



LAPORAN SKRIPSI

**RANCANG BANGUN ALAT *ELECTRONIC NOSE* (E-NOSE)
UNTUK IDENTIFIKASI KESEGARAN DAGING
MENGUNAKAN SENSOR METAL OXIDE
SEMICONDUCTOR (MOS)**

NAMA : RACHMAWAN WIJAYA

NIM : 201452015

DOSEN PEMBIMBING

BUDI GUNAWAN, S.T., M.T.

SOLEKHAN, ST., M.T.

PROGRAM STUDI TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS MURIA KUDUS

2018

HALAMAN PERSETUJUAN

RANCANG BANGUN ALAT *ELECTRONIC NOSE* (E-NOSE) ALAT IDENTIFIKASI KESEGERAN DAGING MENGGUNAKAN SENSOR METAL OXIDE SEMICONDUCTOR (MOS)

RACHMAWAN WIJAYA

NIM. 201452015

Kudus, 29 Agustus 2018

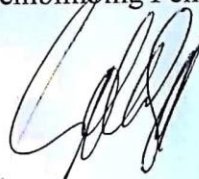
Menyetujui,

Pembimbing Utama,



Budi Gunawan, S.T., M.T.
NIDN: 0613027301

Pembimbing Pendamping,



Solekhan, ST., MT
NIDN: 0619057201

Mengetahui

Koordinator Skripsi/Tugas Akhir



Imam Abdul Rozaq, S.Pd., M.T.
NIDN: 0629088601

HALAMAN PENGESAHAN

**RANCANG BANGUN ALAT *ELECTRONIC NOSE* (E-NOSE) ALAT
IDENTIFIKASI KESEGRAN DAGING MENGGUNAKAN SENSOR
METAL OXIDE SEMICONDUCTOR (MOS)**

RACHMAWAN WIJAYA

NIM. 201452015

Kudus, 29 Agustus 2018

Menyetujui,

Ketua Penguji,



Mohammad Iqbal, S.T., M.T.

NIDN: 0619077501

Anggota Penguji I,



Imam Abdur Rozaq, S.Pd., M.T.

NIDN: 0629088601

Anggota Penguji II,



Budi Gunawan, S.T., M.T.

NIDN: 0613027301

Mengetahui



Dekan Fakultas Teknik

Mohammad Dahlan, ST, MT

NIDN. 0601076901

Ketua Program Studi

Teknik Elektro



Mohammad Iqbal, S.T., M.T.

NIDN: 0619077501

PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini :

Nama : Rachmawan Wijaya
NIM : 201452015
Tempat & Tanggal Lahir : Kudus, 18 September 1996
Judul Skripsi : Rancang Bangun Alat *Electronic Nose* (E-Nose)
Alat Identifikasi Kesegaran Daging Menggunakan
Sensor Metal Oxide Semiconductor (Mos)

Menyatakan dengan sebenarnya bahwa penulisan Skripsi ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan lain yang tercantum sebagai bagian dari Skripsi ini. Seluruh ide, pendapat, atau materi dari sumber lain telah dikutip dalam Skripsi dengan cara penulisan referensi yang sesuai.

Demikian pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar dan sanksi lain sesuai dengan peraturan yang berlaku di Universitas Muria Kudus.

Demikian pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Kudus, 16 Juli 2018

Yang memberi pernyataan,



Rachmawan Wijaya

NIM. 201452015

**RANCANG BANGUN ALAT *ELECTRONIC NOSE* (E-NOSE) ALAT
IDENTIFIKASI KESEGARAN DAGING MENGGUNAKAN SENSOR
METAL OXIDE SEMICONDUCTOR (MOS)**

Nama : Rachmawan Wijaya

NIM : 201452015

Pebimbing :

1. Budi Gunawan, S.T., M.T.
2. Solekhan, S.T., M.T.

RINGKASAN

Seiring dengan kemajuan waktu konsumsi daging dari waktu ke waktu meningkat, dengan meningkatnya konsumsi daging akan banyak orang yang tidak bertanggung jawab untuk menjual daging dengan daging yang kurang segar, untuk mengantisipasi hal tersebut maka dibuatlah alat elektronik yang dapat mendeteksi kesegaran daging atau disebut hidung elektronik. Hidung Elektronik adalah alat yang dapat mengenali gas atau bau atau aroma yang memiliki fungsi seperti indra manusia pada hidung. Penelitian ini memiliki latar belakang karena untuk mempercepat dari pendeteksian suatu daging dengan cepat dan akurat.

Sensor gas MOS yang digunakan pada penelitian ini adalah TGS2600, TGS2602, TGS2620, MQ135, TGS813. Dengan menggunakan kelima sensor MOS output dari sensor kalulasi *neural network feed forward* dimasukkan kedalam mikrokontroller. Metode penelitian yang digunakan meliputi: 1) Merancang dan membuat alat portabel E-NOSE 2) Pengujian Pendeteksian Tidak Langsung dan Pendeteksian Langsung menggunakan data bobot dan bias dari matlab 3) Pengujian Pendeteksian Tidak Langsung dan Pendeteksian Langsung menggunakan data dari berat dan bias menggunakan visual basic.

Hasil dari penelitian ini adalah terciptanya alat E-NOSE yang bersifat portabel dan mudah untuk penggunaannya, dengan hasil Pendeteksian Tidak Langsung dan Pendeteksian Langsung mendapatkan tingkat keberhasilan 100%.

Kata kunci: Arduino, daging, hidung elektronik, *Neural Network*, portabel

**DESIGN OF ELECTRONIC NOSE EQUIPMENT (E-NOSE)
IDENTIFICATION EQUIPMENT OF MEAT SUCCESS USING METAL
OXIDE SEMICONDUCTOR (MOS)**

Name of Student : Rachmawan Wijaya

Student Identify Number : 201452015

Supervisor :

3. Budi Gunawan, S.T., M.T.

4. Solekhan, S.T., M.T.

ABSTRACT

Along with the increasing time of meat consumption from time to time increases, with the increase in meat consumption, there will be many people who are not responsible for selling meat with less fresh meat. To anticipate this, an electronic device that can detect the freshness of the meat is called electronic nose. The Electronic Nose is a device that can recognize gases or smells or scents that have a function like the human senses in the nose. This research has a background because to accelerate the detection of a meat quickly and accurately.

MOS gas sensors used in this study are TGS2600, TGS2602, TGS2620, MQ135, TGS813. By using the five output MOS sensors from the neural network sensor the forward feed is inserted into the microcontroller. The research methods used include: 1) Designing and making E-NOSE portable devices 2) Indirect Detection Testing and Direct Detection using weight data and bias from text 3) Indirect Detection Testing and Direct Detection using data from heavy and biased visual basic .

The results of this study are the creation of E-NOSE devices that are portable and easy to use, with the results of Indirect Detection and Direct Detection getting a 100% success rate.

Keywords: Arduino, meat, electronic nose, Neural Network, portable

KATA PENGANTAR

Alhamdulillah segala puji syukur penulis panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis mampu menyelesaikan penulisan laporan skripsi yang berjudul **"Rancang Bangun Alat *Electronic Nose* (E-Nose) Untuk Identifikasi Kesegaran Daging Menggunakan Sensor Metal Oxide Semiconductor (Mos)"**.

Skripsi ini ditujukan untuk memenuhi salah satu syarat memperoleh gelar Strata-satu di program studi Teknik Elektro Fakultas Teknik Universitas Muria Kudus.

Penyusunan skripsi ini tak lepas dari bantuan dan dukungan beberapa pihak, untuk itu penulis menyampaikan ucapan terima kasih kepada:

1. Allah ﷻ yang telah memberikan rahmad dan hidayah sehingga dapat menyelesaikan skripsi ini.
2. Orang tua yang memberikan dukungan serta doa.
3. Bapak Dr. H Suparno SH. MS selaku Rektor Universitas Muria Kudus.
4. Bapak Moh. Dahlan ST, MT, selaku Dekan Fakultas Teknik Universitas Muria Kudus.
5. Bapak Mohammad Iqbal ST, MT, selaku Ketua Program Studi teknik Elektro S1 Universitas Muria Kudus.
6. Bapak Budi Gunawan ST, MT, selaku Pembimbing I yang telah melibatkan saya dalam penelitian beliau serta memberikan arahan dan usulan dalam penyelesaian skripsi ini.
7. Bapak Solekhan ST., MT, selaku Pembimbing II yang telah memberikan arahan dalam penyelesaian skripsi ini.
8. Bapak Arief Sudarmadji, ST., MT., Ph.D. yang sudah melibatkan saya dalam penelitian dan mengizinkan saya untuk melakukan penelitian di Laboratorium Teknologi Pangan UNSOED serta memberi pengetahuan tentang metode *Neural Network*.
9. Seluruh dosen dan staff Program Studi Teknik Elektro yang telah membantu dan menjadi tempat bertanya jika saya mengalami kesulitan dalam penulisan skripsi ini.

10. Teman – teman mahasiswa Kisyarangga Agung Pribadi dan Malvin Taqqi Derras jurusan TEP UNSOED dan UMK yang terlibat dalam satu penelitian ini.

11. Teman–teman kuliah khususnya Program Studi Teknik Elektro Universitas Muria Kudus yang telah memberikan motivasi, kritik dan saran.

Penulis menyadari adanya kekurangan dan ketidaksempurnaan dalam penulisan skripsi ini, karena itu penulis menerima kritik, saran dan masukan dari pembaca sehingga penulis dapat lebih baik di masa yang akan datang. Akhirnya penulis berharap semoga buku tesis ini bisa bermanfaat khususnya bagi penulis dan umumnya bagi para pembaca.

Kudus, 19 Maret 2018

Penulis



DAFTAR ISI

HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN	iv
RINGKASAN.....	vi
ABSTRACT	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
DAFTAR SIMBOL	xiv
DAFTAR LAMPIRAN	xv
DAFTAR ISTILAH DAN SINGKATAN.....	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah.....	1
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Manfaat.....	3
BAB II TINJAUAN PUSTAKA	4
2.1. Daging sapi.....	4
2.2. Electronic Nose	5
2.3. Struktur <i>Neural Network</i>	7
2.4. Arduino Mega 2560	9
2.5. Sensor Gas Semikonduktor	10
2.5.1. Sensor gas TGS2620	10
2.5.2. Sensor gas TGS2602	11
2.5.3. Sensor gas TGS2600	12
2.5.4. Sensor gas TGS813	13
2.5.5. Sensor gas MQ135	14
BAB III METODOLOGI	15
3.1. Metode penelitian.....	15
3.2. Waktu dan Tempat Penelitian	15
3.3. Tahapan penelitian	15
3.4. Alat dan Bahan Yang Digunakan.....	15
3.5. Perancangan hardware.....	16
3.6. Sensor Gas Tipe MOS Yang Digunakan.....	17
3.7. Rangkaian array sensor	18
3.8. Rangkaian <i>wiring</i> E-NOSE portable	19
3.9. Diagram blok E-NOSE portable	20
3.10. Flowchart sistem pendeteksian.....	20
3.11. Struktur <i>Neural Network</i> yang digunakan.....	22
3.12. Pengujian alat	22
3.12.1. Kategori pengujian	22
3.12.2. Software yang digunakan	23
3.12.3. Prosedur pengujian.....	23

3.12.4.	Perencanaan pengujian secara Pendeteksian Tidak Langsung.....	24
3.12.5.	Perencanaan pengujian secara Pendeteksian Langsung	24
3.13.	Pengujian identifikasi berdasarkan waktu penyimpanan sample	25
3.14.	Pengambilan data training	25
3.15.	Analisa data	25
BAB IV	HASIL DAN PEMBAHASAN	26
4.1.	Alat akuisisi data	26
4.1.1.	Gambar alat akuisisi data	26
4.1.2.	Hasil pembacaan sensitifitas sensor MOS	27
4.2.	Tahap kalkulasi <i>NN feed forward</i> pada E-NOSE.....	28
4.3.	Alat E-NOSE portable.....	29
4.4.	Pengujian Pendeteksian Tidak Langsung.....	33
4.4.1.	Uji Pendeteksian Tidak Langsung menggunakan matlab	33
4.4.2.	Uji Pendeteksian Tidak Langsung menggunakan VB.....	34
4.5.	Pengujian Pendeteksian Langsung	36
4.5.1.	Pengujian real menggunakan Matlab	36
4.5.2.	Pengujian real menggunakan Visual Basic	37
4.5.3.	Hasil pengujian daging dengan jarak waktu 2 jam	38
4.6.	Analisa hasil pengujian	38
BAB V	PENUTUP	39
5.1.	Kesimpulan.....	39
5.2.	Saran.....	39
DAFTAR PUSTAKA		40
Lampiran 1	hasil kalkulasi dan sensitifitas sensor pada tampilan serial monitor	42
Lampiran 2	hasil kalkulasi dan sensitifitas sensor dengan pengambilan gas 2 jam sekali.....	43
Lampiran 3	bobot dan bias pada hasil pelatihan	45
Lampiran 4	foto kegiatan	46
Lampiran 5	code program arduino	47
BIODATA PENULIS		74

DAFTAR GAMBAR

Gambar 2. 1 Bagian Dari Daging Sapi.....	4
Gambar 2. 2 Struktur Dari Multilayer Perceptron.	8
Gambar 2. 3 Arduino Mega 2560	10
Gambar 2. 4 Bentuk Fisik Sensor Gas Tgs2620.	11
Gambar 2. 5 Detail Komponen Gas Tgs2620.	11
Gambar 2. 6 Bentuk Fisik Sensor Gas Tgs2602.	12
Gambar 2. 7 Detail Komponen Sensor Tgs2602.	12
Gambar 2. 8 Bentuk Hardware Sensor Tgs2600.....	13
Gambar 2. 9 Detail Komponen Sensor Tgs2600.	13
Gambar 2. 10 Bentuk Fisik Sensor Tgs813	14
Gambar 2. 11 Hardware Sensor Gas Mq135.	14
Gambar 3. 1 Tahapan Peleitian Yang Dilakukan.....	15
Gambar 3. 2 Rancangan Alat E-Nose Portable Dilihat Dari Atas	16
Gambar 3. 3 Rancangan Alat E-Nose Portable Dilihat Dari Depan	16
Gambar 3. 4 Rancangan Alat E-Nose Portable Dilihat Dari Samping Kiri	17
Gambar 3. 5 Rancangan Alat E-Nose Portable Dilihat Dari Perspektif 3d	17
Gambar 3. 6 Pcb Sensor Mos Bagian Atas	18
Gambar 3. 7 Pcb Sensor Mos Bagian Bawah	19
Gambar 3. 8 Wiring E-Nose Portable	19
Gambar 3. 9 Diagram Blok Aliran Daya E-Nose Portable	20
Gambar 3. 10 Diagram Blok Sistem Komunikasi Dan Kontrol E-Nose Portable	20
Gambar 3. 11 Flowchart Alur Menu Ada E-Nose Portable	21
Gambar 3. 12 Flowchart Pendeteksian A. Pendeteksian Tidak Langsung B. Pendeteksian Langsung.....	22
Gambar 3. 13 Struktur Multilayer Perceptron Dari Nn Yang Dirancang	22
Gambar 3. 14 Software Visual Basic Yang Digunakan Dalam Penelitian Ini.....	23
Gambar 4. 1 Spesifikasi Alat Akuisisi Data	26
Gambar 4. 2 Spesifikasi Alat E-Nose Bagian 1	29
Gambar 4. 3 Spesifikasi Alat E-Nose Bagian 2	30
Gambar 4. 4 Spesifikasi Alat E-Nose Bagian 3	31
Gambar 4. 5 Spesifikasi Alat E-Nose Bagian 4	32

DAFTAR TABEL

Tabel 4. 1 Tabel Hasil Sensitifitas Sensor Mos Terhadap Kondisi Daging	27
Tabel 4. 2 Hasil Data Pendeteksian Tidak Langsung Menggunakan Matlab	33
Tabel 4. 3 Hasil Data Pendeteksian Tidak Langsung Menggunakan Visual Basic	35
Tabel 4. 4 Hasil Data Pendeteksian Langsung Menggunakan Matlab.....	37
Tabel 4. 5 Hasil Data Pendeteksian Langsung Menggunakan Visual Basic.....	37
Tabel 4. 6 Tabel Pengujian Daging Dengan Jarak Waktu Setiap 2 Jam Sekali....	38



DAFTAR SIMBOL

Simbol	Keterangan	Satuan	Nomor Persamaan
R_1	Hambatan R_1 pada board sensor MOS	Ω	2.5
V_{in}	Tegangan input referensi ADC	V	2.5
V_{out}	Tegangan yang dikeluarkan oleh board sensor MOS	V	2.5
R_o	Kondisi pengukuran udara bersih (baseline)	Ω	4.1
R_g	Kondisi pengukuran udara gas	Ω	4.1
S	Sensitivitas sensor gas MOS	-	4.1
Z_j	kalkulasi hidden layer	-	4.2
N	jumlah synapse layer2 (hidden layer)	-	4.2
X_i	masukan nilai sensitivitas dari sensor	-	4.2
V_{ij}	nilai bobot input ke hidden layer	-	4.2
Z_j'	hasil perhitungan hidden layer yang ditempatkan di hidden layer	-	4.3
e^{-z_j}	eksponensial dari $-Z_j$	-	4.3
$Bias$	nilai bias	-	4.3
Y_k	kalkulasi output layer	-	4.4
M	jumlah synapse layer3	-	4.4
W_{jk}	bobot hidden ke output	-	4.4
e^{-y_k}	eksponensial dari $-Y_k$	-	4.5
Y_k'	hasil perhitungan hidden layer yang ditempatkan di output layer	-	4.5

DAFTAR LAMPIRAN

Lampiran 1 Hasil Kalkulasi Dan Sensitifitas Sensor Pada Tampilan Serial Monitor	42
Lampiran 2 Hasil Kalkulasi Dan Sensitifitas Sensor Dengan Pengambilan Gas 2 Jam Sekali	43
Lampiran 3 Bobot Dan Bias Pada Hasil Pelatihan.....	45
Lampiran 4 Foto Kegiatan	46
Lampiran 5 <i>Code</i> Program Arduino.....	47



DAFTAR ISTILAH DAN SINGKATAN

Mos	: <i>Metal Oxide Sensor</i>
VB	: Visual Basic
MATLAB	: <i>Matrix Laboratory</i>
MLP	: Multi Layer Perceptron



BAB I PENDAHULUAN

1.1. Latar Belakang

Kesegaran daging adalah faktor utama untuk menentukan kualitas daging dan kesegaran daging tersebut juga untuk menentukan daging tersebut masih layak konsumsi, saat ini masih banyak menggunakan cara tradisional untuk menentukan daging tersebut segar atau busuk yaitu menggunakan indra hidung sebagai identifikasi kesegaran daging, selain itu juga dapat menggunakan laboratorium untuk menentukan kesegaran daging tersebut tetapi akan memakan banyak waktu lama dan memakan banyak biaya, oleh karena itu dibangunlah sebuah sistem yang dapat memudahkan identifikasi kesegaran daging dengan cepat yaitu elektronik nose, sistem ini ke arduino dengan menggunakan sensor gas sebagai alat pendeteksi kesegaran daging

Pada tugas akhir ini digunakan metode pendekatan *Neural Network* sebagai metode untuk melakukan pengenalan pola tingkat kesegaran daging yang akan diuji berupa nilai tegangan dari output sensor gas yaitu sensor gas TGS2600, TGS2602, TGS2620, TGS813, MQ135. Terdapat 2 kondisi kesegaran daging yang di tampilkan yaitu segar dan tidak segar.

Tujuan dari tugas akhir ini adalah 1) untuk mendeteksi tingkat kesegaran daging menggunakan *Electronic Nose* (e-nose) dengan deret sensor *Metal Oxide Sensor* (MOS), 2) untuk memudahkan masyarakat atau instansi mendeteksi kesegaran daging sapi dengan menyingkat waktu untuk pendeteksian sebuah daging.

Luaran yang diharapkan dalam tugas akhir ini adalah Prototipe e-nose yang bersifat portable.

