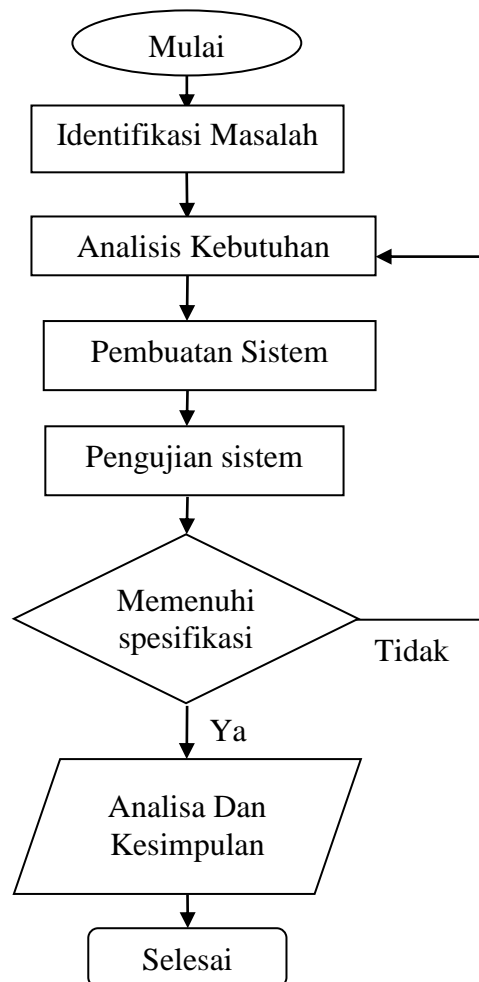


### BAB III METODE PENELITIAN

Dalam upaya pembuatan skripsi dengan judul “Rancang Bangun Alat Identifikasi Diagnosa Penyakit Diabetes Mellitus Melalui Urine Menggunakan Sensor Gas Dan Jaringan Syaraf Tiruan” ini, secara garis besar metode penelitian diawali dengan studi kepustakaan, metode perancangan, metode pengujian, dan analisa data.



**Gambar 3.1.** Flowchart Penelitian

### **3.1 Studi Literatur**

Studi literatur dilakukan untuk menunjang metode wawancara dan observasi yang dilakukan. Dengan cara mengambil intisari dari sumber literatur-literatur berupa buku, artikel dan lain-lain yang berhubungan dengan penelitian yang dilakukan. Referensi didapatkan dari sumber langsung dan tidak langsung. Sumber langsung didapat dari hasil diskusi atau konsultasi dengan dosen dan rekan kuliah, sedangkan sumber tidak langsung didapat dari tulisan laporan penelitian-penelitian (jurnal) yang dilakukan sebelumnya, buku, internet serta referensi-referensi lain yang berkaitan dengan perancangan dan pembuatan sistem.

### **3.2 Pengambilan Sampling Urine**

#### **3.2.1 Pengambilan Sampling untuk pelatihan sistem**

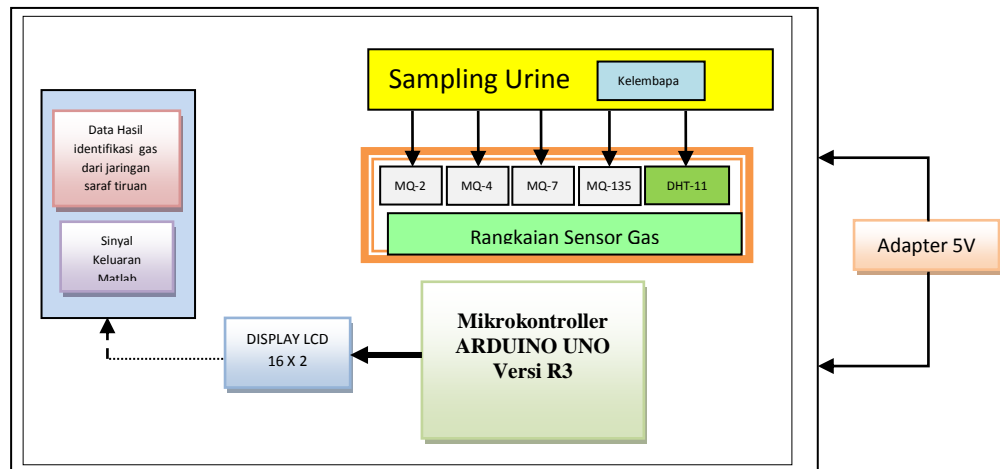
1. Tanyakan pada petugas Laboratorium Rumah Sakit Petrokimia Gresik.
2. Membeli sisa hasil sampling urine dari penderita diabetes mellitus.
3. Menanyakan standart Laborat dalam menentukan adanya penyakit diabetes mellitus pada urine pasien.
4. Membawa pulang sampling urine dengan media yang bersih.

