terlebih dahulu harus disesuaikan dengan tegangan *analog* maksimal yang diizinkan dan juga ADC 0804 merupakan suatu IC CMOS pengubah *analog* ke digital delapan bit dengan satu kanal masukan. Oleh karena

itu, dengan pertimbangan diatas penulis sengaja memilih ADC 0804 sebagai konverter analog ke digital.

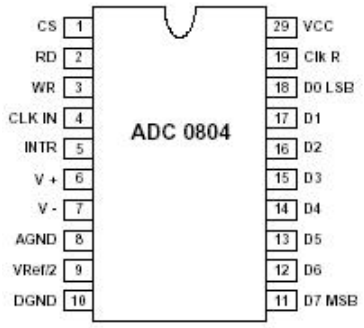
IC ADC 0804 mempunyai masukan *analog*, Vin (+) dan Vin (-), sehingga dapat menerima masukan diferensial. Masukan *analog* sebenarnya (Vin) sama dengan selisih antara tegangan-tegangan yang dihubungkan dengan ke dua pin masukan yaitu Vin= Vin (+) – Vin (-). Kalau masukan *analog* berupa tegangan tunggal, tegangan ini harus dihubungkan dengan Vin (+), sedangkan Vin (-) di*ground*kan. Pada tugas akhir ini, ADC 0804 menggunakan Vcc = +5 Volt dan tegangan referensi adalah 5 Volt. Dalam hal ini jangkauan masukan *analog* mulai dari 0 Volt sampai 5 Volt (skala penuh), karena IC ini adalah SAC 8-bit, resolusinya akan sama dengan :

$$Resolusi = \frac{\text{tegangan skala penuh}}{2^{n}-1} = \frac{5\ \text{Volt}}{255} = 2\ mVolt \quad \ldots\ldots\ldots\ldots\ldots \quad (3.1)$$

(*n menyatakan jumlah bit keluaran biner IC analog to digital converter*)

IC ADC 0804 memiliki generator *clock intenal* yang harus diaktifkan dengan menghubungkan sebuah resistor eksternal (R) antara pin CLK OUT dan CLK IN serta sebuah kapasitor eksternal (C) antara CLK IN dan *ground* digital. Frekuensi *clock* yang diperoleh di pin CLK OUT sama dengan :

$$f = \frac{0.91}{RC} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.2)$$



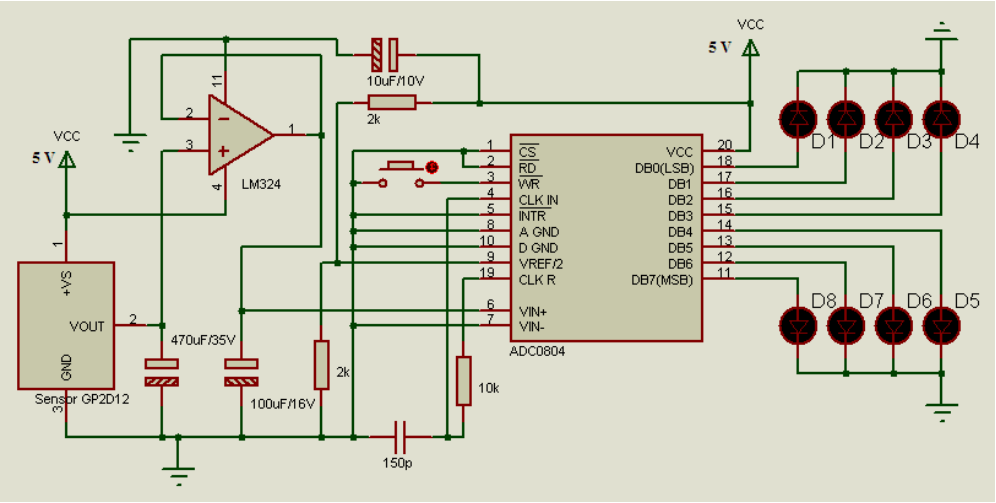
Gambar 3.9 *Pin IC ADC 0804*

Keterangan pada masing-masing pin pada IC ADC 0804 adalah:

1. Pin 1-3 (CS, RD, WR) Merupakan masukan kontrol digital dengan level tegangan logika TTL. Pin CS dan RD jika tidak aktif maka keluaran digital akan berada pada keadaan impedansi tinggi. Pin WR bila dibuat aktif bersamaan dengan CS akan memulai konversi. Konversi akan *reset* bila WR dibuat tidak aktif. Konversi dimulai setelah WR berubah menjadi aktif.
2. Pin 4 dan 19 (*clock* IN dan *clock* R). Merupakan pin masukan dari rangkaian schmit trigger. Pin ini digunakan sebagai *clock internal* dengan menambah rangkaian RC.
3. Pin 5 (INTR) Merupakan pin interupsi keluaran yang digunakan didalam sistem mikroprosesor. Pin 5 menunjukkan bahwa konversi telah selesai. Pin 5 akan mengeluarkan logika tinggi bila konversi dimulai dan mengeluarkan pin rendah bila konversi selesai.
4. Pin 6 dan 7 (Vin (+) dan Vin (-)) Merupakan pin interupsi untuk masukan tegangan *analog*. Vin (+) dan Vin (-) adalah sinyal masukan differensial. Vin (-) digunakan untuk masukan negatif jika Vin (+) dihubungkan

dengan *ground*, dan Vin (+) digunakan untuk masukan positif jika Vin (-) dihubungkan *ground*.

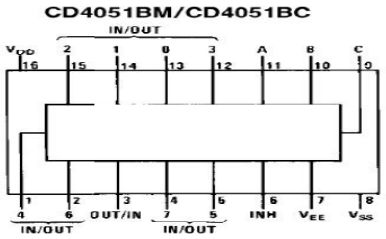
5. Pin 8 dan 10 (AGND dan DGND) Pin ini dihubungkan dengan *ground*.
6. Pin 9 (Vref/2) Merupakan pin masukan tegangan referensi yang digunakan sebagai referensi untuk tegangan masukan dari pin 6 dan 7.
7. Pin 11 sampai 18 (bus data 8 bit) Merupakan jalur keluaran data digital 8 bit. Pin 11 merupakan data MSB dan pin 18 merupakan data LSB.
8. Pin 20 (V+) Pin ini dihubungkan ke VCC (5volt).



Gambar 3.10 *Rangkaian IC ADC0804*

3.3.4 Multiplekser

Multiplekser adalah suatu sirkuit yang berfungsi menggabungkan beberapa atau banyak sinyal elektrik menjadi satu sinyal tunggal. Biasanya *input* multiplekser berupa data yang terdiri dari 8 bit. *Input-input* tersebut akan diseleksi urutan keluarannya oleh suatu pengontrol skema dari IC multipleser CD4051 ditunjukkan pada gambar 3.11.