1.2. Perumusan Masalah

1. Bagaimana sensor gas dapat mengidentifikasi tingkat kesegaran suatu daging.
2. Bagaimana proses pengolahan data tegangan dari output sensor gas agar dapat diketahui tingkat kesegaran suatu daging.
3. Bagaimana cara meimplementasikan metode *Neural Network* untuk mengklasifikasi tingkat kesegaran daging.

1.3. Batasan Masalah

Masalah yg akan diteliti dan dikerjakan dalam skripsi ini adalah;

1. Alat E-NOSE yang akan digunakan khusus untuk mendeteksi tingkat kesegaran daging dan bersifat portabel dengan *supply* daya menggunakan batteray.
2. Tipe arduino yang digunakan adalah arduino mega 2560.
3. Sensor MOS yang digunakan berjumlah 5 dengan tipe; TGS813, TGS2600, TGS2602, TGS2620, MQ135
4. Pengujian alat yang akan dilakukan meliputi 2 kategori yaitu Pendeteksian Tidak Langsung dan Pendeteksian Langsung.
5. Data yang digunakan pada pengujian Pendeteksian Tidak Langsung adalah nilai bobot dan bias dari hasil simulasi 2 software, yaitu; Matlab dan VB
6. Pendeteksian Langsung yang dimaksud adalah pengujian menggunakan data bobot dan bias hasil dari software Matlab dan VB, Pendeteksian Langsung yang dimaksud adalah pengujian menggunakan gas dari daging yang diinjeksikan kedalam chamber.
7. Pada skripsi ini, tidak membuat/membangun dua software yg digunakan untuk simulasi (Matlab dan VB), tetapi hanya menggunakan data hasil simulasinya dengan seijin yang membuat.

1.4. Tujuan

1. Mampu menggunakan sensor TGS813, TGS2600, TGS2602, TGS2620, MQ135 sebagai alat identifikasi tingkat kesegaran daging
2. Mampu menampilkan data dari 5 sensor gas yang dimasukkan kedalam Arduino.
3. Mampu menerapkan metode *Neural Network* untuk mengenali tingkat kesegaran daging.
4. Mampu melakukan pengenalan terhadap tingkat kesegaran daing secara cepat yang diimplementasikan dalam Arduino.

1.5. Manfaat

1. Kemudahan dalam mendeteksi kesegaran daging sapi yang awalnya menggunakan cara konvensional.
2. Diharapkan hasil dari penelitian ini selain menambah pengetahuan juga dapat diterapkan di dalam penelitian lainnya.

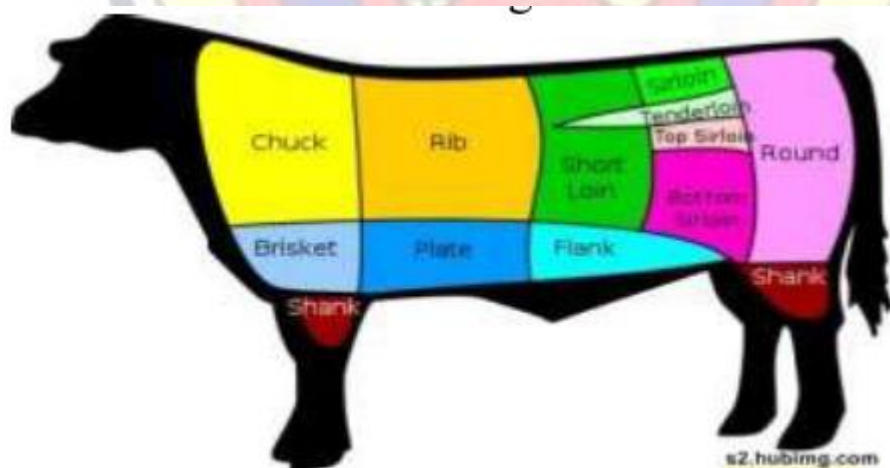


BAB II TINJAUAN PUSTAKA

2.1. Daging sapi

Konsumsi daging sapi di Indonesia pada tahun 2007 konsumsi daging sapi meningkat mencapai 453.844 ton dan selanjutnya tahun 2008 menurun mencapai angka 395.035 ton. Namun demikian konsumsi daging sapi diperkirakan akan meningkat tiap tahunnya dan akan terus meningkat seiring dengan membaiknya perekonomian Indonesia. Meningkatnya permintaan tersebut memberikan celah kepada kelompok tertentu yang memanfaatkan situasi untuk meraup keuntungan yang lebih besar. Salah satunya adalah dengan cara menjual daging sapi gelongongan, yaitu sapi yang diberi minum secara paksa sebelum disembelih. Hal ini mengakibatkan bertambahnya kadar air dalam daging, sehingga daging sapi gelongongan menjadi lebih berat bila dibanding dengan daging sapi segar yang normal. (Kiswanto, 2012).

Daging sapi dikelompokkan dalam beberapa bagian. Has Dalam atau fillet atau tenderloin adalah daging sapi dari bagian tengah badan. Has Luar atau lebih dikenal dengan nama Sirloin adalah bagian daging sapi yang berasal dari bagian bawah daging iga, terus sampai ke bagian sisi luar has dalam. Biasanya daging ini digunakan untuk membuat steak. Tapi di Indonesia biasanya menyebut bagian daging sapi dengan has luar dan has dalam saja. Bagian round, short loin, rib, chunk dan brisket disebut dengan sirloin. (Hadi and Setiawan, 2011)



Gambar 2. 1 bagian dari daging sapi (Hadi and Setiawan, 2011)

2.2. Kesegaran daging sapi

Faktor terpenting yang mempengaruhi kesegaran dan kualitas dari daging adalah aroma, warna, tekstur, dan rasa. Kualitas rasa dari daging itu sendiri ditentukan oleh banyaknya kandungan volatile organic compound (VOC) yang terdapat di dalamnya. Daging dapat diklasifikasikan menggunakan sebuah electronic nose dengan cara yang sama seperti persepsi manusia dalam menentukan kualitas dan tingkat kesegaran. Aroma atau bau dari daging terbentuk dari gabungan kompleks dari beberapa VOC yang berasal dari beragam reaksi kimia yang terjadi dalam daging. Banyak pendapat yang menyatakan jika sebuah daging segar tidak memiliki bau sama sekali (Simamora, 2017).

beberapa faktor yang dapat dijadikan pedoman untuk memilih daging segar antara lain :

a. Warna

Warna daging adalah salah satu kriteria penilaian mutu daging yang dapat dinilai langsung. Warna daging ditentukan oleh kandungan dan keadaan pigmen daging yang disebut mioglobin dan dipengaruhi oleh jenis hewan, umur hewan, pakan, aktivitas otot, penanganan daging dan reaksi-reaksi kimiawi yang terjadi di dalam daging. Warna daging sapi segar yang baik adalah warna merah cerah. Warna daging sapi yang baru dipotong yang belum terkena udara adalah warna merah keunguan, lalu jika telah terkena udara selama kurang lebih 15-30 menit akan berubah menjadi warna merah cerah. Warna merah cerah tersebut akan berubah menjadi merah-coklat atau coklat jika daging dibiarkan lama terkena udara. (Nurfawaidi, 2015)

b. Bau

Bau daging segar tidak berbau masam/busuk, tetapi berbau khas daging segar. Bau daging dipengaruhi oleh jenis hewan, pakan, umur daging, jenis kelamin, lemak, lama waktu, dan kondisi penyimpanan. Bau daging dari hewan yang tua relatif lebih kuat dibandingkan hewan muda, demikian pula daging dari hewan jantan memiliki bau yang lebih kuat daripada hewan betina. Kebusukan akan merusak daging ditandai oleh terbentuknya senyawa-senyawa berbau busuk seperti amonia, H₂S, indol, dan amin, yang merupakan hasil pemecahan protein oleh mikroorganisme. (Nurfawaidi, 2015)

c. Tekstur

Tekstur Daging yang masih segar terasa sangat kenyal, padat dan tidak kaku, jika daging segar apabila ditekan tekstur daging akan kembali seperti semula setelah ditekan. (Mas'ud and Supriyanto, 2015)

d. Kenampakan

Daging segar tidak berlendir, tidak terasa lengket ditangan dan terasa kebasahannya. Daging yang busuk sebaliknya berlendir dan terasa lengket ditangan. Selain itu permukaan daging berwarna kusam, kotor dan terdapat noda merah, hitam, biru, putih kehijauan akibat kegiatan mikroba. (Nurfawaidi, 2015)

2.3. Electronic Nose

Secara manual/tradisional sortasi tingkat kematangan buah dilakukan menggunakan indera manusia. Cara ini memerlukan seorang pakar (orang yang terlatih/terbiasa) untuk mensortasi suatu komoditas. Selain itu cara ini bersifat kurang dapat dipercaya karena subjektifitas yang tinggi dan unreproducible (bila diulangi oleh orang lain akan menghasilkan hasil yang berbeda) dalam melakukan identifikasi atribut mutu buah. Untuk dapat melakukan identifikasi atribut mutu secara tepat, maka dibutuhkan perangkat/instrumen yang digunakan untuk mengukur atribut-atribut mutu (instrumental technique) yang bersifat cepat, dapat dipecah, konsisten, dan mudah pengoperasiannya. (Sudarmaji and Ediati, 2011)

Sebuah *Electronic Nose* atau yang sering disingkat sebagai e-nose adalah instrumen analitik yang dirancang untuk meniru cara manusia dalam merasakan bau. Pada e-nose, proses analisisnya tidaklah berfokus pada identifikasi dan kuantifikasi dari campuran gas yang menguap namun lebih ke arah deskripsi kuantitatif dari profil aroma secara keseluruhan meliputi hubungan antar komponen. (Simamora, 2017)

Electronic Nose (e-nose) atau hidung elektronik merupakan sebuah instrumen yang digunakan sebagai alat pendeteksi aroma yang meniru sistem kerja indera penciuman manusia. Menurut (Pearce *et al.*, 2003) terdapat dua jenis *Electronic Nose* yaitu *direct* dan *indirect*. Pada model *direct* (langsung) pengujian dilakukan dengan menghadapkan langsung pada objek uji dengan jarak yang sangat dekat antara sensor dan aroma objek uji. Namun pada model *direct* ini masih

terdapat gangguan aroma dari udara sekitar sehingga mengganggu pembacaan sensor. Pada model *indirect* pengujian dilakukan dengan mengambil aroma objek uji melalui aliran udara yang di bawa dari ruang sampel objek uji melalui selang dan kran (*valve*) sehingga pengaruh aroma dari udara diluar dapat diminimalisir agar tidak mengganggu pembacaan sensor.

2.4. Struktur *Neural Network*

Neuron mampu melakukan tugas yang sangat kompleks seperti klasifikasi dan pemahaman pola. ANN dapat memperkirakan rentang yang cukup luas suatu model statistika dan fleksibel dalam menggambarkan model (linier maupun nonlinier). ANN dapat digunakan untuk permasalahan yang sama dengan permasalahan statistika multivariat seperti multiple regression, analisa diskriminan, dan analisa kluster. Dalam banyak kasus, hasil yang didapat dengan ANN dapat dibandingkan dengan model statistika multivariat. (Meinanda, 2009)

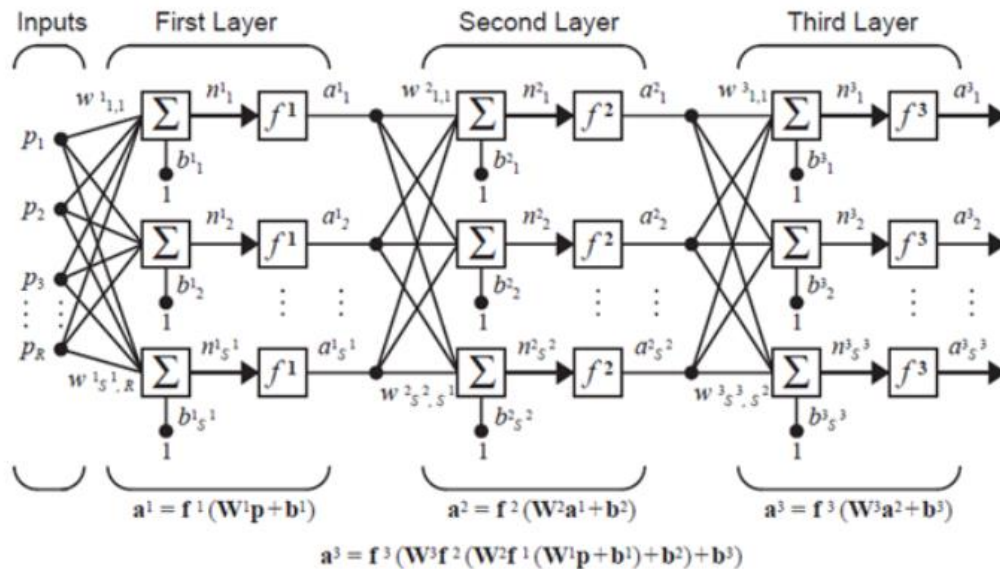
Neural Network merupakan sebuah paradigma pemrosesan informasi yang terinspirasi oleh cara sistem saraf biologis seperti otak dalam memproses informasi. Bagian penting dari paradigma ini adalah struktur baru dari sistem pemrosesan informasi. *Neural Network* terdiri atas sejumlah besar neuron yang saling terhubung yang bekerja sebagai satu kesatuan untuk memecahkan suatu masalah. Sama seperti manusia, *Neural Network* belajar berdasarkan contoh atau pengalaman. *Neural Network* dapat diatur untuk sebuah aplikasi yang spesifik seperti pattern recognition atau data classification melalui proses learning. (Simamora, 2017)

Artificial Neural Network (ANN) atau yang lebih dikenal dengan Jaringan Syaraf Tiruan (JST) telah muncul sebagai alat yang menarik untuk pemodelan proses yang kompleks. Kekuatan dari JST adalah struktur yang umum dan memiliki kemampuan untuk mempelajari dari data historikalnya. Sehingga dalam penelitian ini pengujian analisis blending kopi Arabika dan Robusta menggunakan sistem pengenalan pola Jaringan Syaraf Tiruan. (M. Surya Abdul, 2014)

Sebuah multilayer perceptron (MLP) adalah sebuah model *Neural Network* yang bersifat feedforward yang memetakan sekumpulan data input menuju ke sekumpulan output yang sesuai. Multilayer perceptron terdiri atas beberapa lapisan atau layers dari neuron dimana setiap neuron dari layer sebelumnya akan saling berhubungan dengan neuron pada layer selanjutnya. Multilayer perceptron

menggunakan teknik supervised learning berupa backpropagation untuk melakukan training atau pelatihan. MLP merupakan modifikasi dari perceptron yang tidak dapat mengenali data yang tidak dapat dipisahkan secara linear. (Simamora, 2017)

Sebuah contoh dari multilayer perceptron ditunjukkan pada Gambar 2.2 dimana terdapat 3 layer perceptron yang disusun secara seri. Output dari layer pertama menjadi input bagi layer kedua, dan output dari layer kedua menjadi input bagi layer ketiga. Dalam multilayer perceptron setiap layer dapat memiliki jumlah neuron yang berbeda, dan juga memiliki transfer function yang berbeda. Untuk mengidentifikasi struktur dari sebuah multilayer perceptron dapat digunakan sebuah notasi yang sederhana dimana jumlah input diikuti oleh jumlah neuron pada setiap layer : (Simamora, 2017)



Gambar 2. 2 Struktur dari Multilayer Perceptron.

(Simamora, 2017)

$$R - S^1 - S^2 - S^3 \quad (2.0)$$

Dimana S adalah jumlah input dari *Neural Network*, S^1 adalah jumlah neuron pada layer pertama, S^2 adalah jumlah neuron pada layer kedua dan S^3 adalah jumlah neuron pada layer ketiga. (Simamora, 2017)

Tahap Umpan Maju

Pada *input layer* x_i dari deret ke-1 hingga ke- n mengirimkan sinyal ke *hidden layer*. Masing – masing dari *hidden layer* dikalikan bobot dan dijumlahkan biasanya (Puspitaningrum, 2014).

a) Hidden layer

Nilai sum :

$$Z_j = \sum_{i=0}^N X_i \cdot V_{ij} \quad (2.1)$$

Nilai sigmoid :

$$Z'_j = \frac{1}{1+e^{-z_j+bias}} \quad (2.2)$$

b) Output layer

Nilai sum :

$$Y_k = \sum_{j=0}^M Z'_j \cdot W_{jk} \quad (2.3)$$

Nilai sigmoid :

$$Y'_k = \frac{1}{1+e^{-y_k+bias}} \quad (2.4)$$

Keterangan:

Z_j = kalkulasi hidden layer

N = jumlah synapse layer2 (hidden layer)

X_i = masukan nilai sensitivitas dari sensor

V_{ij} = nilai bobot input ke hidden layer

Z'_j = hasil perhitungan hidden layer yang ditempatkan di hidden layer

e^{-z_j} = eksponensial dari $-Z_j$

Bias = nilai bias

Y_k = kalkulasi output layer

M = jumlah synapse layer3

W_{jk} = bobot hidden ke output

e^{-y_k} = eksponensial dari $-Y_k$

Y'_k = hasil perhitungan hidden layer yang ditempatkan di output layer

2.5. Arduino Mega 2560

Arduino mega adalah sebuah papan elektronik yang dapat diprogram seperti sebuah komputer tetapi hanya saja memiliki memori yang kecil dengan dimensi yang lebih kecil dari ukuran sebuah komputer yang sebenarnya, arduino mega didesain untuk lebih ke project yang kompleks dengan pin arduino sebanyak 54 pin digital input/output, 16 analog input, dan lebih banyak ruang untuk membuat list program dan direkomendasikan untuk project robotika (arduino.cc).



Gambar 2. 3 Arduino mega 2560

(Adzhar, 2015)

Konversi ADC ke satuan hambatan

$$X = \frac{(R1.Vin) - (Vout . R1)}{Vout} \quad (2.5)$$

Keterangan:

X = konversi ADC ke hambatan (ohm)

R1 = hambatan R1 (ohm)

Vin = tegangan input (5v)

Vout = tegangan yang dibaca oleh ADC arduino (0 - 1024)

2.6. Sensor Gas Semikonduktor

2.6.1. Sensor gas TGS2620

Sensor ini dapat mendeteksi beberapa gas, yaitu gas methane, CO, Iso butan, hydrogen dan ethanol. Fitur dari sensor ini adalah :

1. Konsumsi daya rendah.
2. Sensitivitas tinggi terhadap alkohol dan uap pelarut organik.
3. Tahan lama dan biaya rendah.
4. Menggunakan rangkaian listrik sederhana.

Penggunaan :

1. Penguji alkohol.
2. Detektor uap Organik .

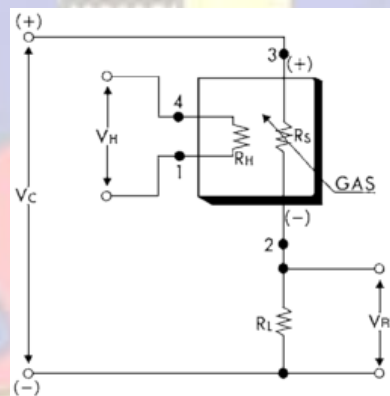
Sensor TGS2620 ini mempunyai elemen-elemen untuk mendeteksi gas, terdiri dari lapisan logam oksida semikonduktor berbentuk substrat alumina dari sebuah chip sensor yang terintegrasi dengan heater. Dengan adanya gas yang

terdeteksi, tingkat konduktivitas sensor akan naik tergantung pada tingkat konsentrasi gas di udara. Sehingga dalam sensor ini akan mengeluarkan output berupa hambatan, TGS2620 memiliki sensitivitas tinggi terhadap uap dan pelarut organik serta uap yang mudah menguap lainnya. Sensor ini juga memiliki kepekaan terhadap berbagai gas yang mudah terbakar seperti karbon monoksida, sensor TGS2620 hanya membutuhkan arus untuk heater sebesar 42mA (Surobramantyo, 2016).



Gambar 2. 4 bentuk fisik sensor gas TGS2620.

(Figaro Engineering Inc, 2014)



Gambar 2. 5 detail komponen gas TGS2620.