### 3.2.2 Prosedure Pengambilan Sampling Urine

1. Menyediakan tempat sampling urine dengan wadah gelas untuk subjek yang ingin diketahui urinenya apakah terindikasi diabetes mellitus atau tidak.
2. Tuangkan gelas sampling urine ke gelas preparat sekitar 6 ml (6 cc) untuk menyamakan konsentrasi sampling saat pengujian dengan sistem yang dibuat.
3. Melakukan pengukuran dan evaluasi.
4. Membersihkan preparat yang telah digunakan dengan mencuci dengan air sampai bersih dan mengeringkannya.
5. Dalam setiap langkah pada prosedure pengambilan sampling urine harus digunakan sarung tangan dan masker agar kesehatan kita terjaga.

### 3.3 Pembuatan Sistem

Pembuatan sistem terbagi atas dua (2) bagian yaitu perancangan perangkat keras (*hardware*) yang meliputi perancangan skema kerja sistem. Perancangan perangkat lunak (*software*) meliputi perancangan program (*coding*), skema kerja sistem dapat dilihat pada diagram dibawah ini.



**Gambar 3.2.** Rancangan Desain Diagram Blok

Prinsip kerja sistem menurut rancangan desain diagram blok pada Gambar 3.2 adalah sebagai berikut :

1. *Power Supply* berasal dari AC 220 V dirubah menggunakan adaptor dengan *output* tegangan DC 5V untuk power mikrokontroler dan sumber tegangan (VCC) untuk modul sensor gas.
2. Sensor gas bekerja pada tegangan DC antara 3 - 5V, dan mendeteksi gas yang akan dikonversikan menjadi sinyal digital dan dikirim ke IC mikrokontroler arduino uno.
3. Arduino uno versi R3 berfungsi sebagai pengolah data utama (*Processor*) yang mana bekerja pada tegangan max 5V. Disini IC arduino uno merubah sinyal ADC dari sensor suara dan menghasilkan *output* berupa sinyal digital untuk dikirim ke modul mikrokontroler dan ditampilkan data setiap proses pada LCD 16 x 2.

4. LCD 16 x 2 berfungsi untuk menampilkan data setiap proses dari sinyal ADC yang masuk dan yang dikelola oleh mikrokontroler arduino uno yang bekerja pada tegangan DC 5V.
5. Jaringan saraf tiruan berupa data hasil uji coba dengan berbagai macam jenis gas dan disimpan pada mikrokontroler digunakan untuk identifikasi urine.
6. Kotak panel pengendali sebagai tempat (*casing*) pelindung *mikrokontroller* dan piranti elektronik lainnya serta tempat sampling urine.

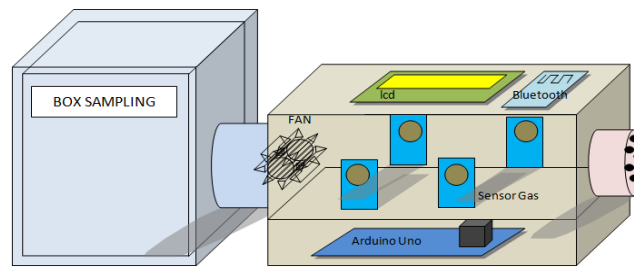
### 3.3.1. Perancangan Hardware

Perancangan Hardware terdiri atas pembuatan perangkat mekanik dan perangkat elektrik. Perangkat elektrik terdiri dari modul Arduino Uno versi R3, LCD 16 x 2, modul sensor gas yang berbeda yaitu MQ-2, MQ-4, MQ-7 dan MQ-135, dan juga beberapa perangkat penunjang elektronika seperti tiga LED penanda, *fan* DC5v sebagai *exhaust*, rangkaian pembagi tegangan (*VCC*) serta rangka (*cover*) yang melindungi seluruh komponen elektrik.

#### 1. Perangkat Mekanik

Secara keseluruhan perangkat mekanik memiliki berbagai macam elemen pendukung seperti :

- a. Tempat sensor
- b. *Box* sampling
- c. Pelindung (*Cover*).