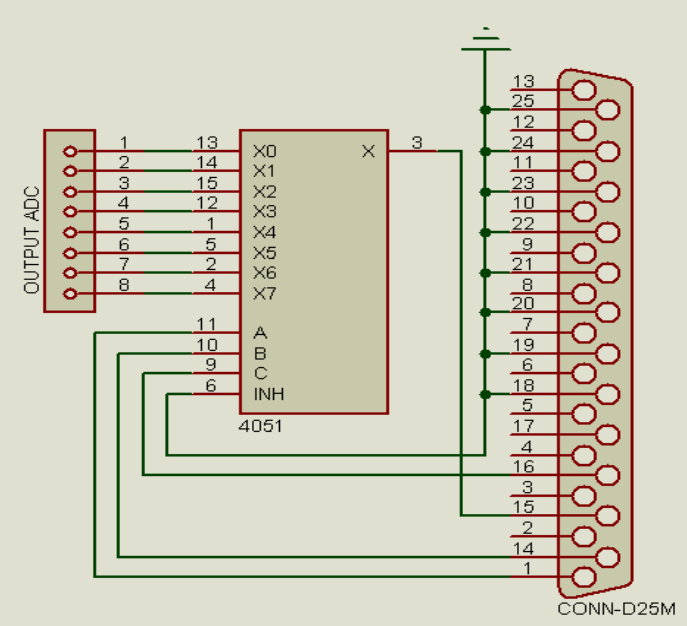
Gambar 3.11 *Skema IC Multiplekser CD4051*

Input pada multiplekser merupakan *output* dari ADC yang kemudian diberikan pada kaki 13, 14, 15, 12, 1, 5, 2 dan 4, hal ini ditunjukkan pada gambar 3.12 rangkaian multiplekser. *Output* akan diperoleh dari kaki 3 dan sinyal yang dikeluarkan pada *output* ditentukan oleh *input* A, B, C pada kaki 11, 10 dan 9. Proses pengontrolan *output* tersebut sesuai tabel 3.1 *Pengontrol Output Multiplekser.*

Tabel 3.1 *Pengontrol Output Multiplekser*

***INPUT* STATES**				**"ON"**
INHIBIT	**C**	**B**	**A**	**CHANNELS**
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	*	*	*	None

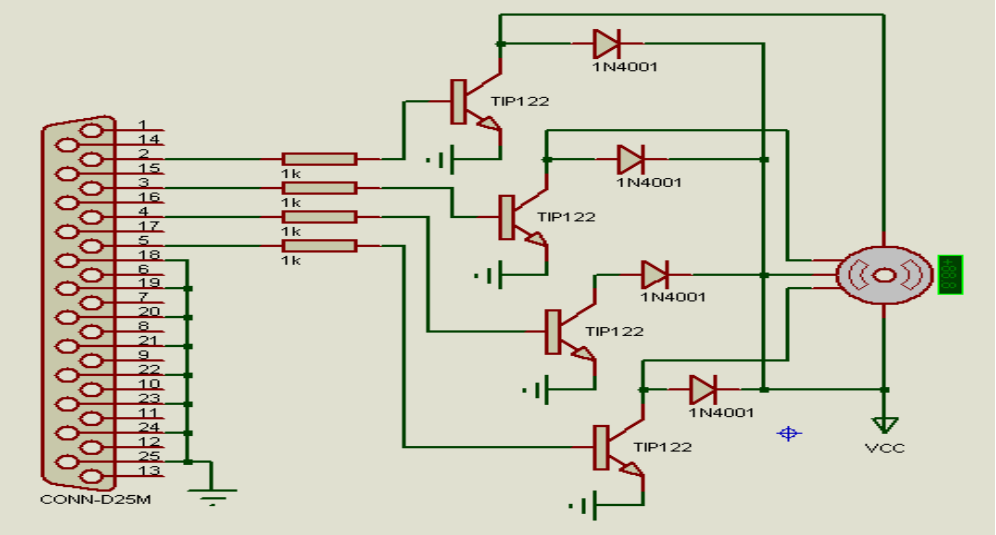
Jika pada *input* C = 0, B = 0, dan A = 0 maka *output* pin 3 akan bernilai sama dengan *input* pin 13, sedangkan jika C = 0, B = 0, dan A = 1 maka *output* pin 13 akan bernilai sama dengan pin 15 dan demikian seterusnya.



Gambar 3.12 *Rangkaian Multiplekser*

3.3.5 Driver Motor Stepper

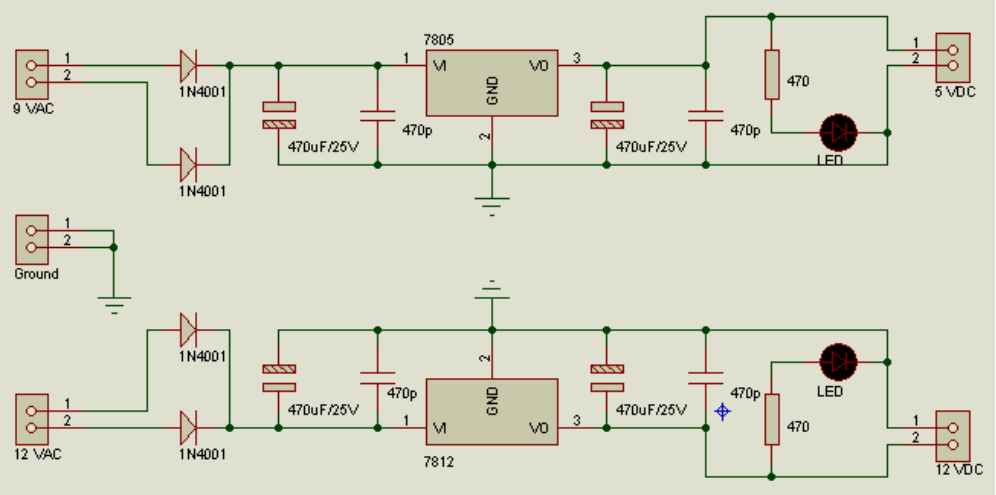
Pada proyek tugas akhir ini, motor yang dipakai adalah motor stepper. Motor dikontrolan menggunakan PC (*Persona Computerl*) dimana arus yang berasal dari komputer berkisar 2.6 mA. Oleh karena itu, untuk mengatasi arus induksi dari motor stepper maka digunakan suatu rangkaian yang dapat mengatasi masalah tersebut. Rangkaian yang dipakai adalah rangkaian driver motor stepper. Rangkaian motor stepper ini dapat kita lihat pada pada gambar 3.13. Pada gambar tersebut diperlihatkan rangkaian alir yang memanfaatkan diode 1N4001 dan TIP122. Rangkaian tersebut dihubungkan langsung ke PC agar dapat dikontrol dengan keinginan kita.