(Figaro Engineering Inc, 2014)

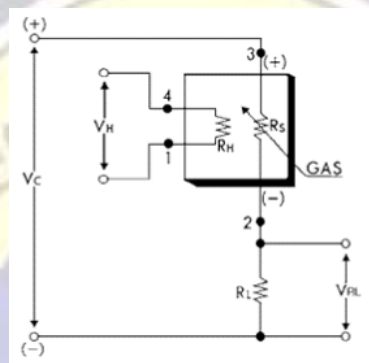
2.6.2. Sensor gas TGS2602

Sensor TGS 2602 memiliki sensitivitas yang tinggi terhadap gas berbau dengan konsentrasi rendah seperti ammonia dan H_2S yang dihasilkan dari limbah rumah tangga dan lingkungan perkantoran. Sensor ini juga memiliki sensitivitas yang tinggi terhadap gas VOC berkonsentrasi rendah seperti toluene yang dihasilkan dari pemolesan kayu dan produk bangunan. TGS 2602 hanya

membutuhkan arus heater sebesar 42 mA dan divais ini dibangun dalam bentuk TO-5 package (Simamora, 2017).



Gambar 2. 6 betuk fisik sensor gas TGS2602.
(Figaro Engineering Inc, 2008)



Gambar 2. 7 detail komponen sensor TGS2602.
(Figaro Engineering Inc, 2008)

2.6.3. Sensor gas TGS2600

Sensor TGS 2600 adalah sensor yang digunakan untuk mendeteksi udara segar (O_2) pada lingkungan yang terkontaminasi. Fitur dari sensor ini adalah :

1. Sensitivitas tinggi untuk senyawa organik yang mudah menguap dan gas berbau.
2. Konsumsi daya rendah.
3. Sensitivitas tinggi untuk kontaminasi pada udara segar (O_2).
4. Tahan lama.
5. Ukuran kecil.

Penggunaan :

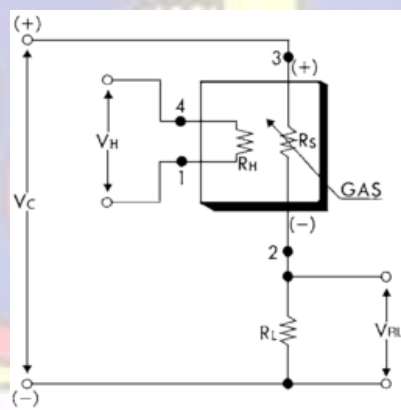
1. Pembersih udara.
2. Kontrol Ventilasi.
3. Memantau Kualitas udara.

Sensor TGS2600 ini mempunyai elemen-elemen untuk mendeteksi gas, terdiri dari lapisan logam oksida semikonduktor berbentuk substrat alumina dari sebuah chip sensor yang terintegrasi dengan heater. Dengan adanya gas yang terdeteksi, tingkat konduktivitas sensor akan naik tergantung pada tingkat konsentrasi gas di udara. Sehingga dalam sensor ini akan mengeluarkan output berupa hambatan. TGS 2600 memiliki sensitivitas yang tinggi untuk konsentrasi gas yang rendah di udara seperti hidrogen dan karbon monoksida yang ada dalam asap rokok. Sensor dapat mendeteksi hidrogen pada beberapa tingkat ppm, TGS 2600 hanya membutuhkan arus heater sebesar 42mA (Surobramantyo, 2016).



Gambar 2. 8 bentuk hardware sensor TGS2600.

(Figaro, 2005)



Gambar 2. 9 detail komponen sensor TGS2600.

(Figaro, 2005)

2.6.4. Sensor gas TGS813

Bahan detektor gas dari sensor gas semikonduktor adalah metal oksida, khususnya senyawa SnO_2 . Struktur sensor ini dapat dilihat pada Gambar.1. Ketika kristal metal oksida (SnO_2) dihangatkan pada temperatur tertentu, oksigen akan diserap pada permukaan kristal dan oksigen di udara akan terionisasi dan terikat pada SnO_2 dalam bentuk ion-ion negatif. (Utami *et al.*, no date)



Gambar 2. 10 bentuk fisik sensor TGS813
(Figaro Engineering Inc, 2002)

2.6.5. Sensor gas MQ135

Sensor MQ-135 merupakan sensor gas yang bisa digunakan dalam peralatan kontrol kualitas udara untuk bangunan / kantor untuk medeteksi gas amonia (NH_3), Nitrogen Oxide (NO_x), bensol, asap, Carbon Dioxide (CO_2). Materi sensitif dari sensor gas MQ-135 ini adalah SnO_2 (Wardhani. Andri, Susilo.Bambang, 2013).



Gambar 2. 11 hardware sensor gas MQ135.
(olimex, 2016)

BAB III METODOLOGI

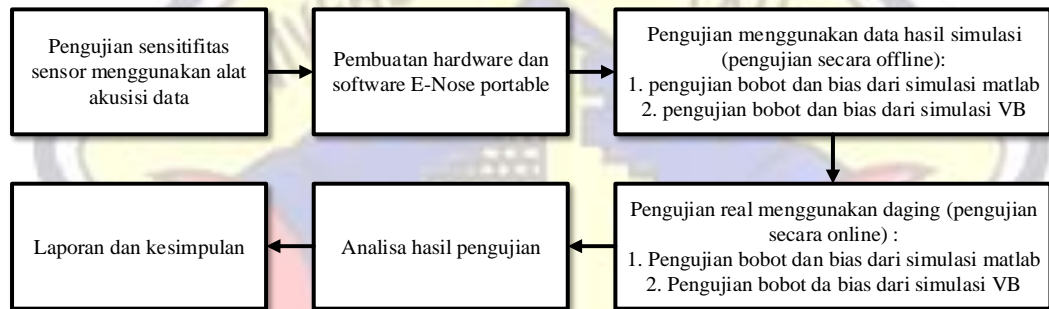
3.1. Metode penelitian

Metologi yang digunakan adalah riset development yang dihasilkan adalah prototipe alat e-nose untuk mendeteksi kondisi daging

3.2. Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan pada bulan November 2017 s/d Mei 2018 di Laboratorium Teknologi Pangan Universitas Jendral Sudirman Purwokerto dan Laboratorium Teknik Elektro Universitas Muria Kudus pada bulan november 2017 sampai juli 2018.

3.3. Tahapan penelitian



Gambar 3. 1 tahapan peleitian yang dilakukan

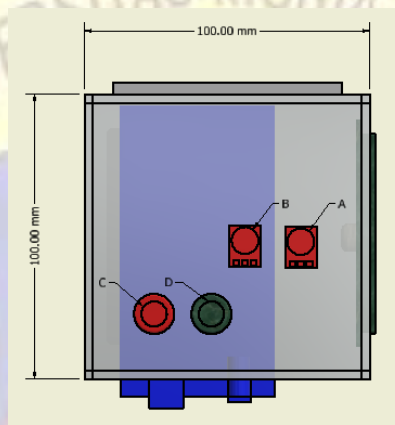
3.4. Alat dan Bahan Yang Digunakan

1. Perangkat keras alat penelitian
 - a) Sensor MOS
 - b) Arduino Mega 2560
 - c) *Push button*
 - d) Sensor sentuh
 - e) Relay
 - f) Led
 - g) Modul charger
 - h) Batteray 18650
 - i) Stepup DC
 - j) LCD 1602
 - k) Voltmeter
 - l) Personal Computer atau laptop

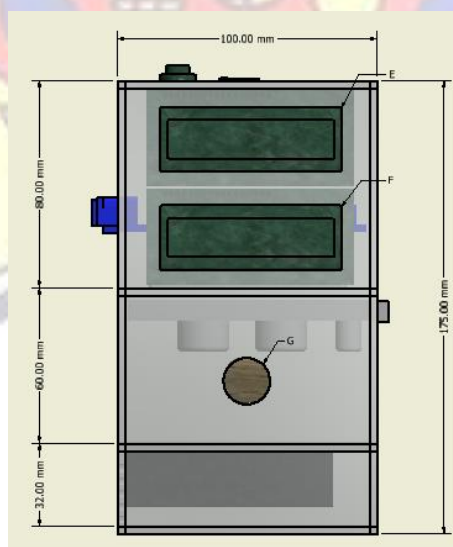
2. Perangkat lunak
 - a) Arduino IDE
 - b) Matlab 2016
 - c) VB v6

3.5. Perancangan hardware

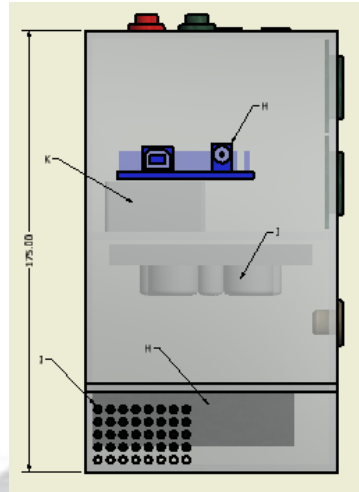
Hardware yang dirancang adalah rancangan wadah untuk memasukkan semua alat dan bahan menjadi satu alat yang bersifat portable, yang terdiri dari LCD, batteray, *push botton*, sensor sentuh, mikroprosesor, relay, charger dan lainnya, berikut adalah gambar rancangan hardware ;



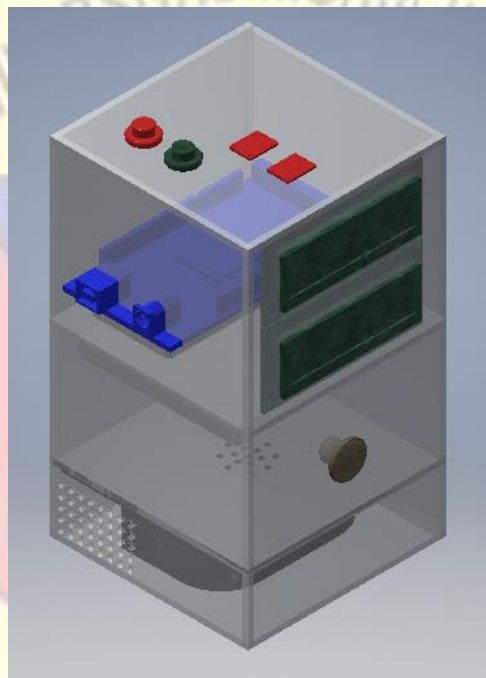
Gambar 3. 2 rancangan alat E-NOSE portable dilihat dari atas



Gambar 3. 3 rancangan alat E-NOSE portable dilihat dari depan



Gambar 3. 4 Rancangan alat E-NOSE Portable dilihat dari samping kiri



Gambar 3. 5 Rancangan alat E-NOSE Portable dilihat dari perspektif 3D

Gambar tersebut adalah gambar yang dibuat menggunakan software 3D inventor 2017 untuk memperkirakan dimensi dan penempatan pada alat yang akan dibuat.

3.6. Sensor Gas Tipe MOS Yang Digunakan

Ada 5 tipe sensor gas MOS yang digunakan untuk pengumpulan data dari pembacaan sensor yang terpasang pada *electronic nose*. Pada dasarnya penggunaan 5 sensor ini adalah untuk memperbesar dan memperbanyak dimensi dalam pengenalan pola aroma yang dimasukkan ke dalam *Neural Network* sehingga terjadi

peningkatan pengenalan aroma dengan tingkat kesulitan dan karakteristik yang berbeda – beda.

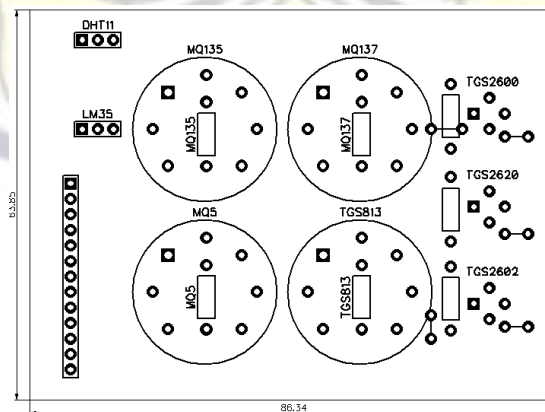
Tabel 3.1. Tipe sensor MOS yang di pakai dalam *electronic nose*

No	Gambar Sensor	Nama Sensor	Range hambatan pada sensor	Suhu kerja yang digunakan
1		TGS 2600	10k Ω -90k Ω pada udara	35 $^{\circ}$ C
2		TGS 2602	10k Ω -100k Ω pada udara	35 $^{\circ}$ C
3		TGS 2620	1k Ω -5k Ω pada ethanol	35 $^{\circ}$ C
4		MQ 135	30k Ω -200k Ω pada udara	35 $^{\circ}$ C
5		TGS 813	5k Ω -15k Ω pada udara	35 $^{\circ}$ C

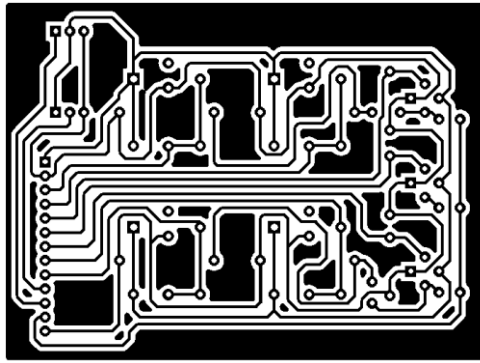
Sumber didapatkan dari datasheet untuk setiap tipenya

3.7. Rangkaian array sensor

Rangkaian sensor ini berupa rangkaian sensor yang berupa jalur pcb, rangkaian sensor ini mempunyai 9 sensor tetapi yang digunakan hanya 5 sensor.



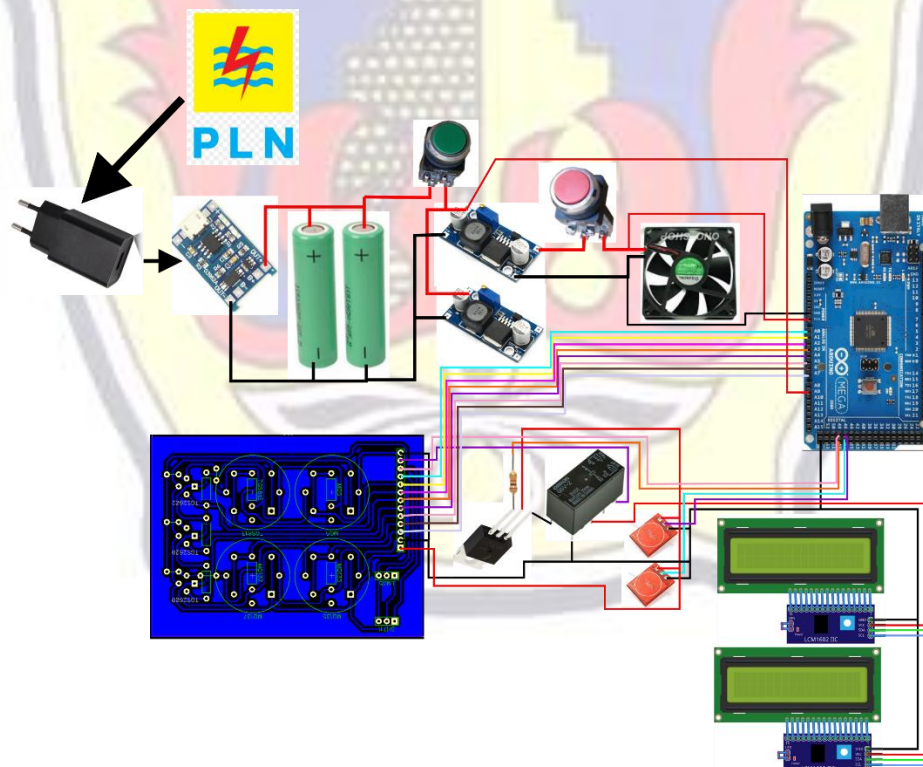
Gambar 3. 6 PCB sensor MOS bagian atas



Gambar 3. 7 PCB sensor MOS bagian bawah

Pada gambar 4.6 dan gambar 4.7 adalah gambar layout dari PCB sensor dari sensor MOS yang berfungsi dari pendeteksian untuk gas yang akan disensor, dimana pada gambar 4.6 adalah layuot atas dan 4.7 adalah layout bawah. Pcb ini berada didalam chamber pengujian yang berfungsi untuk mendeteksi gas dari daging, keluaran dari pcb ini adalah tegangan analog yang dihubungkan ke pin analog pada arduino mega 2560.