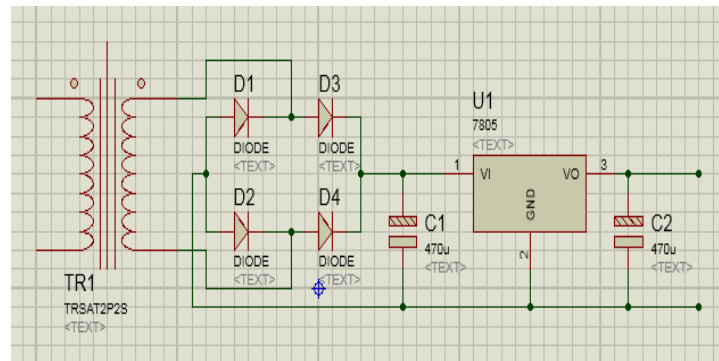


**Gambar 3.3.** Perangkat Mekanik

## 2. Perangkat elektrik

### a. Rangkaian *Power Supply*

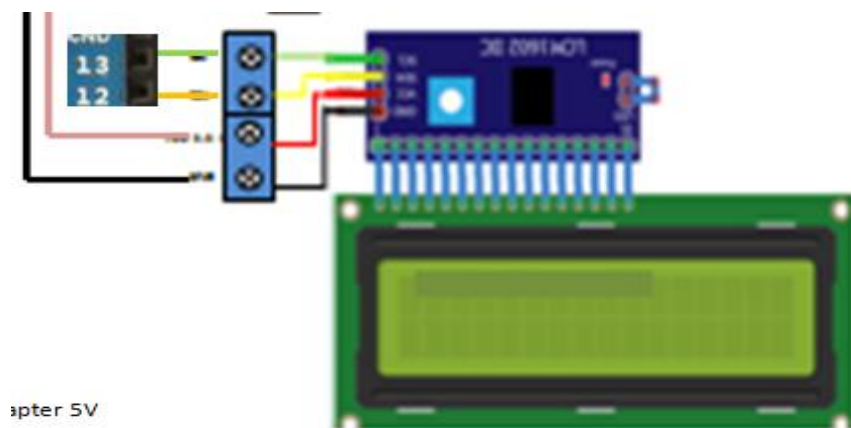
Rangkaian *power supply* berfungsi untuk memberikan *supply* tegangan ke alat. Rangkaian *power supply* terdiri dari trafo, penyearah, filter dan regulator. Adapun cara kerja dari *power supply* adalah tegangan 220V AC diturunkan menjadi 12V AC menggunakan trafo 12V/2A kemudian disearahkan oleh dioda. Sehingga tegangan menjadi 12V DC dan difilter oleh kapasitor 470 uF / 50V dan kapasitor akhir 470 uF / 25V. Kebutuhan rangkaian alat ini adalah sebesar 5V DC, sehingga digunakan ic regulator (*stabilizer*) LM 7805 yang mana akan mendistribusikan pada dua cabang. Cabang pertama pada pemberian tegangan untuk rangkaian sensor dan fan, cabang kedua untuk mikrokontroler arduino uno yang akan didistribusikan lagi ke LCD.



**Gambar 3.4.** Rangkaian Power Supply

### b. Rangkaian LCD 16x2

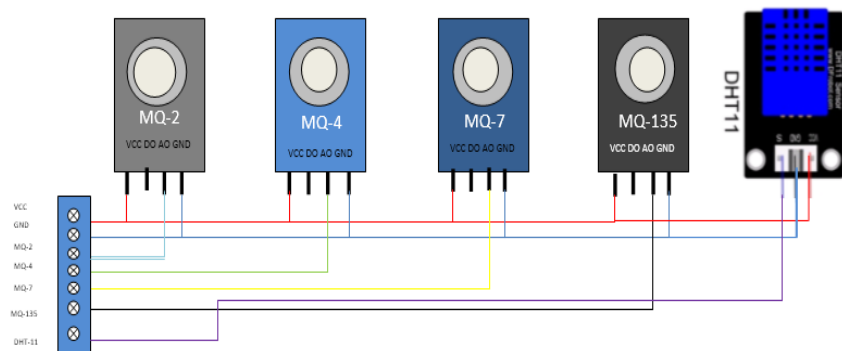
Semua PIN pada LCD 16 x 2 disambungkan pada ic tambahan yaitu ic I2C yang mana pada IC ini hanya memiliki empat outputan saja sehingga efisien dalam hal Pin output LCD, rangkaian sederhana dan tidak memakan banyak ruang pin pada Arduino Uno Board. Pin 1 dari LCD terhubung dengan *ground*. Hubungan Pin antara lain, untuk Pin SDA masuk ke Pin 12, dan Pin SQL masuk ke Pin 13 pada Arduino uno.



**Gambar 3.5.** Rangkaian LCD 16 X 2 Mikrokontroler

### c. Rangkaian Sensor Kelembapan

rangkaian sensor gas dan sensor kelembapan udara (DHT-11) , pin-pin yang digunakan dari setiap sensor hampir sama , yaitu pin tegangan (Vcc) yang mana diberikan 5V dan arus 3A yang cukup besar untuk aktivasi sensor (*Heating*) , *Ground* (Gnd) dan *Digital Output* (DO). Untuk menghubungkan antara sensor ke mikrokontroler Arduino Uno maka pin output dari sensor dihubungkan ke pin digital pada arduino uno yaitu pin digital nomer (2-6) seperti gambar rangkaian diatas. Penggunaan *mini fan* DC 5 V untuk menyedot gas sampling disambungkan power (Vcc) dan Ground (GND) dari rangkaian sensor ini. Penambahan komponen heat sink pada rangkaian ini perlu dilakukan agar mencegah pemanasan yang berlebih dari *power supply* melalui regulator.