Gambar 3.13 *Rangkaian Motor Stepper Driver*

3.3.6 Rangkaian Catu Daya

Rangkaian catu daya adalah suatu rangkaian yang memberikan tegangan pada alat pengendali. Tegangan catu daya ini memanfaatkan tegangan PLN (Perusahaan Listrik Negara) sebesar 220 VAC, tegangan ini kemudian diturunkan dengan trafo penurun tegangan yaitu trafo 2A jenis CT dengan tegangan maksimum keluaran 15 VAC. Tegangan ini kemudian disearahkan oleh diode menjadi tegangan DC yang kemudian distabilkan oleh IC Regulator. Karena tegangan yang diperlukan pada pada setiap rangkaian tidak sama maka IC Regulator yang dipakai dalam pembuatan rangkaian catu daya ini adalah IC L7805 dan IC L78012. IC regulator ini menghasilkan tegangan 5 dan 12 VDC. Tegangan 5 VDC dipakai pada rangkaian multiplekser, ADC 0804 dan pada sensor GP2D12. Sedangkan tegangan 12 VDC dipakai untuk menjalankan rangkaian driver motor stepper. Rangkaian catu daya ini dapat kita lihat pada gambar 3.14



Gambar 3.14 *Rangkaian Catu Daya*

3.3.7 Komunikasi Data Parallel

PC (*Perconal Computer*) memiliki suatu koneksi yang disebut dengan parallel port. Pada dasarnya parallel port ini sering digunakan untuk koneksi printer. Tetapi semakin perkembangan teknologi, parallel port ini dapat digunakan sebagai *output* dan *input* sistem.

Pada keadaan normal (tidak aktif) tegangankan yang dikekuarkan adalah 0 volt, sedangkan jika pada saat kondisi *high* (aktif) maka tengangan yang dikeluarkan adalah 5 volt.

Ada 3 (tiga) bagian dalam parallel port yaitu :

1. Port status

   Port status mempunyai fungsi untuk mengirimkan kode-kode dari sistem ke PC.

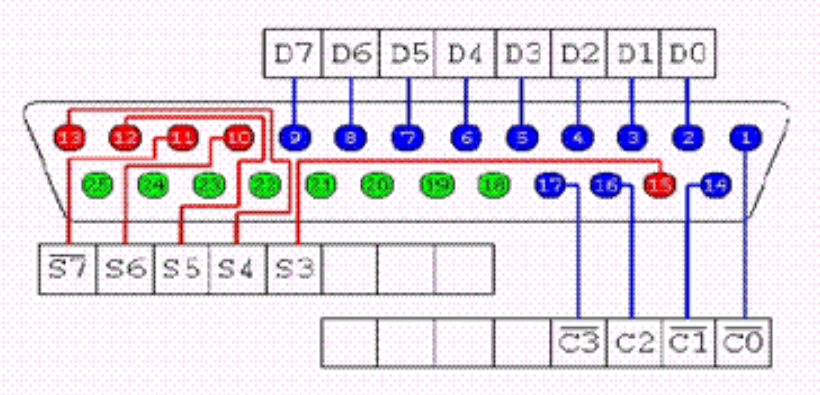
2. Port control

   Port control berfungsi sebagai pengirim kode-kode kontrol dari PC ke sistem yang akan dikendalikan.

3. Data port

   Data port digunakan untuk mengirimkan data ke sistem. Data-data inilah yang nanti dipakai untuk mengendalikan sistem tersebut.

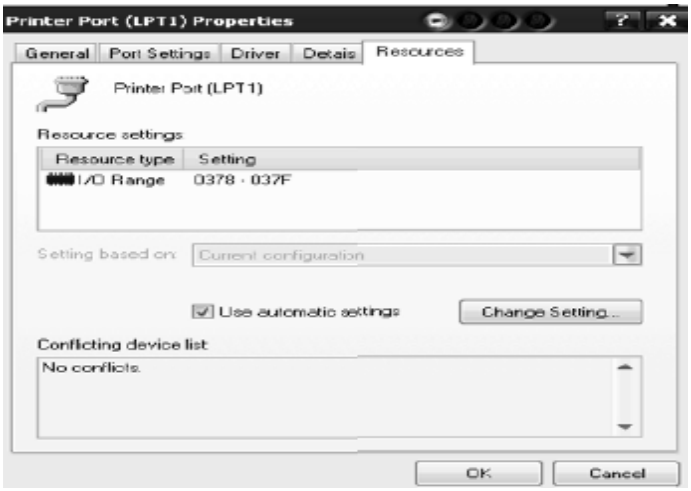
   Di bawah ini gambar konfigurasi pin DB 25 *male*



Gambar 3.15 *Konfigurasi Pin Paralel Port* (*Male*)

Apabila ketiga bagian tersebut terhubung dengan suatu alamat pada PC, maka dengan mudah data dapat dikirim dan dibaca melalui parallel port. Untuk mengidentifikasi *base address* dari parallel port ini maka perlu untuk membuat suatu *device manager* pada *window*nya, pada windows 98 :

1. Pada desktop, klik kanan pada *My Computer* dan pilih *properties.*
2. Klik pada tab *device manager* dan cari LPT1.
3. Setelah memilih LPT1 pilih *propertie.*
4. Kemudian pilih *tab resources* dan *address* selanjutnya dapatt terlihat pada *input* dan *output range.*



Gambar 3.16 *Properties LPT1 pada Device Manager*

Tabel 3.2 *Base Address*

Register	LPT1
Data register (*base address* +0)	0*378
Status register (*base address* +1)	0*379
Control register (*base address* +2	0*37a

Untuk menghubungkan sistem dengan PC, maka perlu dirancang beberapa rangkaian elektronika dengan menggabungkan port parallel DB 25 sebagai *interface*-nya. Rangkaian tersebut dapat dilihat pada rangkaian multiplekser dengan port parallel DB 25 seperti telihat pada gambar 3.11 dan pada rangkaian driver motor stepper dengan port parallel DB 25 seperti terlihat pada Gambar 3.12