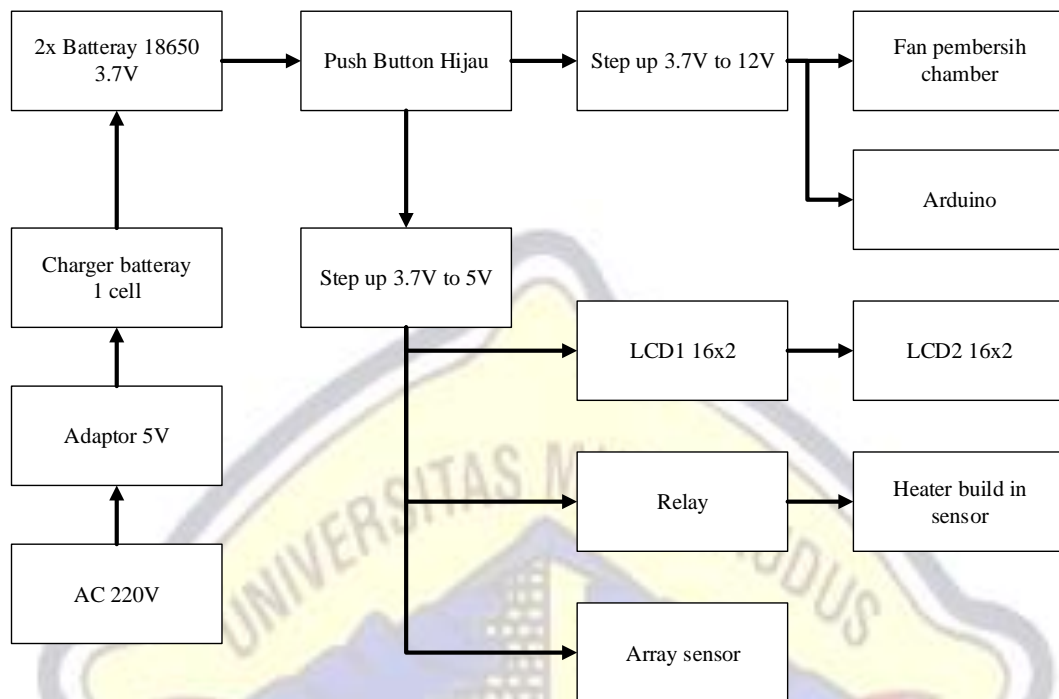
3.8. Rangkaian *wiring* E-NOSE portable



Gambar 3. 8 Wiring E-NOSE PORTABLE

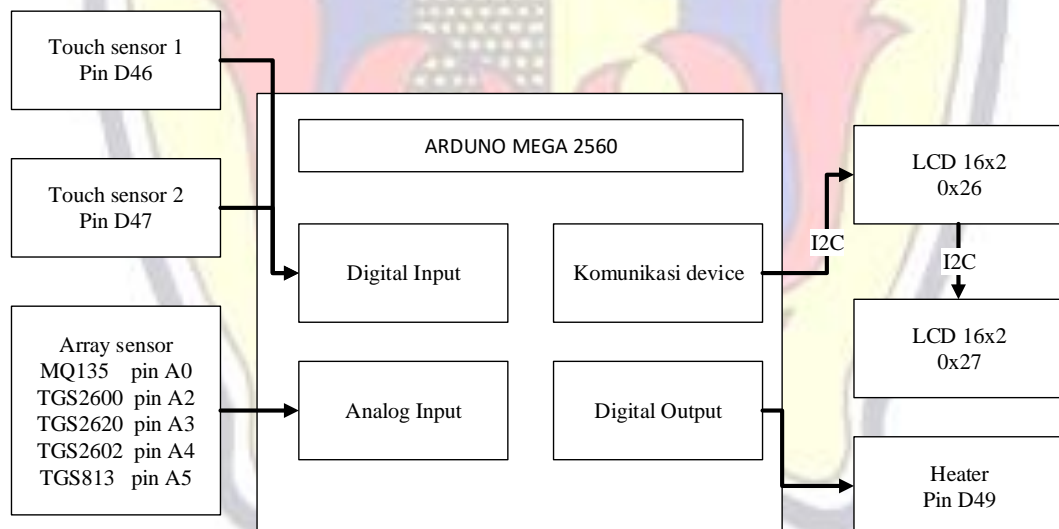
3.9. Diagram blok E-NOSE portable

3.9.1. Diagram blok aliran daya E-NOSE PORTABLE



Gambar 3. 9 diagram blok aliran daya E-NOSE portable

3.9.2. Diagram blok sistem komunikasi dan kontrol pada E-NOSE PORTABLE

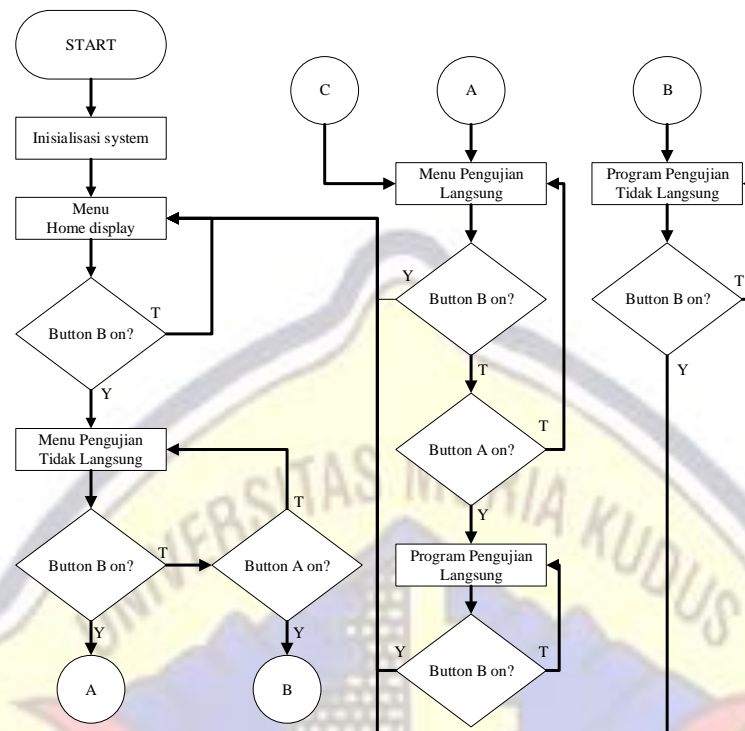


Gambar 3. 10 Diagram blok sistem komunikasi dan kontrol E-NOSE Portable

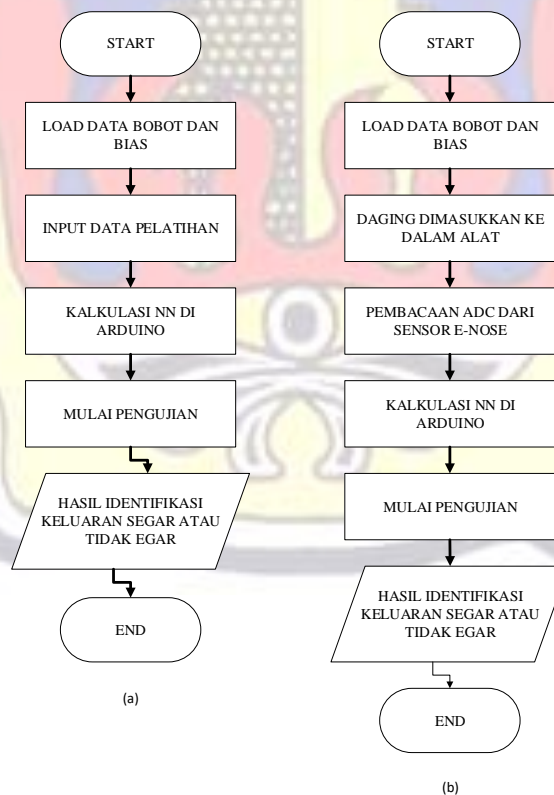
3.10. Flowchart sistem pendeteksian

Pada flowchart ini akan dibedakan antara pendeteksian Pendeteksian Tidak Langsung dan Pendeteksian Langsung. Dimana yang dimaksud Pendeteksian Tidak Langsung adalah pengujian dari data hasil simulasi. Sedangkan yang dimaksud

Pendeteksian Langsung adalah pendeteksian menggunakan sample daging secara real. Adapun flowchart pendeteksian seperti dibawah



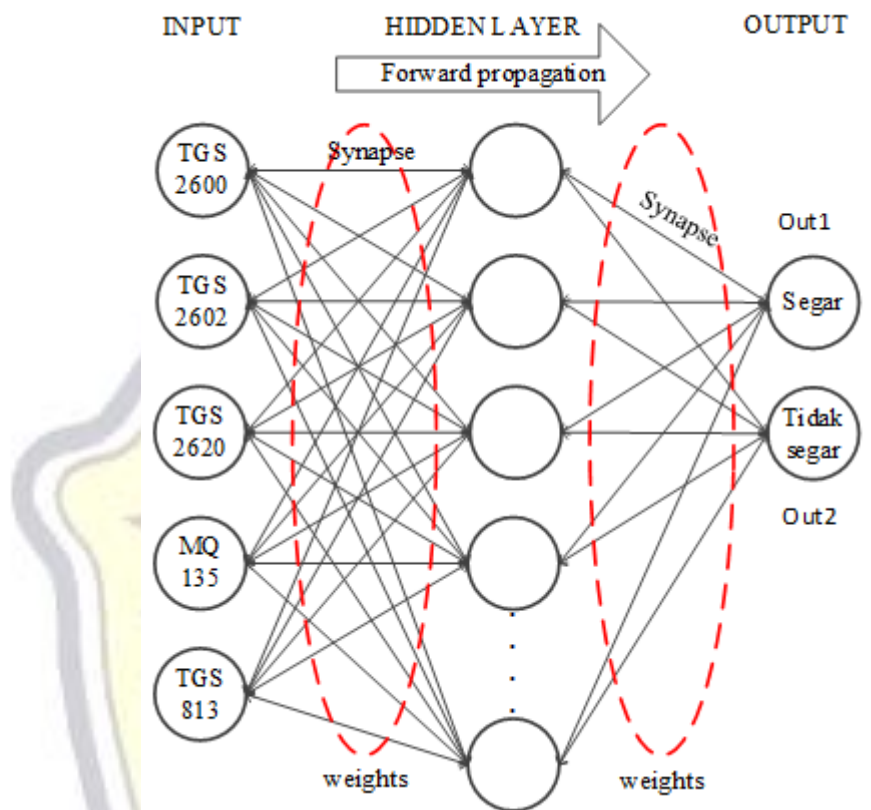
Gambar 3. 11 flowchart alur menu ada E-NOSE portable



Gambar 3. 12 flowchart pendeteksian a. Pendeteksian Tidak Langsung b.
Pendeteksian Langsung

3.11. Struktur *Neural Network* yang digunakan

Struktur *Neural Network* yang dirancang dengan formasi 5 – 8 – 2 yaitu 5 input 8 hidden layer dan 2 output



Gambar 3. 13 Struktur multilayer perceptron dari NN yang dirancang

3.12. Pengujian alat

3.12.1. Kategori pengujian

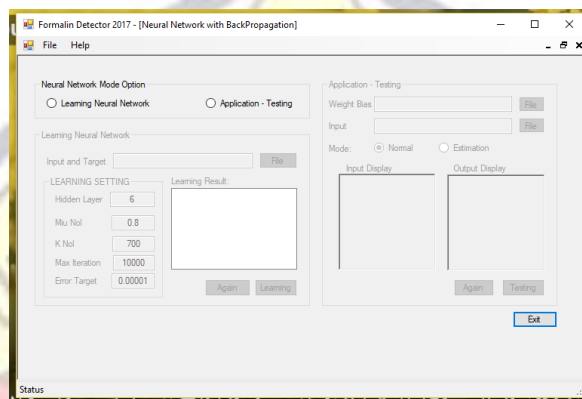
Kategori pengujian dimaksudkan untuk melihat presentase dari keberhasilan alat, pengujian ini dibagi menjadi 2;

1. Pengujian tidak langsung
2. Pengujian langsung

Pengujian tidak langsung adalah menguji nilai bobot dan bias yang dihasilkan dari simulasi software. Dan pengujian langsung adalah pengujian yang menggunakan sample gas daging yang dimasukkan kedalam alat.

3.12.2. Software yang digunakan

Software yang digunakan menggunakan 2 software yaitu Matlab dan Visual Basic, pada program matlab menggunakan tool pada matlab yaitu *nntool*, dan program dari visual basic menggunakan software yang telah dibuat oleh Arief Sudarmadji, ST., MT., Ph.D. selaku dosen teknologi pangan UNSOED. berikut *screenshoot* dari software tersebut.



Gambar 3. 14 software visual basic yang digunakan dalam penelitian ini

Pada software visual basic tersebut bertujuan untuk mendapatkan bobot dan bias yang akan dimasukkan kedalam alat E-NOSE portable.

3.12.3. Prosedur pengujian

1. Pengujian tidak langsung

Prosedur pengujian tidak langsung sebagai berikut;

- Menyalakan alat E-NOSE
- Pilih menu uji tidak langsung
- 22 data uji akan tampil pada LCD 1602 bagian atas, dan hasil dari kalkulasi akan tampil pada LCD 1602 bagian bawah

2. Pengujian langsung

Prosedur pengujian langsung sebagai berikut;

- Menyalakan alat E-NOSE.
- Lakukan pembersihan pada chamber dengan menekan tombol merah yang berada disebelah tombol power selama 1 menit.
- Pilih menu uji langsung.
- Tunggu hingga suhu pada chamber hingga 35°C.

- e. Tunggu hingga pengambilan Ro selesai dan pada LCD tertulis suntikkan gas kedalam chamber.
- f. Masukkan gas dari daging menggunakan suntik berukuran 10ml dengan tigas kali penyuntikan gas jika sudah tekan sensor sentuh sebelah kanan untuk melanjutkan ke proses selanjutnya.
- g. Tunggu hingga proses pengambilan Rg selesai.
- h. Dan hasil dari gas yang dimasukkan akan keluar sebagai daging segar atau daging tidak segar

3.12.4. Perencanaan pengujian secara Pendeteksian Tidak Langsung

Pada pengujian Pendeteksian Tidak Langsung akan dibagi menjadi 2 bagian yaitu menguji simulasi dari Matlab dan dari VB. Pada pengujian ini akan diuji nilai bobot dan bias yang merupakan output dari simulasi dari matlab dari data sensitifitas sensor mos yang telah diuji menggunakan akuisisi data. Pada pengujian Pendeteksian Tidak Langsung ini akan diuji nilai bobot dan bias dari 2 kondisi daging yaitu segar dan tidak segar

Prosedur pada pengujian Pendeteksian Tidak Langsung menggunakan matlab;

- a. Memasukkan nilai sensitifitas sensor ke masing masing program (Matlab dan VB) sebagai data training
- b. Dari simulasi data training kedua pogram (Matlab dan VB) menghasilkan nilai bobot dan bias
- c. Niai bobot dan bias dari masing masing program ini akan dimasukkan ke arduino di alat E-NOSE portable
- d. Akan dilihat hasil identifikasi alat utuk masing masing daging
- e. Pengujian akan dilakukan pengulangan sebanyak 3 kali pengujian.

3.12.5. Perencanaan pengujian secara Pendeteksian Langsung

Pada pengujian Pendeteksian Langsung ini akan diujikan sample daging secara real kedalam alat E-NOSE portable dengan caa memasukkan gas kedalam chamber. Nilai bobot dan bias di program E-NOSE ini diambil nilai bobot dan bias dari hasil simulasi 2 program yaitu matlab dan VB. Prosedur pada pengujian Pendeteksian Langsung;

- a. Memasukkan nilai bobot dan bias dari hasil simulasi kedua progam yaitu Matlab dan VB kedalam arduino dalam alat E-NOSE.

- b. Menguji sample daging secara real kedalam alat E-NOSE portable.
- c. Akan dilihat hasil identifikasi secara real di alat E-NOSE untuk masing masing daging.
- d. Pada saat pengujian daging yang lain akan ada prosedur pembersihan daging dengan cara menghidupkan fan yang berada didalam alat E-NOSE.

Pengujian akan dilakukan pengulangan sebanyak 3 kali pengujian.

3.13. Pengujian identifikasi berdasarkan waktu penyimpanan sample

daging akan diuji setiap 2 jam sekali mulai jam ke 0 hingga jam ke 12.

3.14. Pengambilan data training

Data training yang digunakan merupakan nilai sensitivitas dari masing masing sensor yang diambil dengan alat akuisisi data dari pembacaan sample saat terkena gas dari daging segar maupun daging tidak segar dengan masing masing pengambilan data selama 2 menit dengan jarak pengambilan data setiap 2.5 detik.

3.15. Analisa data

Pada analisa data ini akan dilihat presentase keberhasilan pendeteksian alat baik Pendeteksian Tidak Langsung dan Pendeteksian Langsung dari masing masing program

BAB IV HASIL DAN PEMBAHASAN

4.1. Alat akuisisi data

Alat akuisisi data adalah alat untuk mendapatkan hasil data dari sebuah pengujian yang dilakukan beberapa kali untuk mendapatkan hasil data yang baik. Untuk mendapatkan nilai dari alat akuisisi data dapat digunakan alat yang telah ada atau alat yang telah digunakan sebelumnya untuk dijadikan alat. Pembacaan nilai resistansi sensor dalam pendeteksian daging dibaca melalui alat akuisisi data yang terdiri dari ; sensor, pengkondisi sinyal, ADC, dan komputer yang merupakan alat E-NOSE yang tidak portable.

4.1.1. Gambar alat akuisisi data



Gambar 4. 1 Spesifikasi alat akuisisi data

Keterangan :

1. Laptop / PC untuk mengambil data dari sensor ke laptop
2. Chamber berisi deret sensor untuk pendeteksian gas daging
3. Kipas pembersih chamber
4. Kontroller / arduino sebagai ADC konverter dan komunikasi ke laptop
5. Pembagi tegangan untuk sumber tenaga sensor dan arduino
6. Vacuum untuk pembersihan chamber
7. Pemberian udara bersih ke chamber
8. Silica gel untuk membersihkan udara yang akan masuk ke chamber
9. Kabel USB penghubung alat ke laptop
10. Power supply untuk supply alat

4.1.2. Hasil pembacaan sensitifitas sensor MOS

Sensitifitas sensor dapat dihasilkan dari pengukuran nilai ADC yang rubah menjadi nilai hambatan dan dikalkulasi dengan rumus sensitivitas. Sensitifitas sensor dihitung dengan rumus

$$S=R_g/R_o \quad (4.1)$$

Dimana R_o adalah resistansi awal sensor sebelum terpapar gas dari daging, dan R_g adalah resistansi setelah terpapar gas dari daging sapi. Dari pengujian didapatkan hasil seperti tabel dibawah.

Tabel 4. 1 Tabel hasil sensitifitas sensor MOS terhadap kondisi daging

Ulangan	TGS2600	TGS2602	TGS2620	MQ135	TGS813	Logika training daging	
1	1.077	1.129	1.082	1.029	1.112	1	0
2	0.988	1.124	1.005	0.97	1.012	1	0
3	0.961	1.067	0.976	0.99	0.964	1	0
4	0.997	1.119	1.009	1.025	1.021	1	0
5	0.989	1.092	0.995	0.96	1.005	1	0
6	0.992	1.085	0.995	0.999	1.001	1	0
7	0.997	1.086	0.996	1.013	1.025	1	0
8	1.001	1.088	1.004	1.017	1.022	1	0
9	1.001	1.089	1.013	0.991	0.998	1	0
10	1.006	1.034	1.025	1.01	0.999	1	0
11	1.012	1.068	1.015	1.008	1.012	1	0
12	1.011	1.139	1.02	1	1.024	0	1
13	1.108	1.114	1.12	0.999	1.117	0	1
14	1.374	0.996	1.266	1.228	1.417	0	1
15	2.144	0.983	1.944	1.762	2.518	0	1
16	2.279	1.298	2.294	2.24	2.639	0	1
17	3.589	1.109	3.419	2.922	5.091	0	1
18	4.198	1.152	4.102	3.336	5.625	0	1
19	3.031	1.161	2.978	2.597	3.788	0	1
20	3.477	1.074	3.428	2.857	4.305	0	1
21	2.259	1.51	2.207	2.135	2.789	0	1
22	3.939	1.138	3.894	3.109	5.144	0	1

4.2. Tahap kalkulasi NN *feed forward* pada E-NOSE

Langkah-langkah dalam alur maju (*forward*) adalah:

- a. normalisasi input dan nilai desire output (menjadi dalam range 0 – 1).
- b. memberi nilai weight secara acak/random pada nilai -1 s/d +1
- c. memberi inisialisasi nilai bias = 1
- d. mencari nilai sum dan sigmoid untuk Hidden layer dan Ouput layer

1. Kalkulasi adjusment pada perhitungan

$$X = S_i / \max_i \quad (4.1)$$

2. Hidden layer

Nilai sum :

$$Z_j = \sum_{i=0}^N X_i \cdot V_{ij} \quad (4.2)$$

Nilai sigmoid :

$$Z'_j = \frac{1}{1 + e^{-z_j + \text{bias}}} \quad (4.3)$$

Konversi menjadi code arduino ;

```
for( j = 0 ; j < HiddenNodes ; j++ ) {  
    sum2=0;  
    for( i = 0 ; i < HiddenNodes ; i++ ) { //nilai sum  
        sum2 += Masukan[i] * BobotInHid[i][j] ;  
        delay(10);  
    }  
    Hidden[j] = 1.0/(1.0 + exp(-(sum2+Bias[j]))); //nilai  
    sigmoid  
}
```

Dimana *InputNodes* dan *HiddenNodes* diambil dari bobot hasil kalkulasi dari software Matlab maupun VB berupa data array dengan array 5x8, 5 adalah dari jumlah input sensor dan 8 adalah jumlah hidden layer.

3. Output layer

Nilai sum :

$$Y_k = \sum_{i=0}^M Z'_j \cdot W_{jk} \quad (4.4)$$

Nilai sigmoid :

$$Y'_k = \frac{1}{1 + e^{-y_k + \text{bias}}} \quad (4.5)$$

Konversi menjadi code arduino ;

```
for( k = 0 ; k < OutputNodes ; k++ ) {  
    sum2=0;  
    for( j = 0 ; j < HiddenNodes ; j++ ) { //nilai sum  
        sum2 += Hidden[j] * BobotHidOut[j][k] ;  
        delay(10);  
    }  
}
```



```

    }
    Keluaran[k] = 1.0/(1.0 + exp(-(sum2 + Bias[k+8]))); //nilai
sigmoid
}

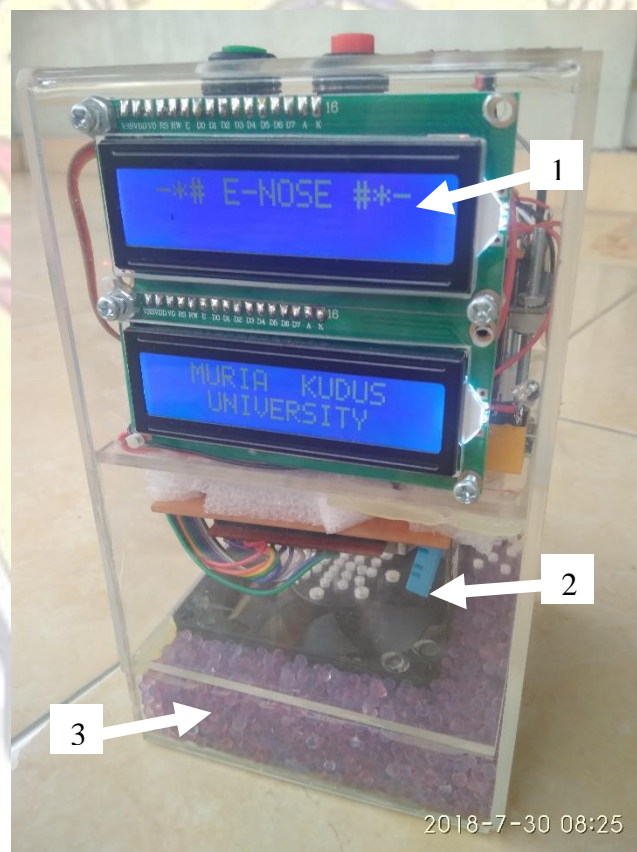
```

Dimana output layer merupakan rumus untuk kalkulasi dari hidden layer ke output layer yang berupa array 8x2 dimana 8 adalah dari hidden layer dan 2 adalah output layer.