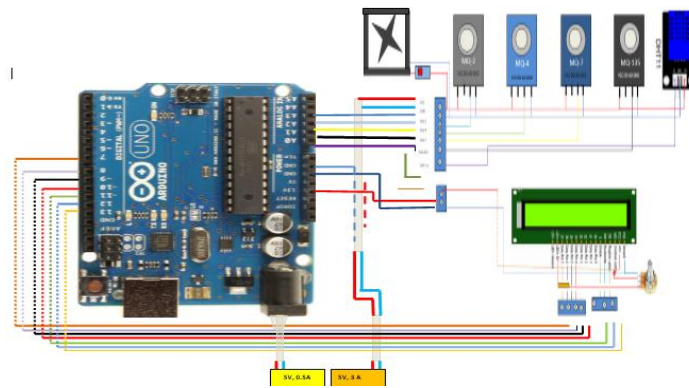
**Gambar 3.6.** Rangkaian Sensor Kelembapan

### d. Rangkaian Keseluruhan Sistem

Penggabungan rangkaian antara komponen elektrik yang digunakan bertujuan agar keseluruhan sistem hardware siap

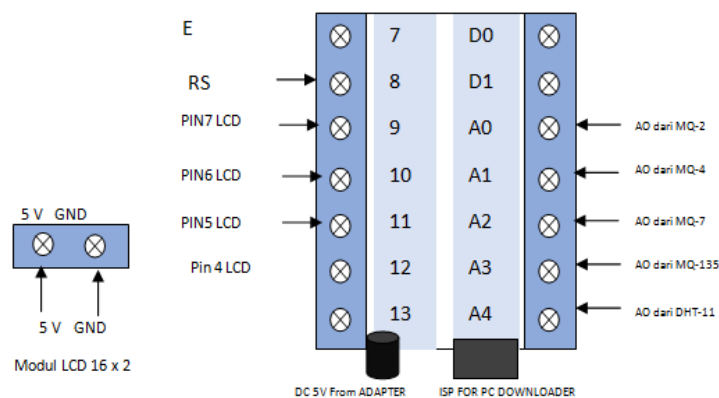


digunakan selain itu dalam pemasangan komponen elektrik terhadap pin-pin input dan output mikrokontroler perlu memperhatikan data sheet agar penempatan pin sesuai dengan fungsinya. Setelah menggabungkan rangkaian elektrik ini diharapkan dalam proses memprogram dapat lebih mudah dalam mendefinisikan pin – pin pada koding.



**Gambar 3.7.** Rangkaian Keseluruhan Sistem

Berikut Data PIN pada mikrokontroler arduino yang digunakan terhadap sejumlah komponen elektrik yang digunakan.



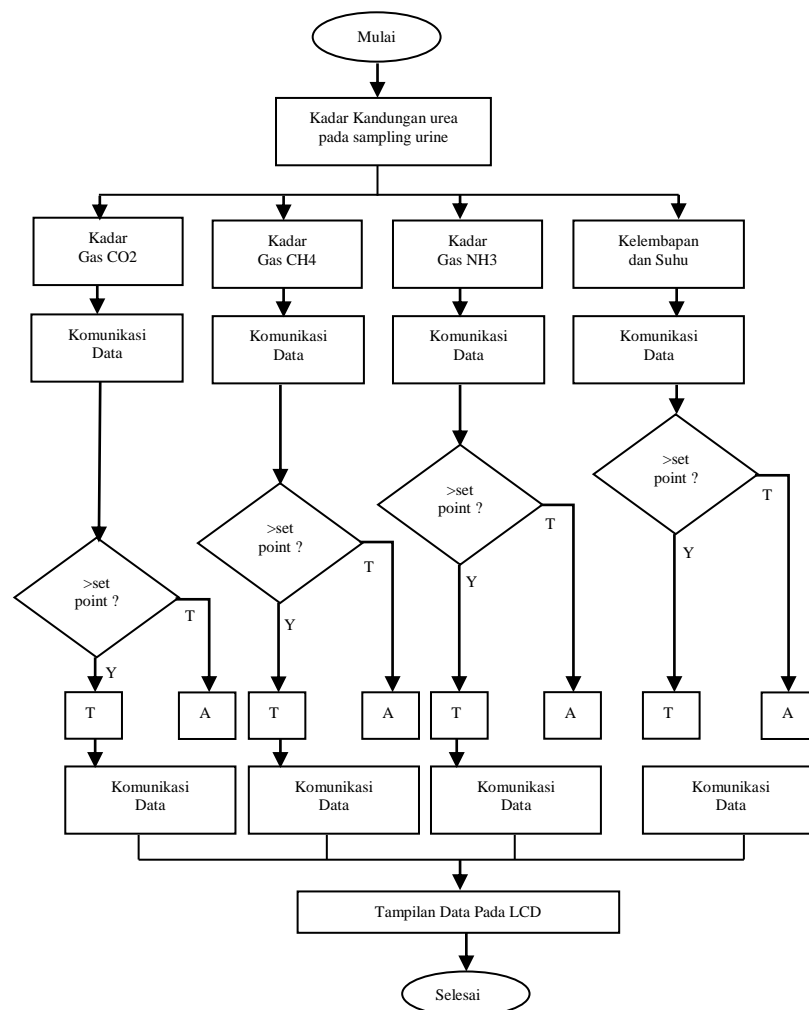
**Gambar 3.8.** Data Pin Arduino Uno yang Digunakan

Pada skema data pin arduino uno yang digunakan untuk *output* sensor PIN (0-2) adalah pin inputan Analogue PIN D (0-1) adalah

$R_x$  dan  $T_x$ , sehingga tegangan disupply dari pin 3.3 V Arduino, LCD juga disupply tegangan dari Pin 5V dari arduino. ISP dihubungkan dengan PC untuk proses pemrograman.