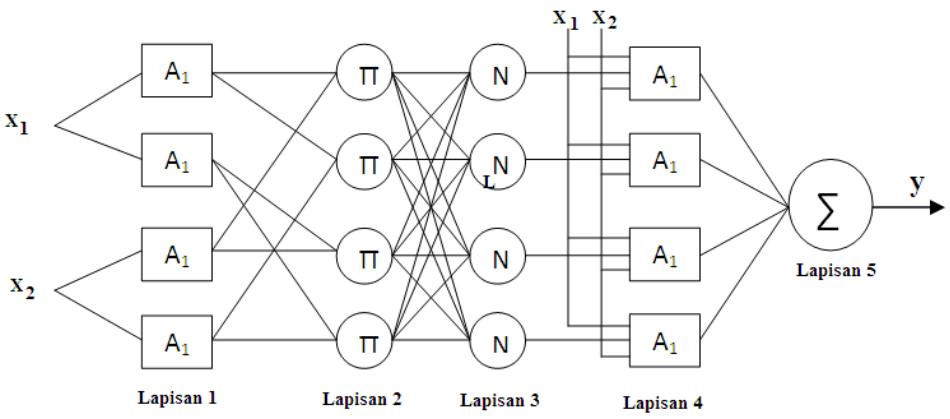
3.4 Perancangan Perangkat Lunak (*Software*)

Untuk melakukan pengontrolan tinggi level air, perangkat lunak merupakan suatu komponen penting sehingga diharuskan dirancang dengan sistem yang diinginkan yaitu mengontrol ketinggian air dari 0 sampai 30 cm dengan memanfaatkan buka tutup keran.

3.4.1 Perancangan Program *ANFIS*

Dalam perancangan perangkat lunak ini dipakai suatu logika yang dikenal dengan *Adaptive Neuro Fuzzy Interference System* (*ANFIS*). Sistem interferensi fuzzy model TSK orde satu dengan pertimbangan kesederhanaan serta kemudahan komputansi. Pertimbangan ini penting karena sistem tersebut akan melalui suatu proses belajar yang mempunyai beban komputansi besar.

Pada sistem interferensi fuzzy TSK orde satu dengan masukan, aturan yang digunakan diekivalenkan dengan struktur jaringan dengan lima lapisan seperti gambar 3.17. Tiap lapisan mempunyai fungsi yang berbeda dan terdiri atas beberapa simpul. Lapisan *ANFIS* tersebut dijelaskan sebagai berikut.



Gambar 3.17 *Struktur Neuro-Fuzzy*

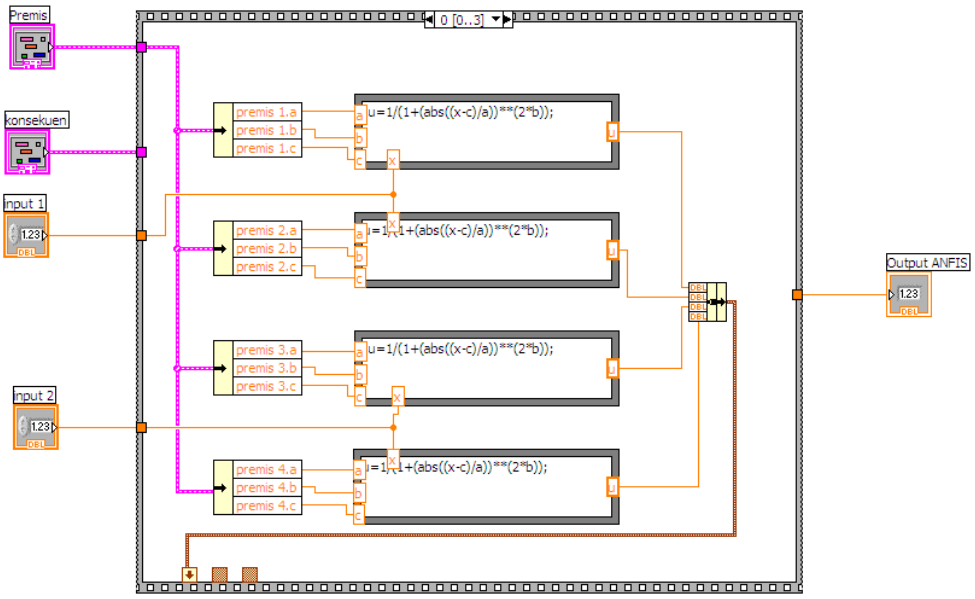
1. Lapisan 1

$$O_{1,i} = \begin{cases} \mu_{A_i}(X_2); i = 1,2 \\ \mu_{B_{i-2}}(X_2); i = 3,4 \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.1)$$

Dengan *O1,I* adalah keluaran ke-I pada lapisan ke-1. Fungsi keanggotaan yang digunakan adalah *generalized bell*

$$\mu_i(x_n) = \frac{1}{1+\left|\frac{x_n - c_i}{a_i}\right|^{2b_i}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.2)$$

Untuk n = 1, 2 dan I = 1, …,4, dengan $\{a_i,b_i,c_i\}$ adalah himpunan parameter premis.

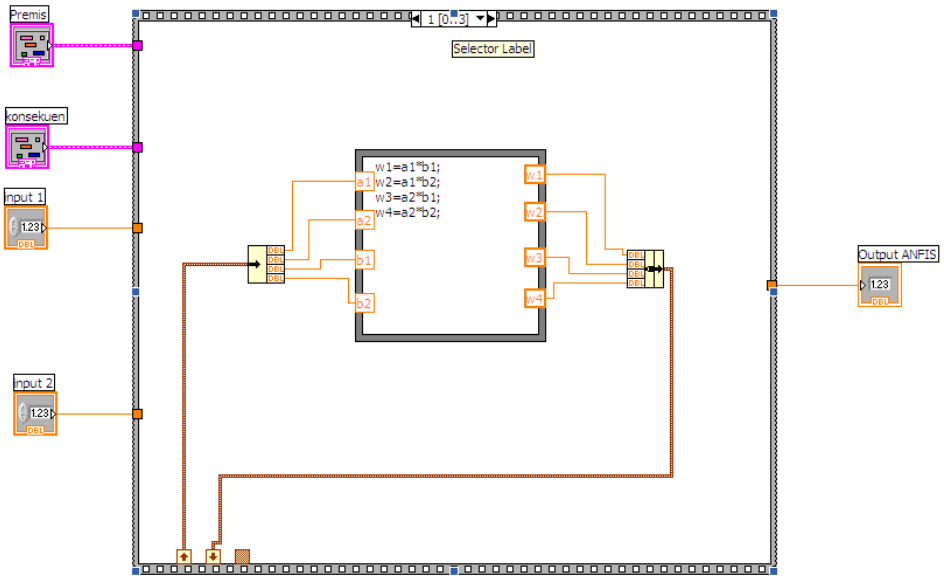


Gambar 3.18 *Frame 0 Sequence ANFIS*

2. Lapisan 2

$$O_{2,i} = w_i = \mu_{A_j}(x_1).\mu_{B_k}(x_2) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.3)$$

Untuk j = k = 1, 2 dan i = 1, ….., 4.