4.3. Alat E-NOSE portable

Alat ini merupakan pengembangan dari alat akuisisi data yang telah digunakan, alat ini telah menggabungkan semua menjadi satu alat dan memperkecil ukurannya sehingga mudah dibawa dan mudah penggunaanya dengan fungsi yang sama yaitu dapat mendeteksi tingkat kesegaran daging.

4.3.1. Gambar dan spesifikasi alat E-NOSE portable



Gambar 4. 2 spesifikasi alat E-NOSE bagian 1

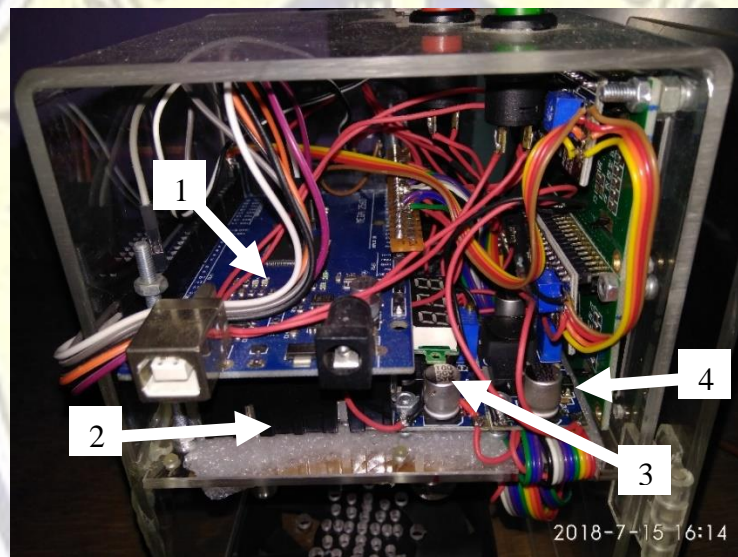
Keterangan :

1. Ruang kontrol
2. Chamber pengujian gas daging
3. Ruang pembersihan

Pada gambar 4.2 Ruang kontrol pada alat *E-NOSE* portable ini berupa arduino, I2C, LCD1602, battre 18650, relay, charger battre, stepup DC to DC, tombol dan sensor sentuh. Semua perangkat ini digunakan untuk mengendalikan dan memproses dari sistem kerja dari *E-NOSE* portable.

Chamber pengujian gas adalah sebuah ruang untuk tempat untuk pengujian berlangsung dimana diruang ini akan dimasukannya gas dari daging, ruang ini terdapat deret sensor yang berfungsi untuk mendeteksi gas dari daging yang disuntikkan kedalam ruang ini.

Ruang embersihan adalah ruang untuk membersihkan ruang chamber dan membersihkan dari gas yang akan masuk pada ruang chamber, dimana silca gel berfungsi untuk membersihkan udara dari lingkungan sekitar dengan fan 9cm untuk membersihkan sekaligus memberikan gas yang bersih kedalam ruang chamber.



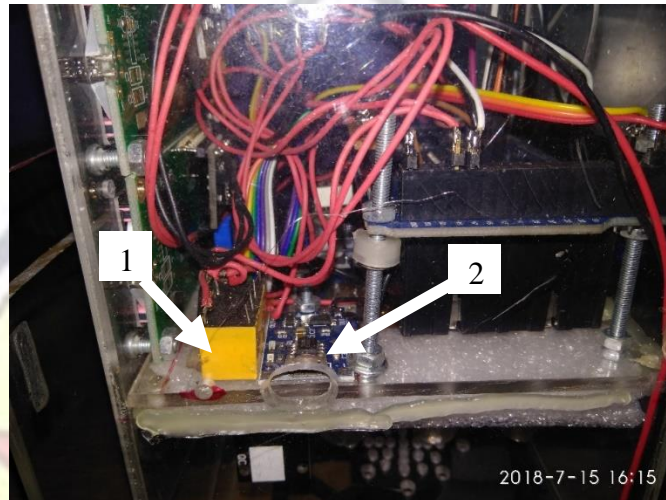
Gambar 4. 3 Spesifikasi alat E-NOSE bagian 2

Keterangan :

1. Arduino
2. Battre 18650
3. Steup up 3.7 to 5v DC
4. Step up 3.7 to 12v DC

Pada ganbar 4.3 Arduino mega 2560 ini digunakan untuk memproses semua pekerjaan yang dibutuhkan pada E-NOSE portable yaitu *UI*, kalkulasi untuk *neural network feed forward* dan kontrol heater pada chamber. Battre 18650 ini adalah sumber daya untuk mensupply dari alat *E-NOSE* portable ini, dengan adanya stepup

DC to DC ini akan menaikkan tegangan dari 3.7 ke 5v DC yang digunakan untuk memanaskan heater yang dikendalikan dengan relay untuk mematikan dan menyalakan heater, dan stepup 3.7 ke 12v DC ini digunakan untuk menghidupkan arduino mega 2560 yang dimasukkan kedalam regulator sehingga tegangan yang ada didalam arduino mega adalah 5vDC yang dihubungkan ke device I2C, LCD1602, sensor sentuh, dan ADC untuk sensor pada chamber.

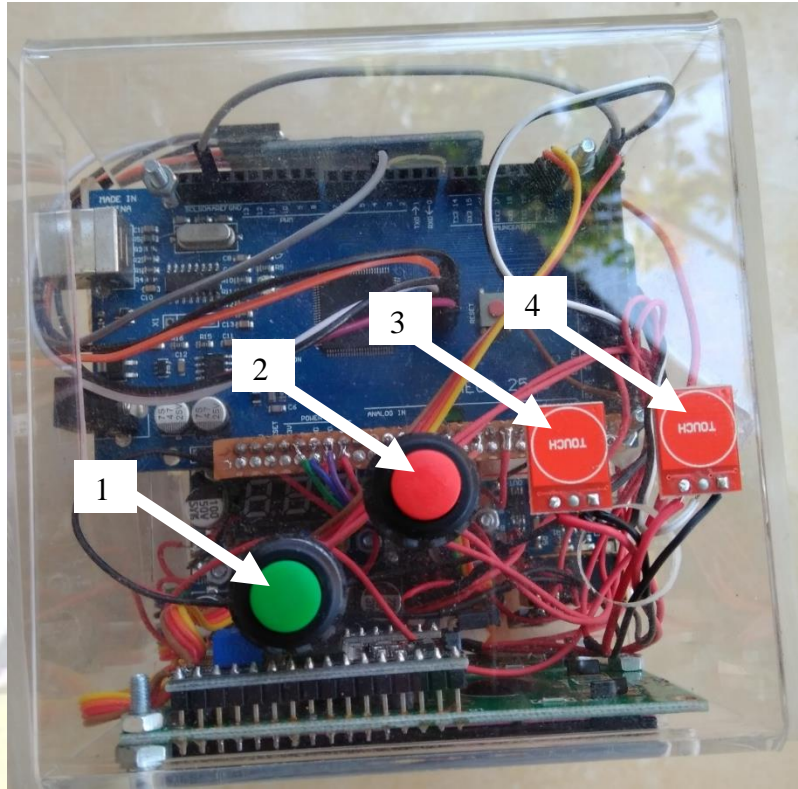


Gambar 4. 4 spesifikasi alat E-NOSE bagian 3

Keterangan :

1. Relay
2. Charger battre 1 cell

Pada gambar 4.4 Relay yang digunakan adalah relay 5v dengan menggunakan transistor BD140 untuk penguat hingga dapat menyalakan dan mematikan relay 5v, relay ini berfungsi untuk menyalakan dan mematikan heater pada chamber. Charger battre 1 cell ini berfungsi untuk mengisi dari battre 18650 ketika battre habis atau akan habis, jadi tidak melepas atau membongkar ketika mengisi dari battre tersebut, pada charger ini dihubungkan dengan battre 18650 yang dihubungkan paralel sehingga charger 1 cell tersebut dapat mengisi pada kedua battre tersebut.



Gambar 4. 5 spesifikasi alat E-NOSE bagian 4

Keterangan :

1. Tombol ON/OFF alat E-NOSE
2. Tombol ON/OFF kipas
3. Tombol A
4. Tombol B

Pada gambar 4.5 Terdapat 2 buah tombol dan 2 buah sensor sentuh yang memiliki fungsi yang berbeda beda, tombol hijau yang ditunjukkan nomor 1 ini memiliki fungsi untuk menyalakan dan mematikan seluruh sistem dari alat E-NOSE portable, tombol merah yang ditunjukkan pada nomor 2 memiliki fungsi untuk menyalakan dan mematikan fungsi dari pembersihan ruang chamber dan sekaligus memberikan udara bersih didalam chamber.

Sensor sentuh B yang ditunjukkan nomor 4 ini memiliki fungsi untuk memasuki menu UI pada E-NOSE portable ini, menu yang didalamnya berupa menu untuk pengujian Pendeteksian Tidak Langsung dan menu untuk pengujian Pendeteksian Langsung, tombol A yang ditunjukkan pada nomor 3 digunakan untuk memulai dari menu yang dipilih yang berupa tombol YES pada menu.

4.4. Pengujian Pendeteksian Tidak Langsung

Pengujian Pendeteksian Tidak Langsung adalah pengujian alat E-NOSE portable menggunakan data latih sebanyak 22 data, setelah melakukan pengujian ini akan terlihat untuk besar kecilnya pengenalan daging dengan menggunakan 2 buah software.

4.4.1. Uji Pendeteksian Tidak Langsung menggunakan matlab

Data simulasi dengan 22 data uji terdiri dari 11 data segar dan 11 data tidak segar dan diolah dari Matlab untuk menghasilkan bobot dan bias, bobot dan bias tersebut akan dimasukkan ke program arduino dengan kalkulasi *Neural Network feed forward*. Berikut Hasil menggunakan matlab

Tabel 4. 2 hasil data Pendeteksian Tidak Langsung menggunakan Matlab

pengujian ke	Sampel daging	Nilai sensitifitas 5 sensor	Output	Target	keberhasilan pengenalan
1	Daging segar	(0.9950) (1.0810) (1.0030) (1.0010) (1.0050)	0.5018 0.0808	1 0	Benar
2	Daging segar	(0.9900) (1.1200) (1.0100) (0.9700) (1.0100)	0.5020 0.0811	1 0	Benar
3	Daging segar	(0.9610) (1.0670) (0.9760) (0.9900) (0.9640)	0.5018 0.0808	1 0	Benar
4	Daging segar	(0.9970) (1.1190) (1.0090) (1.0250) (1.0210)	0.5020 0.0810	1 0	Benar
5	Daging segar	(0.9890) (1.0920) (0.9950) (0.9600) (1.0050)	0.5019 0.0809	1 0	Benar
6	Daging segar	(0.9920) (1.0850) (0.9950) (0.9990) (1.0010)	0.5018 0.0809	1 0	Benar
7	Daging segar	(0.9970) (1.0860) (0.9960) (1.0130) (1.0250)	0.5018 0.0808	1 0	Benar
8	Daging segar	(1.0010) (1.0880) (1.0040) (1.0170) (1.0220)	0.5018 0.0809	1 0	Benar
9	Daging segar	(1.0010) (1.0890) (1.0130) (0.9910) (0.9980)	0.5019 0.0809	1 0	Benar

Pengujian Ke	Sampel Daging	Nilai Sensitifitas 5 Sensor	Output	Target	Keberhasilan Pengenalan
10	Daging segar	(1.0060) (1.0340) (1.0250) (1.0100) (0.9990)	0.5016 0.0806	1 0	Benar
11	Daging segar	(1.0120) (1.0680) (1.0150) (1.0080) (1.0120)	0.5018 0.0808	1 0	Benar
12	Daging tidak segar	(2.9200) (1.1500) (2.8300) (2.4600) (3.7000)	0.5014 0.0806	0 1	Salah
13	Daging tidak segar	(1.1080) (1.1140) (1.1200) (0.9990) (1.1170)	0.5020 0.0811	0 1	Salah
14	Daging tidak segar	(1.3740) (0.9960) (1.2660) (1.2280) (1.4170)	0.5014 0.0804	0 1	Salah
15	Daging tidak segar	(2.1430) (0.9000) (1.9400) (1.7600) (2.5100)	0.5008 0.0797	0 1	Salah
16	Daging tidak segar	(2.2790) (1.2980) (2.2940) (2.2400) (2.6390)	0.5024 0.0817	0 1	Salah
17	Daging tidak segar	(3.5890) (1.1090) (3.4190) (2.9220) (5.0910)	0.5007 0.0798	0 1	Salah
18	Daging tidak segar	(4.1980) (1.1520) (4.1020) (3.3360) (5.6250)	0.5010 0.0801	0 1	Salah
19	Daging tidak segar	(3.0310) (1.1610) (2.9780) (2.5970) (3.7880)	0.5015 0.0807	0 1	Salah
20	Daging tidak segar	(3.4800) (1.0700) (3.4300) (2.8600) (4.3100)	0.5011 0.0802	0 1	Salah
21	Daging tidak segar	(2.2600) (1.5100) (2.2070) (2.1350) (2.7890)	0.5034 0.0829	0 1	Salah
22	Daging tidak segar	(3.9300) (1.1400) (3.9000) (3.1100) (5.1400)	0.5011 0.0803	0 1	Salah

4.4.2. Uji Pendeteksian Tidak Langsung menggunakan VB

Data simulasi dengan 22 data uji terdiri dari 11 data segar dan 11 data tidak segar dan diolah dengan software hasil dari Visual Basic untuk menghasilkan bobot dan bias, bobot dan bias tersebut akan dimasukkan ke program arduino dengan

kalkulasi *Neural Network feed forward*. Berikut Hasil dari pengujian Pendeteksian Tidak Langsung menggunakan Visual Basic

Tabel 4. 3 hasil data Pendeteksian Tidak Langsung menggunakan Visual Basic

Pengujian ke	Sampel daging	Nilai sensitifitas 5 sensor	Hasil kalkulasi	Target	Keberhasilan pengenalan
1	Daging segar	(0.9950) (1.0810) (1.0030) (1.0010) (1.0050)	1.0000 0.0000	1 0	Benar
2	Daging segar	(0.9900) (1.1200) (1.0100) (0.9700) (1.0100)	0.9999 0.0001	1 0	Benar
3	Daging segar	(0.9610) (1.0670) (0.9760) (0.9900) (0.9640)	1.0000 0.0000	1 0	Benar
4	Daging segar	(0.9970) (1.1190) (1.0090) (1.0250) (1.0210)	0.9998 0.0002	1 0	Benar
5	Daging segar	(0.9890) (1.0920) (0.9950) (0.9600) (1.0050)	1.0000 0.0000	1 0	Benar
6	Daging segar	(0.9920) (1.0850) (0.9950) (0.9990) (1.0010)	1.0000 0.0000	1 0	Benar
7	Daging segar	(0.9970) (1.0860) (0.9960) (1.0130) (1.0250)	0.9999 0.0001	1 0	Benar
8	Daging segar	(1.0010) (1.0880) (1.0040) (1.0170) (1.0220)	0.9999 0.0001	1 0	Benar
9	Daging segar	(1.0010) (1.0890) (1.0130) (0.9910) (0.9980)	0.9999 0.0001	1 0	Benar
10	Daging segar	(1.0060) (1.0340) (1.0250) (1.0100) (0.9990)	1.0000 0.0000	1 0	Benar
11	Daging segar	(1.0120) (1.0680) (1.0150) (1.0080) (1.0120)	0.9999 0.0001	1 0	Benar
12	Daging tidak segar	(2.9200) (1.1500) (2.8300) (2.4600) (3.7000)	0.0000 1.0000	0 1	Benar
13	Daging tidak segar	(1.1080) (1.1140) (1.1200) (0.9990) (1.1170)	0.2444 0.7605	0 1	Benar

Pengujian Ke	Sampel Daging	Nilai Sensitifitas 5 Sensor	Output	Target	Keberhasilan Pengenalan
14	Daging segar	(1.3740) (0.9960)	0.0000	0	Benar
	tidak segar	(1.2660) (1.2280)	1.0000	1	
		(1.4170)			
15	Daging segar	(2.1430) (0.9000)	0.0000	0	Benar
	tidak segar	(1.9400) (1.7600)	1.0000	1	
		(2.5100)			
16	Daging segar	(2.2790) (1.2980)	0.0000	0	Benar
	tidak segar	(2.2940) (2.2400)	1.0000	1	
		(2.6390)			
17	Daging segar	(3.5890) (1.1090)	0.0000	0	Benar
	tidak segar	(3.4190) (2.9220)	1.0000	1	
		(5.0910)			
18	Daging segar	(4.1980) (1.1520)	0.0000	0	Benar
	tidak segar	(4.1020) (3.3360)	1.0000	1	
		(5.6250)			
19	Daging segar	(3.0310) (1.1610)	0.0000	0	Benar
	tidak segar	(2.9780) (2.5970)	1.0000	1	
		(3.7880)			
20	Daging segar	(3.4800) (1.0700)	0.0000	0	Benar
	tidak segar	(3.4300) (2.8600)	1.0000	1	
		(4.3100)			
21	Daging segar	(2.2600) (1.5100)	0.0000	0	Benar
	tidak segar	(2.2070) (2.1350)	1.0000	1	
		(2.7890)			
22	Daging segar	(3.9300) (1.1400)	0.0000	0	Benar
	tidak segar	(3.9000) (3.1100)	1.0000	1	
		(5.1400)			

4.5. Pengujian Pendeteksian Langsung

Pengujian Pendeteksian Langsung ini adalah pengujian menggunakan daging asli dengan cara meinjeksikan gas dari daging atau di suntikkan gas dari daging ke dalam chamber alat E-NOSE portable.

4.5.1. Pengujian real menggunakan Matlab

Pengujian real ini dimaksudkan untuk pengujian menggunakan daging asli menggunakan gas atau bau dari daging tersebut dimasukkan ke alat E-NOSE portable menggunakan suntik sehingga gas tersebut berada didalam ruang chamber dan mendapatkan hasil yang tampil pada LCD segar atau tidak segar, dengan pengujian daging sebanyak tiga kali pengujian daging segar dan tiga kali dengan daging tidak segar dan ditampilkan ke tabel berikut:

Tabel 4. 4 hasil data Pendeteksian Langsung menggunakan Matlab

Pengujian ke	Sampel	Hasil pembacaan LCD	Hasil kalkulasi NN	Keberhasilan pengenalan
1	Daging Segar	Daging Segar	0.501475763 0.080427094	Benar
2	Daging Segar	Daging Segar	0.50151968 0.08048192	Benar
3	Daging Segar	Daging Segar	0.501683664 0.080682945	Benar
4	Daging Tidak Segar	Daging Segar	0.502618265 0.081837654	Salah
5	Daging Tidak Segar	Daging Segar	0.502603483 0.081817455	Salah
6	Daging Tidak Segar	Daging Segar	0.502673817 0.081906796	Salah

4.5.2. Pengujian real menggunakan Visual Basic

Pengujian real ini dimaksudkan untuk pengujian menggunakan daging asli menggunakan gas atau bau dari daging tersebut dimasukkan ke alat E-NOSE portable menggunakan suntik sehingga gas tersebut berada didalam ruang chamber dan mendapatkan hasil yang tampil pada LCD segar atau tidak segar, dengan pengujian daging sebanyak tiga kali pengujian daging segar dan tiga kali dengan daging tidak segar dan ditampilkan ke tabel berikut:

Tabel 4. 5 hasil data Pendeteksian Langsung menggunakan Visual Basic

Pengujian ke	Sampel	Hasil pembacaan LCD	Hasil kalkulasi NN	Keberhasilan pengenalan
1	Daging segar	Daging segar	1.0000 0.0000	Benar
2	Daging segar	Daging segar	0.9999 0.0000	Benar
3	Daging segar	Daging segar	1.0000 0.0000	Benar
4	Daging tidak segar	Daging tidak segar	0.0000 1.0000	Benar
5	Daging tidak segar	Daging tidak segar	0.0000 1.0000	Benar
6	Daging tidak segar	Daging tidak segar	0.0000 1.0000	Benar

4.5.3. Hasil pengujian daging dengan jarak waktu 2 jam

Pada pengujian ini dimaksudkan untuk melihat sampai jam ke berapa daging dinyatakan tidak segar oleh alat E-NOSE portable ini, pada pengujian ini menggunakan jarak waktu 2 jam sekali untuk pengambilan data daging.