### 3.3.2 Perancangan Software

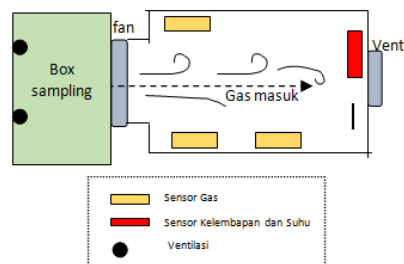
Perancangan software menggunakan aplikasi Arduino IDE 1.8.5, yang berfungsi sebagai aplikasi programming atau koding dari Arduino Uno. Program akan bekerja dengan menganalisa data hasil uji coba dan menyimpan dalam memori untuk diolah dan diukur pada aplikasi MATLAB di PC.



**Gambar 3.9.** Flowchart Program Pengukuran Bau Urea Pada Urine

## 1. Pendeteksian Gas

Untuk sistem pengukur kandungan bau urea dalam urine ini menggunakan 4 buah sensor yang mana berfungsi untuk memastikan bau urea pada urine dapat diukur secara akurat. Dalam proses pengukuran sampling menggunakan ke empat sensor dengan penempatan saling berhadapan pada sisi kanan, kiri, dan depan sesuai aliran udara yang di sedot *fan*. Berikut ini gambar sistem pendeteksian gas sebelum disusunnya *software*.



**Gambar 3.10.** Sistem Pendeteksian Gas

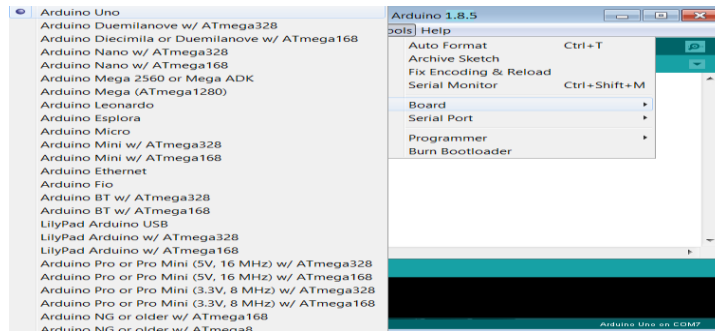
## 2. Proses Pemrograman

- a. Proses pemrograman diawali dengan memprogram arduino Uno agar mampu membaca sensor ADC yang digunakan, hal pertama yang dilakukan adalah membuka aplikasi arduino yang telah terinstall.

```
File Edit Sketch Tools Help
sketch_may16b
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
```

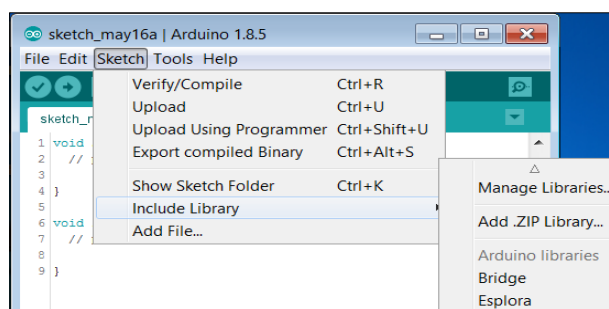
**Gambar 3.11.** Sketch Board Arduino Uno

Pilih menu Tools, lalu Board dan Pilih Arduino Uno dan apabila tidak tersedia maka pilih Arduino Genui Uno.



**Gambar 3.12.** Pemilihan Board Arduino Uno

- b. Setelah terpilih board arduino uno maka klik Sketch, pilih Include library dan pilih library yang dibutuhkan, seperti library LCD 16 x 2, dan Sensor gas. Apabila tidak ada maka dapat ditambah secara manual pada menu manage library atau add zip library dan mengikuti intruksi pada aplikasi.



**Gambar 3.13.** Penambahan Komponen Library

- c. Setelah penambahan *library*, yang harus dilakukan adalah menentukan *port-port* dari mikrokontroler yang digunakan agar pendefinisian pada koding tidak salah dan alat dapat berfungsi dengan baik. Selanjutnya adalah proses pelatihan jaringan saraf

tiruan. Pelatihan dilakukan dengan mengamati keluaran sinyal analog pada sensor gas satu persatu dan mencatatnya menggunakan aplikasi dari delphi yang telah dibuat. Data ADC acak yang timbul perlu dicocokkan dengan nilai range,  $R_s$ ,  $V_{in}$  dan  $V_{reff}$  data sheet pada sensor sehingga dapat mengidentifikasi jenis gas apa yang disensing sebelum pengukuran, Pada pengujian jaringan saraf tiruan ini diharapkan sistem dapat mengenali jenis gas dan mencocokkan terhadap target dari ispu yang telah dinormalisasi. Tetapi sebelum masuk pemrograman di MATLAB, data ADC keluaran dari DELPHI harus dikonversi menuju PPM, disini hasil pengukuran kandungan gas beracun dapat diketahui. setelah itu dilanjutkan.