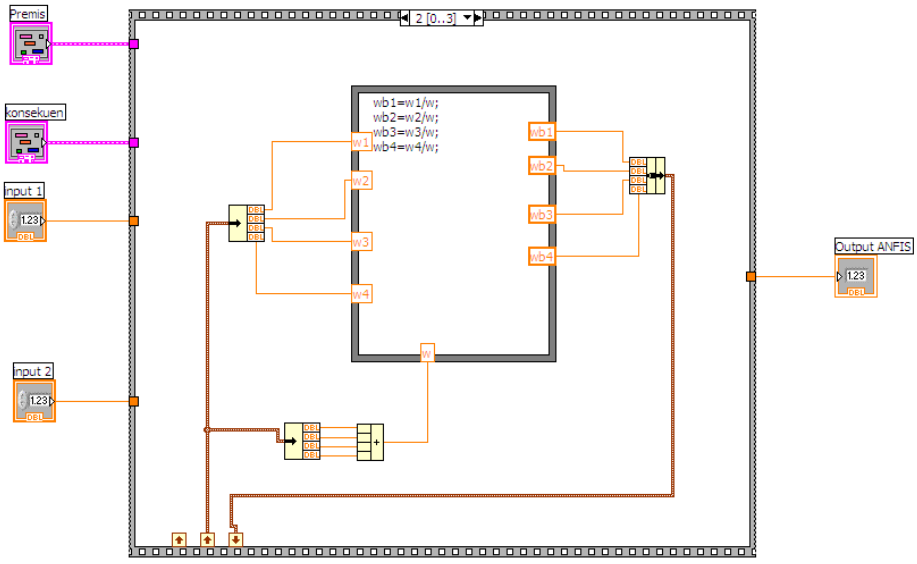


Gambar 3.19 *Frame 1 Sequence ANFIS*

3. Lapisan 3

$$O_{3,i} = \overline{w}_i = \frac{w_i}{\sum w} = \frac{w_i}{w_1 + \cdots + w_4} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.4)$$

Untuk i=1, …, 4.

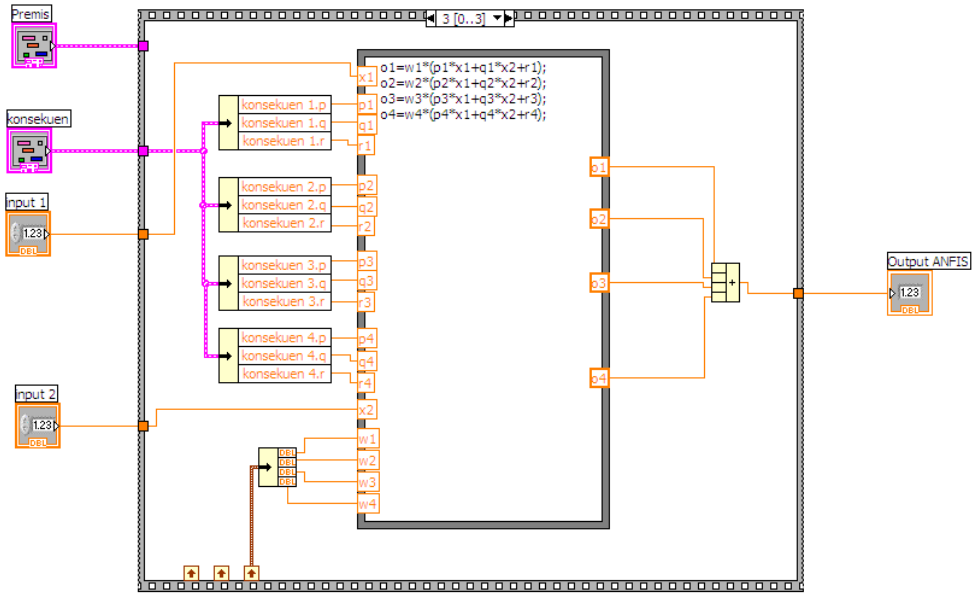


Gambar 3.20 *Frame 2 Sequence ANFIS*

4. Lapisan 4

$$O_{4,i} = \overline{w}_i . f_i = \overline{w}_i (p_i x_1 + q_1 x_2 + r_1) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.5)$$

Untuk i = 1, …, 4, dengan $\{p_i,q_i,r_i\}$ adalah himpunan parameter konsekuen.

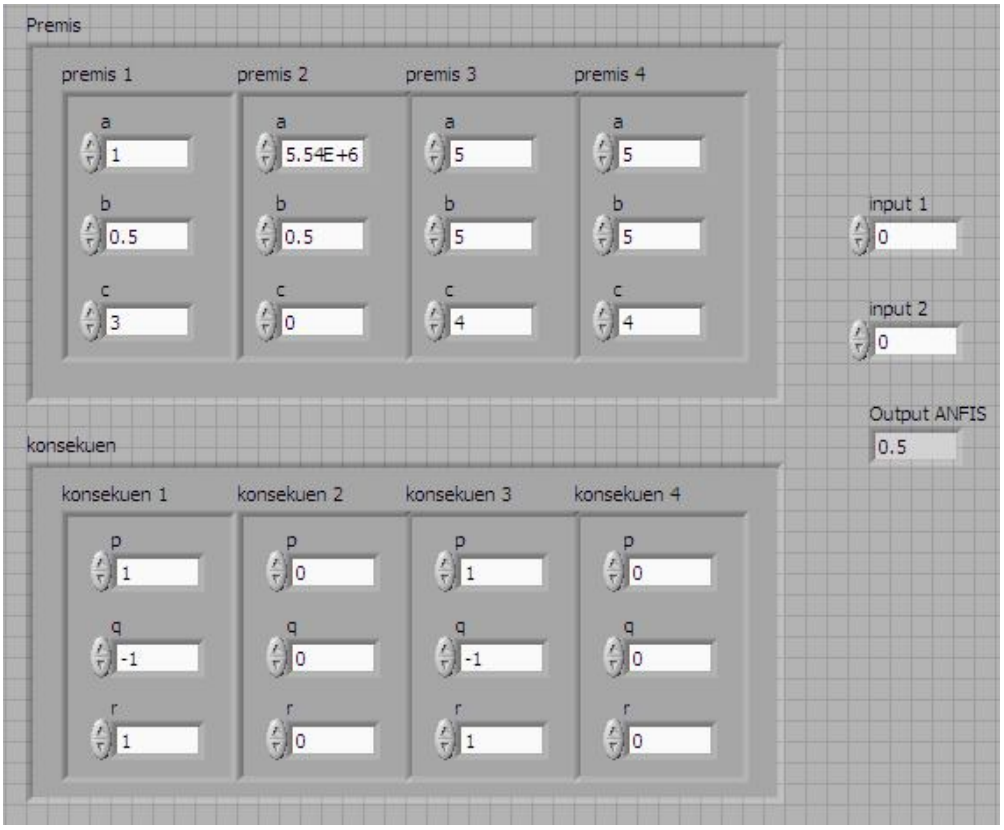


Gambar 3.21 *Frame 3 Sequence ANFIS*

5. Lapisan 5

$$O_5 = \sum \overline{w}_i . f_i = y \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.6)$$