Tabel 4. 6 tabel pengujian daging dengan jarak waktu setiap 2 jam sekali

No	Waktu	Hasil Pendeteksian	Hasil Kalkulasi Alat	
1	0 Jam	Daging Segar	1.0000	0.0000
2	2 Jam	Daging Segar	1.0000	0.0000
3	4 Jam	Daging Segar	1.0000	0.0000
4	6 Jam	Daging Segar	0.9999	0.0000
5	8 Jam	Daging Segar	0.7179	0.2864
6	10 Jam	Daging Tidak Segar	0.0000	1.0000
7	12 Jam	Daging Tidak Segar	0.0000	1.0000
8	14 Jam	Daging Tidak Segar	0.0000	1.0000

4.6. Analisa hasil pengujian

Hasil data dari Pendeteksian Tidak Langsung menggunakan data bobot dan bias matlab menunjukkan tingkat keberhasilan 50%, dengan 100% mengenali kesegaran segar dan 0% tingkat kesegaran tidak segar, dan pengujian Pendeteksian Langsung dengan hasil 100% dapat mengenali tingkat kesegaran segar, dan 0% pengenalan kesegaran tidak segar. Hasil data dari pengujian Pendeteksian Tidak Langsung data dengan bobot dan bias dari VB menunjukkan tingkat keberhasilan pengenalan sebesar 100% untuk segar dan 100% tidak segar, dan 3 kali pengujian secara Pendeteksian Langsung mendapatkan hasil 100% pengenalan secara segar dan tidak segar.

Pada hasil pada tabel 4.6 pada jam ke 0 sampai jam ke 6 alat E-NOSE portable masih mendeteksi dengan daging segar dengan hasil mendekati 1,0. Pada jam ke 8 hasil pendeteksian dari alat E-NOSE masih menampilkan hasil daging segar dengan hasil kalkulasi NN 0.71 dan 0.28 dengan hasil ini masih dinyatakan segar karena pembulatan dari 0.71 adalah 1 dan pembulatan untuk 0.28 adalah 0 karena itulah pada output masih dinyatakan dengan output 1,0 atau dinyatakan dengan daging segar pada LCD E-NOSE portable. Pada hasil jam ke 12 menunjukkan hasil dengan daging tidak segar, dengan hasil kalkulasi 0.00353 dan 0.996 dengan pembulatan hasil adalah 0,1 dengan demikian pada LCD tertampil daging tidak segar.

BAB V PENUTUP

5.1. Kesimpulan

Berdasarkan hasil dari penelitian yang dilakukan dapat ditarik beberapa kesimpulan

1. Telah dibuatnya E-NOSE portable yang berfungsi untuk membedakan kategori tingkat kesegaran daging dengan pengenalan pola NN feed forward dengan menggunakan sensor 5 MOS dengan tingkat kesegaran daging sebesar 100% pengenalan kesegaran.
2. Pendeteksian secara tidak langsung mendapatkan hasil 100% dari 22 buah data meliputi 11 data daging segar dan 11 data daging tidak segar dan pendeteksian secara langsung mendapatkan hasil 100% keberhasilan dengan 6 gas dari daging meliputi tiga buah daging tidak segar dan tiga buah daging tidak segar.
3. Pendeteksian daging segar dari jam ke 0 hingga jam ke 6 masih dengan hasil pembulatan mendekati 1,0 dan mulai berkurang pada waktu jam ke 8 dengan hasil daging segar dan hasil dari kalkulasi 0.71 dan 0.28 dan pada jam ke 12 hasil dari LCD mendapatkan daging tidak segar.

5.2. Saran

1. Untuk monitoring hasil yang di koneksikan dengan internet sehingga dapat dilihat kesegaran daging tersebut melalui jarak jauh.
2. Proses pengenalan uji Pendeteksian Tidak Langsung maupun Pendeteksian Langsung menggunakan waktu sekitar 2 detik untuk kalkulasi Pendeteksian Tidak Langsung maupun Pendeteksian Langsung, penghitungan kalkulasi ini dapat ditingkatkan dengan mengupgrade mikrokontroler atau menyingkat list program dari kalkulasi *Neural Network feed forward*.
3. Pengujian Pendeteksian Langsung memakan waktu 2 menit untuk Ro dan Rg, disarankan untuk menyingkat waktu tersebut agar dapat mengenali dengan cepat.
4. Penggunaan 2 buah battre 18650 ini untuk melakukan pengujian Pendeteksian Langsung tidak cukup untuk pengujian hingga selesai disarankan untuk mengupgrade battre atau menggantinya dengan daya yang lebih besar.

DAFTAR PUSTAKA

- Adzhar, H., 2015. "Sistem Penyeteman Nada Dawai Gitar Otomatis Dengan Motor Servo Continuous Menggunakan Kontroler Pid Berbasis Arduino Mega 2560".
- Anonim, 2005 "TGS 2600 - for the detection of Air Contaminants," Figaro Engineering Inc., pp. 1–2.
- Anonim, 2002. "TGS 813 - for the detection of Combustible Gases," pp. 1–2. Available at: <http://www.figarosensor.com/products/813pdf.pdf>.
- Anonim, 2008. "TGS2602-for the detection of Air Contaminants," pp. 1–2. Available at: papers3://publication/uuid/90B09016-5527-4DE2-A8DC-E729AB474FD1.
- Anonim, 2014. "TGS 2620 - for the detection of Solvent Vapors." Available at: http://www.figaro.co.jp/en/product/docs/tgs2620_product_information%28en%29_rev02.pdf.
- Anonim, 2016. "Mq-135 Gas Sensor," 1, pp. 3–4.
- Hadi, R. W. and Setiawan, I., 2011. "Perancangan Alat Pendeteksi Kualitas Daging Sapi Berdasar Warna dan Bau Berbasis Mikrokontroler Atmega32 Menggunakan Logika Fuzzy," 13(1), pp. 21–26.
- Kiswanto, 2012. "Identifikasi Citra Untuk Mengidentifikasi Jenis Daging Sapi Dengan Menggunakan Transformasi Wavelet Haar," Universitas Diponegoro, 1(2), pp. 73–79. doi: 10.21456/vol1iss2pp73-79.
- M. Surya Abdul, 2014. "Robusta Dengan Electronic Nose Menggunakan Sistem," 2(2), pp. 73–78.
- Mas'ud and Supriyanto, C. (2015) "Implementasi Principal Component Analysis (PCA) dan Euclidean Distance untuk Identifikasi Citra Daging Sapi dan Daging Babi," Skripsi, Fakultas Ilmu Komputer.
- Meinanda, M. H., 2009. "Prediksi masa studi sarjana dengan artificial neural network," Internetworking Indonesia Journal, 1(2), pp. 31–35.
- Nurfawaidi, A. (2015) Pengembangan Label Pintar untuk Indikator Kesegaran Daging Sapi pada Kemasan.
- Puspitaningrum, D., 2014. "Pengantar Jaringan Syaraf Tiruan."
- Simamora, J., 2017. "Rancang bangun sistem pendeteksi kesegaran daging berdasarkan sensor bau dan warna."
- Sudarmaji, A. and Ediati, R., 2011. "Identifikasi Kematangan Buah Tropika Berbasis Sistem Penciuman Elektronik Menggunakan Deret Sensor Gas Semikonduktor Dengan Metode Jaringan Syaraf Tiruan," Jtep Jurnal Keteknikan Pertanian, 25(1), pp. 48–59. Available at: <http://journal.ipb.ac.id/index.php/jtep/article/view/7378>.

- Surobramantyo, J., 2016. "Rancang bangun alat pendeteksi kematangan buah durian menggunakan sensor tgs 2620 dan tgs 2600 berbasis arduino," (2016), pp. 5–42.
- Utami, E. et al., no date. "Identifikasi Penyakit Diabetes Melalui Bau Urine Dengan Sensor Gas Menggunakan Metoda Pembelajaran."
- Wardhani. Andri, Susilo.Bambang, Y. R., 2013. "Gas Amonia Berbahaya Pada Kandang Ayam Broiler," Jurnal Keteknikan Pertanian Tropis dan Biosistem, 3(1), pp. 86–94.



Lampiran 1 hasil kalkulasi dan sensitifitas sensor pada tampilan serial monitor

Hasil keluaran sensitifitas sensor dari serial monitor dari pengujian Pendeteksian Tidak Langsung menggunakan data *learning* matlab yang telah di rubah menjadi tabel seperti ini.

Tabel pengujian tidak langsung menggunakan gas daging.

No	Sensitifitas Sensor (Ro/Rg)					Output	
	MQ135	TGS2600	TGS2620	TGS2602	TGS813		
1	0.995	1.081	1.003	1.001	1.005	0.501818	0.080842
2	0.99	1.12	1.01	0.97	1.01	0.502002	0.081059
3	0.961	1.067	0.976	0.99	0.964	0.501762	0.080773
4	0.997	1.119	1.009	1.025	1.021	0.501968	0.081025
5	0.989	1.092	0.995	0.96	1.005	0.501874	0.080904
6	0.992	1.085	0.995	0.999	1.001	0.501833	0.080859
7	0.997	1.086	0.996	1.013	1.025	0.501815	0.080838
8	1.001	1.088	1.004	1.017	1.022	0.501832	0.080859
9	1.001	1.089	1.013	0.991	0.998	0.501873	0.080906
10	1.006	1.034	1.025	1.01	0.999	0.50164	0.080628
11	1.012	1.068	1.015	1.008	1.012	0.50177	0.080784
12	2.92	1.15	2.83	2.46	3.7	0.501428	0.080561
13	1.108	1.114	1.12	0.999	1.117	0.502005	0.081066
14	1.374	0.996	1.266	1.228	1.417	0.501393	0.080352
15	2.143	0.9	1.94	1.76	2.51	0.500784	0.079675
16	2.279	1.298	2.294	2.24	2.639	0.502417	0.081746
17	3.589	1.109	3.419	2.922	5.091	0.500719	0.07977
18	4.198	1.152	4.102	3.336	5.625	0.500953	0.080148
19	3.031	1.161	2.978	2.597	3.788	0.501492	0.080664
20	3.48	1.07	3.43	2.86	4.31	0.501107	0.080235
21	2.26	1.51	2.207	2.135	2.789	0.503359	0.082905
22	3.93	1.14	3.9	3.11	5.14	0.501125	0.08031

Tabel pengujian langsung menggunakan gas daging.

No	Sensitifitas Sensor (Ro/Rg)					Output	
	MQ135	TGS2600	TGS2620	TGS2602	TGS813		
1	1.021935	0.991427	1.019198	0.981402	1.003718	0.501476	0.080427
2	1.016344	1.004866	1.018457	1.001833	1.003873	0.50152	0.080482
3	1.034202	1.041314	1.065791	1.044263	1.017385	0.501684	0.080683
4	1.555247	1.217309	1.0373	1.220711	1.061114	0.502618	0.081838
5	1.466786	1.217484	1.063329	1.20786	1.057549	0.502603	0.081817
6	1.552692	1.230301	1.026542	1.227979	1.058272	0.502674	0.081907

Hasil keluaran dari serial monitor dari pengujian Pendeteksian Tidak Langsung menggunakan data *learning* VB yang telah dirubah menjadi tabel seperti dibawah ini

Tabel pengujian tidak langsung menggunakan gas daging.

No	Sensitifitas Sensor (Ro/Rg)					Output	
	MQ135	TGS2600	TGS2620	TGS2602	TGS813		
1	0.995	1.081	1.003	1.001	1.005	1.000	0.000
2	0.990	1.120	1.010	0.970	1.010	1.000	0.000
3	0.961	1.067	0.976	0.990	0.964	1.000	0.000
4	0.997	1.119	1.009	1.025	1.021	1.000	0.000
5	0.989	1.092	0.995	0.960	1.005	1.000	0.000
6	0.992	1.085	0.995	0.999	1.001	1.000	0.000
7	0.997	1.086	0.996	1.013	1.025	1.000	0.000
8	1.001	1.088	1.004	1.017	1.022	1.000	0.000
9	1.001	1.089	1.013	0.991	0.998	1.000	0.000
10	1.006	1.034	1.025	1.010	0.999	1.000	0.000
11	1.012	1.068	1.015	1.008	1.012	1.000	0.000
12	2.920	1.150	2.830	2.460	3.700	0.000	1.000
13	1.108	1.114	1.120	0.999	1.117	0.244	0.761
14	1.374	0.996	1.266	1.228	1.417	0.000	1.000
15	2.143	0.900	1.940	1.760	2.510	0.000	1.000
16	2.279	1.298	2.294	2.240	2.639	0.000	1.000
17	3.589	1.109	3.419	2.922	5.091	0.000	1.000
18	4.198	1.152	4.102	3.336	5.625	0.000	1.000
19	3.031	1.161	2.978	2.597	3.788	0.000	1.000
20	3.480	1.070	3.430	2.860	4.310	0.000	1.000
21	2.260	1.510	2.207	2.135	2.789	0.000	1.000
22	3.930	1.140	3.900	3.110	5.140	0.000	1.000

Tabel pengujian langsung menggunakan gas daging.

No	Sensitifitas Sensor (Ro/Rg)					Output	
	MQ135	TGS2600	TGS2620	TGS2602	TGS813		
1	1.0177	0.9835	0.9858	0.9848	0.9896	1.0000	0.0000
2	1.0242	1.0194	1.0288	1.0270	1.0101	0.9999	0.0000
3	1.1359	0.9385	0.9954	0.9261	0.9849	1.0000	0.0000
4	1.7536	1.2888	1.0376	1.2724	1.1025	0.0000	1.0000
5	2.2690	1.5033	1.0260	1.4504	1.1511	0.0000	1.0000
6	1.8789	1.3810	1.0251	1.3295	1.1096	0.0000	1.0000

Lampiran 2 hasil kalkulasi dan sensitifitas sensor dengan pengambilan gas 2 jam sekali

No	Waktu	Sensitifitas Sensor					Output	
		MQ135	TGS2600	TGS2620	TGS2602	TGS813		
1	0 Jam	0.9421	0.9907	1.0062	0.9879	0.9814	1.0000	0.0000
2	2 Jam	1.0101	0.9982	1.0027	0.9973	0.9988	1.0000	0.0000
3	4 Jam	1.0025	1.0039	1.0028	0.9988	1.0012	1.0000	0.0000
4	6 Jam	1.0242	1.0194	1.0288	1.0270	1.0101	0.9999	0.0000
5	8 Jam	1.1889	1.1541	1.0332	1.1497	1.0563	0.718	0.286
6	10 Jam	1.7536	1.2888	1.0376	1.2724	1.1025	0.0000	1.0000
7	12 Jam	2.1786	1.4308	1.0494	1.4245	1.1626	0.0000	1.0000
8	14 Jam	3.2859	1.8337	1.0679	1.7556	1.3145	0.0000	1.0000



Lampiran 3 bobot dan bias pada hasil pelatihan

1. Nilai bobot dan bias input - hidden dari matlab yang diambil untuk penelitian ini
Dari penelitian menggunakan data Matlab

Jumlah neuron	Bobot hidden				Bias hidden	
8	0.643	11.173	1.177	1.578	-0.501	-24.560
	-0.653	6.882	0.605	2.274	0.998	-13.571
	1.174	12.772	0.430	-1.363	0.012	-18.971
	-0.408	-13.634	0.788	1.493	-0.034	13.561
	-0.028	-3.265	-1.740	-1.833	0.972	8.687
	1.083	5.558	-0.871	2.552	-0.567	-9.304
	1.944	-5.518	-0.207	-2.024	-0.144	10.246
	-0.721	-6.935	-1.123	0.503	1.348	3.591

Nilai bobot dan bias dari hidden layer ke output layer

Bobot keluaran								Bias keluaran
-0.915	-0.622	0.173	-0.278	0.622	-0.832	0.303	-0.193	-0.463
0.946	0.334	0.35	0.241	-0.961	0.95	-0.538	-0.756	-0.484

2. Nilai bobot dan bias input ke hidden dari visual basic yang digunakan untuk penelitian ini




Jumlah neuron	Bobot hidden					Bias hidden	
8	30.70241	27.04392	28.58146	2.867154	28.57826	-21.489	
	-3.71385	-4.4445	-4.38188	-0.87791	-2.51963	2.772899	
	8.947571	6.228763	8.939346	1.399313	9.453681	-6.76709	
	-2.37759	-3.49154	-2.09213	0.270878	-2.09637	1.161926	
	-3.42602	-3.07484	-2.75698	-0.92072	-2.3943	1.690262	
	22.25978	20.07708	20.48993	1.73795	21.44807	-15.7676	
	-5.18146	-6.84166	-4.79421	0.633344	-4.12798	4.040079	
	-1.0132	-0.67861	-1.27032	0.022247	-1.40597	-1.39219	

Nilai bobot dan bias dari hidden layer ke output layer

Bobot keluaran								Bias keluaran
-15.828	2.901	-4.702	2.424	2.024	-11.757	4.446	1.703	10.970
16.319	-3.837	4.159	-2.963	-3.748	11.363	-4.159	-1.408	-9.764

Lampiran 4 foto kegiatan

Foto kegiatan penelitian

No	Gambar	Keterangan
1		Preparasi sampel daging sapi potong
2		Sampel daging sapi potong dari pasar siap uji
3		Hasil dari kalkulasi <i>Neural Network</i>

Lampiran 5 *code* program arduino

1. Program arduino *Neural Network* Pendeteksian Tidak Langsung dan Pendeteksian Langsung menggunakan data Matlab

```
/*
 * dht11      pin 48 -->
 * batt       pin A9 -->
 * LM35       pin A7
 * mq5        pin A6
 * tgs813     pin A5 -->
 * tgs2602    pin A4 -->
 * tgs2620    pin A3 -->
 * tgs2600    pin A2 -->
 * mq137      pin A1
 * mq135      pin A0 -->
 * MOSI - pin 11
 * MISO - pin 12
 * CLK - pin 13
 * CS - pin 4
 * BTN1 - 46
 * BTN2 - 47
 * REL - 49
 */
//#include <SPI.h>
//#include <SD.h>
//const int chipSelect = 4;
#include <Wire.h>
#include <dht.h>
#include <LiquidCrystal_I2C.h>
#include <math.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
LiquidCrystal_I2C lcd2(0x26, 16, 2);
dht DHT;
#define DHT11_PIN 48
const int analogInPin1 = A0, analogInPin3 = A2, analogInPin4 = A3,
analogInPin5 = A4;
const int analogInPin6 = A5, analogInPin9 = A8;
int
sensorValue1, sensorValue2, sensorValue3, sensorValue4, sensorValue5, s
ensorValue6, sensorValue7, sensorValue8, sensorValue9;
double outputValue=0;
double battre = A9;
double vbat = 0;
double vbat2;
const int buttonPin = 46;
const int button2 = 47;
const int heater = 49;
int waktudelay = 2000;
int buttonState = 0;
int buttonState2 = 0;
int ia=0, ja=0, ka=0, la=0;
int a=0, b=0, c=0, d=0;
double adc1, adc2, adc3, adc4, adc5;
double r1, r2, r3, r4, r5;
double r1ro, r2ro, r3ro, r4ro, r5ro;
double r1rg, r2rg, r3rg, r4rg, r5rg;
int vr1 = 43000;
```

```

int vr2 = 10000;
int vr3 = 10000;
int vr4 = 10000;
int vr5 = 10000;
int z;
int menit = 120; //2 menit
const int InputNodes = 5;
const int HiddenNodes = 8;
const int OutputNodes = 2;
const float maxi = 6;
//double sensitivitas=0;
//File dataFile = SD.open("log.txt", FILE_WRITE);

float sensi[InputNodes];    // 5 input nilai Sensitivitas

const float BobotInHid[InputNodes][HiddenNodes] = {    // Bobot Input
ke Hidden layer (5,8)
{ 0.643 , -0.653 , 1.174 , -0.408 , -0.028 , 1.083 , 1.944 , -
0.721 },
{ 11.173 , 6.882 , 12.772 , -13.634 , -3.265 , 5.558 , -5.518 ,
-6.935 },
{ 1.177 , 0.605 , 0.43 , 0.788 , -1.74 , -0.871 , -0.207 , -
1.123 },
{ 1.578 , 2.274 , -1.363 , 1.493 , -1.833 , 2.552 , -2.024 ,
0.503 },
{ -0.501 , 0.998 , 0.012 , -0.034 , 0.972 , -0.567 , -0.144 ,
1.348 }
};

const float BobotHidOut[HiddenNodes][OutputNodes] = {    // Bobot
Hidden layer ke Ouput(8,2)
{ -0.915 , 0.946 },
{ -0.622 , 0.334 },
{ 0.173 , 0.35 },
{ -0.278 , 0.241 },
{ 0.622 , -0.961 },
{ -0.832 , 0.95 },
{ 0.303 , -0.538 },
{ -0.193 , -0.756 }
};

const float Bias[HiddenNodes+OutputNodes] = { //(1,10)
-24.56 , -13.571 , -18.971 , 13.561 , 8.687 , -9.304 , 10.246
, 3.591 , -0.463 , -0.484
};

float Masukan[InputNodes];
float Hidden[HiddenNodes];
float Keluaran[OutputNodes];

int i, j, k; //layer
float sum2;
//float aktivasi;

void setup() {
    Serial.begin(115200);
    lcd.begin();