**Tabel 3.1.** Data Normalisasi Sebelum Diproses JST

No	Kategori	MQ-2	MQ-4	MQ-7	MQ-135	Target
1	Sakit	0,23	0,46	0,90	0,10	1,00
2	Sakit	0,25	0,43	0,90	0,10	1,00
3	Sakit	0,22	0,45	0,90	0,10	1,00
4	Sehat	0,43	0,38	0,90	0,10	0,00
5	Sakit	0,22	0,42	0,90	0,10	1,00
6	Sakit	0,24	0,43	0,90	0,10	1,00
7	Sakit	0,22	0,54	0,90	0,10	1,00
8	Sehat	0,42	0,24	0,90	0,10	0,00
9	Sakit	0,21	0,61	0,90	0,10	1,00
10	Sakit	0,21	0,70	0,90	0,10	1,00
11	Sakit	0,17	0,53	0,90	0,10	1,00
12	Sehat	0,59	0,25	0,90	0,10	0,00
13	Sakit	0,25	0,44	0,90	0,10	1,00
14	Sakit	0,31	0,30	0,90	0,10	1,00
15	Sakit	0,23	0,46	0,90	0,10	1,00
16	Sehat	0,72	0,28	0,90	0,10	0,00
17	Sakit	0,25	0,43	0,90	0,10	1,00
18	Sakit	0,26	0,49	0,90	0,10	1,00
19	Sakit	0,27	0,44	0,90	0,10	1,00
20	Sehat	0,90	0,24	0,90	0,10	0,00

**Tabel 3.1.** Data Normalisasi Sebelum Diproses JST

No	Kategori	MQ-2	MQ-4	MQ-7	MQ-135	Target
21	Sakit	0,27	0,46	0,90	0,10	1,00
22	Sakit	0,21	0,46	0,90	0,10	1,00
23	Sakit	0,23	0,41	0,90	0,10	1,00
24	Sehat	0,90	0,23	0,84	0,10	0,00

konversi data ADC ke satuan PPM, harus memperhatikan *datasheet* sensor yang mana pada setiap sensor memiliki tingkat kesensitifan yang berbeda-beda terhadap jenis gas tertentu yang dinyatakan dalam satuan *range*, dan dinyatakan program arduino 10 bit ( $2^{10} = 1024$ ), dengan nilai X yang didapat dengan formula sebagai berikut :

$$X = \frac{BIT\ RATE}{RANGE} \dots\dots\dots(1)$$

$$\text{Konversi ADC} = \frac{V_{in}}{V_{ref}} \times \text{Bit Rate} \dots\dots\dots(2)$$

$$\text{Kadar Gas (ppm)} = \text{Konv ADC} \times X \dots\dots\dots(3)$$

Dengan tiga tahap formula diatas, maka nilai konversi ppm dapat dicari, berikut tabel perhitungan sensor gas yang digunakan berdasarkan data sheetnya.

**Tabel 3.2.** Nilai Konversi Sensor Gas

MQ-7	CO		MQ-135	NOX		MQ-4	CH4		MQ-2	SOX	
BIT RATE	1024	BIT	BIT RATE	1024	BIT	BIT RATE	1024	BIT	BIT RATE	256	BIT
RANGE	1980	Ppm	RANGE	290	Ppm	RANGE	9700	Ppm	RANGE	199	Ppm
X	1,933594	Satuan	X	0,283203	Satuan	X	9,472656	Satuan	X	0,777344	Satuan
V.ref	5	V	V.ref	5	V	V.ref	5	V	V.ref	5	V
V.in	2,5	V	V.in	2,5	V	V.in	2,5	V	V.in	2,5	V
Konversi (ADC)	512	Satuan	Konversi (ADC)	512	Satuan	Konversi (ADC)	512	Satuan	Konversi (ADC)	128	Satuan
kadar gas	990	Ppm	kadar gas	145	Ppm	kadar gas	4850	Ppm	kadar gas	99,5	Ppm

Sebagai contoh pengujian konversi gas dari data ADC ke PPM dan Normalisasi akan diberikan contoh dibawah ini :

Diketahui :

- Objek penelitian = urin penderita Diabetes Mellitus
- Nilai stabil ADC MQ-7 yang dirata-rata = 82,92
- Range Sensor MQ-7 terhadap gas CO = 2000–20 = 1980 ppm
- Bit Rate ( $2^{10}$ ) = 1024
- V.in = 2.5 V
- V.ref = 5 V

Dari informasi data yang didapat dari sensor MQ-2 maka dapat dicari nilai kadar gasnya (ppm) dengan cara sebagai berikut :

- $X = \frac{Range}{Byte Rate} = \frac{1980}{1024} = 1,93$
- Konv.ADC =  $(\frac{V.in}{V.ref}) \times Byte rate = (\frac{2,5}{5}) \times 1024 = 512$
- $PPM = (\frac{ADC}{\frac{100}{V.ref}}) \times Byte rate \times X$
- $= (\frac{80}{5}) \times 1024 \times 1,93$
- $= 316,21$