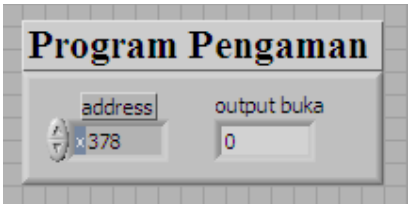
Dalam struktur *ANFIS*, simpul-simpul adaptif terdapat pada lapisan pertama dan keempat. Simpul pada lapisan pertama mengandung parameter premis yang nonlinier sedangkan simpul pada lapisan ke-empat mengandung parameter-parameter tersebut yang tepat melalui suatu proses pembelajaran.



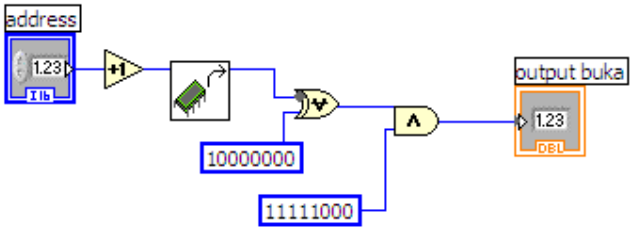
Gambar 3.22 *Front Panel ANFIS*

3.4.2 Perancangan Program Pengaman

Perancangan progam pengaman digunakan sebagai sakelar, prinsip tersebut memanfaatkan status 4 dan status 5 pada pin port parallel DB 25, yaitu pin 13 dan pin 12, jika status 4 aktif maka putaran keran akan mati hal ini menandakan bahwa keran telah membuka penuh sedangkan jika status 5 aktif maka putaran keran akan mati hal ini menandakan keran telah menutup penuh. Program pengaman ini dapat dilihat pada gambar 3.23 dan 3.24 menunjukkan *front panel* dan blok diagram program pengaman.



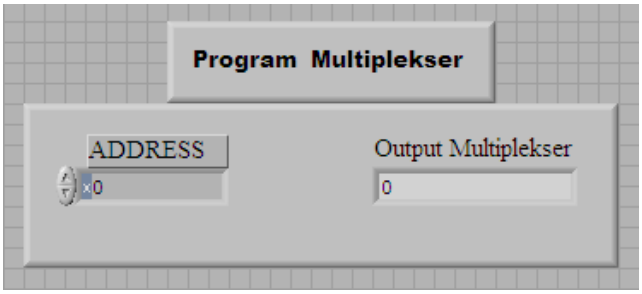
Gambar 3.23 *Front Panel Program Pengaman*



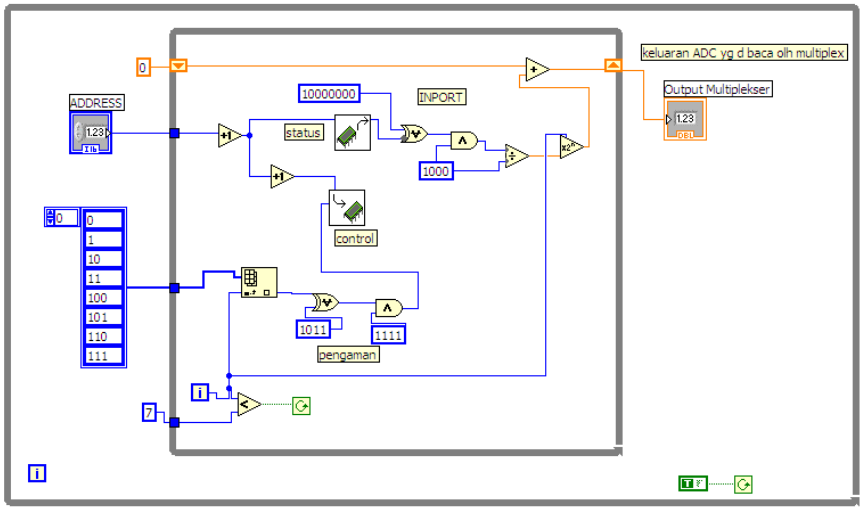
Gambar 3.24 *Blok Diagram Program Pengaman*

3.4.3 Perancangan Program Multiplekser

Untuk mengetahui nilai keluaran dari rangkaian multiplekser apakah sesuai dengan nilai dari keluaran rangkaian ADC (*analog to digital converter*), Maka dibuat program yang bisa merealisasikan permasalah tersebut. Program ini dapat dilihat pada *front panel* program Multiplekser ditunjukkan pada gambar 3.25 dan blok diagram program Multiplekser yang ditunjukan pada gambar 3.26.



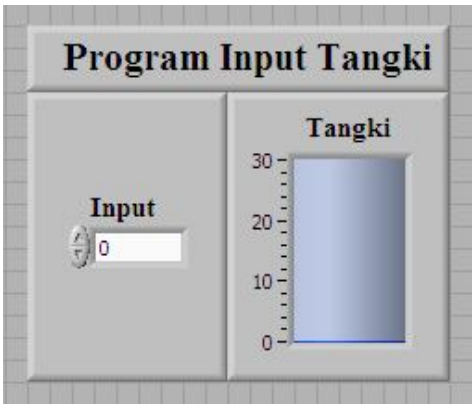
Gambar 3.25 *Front Panel Program Multiplekser*