```



```

    lcd2.begin();
    // if (!SD.begin(chipSelect)) {
    //   lcd2.println("Card failed");
    //   return;
    // }
    // Serial.println("card ok");
    lcd.noBacklight();
    lcd2.noBacklight();
    delay(100);
    pinMode(heater, OUTPUT);
    pinMode(buttonPin, INPUT_PULLUP);
    pinMode(button2, INPUT_PULLUP);
    delay(200);
    lcd.backlight();
    lcd2.backlight();
    menit = menit*2;
    homescreen();
}

void loop() {

updatesensor();
updatepb();
    if (buttonState == HIGH)   ia++; if (buttonState2 == HIGH)   ja++;
    if (ja == 1) {
        if (buttonState2 == HIGH ) {
            lcd.setCursor(0,0); lcd.print("UJI TDK LANGSUNG");
            lcd.setCursor(0,1); lcd.print("   Yes       No   ");
            lcd2.noBacklight();
            lcd2.setCursor(0,0); lcd2.print("                                ");
            lcd2.setCursor(0,1); lcd2.print("                                ");
        }
        if (ia == 1) {
            lcd2.backlight();
            NNoff();   ia=0;
        }
    }
    //-----
    if (ja == 2) {
        if (buttonState2 == HIGH) {
            lcd.setCursor(0,0); lcd.print(" UJI LANGSUNG? ");
            lcd.setCursor(0,1); lcd.print("   Yes       No   ");
            lcd2.noBacklight();
            lcd2.setCursor(0,0); lcd2.print("                                ");
            lcd2.setCursor(0,1); lcd2.print("                                ");
        }
        if (ia == 1) {
            lcd2.backlight();
            NNNon();   ia=0;
        }
        delay(50);
    }
    //-----
    if (ja == 3) {
        if (buttonState2 == HIGH) {
            lcd.setCursor(0,0); lcd.print("      THANKS TO ");
            lcd.setCursor(0,1); lcd.print("      YES       NO ");
            lcd2.noBacklight();
            lcd2.setCursor(0,0); lcd2.print("                                ");

```

```

    lcd2.setCursor(0,1);  lcd2.print("
    }
    if (ia == 1) {
        about();
        homescreen(); ja=0; ia=0;
    }

    }

    if (ja == 4) {
        homescreen();
// thanks to
    ja = 0;
    }
    ia=0;
}
//-----
//-----HOME SCREEN-----
//-----

void homescreen(){
    lcd.setCursor(0,0);  lcd.print(" -*# E-NOSE #- ");
    lcd.setCursor(0,1);  lcd.print("
    lcd2.backlight();
    lcd2.setCursor(0,0);  lcd2.print("  MURIA  KUDUS  ");
    lcd2.setCursor(0,1);  lcd2.print("    UNIVERSITY  ");
}
//-----
//-----
//-----update status pb-----
//-----

void updatepb(){
    buttonState = digitalRead(buttonPin);
    buttonState2 = digitalRead(button2);
}
//-----
//-----updatesensor-----
//-----

void updatesensor(){
    analogReference(EXTERNAL);
    vbat = analogRead(battre);
    vbat = vbat/1024*5;
    sensorValue1  =  analogRead(analogInPin1);      sensorValue2  =
analogRead(analogInPin3);
    sensorValue3  =  analogRead(analogInPin4);      sensorValue4  =
analogRead(analogInPin5);
    sensorValue5  =  analogRead(analogInPin6);      sensorValue9  =
analogRead(analogInPin9);

    sensorValue1 = 1024 - sensorValue1;    sensorValue2 = 1024 -
sensorValue2;
    sensorValue3 = 1024 - sensorValue3;    sensorValue4 = 1024 -
sensorValue4;
    sensorValue5 = 1024 - sensorValue5;

//          Serial.print(sensorValue1);    Serial.print("
Serial.print(sensorValue2);                Serial.print("
Serial.print(sensorValue3); Serial.print(" ");

```

```

//          Serial.print(sensorValue4);          Serial.print(" ");
Serial.print(sensorValue5); Serial.print(" "); Serial.println();
    int chk = DHT.read11(DHT11_PIN);
//  switch (chk)
//  {
//      case DHTLIB_OK:
//          Serial.print("OK,\t");
//          break;
//      case DHTLIB_ERROR_CHECKSUM:
//          Serial.print("Checksum error,\t");
//          break;
//      case DHTLIB_ERROR_TIMEOUT:
//          Serial.println("Time out error,\t");
//          break;
//      case DHTLIB_ERROR_CONNECT:
//          Serial.print("Connect error,\t");
//          break;
//      case DHTLIB_ERROR_ACK_L:
//          Serial.print("Ack Low error,\t");
//          break;
//      case DHTLIB_ERROR_ACK_H:
//          Serial.print("Ack High error,\t");
//          break;
//      default:
//          Serial.println("Unknown error,\t");
//          break;
//  }
}

//-----
//-----//
//-----convert adc ke satuan hambatan-----
//-----//

void cres(){
    updatesensor();
    adc1 = sensorValue1*0.0048828125;
    r1=adc1*vr1;    r1= 215000 - r1;    r1= r1/adc1;

    adc2 = sensorValue2*0.0048828125;
    r2=adc2*vr2;    r2= 215000 - r2;    r2= r2/adc2;

    adc3 = sensorValue3*0.0048828125;
    r3=adc3*vr3;    r3= 215000 - r3;    r3= r3/adc3;

    adc4 = sensorValue4*0.0048828125;
    r4=adc4*vr4;    r4= 215000 - r4;    r4= r4/adc4;

    adc5 = sensorValue5*0.0048828125;
    r5=adc5*vr5;    r5= 215000 - r5;    r5= r5/adc5;

//    lcd.setCursor(0,0);  lcd.print(r1); lcd.print(" ");
//    lcd.setCursor(9,1);  lcd.print(r2); lcd.print(" ");
//    lcd.setCursor(0,1);  lcd.print(r3); lcd.print(" ");
//    lcd2.setCursor(0,0);  lcd2.print(r4); lcd2.print(" ");
//    lcd2.setCursor(0,1);  lcd2.print(r5); lcd2.print(" ");
}
//-----
//-----//

```

```

//-----rata rata cres Ro-----
//-----//
void cresro(){ // 24 data untuk 2 menit
    lcd.setCursor(0,1); lcd.print("    GET Ro DATA    ");
    lcd2.setCursor(0,0); lcd2.print("    PROGRESS    ");
    for (c = 0; c <= 48; c += 1) { //2.5 detik*48 = 120 detik (2menit)
        updatesensor();
        cres();

        r1ro=r1ro+r1; r1ro=r1ro/2;
        r2ro=r2ro+r2; r2ro=r2ro/2;
        r3ro=r3ro+r3; r3ro=r3ro/2;
        r4ro=r4ro+r4; r4ro=r4ro/2;
        r5ro=r5ro+r5; r5ro=r5ro/2;

        // lcd.setCursor(0,0); lcd.print(r1ro); lcd.print("    ");
        // lcd.setCursor(9,1); lcd.print(r2ro); lcd.print("    ");
        // lcd.setCursor(0,1); lcd.print(r3ro); lcd.print("    ");
        // lcd2.setCursor(0,0); lcd2.print(r4ro); lcd2.print("    ");
        // lcd2.setCursor(0,1); lcd2.print(r5ro); lcd2.print("    ");
        a = c*100/48;
        lcd2.setCursor(0,1); lcd2.print("    "); lcd2.print(a);
        lcd2.print("%    ");
        delay(2500);
    };
    Serial.print(r1ro); Serial.print("    "); Serial.print(r2ro);
    Serial.print("    ");
    Serial.print(r3ro); Serial.print("    "); Serial.print(r4ro);
    Serial.print("    ");
    Serial.print(r5ro); Serial.print("    ");
    lcd.setCursor(0,1); lcd.print("Ro DATA COMPLETE");
    lcd2.noBacklight();
    lcd2.setCursor(0,0); lcd2.print("    ");
    lcd2.setCursor(0,1); lcd2.print("    ");
    a=0;
}
//-----//
//-----rata rata cres Rg-----
//-----//
void cresrg(){ // 24 data untuk 2 menit

    lcd.setCursor(0,0); lcd.print("    GET Rg DATA    ");
    lcd.setCursor(0,1); lcd.print("    ");
    lcd2.backlight();
    lcd2.setCursor(0,0); lcd2.print("    PROGRESS    ");
    for (c = 0; c <= 48; c += 1) { //2.5 detik*48 = 120 detik (2menit)
        updatesensor();
        cres();

        r1rg=r1rg+r1; r1rg=r1rg/2;
        r2rg=r2rg+r2; r2rg=r2rg/2;
        r3rg=r3rg+r3; r3rg=r3rg/2;
        r4rg=r4rg+r4; r4rg=r4rg/2;
        r5rg=r5rg+r5; r5rg=r5rg/2;
        a = c*100/48;
        lcd2.setCursor(0,1); lcd2.print("    "); lcd2.print(a);
        lcd2.print("%    ");
        delay(2500);
    };
}

```



```

    }
    Serial.print(r1rg);    Serial.print("    ");    Serial.print(r2rg);
Serial.print(" ");
    Serial.print(r3rg);    Serial.print("    ");    Serial.print(r4rg);
Serial.print(" ");
    Serial.print(r5rg); Serial.print(" ");
    lcd.setCursor(0,1);  lcd.print("Rg DATA COMPLETE");
    lcd2.noBacklight();
    lcd2.setCursor(0,0);  lcd2.print("                ");
    lcd2.setCursor(0,1);  lcd2.print("                ");
    a=0;
}
//-----
//-----//
//-----HEATER-----
//-----
void kontrolheater(){
    if (DHT.temperature <= 35) {
        lcd2.setCursor(0,0);          lcd2.print("    T    chamber    ");
        lcd2.print(DHT.temperature);
        lcd2.setCursor(0,1);  lcd2.print("    HEATER ON    "); //heater on
        digitalWrite(heater, HIGH);
        delay(1000);
        for (c = 0; c <= 5 ; c += 1) {
            updatesensor();
            lcd2.setCursor(0,0);          lcd2.print("    T    chamber    ");
        lcd2.print(DHT.temperature);
            if (c == 3) {c=0;}
            if (DHT.temperature >= 35) {c=6;}
        }
    }
}
//-----
//-----//
//-----DATA NN ONLINE-----
//-----
void NNon() {
    lcd.setCursor(0,0);  lcd.print("    UJI DAGING    ");
    lcd.setCursor(0,1);  lcd.print("    LANGSUNG    ");
    // heater on, tunggu 2 menit / suhu diatas 30
    // read adc -> convert to res
    // sensitivitas
    // kalkulasi nn
    kontrolheater();
    updatesensor();
    // ----- 2
    digitalWrite(heater, HIGH);
    cresro();
    lcd.setCursor(0,0);  lcd.print("    SUNTIKKAN GAS ");
    lcd.setCursor(0,1);  lcd.print("    KEDALAM CHAMBER");
    for (c = 0; c <= 52 ; c += 1) {
        updatepb();
        if (buttonState2 == HIGH) {
            c = 52;
        }
        if (c == 50) {
            c = 0;
        }
        delay(50);
    }
}

```

```

    }
    cresrg();
    lcd2.backlight();
    sensitivitas();
    NN();

    lcd.setCursor(0,0);lcd.print(" HASIL CALC NN  ");
    //if( Keluaran[0],DEC >= Keluaran[1],DEC){
    // lcd.setCursor(0,1); lcd.print("  DAGING SEGAR  ");
    //}
    //else if( Keluaran[0],DEC <= Keluaran[1],DEC){
    // lcd.setCursor(0,1); lcd.print("DAGING TDK SEGAR");
    //}
    //else {
    // lcd.setCursor(0,1); lcd.print("  INVALID DATA  ");
    //};
    }

    //-----
    //-----//
    //-----tampil ke serial monitor-----
    //-----//
    void sensitivitas(){
        lcd.setCursor(0,0); lcd.print("  SENSITIVITAS  ");
        lcd.setCursor(0,1); lcd.print("  ");
        sensi[0]=r1rg/r1ro; sensi[1]=r2rg/r2ro; sensi[2]=r3rg/r3ro;
        sensi[3]=r4rg/r4ro; sensi[4]=r5rg/r5ro;
        Serial.print(sensi[0]); Serial.print("  ");
        Serial.print(sensi[1]); Serial.print("  ");
        Serial.print(sensi[2]); Serial.print("  ");
        Serial.print(sensi[3]); Serial.print("  ");
        Serial.print(sensi[4]); Serial.print("  ");

    }
    //-----
    //-----//
    //-----tampil ke serial monitor-----
    //-----//
    void serial(){
        lcd.setCursor(0,0); lcd.print(sensorValue1); lcd.print("  ");
        lcd.setCursor(8,0); lcd.print(sensorValue2); lcd.print("  ");
        lcd.setCursor(0,1); lcd.print(sensorValue3); lcd.print("  ");
        lcd.setCursor(6,1); lcd.print(sensorValue4); lcd.print("  ");
        lcd.setCursor(12,1); lcd.print(sensorValue5); lcd.print("  ");
        lcd2.setCursor(12,0); lcd2.print(vbat); lcd2.print("V ");
        lcd2.setCursor(0,1); lcd2.print("H "); lcd2.print(DHT.humidity,
1);
        lcd2.setCursor(7,1); lcd2.print("T ");
        lcd2.print(DHT.temperature, 1); lcd2.print(" "); lcd2.print(ia);
        lcd2.print(ja);
        lcd2.setCursor(0,0);
        lcd2.print(ka); lcd2.print(" "); lcd2.print(la); lcd2.print("
"); lcd2.print(menit);
        lcd2.print("  ");
    }

    //-----
    //-----//

```

```

//-----isi data ke SDCARD data online-----
//-----//
void savesdon(){
//    File dataFile = SD.open("DataOnLine.txt", FILE_WRITE);
//    dataFile.println("");
//    dataFile.print(sensi[0]);    dataFile.print(" , ");
//    dataFile.print(sensorValue2);    dataFile.print(" , ");
//    dataFile.print(sensorValue3);    dataFile.print(" , ");
//    dataFile.print(sensorValue4);    dataFile.print(" , ");
//    dataFile.print(sensorValue5);    dataFile.print(" , ");
//    dataFile.print(DHT.humidity, 1);    dataFile.print(" , ");
//    dataFile.print(DHT.temperature, 1);    dataFile.print(" , ");
//    dataFile.print(vbat);
//    dataFile.close();
}

//-----//
//-----//
//-----kalkulasi NN-----
//-----//
void NN(){

    lcd.setCursor(0,0); lcd.print("    WAIT DATA    ");
    lcd.setCursor(0,1);

//masukan
    Serial.println (" Input ");

    lcd2.setCursor(0,0); lcd2.print ("    loading    ");
    lcd2.setCursor(0,1); lcd2.print ("    ");
    for( i = 0 ; i < InputNodes ; i++ ) {
        Masukan[i] = sensi[i] / maxi;
        Serial.print ( sensi[i], DEC);
        Serial.print (" ");
        Serial.print ( Masukan[i], DEC);
        Serial.print (" ");
//        dataFile.print(Masukan[i], DEC);    dataFile.print("
, ");
        delay(10);
    }
    Serial.println ();

/*****
*
* Compute hidden layer activations
*****/
/
    for( j = 0 ; j < HiddenNodes ; j++ ) {
        sum2=0;
        for( i = 0 ; i < HiddenNodes ; i++ ) {            //nilai sum
            sum2 += Masukan[i] * BobotInHid[i][j] ;
            delay(10);
        }
        Hidden[j] = 1.0/(1.0 + exp(-(sum2+Bias[j]))); //nilai sigmoid
    }
}

```

```

/*****
*
* Compute output layer
*****/
/
for( k = 0 ; k < OutputNodes ; k++ ) {
    sum2=0;
    for( j = 0 ; j < HiddenNodes ; j++ ) {                //nilai
sum
        sum2 += Hidden[j] * BobotHidOut[j][k] ;
        delay(10);
    }
    Keluaran[k] = 1.0/(1.0 + exp(-(sum2 + Bias[k+8]))); //nilai
sigmoid
}

//luaran
    lcd2.print("");
    Serial.println (" Output ");
    lcd2.setCursor(0,1);
    lcd2.print(" "); //clear lcd
    lcd2.setCursor(0,0);
    lcd2.print(" "); //clear lcd
    lcd2.setCursor(0,0);
    for( i = 0 ; i < OutputNodes ; i++ ) {
        Serial.print (Keluaran[i],DEC);
        Serial.print (" ");

        lcd2.print(" "); lcd2.print(Keluaran[i],DEC);
        lcd2.setCursor(0,1);
//        dataFile.print(sensi[i]);    dataFile.print(" , ");
    }

    Serial.println ();
    if(ja == 2){
        if( Keluaran[0] >= Keluaran[1]){
            lcd.setCursor(0,1); lcd.print(" DAGING SEGAR
");
        }
        else {
            lcd.setCursor(0,1);    lcd.print("DAGING TDK
SEGAR");
        }
    }
    //reset input, hidden dan luaran
    sum2=0;
    //for( i = 0 ; i < InputNodes ; i++ ) {
    //    sensi[i]=0;
    //    Masukan[i]=0;
    //    }
    //    for( j = 0 ; j < HiddenNodes ; j++ ) {
    //        Hidden[j]=0;
    //    }
    //    for( k = 0 ; k < OutputNodes ; k++ ) {
    //        Keluaran[k] =0;
    //    }

}

```



```

//-----
-----//
//-----DATA NN OFFLINE-----
-----//

void NNoff(){

    lcd.setCursor(0,0);  lcd.print("    UJI DAGING  ");
    lcd.setCursor(0,1);  lcd.print(" TIDAK LANGSUNG");
    lcd.setCursor(0,1);

        for (a = 0; a <= 16; a += 1) {
            lcd.print("#");
            delay(200);
        }

    lcd.setCursor(0,0);  lcd.print(" Load data Input");
    lcd.setCursor(0,1);  lcd.print("                ");
    char val;
    for (c = 1; c <= 22; c += 1) {
        d=1;
        val = c;
        if (val==1) {
            sensi[0] = 0.995; sensi[1] = 1.081; sensi[2] = 1.003; sensi[3] =
1.001; sensi[4] = 1.005;
        }
        else if (val==2) {
            sensi[0] = 0.99; sensi[1] = 1.12; sensi[2] = 1.01; sensi[3] = 0.97;
sensi[4] = 1.01;
        }
        else if (val==3) {
            sensi[0] = 0.961; sensi[1] = 1.067; sensi[2] = 0.976; sensi[3] =
0.99; sensi[4] = 0.964;
        }
        else if (val==4) {
            sensi[0] = 0.997; sensi[1] = 1.119; sensi[2] = 1.009; sensi[3] =
1.025; sensi[4] = 1.021;
        }
        else if (val==5) {
            sensi[0] = 0.989; sensi[1] = 1.092; sensi[2] = 0.995; sensi[3] =
0.96; sensi[4] = 1.005;
        }
        else if (val==6) {
            sensi[0] = 0.992; sensi[1] = 1.085; sensi[2] = 0.995; sensi[3] =
0.999; sensi[4] = 1.001;
        }
        else if (val==7) {
            sensi[0] = 0.997; sensi[1] = 1.086; sensi[2] = 0.996; sensi[3] =
1.013; sensi[4] = 1.025;
        }
        else if (val==8) {
            sensi[0] = 1.001; sensi[1] = 1.088; sensi[2] = 1.004; sensi[3] =
1.017; sensi[4] = 1.022;
        }
        else if (val==9) {
            sensi[0] = 1.001; sensi[1] = 1.089; sensi[2] = 1.013; sensi[3] =
0.991; sensi[4] = 0.998;
        }
        else if (val==10) { //a 10
            sensi[0] = 1.006; sensi[1] = 1.034; sensi[2] = 1.025; sensi[3] =
1.01; sensi[4] = 0.999;
        }
    }
}