Tahap berikutnya adalah menormalisasi nilai-nilai yang ada pada hasil percobaan dan data ISPU sebagai target kedalam range (0-1), hal ini dilakukan untuk memudahkan pemrograman pada MATLAB dalam menentukan nilai input ke nilai target dan dalam membuat suatu matrix pada pemrograman, sehingga didapatkan suatu bobot. Nilai normalisasi dapat dicari dengan formula ;

$$X \text{ (Normalisasi)} = \frac{(N.aktual - Xb) \times 0,8}{Xa - Xb} + 0,1 \dots \dots \dots (4)$$

Keterangan : N.aktual = nilai sebenarnya,

:  $X_a$  = nilai batas atas dan  $X_b$  = nilai batas bawah

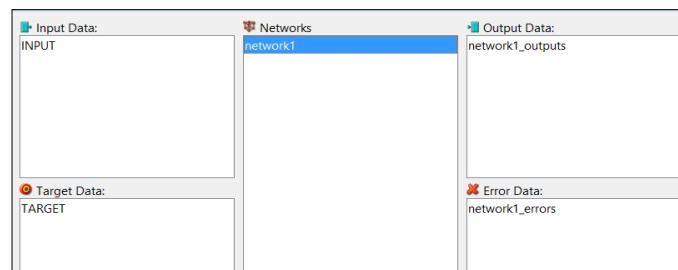
Data yang akan diinputkan kedalam MATLAB adalah data normalisasi hasil pelatihan terhadap berbagai jenis urine yang dilakukan berulang kali agar sistem mampu mengenali variable urine yang berbeda saat pengujian. Berikut ditampilkan data sampling dan data target.

**Tabel 3.3.** Data Latih sebagai Input JST

No	Kategori	MQ 2	MQ 4	MQ 7	MQ 135	Target
1	Sakit	0,59	0,25	0,90	0,10	1,00
2	Sakit	0,23	0,41	0,90	0,10	1,00
3	Sakit	0,27	0,44	0,90	0,10	1,00
4	Sehat	0,67	0,21	0,90	0,10	0,00
5	Sakit	0,62	0,29	0,90	0,10	1,00
6	Sakit	0,21	0,46	0,90	0,10	1,00
7	Sakit	0,26	0,49	0,90	0,10	1,00
8	Sehat	0,90	0,23	0,84	0,10	0,00
9	Sakit	0,48	0,21	0,90	0,10	1,00
10	Sakit	0,27	0,46	0,90	0,10	1,00
11	Sakit	0,25	0,43	0,90	0,10	1,00
12	Sehat	0,90	0,24	0,90	0,10	0,00
13	Sakit	0,27	0,44	0,90	0,10	1,00
14	Sakit	0,23	0,46	0,90	0,10	1,00
15	Sakit	0,17	0,53	0,90	0,10	1,00
16	Sehat	0,72	0,28	0,90	0,10	0,00
17	Sakit	0,31	0,30	0,90	0,10	1,00
18	Sakit	0,21	0,70	0,90	0,10	1,00
19	Sakit	0,24	0,43	0,90	0,10	1,00
20	Sehat	0,59	0,25	0,90	0,10	0,00
21	Sakit	0,25	0,44	0,90	0,10	1,00
22	Sakit	0,21	0,61	0,90	0,10	1,00
23	Sakit	0,22	0,42	0,90	0,10	1,00
24	Sehat	0,42	0,24	0,90	0,10	0,00



- d. Selanjutnya adalah melatih sistem dengan pelatihan menggunakan aplikasi MATLAB 2013 dengan fitur implementasi *nntool* (*neural network tool*). Pada menu *command* “*nntool*” dan *run*, akan muncul *nntool* manager, setelah itu masukkan (*import*) file input dan target yang telah disiapkan, dan juga buat *network* file di pilihan *new* dengan mensetting jumlah *layer* dan *neuron* sesuai data yang kita inputkan, sehingga akan muncul pada *nntool* manager sebagai berikut :

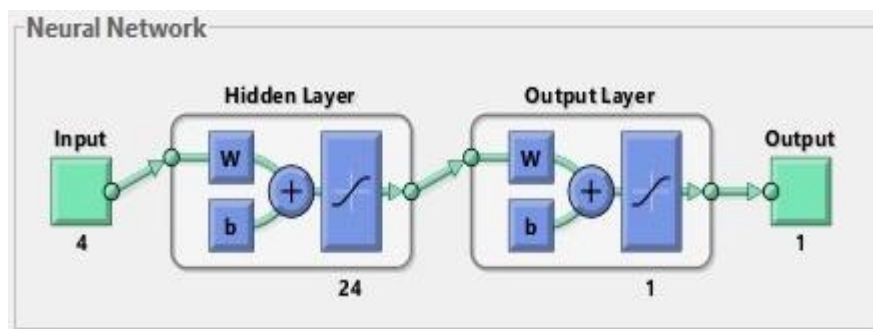


**Gambar 3.14.** Halaman *workspace nntool manager*

- e. Tahap berikutnya adalah melatih jaringan saraf tiruan dengan memasukan parameter yang ditentukan pada *network* yang dibuat. Pada *sheet train*, *simulate* serta *adapt* perlu ditentukan file *input* dan targetnya.