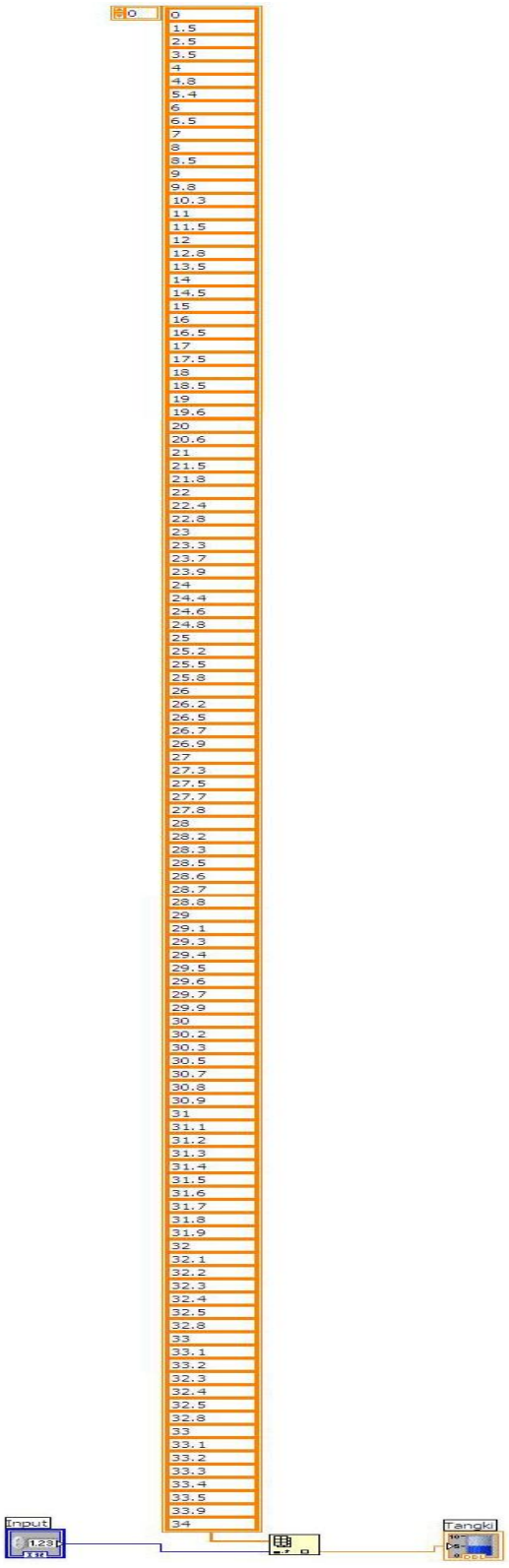
Gambar 3.26 *Blok Diagram Program Multiplekser*

3.4.4 Perancangan Program *Input* Tangki

Perancangan program *input* tangki ini dimaksudkan agar perancangan program tangki berjalan sesuai rencana. Perancangan program *input* tangki merupakan input data jarak tangki dari data multiplekser, program ini dapat dilihat dalam *front panel* dan blok diagram yang terdapat pada gambar 3.27 dan gambar 3.28.



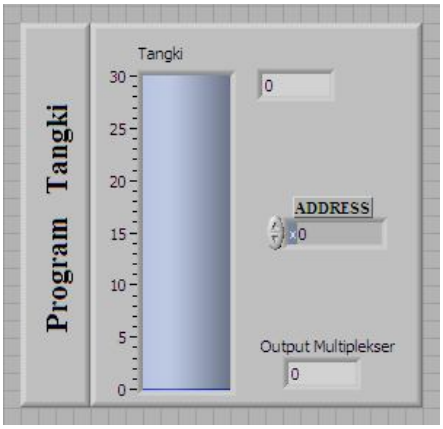
Gambar 3.27 *Front Panel Program Input Tangki*



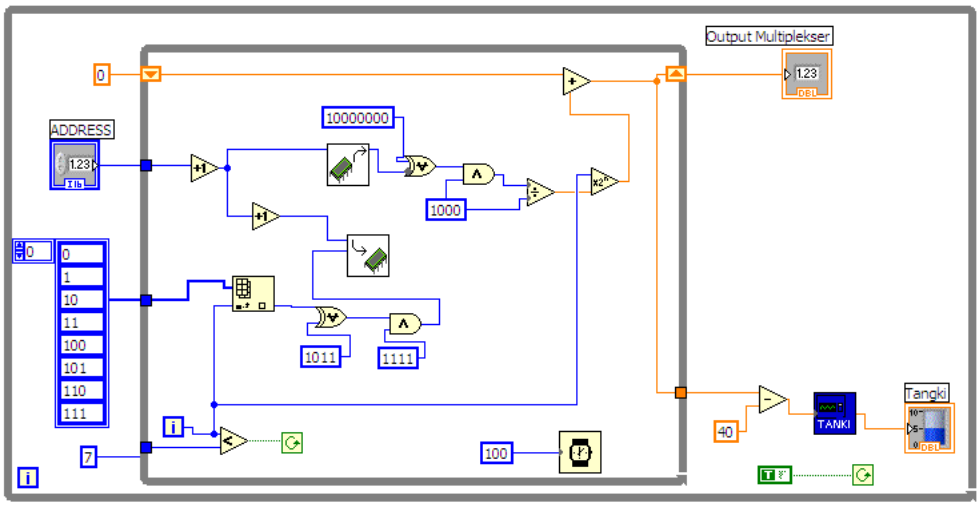
Gambar 3.28 *Blok Diagram Program Input Tangki*

3.4.5 Perancangan Program Tangki

Perancangan program tangki dibuat untuk mengamati perubahan level ketinggian yang direncanakan yaitu 0 cm sampai dengan 30 cm, dimana data level ketinggian air diambil dari data sensor dari sistem yang sudah dirancang. Program tangki ini dapat dilihat pada gambar 3.29 dan gambar 3.30 yang menunjukkan *front panel* dan *blok diagram* program tangki.



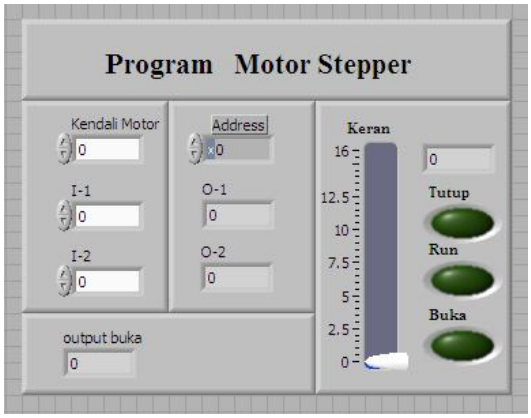
Gambar 3.29 *Front Panel Program Tangki*