- Nilai epoch = 1000
- Nilai goals = 0,001
- Hidden layer = 2
- Jenis algoritma = Tansig
- Nilai validation check = 10
- Time = inf (tidak dibatasi)

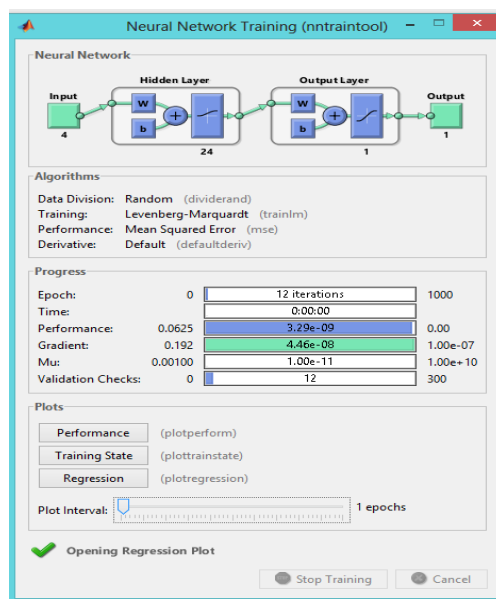
- Neuron = 24 (jml data)



**Gambar 3.15** Arsitektur *Network Backpropagation*

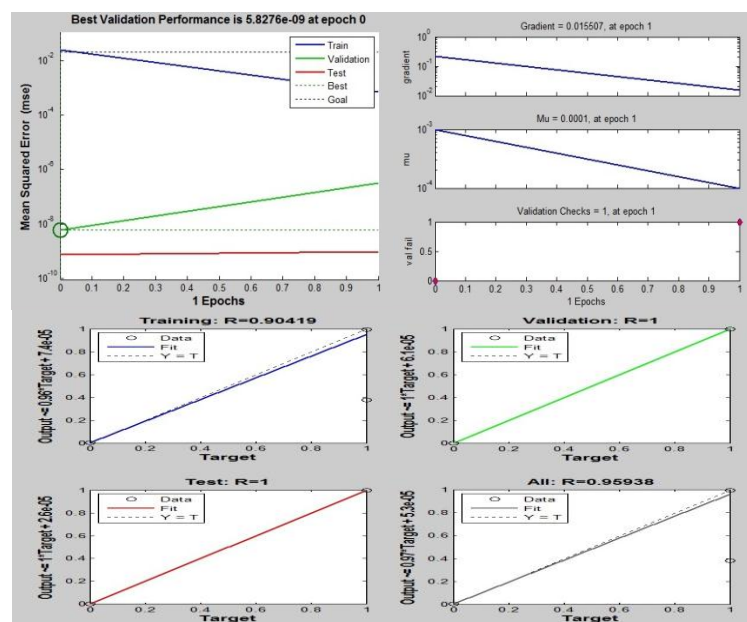
- f. Setelah parameter ditentukan maka klik tombol “*Train Network*” sehingga akan muncul tampilan yang menunjukkan *Performance*, *Training state* dan regresi dengan hasil :

- Nilai performance =  $3.29e-09$
- Gradient =  $4.46e-08$
- Epoch = 12
- Best validation = 12



**Gambar 3.16** Proses *Training Backpropagation*

- g. Selanjutnya kembali ke “*tool manager*” dan pada kolom *output* dan *error* sudah terdapat data hasil pemrosesan Jaringan Saraf Tiruan *Backpropagation*. Untuk menentukan keakurasian hasil output terhadap nilai target maka data harus dibandingkan dengan *microsoft excel*. Untuk memprogram pada mikrokontroler arduino, maka nilai bobot pada *sheet “network”* bisa dimasukan pada *notepad* sementara untuk proses pemrograman pada arduino. Data koding pemrograman ada pada halaman lampiran. Hasil dari pemrosesan JST dikatakan akurat jika hasil perbandingan antara nilai target dan output JST dalam range 90 sampai 99 persen.
- h. Selanjutnya adalah menguji perbandingan antara nilai kelaran JST pada proses pengujian dengan data target dengan melakukan langkah satu sampai empat.



**Gambar 3.17** Pengujian target latih terhadap output JST

### 3. Mengimplementasikan JST Pada Arduino

Setelah melakukan pelatihan dan pengujian jaringan saraf backpropagation tahapan terakhir adalah mengimplementasikan nilai weight (bobot) pada mikrokontroler arduino uno, nilai bobot ini bisa didapat setelah pengujian sistem pada worksheet *network* . Nilai bobot ini sangatlah penting dalam penggunaannya saat sistem ini diberi sampling dengan kadar gas urin. Pada tahap inilah JST *Backpropagation* bekerja. Dengan nilai bobot (*weight*) nilai bias serta fungsi aktivasi yang disertakan pada lembar lampiran dimasukkan kedalam program fungsi GUI pada MATLAB dengan tampilan sebagai berikut :



**Gambar 3.18.** Tampilan Gui Pendeteksi Gas