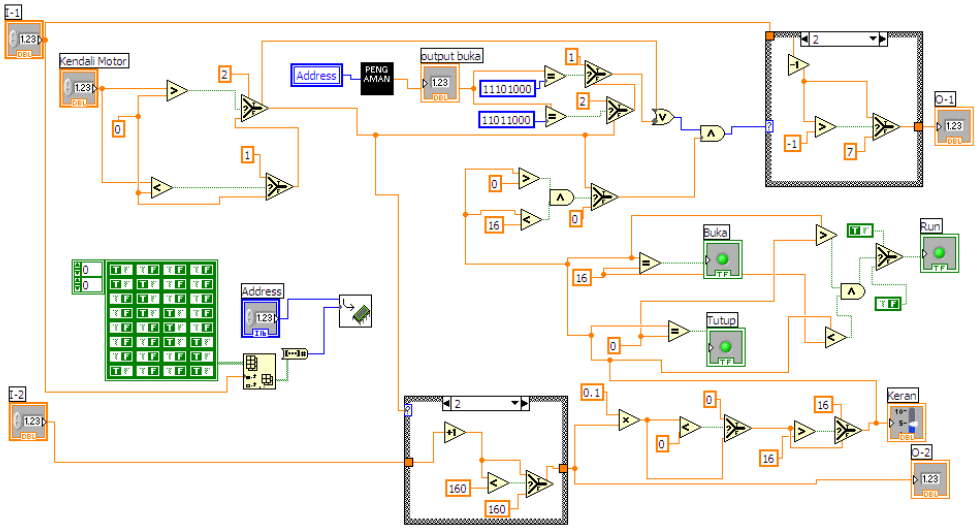
Gambar 3.30 *Blok Diagram Program Tangki*

3.4.6 Perancangan Program Motor Stepper

Perancangan program motor stepper dimaksudkan untuk untuk mengendalikan putaran motor stepper sesuai dengan keinginan yang direncanakan yaitu memutar ke kanan, ke kiri dan berhenti. Program motor stepper ini dapat kita lihat dalam *front panel* dan blok diagram yang terdapat pada gambar 3.31 dan gambar 3.32.



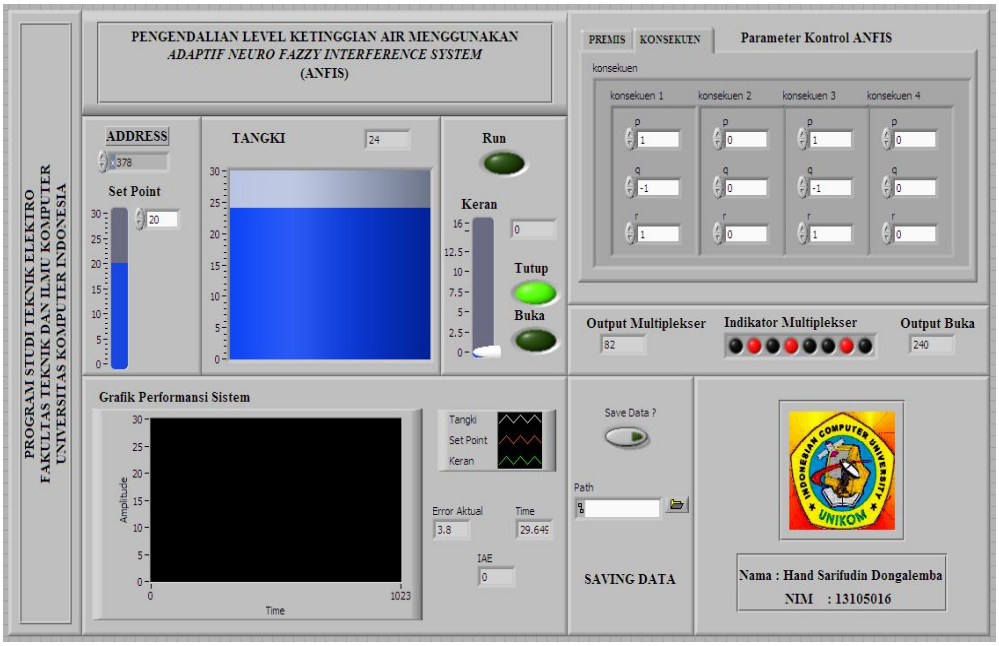
Gambar 3.31 *Front Panel Program Motor Stepper*



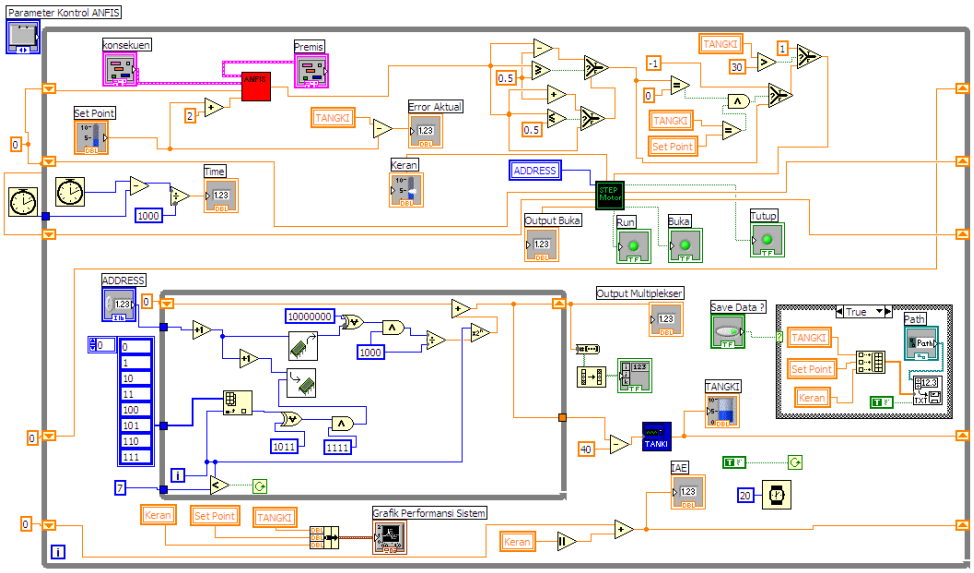
Gambar 3.32 *Blok Diagram Program Motor Stepper*

3.4.7 Perancangan Program Pengontrol *ANFIS*

Perancangan program pengontrol *ANFIS* dilakukan untuk mengendalikan level ketinggian air dari *plant* sehingga level ketinggian air tersebut sesuai dengan level ketinggian yang diharapkan (*setpoint*), perancangan program dilakukan dengan menggabungkan program-program yang sudah dirancang sehingga menjadi suatu kesatuan program pengontrol *ANFIS* yang akan mengendalikan *plant*. Program pengontrol *ANFIS* dapat kita lihat dalam gambar 3.33 (*front panel* program pengontrol *ANFIS*) dan gambar 3.34 (blok diagram program pengontrol *ANFIS*).



Gambar 3.33 *Front Panel Program Pengontrol ANFIS*



Gambar 3.34 *Blok Diagram Program Pengontrol ANFIS*