```

```

else if (val==11) { //b 11
    sensi[0] = 1.012; sensi[1] = 1.068; sensi[2] = 1.015; sensi[3] =
1.008; sensi[4] = 1.012;
}
else if (val==12) { //c 12
    sensi[0] = 2.92; sensi[1] = 1.15; sensi[2] = 2.83; sensi[3] = 2.46;
sensi[4] = 3.70;
}
else if (val==13) { //d 13
    sensi[0] = 1.108; sensi[1] = 1.114; sensi[2] = 1.12; sensi[3] =
0.999; sensi[4] = 1.117;
}
else if (val==14) { //e 14
    sensi[0] = 1.374; sensi[1] = 0.996; sensi[2] = 1.266; sensi[3] =
1.228; sensi[4] = 1.417;
}
else if (val==15) { //f 15
// sensi[0] = 2.143; sensi[1] = 0.980; sensi[2] = 1.940; sensi[3]
= 1.760; sensi[4] = 2.510;
    sensi[0] = 2.143; sensi[1] = 0.9; sensi[2] = 1.940; sensi[3] =
1.760; sensi[4] = 2.510;
}
else if (val==16) { //g 16
    sensi[0] = 2.279; sensi[1] = 1.298; sensi[2] = 2.294; sensi[3] =
2.24; sensi[4] = 2.639;
}
else if (val==17) { //h 17
    sensi[0] = 3.589; sensi[1] = 1.109; sensi[2] = 3.419; sensi[3] =
2.922; sensi[4] = 5.091;
}
else if (val==18) { //i 18
    sensi[0] = 4.198; sensi[1] = 1.152; sensi[2] = 4.102; sensi[3] =
3.336; sensi[4] = 5.625;
}
else if (val==19) { //j 19
    sensi[0] = 3.031; sensi[1] = 1.161; sensi[2] = 2.978; sensi[3] =
2.597; sensi[4] = 3.788;
}
else if (val==20) { //k 20
    sensi[0] = 3.48; sensi[1] = 1.07; sensi[2] = 3.43; sensi[3] = 2.86;
sensi[4] = 4.31;
}
else if (val==21) { //l 21
    sensi[0] = 2.26; sensi[1] = 1.51; sensi[2] = 2.207; sensi[3] =
2.135; sensi[4] = 2.789;
}
else if (val==22) { //m 22
    sensi[0] = 3.93; sensi[1] = 1.14; sensi[2] = 3.90; sensi[3] = 3.11;
sensi[4] = 5.14;
}
// -----data uji segar-----
//
else if (val==23) { //m
    sensi[0] = 0.702839407; sensi[1] = 0.88890896; sensi[2] =
0.959802318; sensi[3] = 0.899047367; sensi[4] = 0.987455068;
}
else if (val==24) { //n
    sensi[0] = 1.145625678; sensi[1] = 1.119375457; sensi[2] =
1.002090354; sensi[3] = 1.140697643; sensi[4] = 1.01452403;
}

```

```

}
else if (val==254) { //o
  sensi[0] = 1.019807489; sensi[1] = 1.108160731; sensi[2] =
0.996963351; sensi[3] = 1.121426149; sensi[4] = 1.005300521;
}
// -----data uji tidak
segar-----//
else if (val==26) { //p
  sensi[0] = 1.137050283; sensi[1] = 1.03709916; sensi[2] =
1.006642131; sensi[3] = 1.058695014; sensi[4] = 1.025453744;
}
else if (val==27) { //q
  sensi[0] = 1.217713189; sensi[1] = 1.576189945; sensi[2] =
1.606527094; sensi[3] = 1.642779849; sensi[4] = 1.311457964;
}
else if (val==28) { //r
  sensi[0] = 1.202653492; sensi[1] = 1.62285434; sensi[2] =
1.778264882; sensi[3] = 1.631823461; sensi[4] = 1.277608188;
};

NN();

  lcd.setCursor(0,1); lcd.print ("    DATA KE "); lcd.print (c);
  updatepb();
  if (buttonState == HIGH)    ia++;  if (buttonState2 == HIGH)
ja++;
  if (ja == 2) {
    lcd2.setCursor(0,0);  lcd2.print(" cancel by user ");
    lcd2.setCursor(0,1);  lcd2.print("                ");
    delay(2000);
    homescreen();  ja = 0;  ia = 0;  c = 22;
  }
  delay(1500);
}
}

//-----
//-----//
//----- thanks to -----
//-----//

void about(){
  lcd.setCursor(0,0);  lcd.print("                ");
  lcd.setCursor(0,1);  lcd.print("                ");
  for (c = 16; c >= 0; c -= 1) {
    lcd.setCursor(c,0);  lcd.print("ALLAH SUBHANAHU ");
    lcd.setCursor(c,1);  lcd.print("    WA TAALA    ");
    delay(200);
  }
  delay(2000);

  for (c = 16; c >= 0; c -= 1) {
    lcd.setCursor(c,0);  lcd.print("AYAH DAN ALM IBU");
    lcd.setCursor(c,1);  lcd.print("                ");
    delay(200);
  }
  delay(2000);
  lcd2.backlight();
  for (c = 16; c >= 0; c -= 1) {
    lcd.setCursor(c,0);  lcd.print("  BUDI GUNAWAN  ");

```

```

lcd.setCursor(c,1); lcd.print("      ST.,MT.      ");
lcd2.setCursor(c,0); lcd2.print(" ARIEF SUDARMAJI");
lcd2.setCursor(c,1); lcd2.print(" ST.,MT.,PhD.  ");
delay(200);
}
delay(2000);
for (c = 16; c >= 0; c -= 1) {
  lcd.setCursor(c,0); lcd.print("      SOLEKHAN      ");
  lcd.setCursor(c,1); lcd.print("      ST.,MT.      ");
  lcd2.setCursor(c,0); lcd2.print("      DAN      ");
  lcd2.setCursor(c,1); lcd2.print("      DOSEN UMK      ");
  delay(200);
}
delay(2000);
for (c = 16; c >= 0; c -= 1) {
  lcd.setCursor(c,0); lcd.print("      TEMAN TEMAN      ");
  lcd.setCursor(c,1); lcd.print("      UMK & UNSOED      ");
  lcd2.noBacklight();
  lcd2.setCursor(c,0); lcd2.print("      ");
  lcd2.setCursor(c,1); lcd2.print("      ");
  delay(200);
}
delay(2000);
}

```

2. Program arduino *Neural Network* Pendeteksian Tidak Langsung dan Pendeteksian Langsung menggunakan data Matlab

```

/*
 * dht11      pin 48 -->
 * batt      pin A9 -->
 * LM35      pin A7
 * mq5       pin A6
 * tgs813     pin A5 -->
 * tgs2602    pin A4 -->
 * tgs2620    pin A3 -->
 * tgs2600    pin A2 -->
 * mq137      pin A1
 * mq135      pin A0 -->
 * MOSI - pin 11
 * MISO - pin 12
 * CLK - pin 13
 * CS - pin 4
 * BTN1 - 46
 * BTN2 - 47
 * REL - 49
 */
#include <SPI.h>
#include <SD.h>
//const int chipSelect = 4;
#include <Wire.h>
#include <dht.h>
#include <LiquidCrystal_I2C.h>
#include <math.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
LiquidCrystal_I2C lcd2(0x26, 16, 2);
dht DHT;
#define DHT11_PIN 48

```



```

const int analogInPin1 = A0, analogInPin3 = A2, analogInPin4 = A3,
analogInPin5 = A4;
const int analogInPin6 = A5, analogInPin9 = A8;
int
sensorValue1, sensorValue2, sensorValue3, sensorValue4, sensorValue5, s
ensorValue6, sensorValue7, sensorValue8, sensorValue9;
double outputValue=0;
double battre = A9;
double vbat = 0;
double vbat2;
const int buttonPin = 46;
const int button2 = 47;
const int heater = 49;
int waktudelay = 2000;
int buttonState = 0;
int buttonState2 = 0;
int ia=0, ja=0, ka=0, la=0;
int a=0, b=0, c=0, d=0;
double adc1, adc2, adc3, adc4, adc5;
double r1, r2, r3, r4, r5;
double r1ro, r2ro, r3ro, r4ro, r5ro;
double r1rg, r2rg, r3rg, r4rg, r5rg;
int vr1 = 43000;
int vr2 = 10000;
int vr3 = 10000;
int vr4 = 10000;
int vr5 = 10000;
int z;
int menit = 120; //2 menit
const int InputNodes = 5;
const int HiddenNodes = 8;
const int OutputNodes = 2;
const float maxi = 6;
//double sensitivitas=0;
//File dataFile = SD.open("log.txt", FILE_WRITE);

float sensi[InputNodes]; // 5 input nilai Sensitivitas

const float BobotInHid[InputNodes][HiddenNodes] = { // Bobot
Input ke Hidden layer (5,8)
{ 30.70241165 , -3.713850021 , 8.947570801 , -2.377589941 , -
3.426021576 , 22.2597847 , -5.18145752 , -1.013198495 },
{ 27.04392242 , -4.444502354 , 6.228762627 , -3.49154377 , -
3.074835062 , 20.07707787 , -6.841657639 , -0.678613126 },
{ 28.58146095 , -4.381884575 , 8.939346313 , -2.092128754 , -
2.756980181 , 20.48993301 , -4.794213772 , -1.270322084 },
{ 2.867153883 , -0.877912819 , 1.399313092 , 0.270878077 , -
0.9207201 , 1.737950087 , 0.633344233 , 0.022247078 },
{ 28.57825851 , -2.519629478 , 9.453680992 , -2.096366882 , -
2.394304514 , 21.44807434 , -4.127983093 , -1.405970693 }
};

const float BobotHidOut[HiddenNodes][OutputNodes] = { // Bobot
Hidden layer ke Ouput (8,2)
{ -15.82777119 , 16.31928825 },
{ 2.900993109 , -3.837117672 },
{ -4.701883793 , 4.159037113 },
{ 2.423502207 , -2.96257925 },

```

```

{ 2.023983955 , -3.747912169 },
{ -11.7565937 , 11.3633604 },
{ 4.446260452 , -4.158718586 },
{ 1.702514768 , -1.408094049 }
};

const float Bias[HiddenNodes+OutputNodes] = { //(1,10)
-21.48898888 , 2.772899151 , -6.767085075 , 1.16192627 ,
1.690261841 , -15.76756477 , 4.04007864 , -1.39218843 ,
10.9704628 , -9.764332771
};

float Masukan[InputNodes];
float Hidden[HiddenNodes];
float Keluaran[OutputNodes];

int i, j, k; //layer
float sum2;
//float aktivasi;

void setup() {
  Serial.begin(115200);
  lcd.begin();
  lcd2.begin();
  // if (!SD.begin(chipSelect)) {
  //   lcd2.println("Card failed");
  //   return;
  // }
  // Serial.println("card ok");
  lcd.noBacklight();
  lcd2.noBacklight();
  delay(100);
  pinMode(heater, OUTPUT);
  pinMode(buttonPin, INPUT_PULLUP);
  pinMode(button2, INPUT_PULLUP);
  delay(200);
  lcd.backlight();
  lcd2.backlight();
  menit = menit*2;
  homescreen();
}

void loop() {
  updatesensor();
  updatepb();
  if (buttonState == HIGH) ia++; if (buttonState2 == HIGH)
ja++;
if (ja == 1) {
  if (buttonState2 == HIGH ) {
    lcd.setCursor(0,0); lcd.print("UJI TDK LANGSUNG");
    lcd.setCursor(0,1); lcd.print("   Yes       No   ");
    lcd2.noBacklight();
    lcd2.setCursor(0,0); lcd2.print("                ");
    lcd2.setCursor(0,1); lcd2.print("                ");
  }
  if (ia == 1) {
    lcd2.backlight();
  }
}

```

```

        NNoff();    ia=0;
    }
}
//-----
if (ja == 2) {
    if (buttonState2 == HIGH) {
        lcd.setCursor(0,0);  lcd.print(" UJI LANGSUNG? ");
        lcd.setCursor(0,1);  lcd.print("   Yes       No   ");
        lcd2.noBacklight();
        lcd2.setCursor(0,0);  lcd2.print(" ");
        lcd2.setCursor(0,1);  lcd2.print(" ");
    }
    if (ia == 1) {
        lcd2.backlight();
        NNNon();    ia=0;
    }
    delay(50);
}
//-----
if (ja == 3) {
    if (buttonState2 == HIGH) {
        lcd.setCursor(0,0);  lcd.print("   THANKS TO ");
        lcd.setCursor(0,1);  lcd.print("   YES       NO   ");
        lcd2.noBacklight();
        lcd2.setCursor(0,0);  lcd2.print(" ");
        lcd2.setCursor(0,1);  lcd2.print(" ");
    }
    if (ia == 1) {
        about();
        homescreen(); ja=0; ia=0;
    }
}

if (ja == 4) {
    homescreen();
//  thanks to
    ja = 0;
}
    ia=0;
}
//-----
//-----//
//-----HOME SCREEN-----
//-----
void homescreen() {
    lcd.setCursor(0,0);  lcd.print(" -*# E-NOSE #- ");
    lcd.setCursor(0,1);  lcd.print(" ");
    lcd2.backlight();
    lcd2.setCursor(0,0);  lcd2.print("  MURIA  KUDUS  ");
    lcd2.setCursor(0,1);  lcd2.print("   UNIVERSITY   ");
}
//-----
//-----//
//-----update status pb-----
//-----//
void updatepb() {
    buttonState = digitalRead(buttonPin);
    buttonState2 = digitalRead(button2);

```

```

    }
//-----
//-----//
//-----updatesensor-----
//-----//
    void updatesensor(){
        analogReference(EXTERNAL);
        vbat = analogRead(battre);
        vbat = vbat/1024*5;
        sensorValue1 = analogRead(analogInPin1);  sensorValue2 =
        analogRead(analogInPin3);
        sensorValue3 = analogRead(analogInPin4);  sensorValue4 =
        analogRead(analogInPin5);
        sensorValue5 = analogRead(analogInPin6);  sensorValue9 =
        analogRead(analogInPin9);

        sensorValue1 = 1024 - sensorValue1;  sensorValue2 = 1024 -
        sensorValue2;
        sensorValue3 = 1024 - sensorValue3;  sensorValue4 = 1024 -
        sensorValue4;
        sensorValue5 = 1024 - sensorValue5;

        // Serial.print(sensorValue1); Serial.print(" ");
        Serial.print(sensorValue2); Serial.print(" ");
        Serial.print(sensorValue3); Serial.print(" ");
        // Serial.print(sensorValue4); Serial.print(" ");
        Serial.print(sensorValue5); Serial.print(" "); Serial.println();
        int chk = DHT.read11(DHT11_PIN);
        // switch (chk)
        // {
        //     case DHTLIB_OK:
        //         Serial.print("OK,\t");
        //         break;
        //     case DHTLIB_ERROR_CHECKSUM:
        //         Serial.print("Checksum error,\t");
        //         break;
        //     case DHTLIB_ERROR_TIMEOUT:
        //         Serial.println("Time out error,\t");
        //         break;
        //     case DHTLIB_ERROR_CONNECT:
        //         Serial.print("Connect error,\t");
        //         break;
        //     case DHTLIB_ERROR_ACK_L:
        //         Serial.print("Ack Low error,\t");
        //         break;
        //     case DHTLIB_ERROR_ACK_H:
        //         Serial.print("Ack High error,\t");
        //         break;
        //     default:
        //         Serial.println("Unknown error,\t");
        //         break;
        // }
    }

//-----
//-----//
//-----convert adc ke satuan hambatan-----
//-----//

```



```

void cres(){
    updatesensor();
    adc1 = sensorValue1*0.0048828125;
    r1=adc1*vr1;    r1= 215000 - r1;    r1= r1/adc1;

    adc2 = sensorValue2*0.0048828125;
    r2=adc2*vr2;    r2= 215000 - r2;    r2= r2/adc2;

    adc3 = sensorValue3*0.0048828125;
    r3=adc3*vr3;    r3= 215000 - r3;    r3= r3/adc3;

    adc4 = sensorValue4*0.0048828125;
    r4=adc4*vr4;    r4= 215000 - r4;    r4= r4/adc4;

    adc5 = sensorValue5*0.0048828125;
    r5=adc5*vr5;    r5= 215000 - r5;    r5= r5/adc5;

    //    lcd.setCursor(0,0);    lcd.print(r1); lcd.print("    ");
    //    lcd.setCursor(9,1);    lcd.print(r2); lcd.print("    ");
    //    lcd.setCursor(0,1);    lcd.print(r3); lcd.print("    ");
    //    lcd2.setCursor(0,0);    lcd2.print(r4); lcd2.print("    ");
    //    lcd2.setCursor(0,1);    lcd2.print(r5); lcd2.print("    ");
    }
    //-----//
    //-----rata rata cres Ro-----//
    void cresro(){ // 24 data untuk 2 menit
        lcd.setCursor(0,1);    lcd.print("    GET Ro DATA    ");
        lcd2.setCursor(0,0);    lcd2.print("    PROGRESS    ");
        for (c = 0; c <= 48; c += 1) { //2.5 detik*48 = 120 detik
            (2menit)
            updatesensor();
            cres();

            r1ro=r1ro+r1;    r1ro=r1ro/2;
            r2ro=r2ro+r2;    r2ro=r2ro/2;
            r3ro=r3ro+r3;    r3ro=r3ro/2;
            r4ro=r4ro+r4;    r4ro=r4ro/2;
            r5ro=r5ro+r5;    r5ro=r5ro/2;

            //    lcd.setCursor(0,0);    lcd.print(r1ro); lcd.print("    ");
            //    lcd.setCursor(9,1);    lcd.print(r2ro); lcd.print("    ");
            //    lcd.setCursor(0,1);    lcd.print(r3ro); lcd.print("    ");
            //    lcd2.setCursor(0,0);    lcd2.print(r4ro); lcd2.print("    ");
            //    lcd2.setCursor(0,1);    lcd2.print(r5ro); lcd2.print("    ");
            a = c*100/48;
            lcd2.setCursor(0,1);    lcd2.print("    "); lcd2.print(a);
            lcd2.print("%    ");
            delay(2500);
        };
        Serial.print(r1ro); Serial.print(" "); Serial.print(r2ro);
        Serial.print(" ");
        Serial.print(r3ro); Serial.print(" "); Serial.print(r4ro);
        Serial.print(" ");
        Serial.print(r5ro); Serial.print(" ");
        lcd.setCursor(0,1);    lcd.print("Ro DATA COMPLETE");
        lcd2.noBacklight();
        lcd2.setCursor(0,0);    lcd2.print("    ");
    }
}

```

```

    lcd2.setCursor(0,1);  lcd2.print("                ");
    a=0;
}
//-----
//-----//
//-----rata rata cres Rg-----
//-----//

void cresrg(){ // 24 data untuk 2 menit

    lcd.setCursor(0,0);  lcd.print("    GET Rg DATA    ");
    lcd.setCursor(0,1);  lcd.print("                ");
    lcd2.backlight();
    lcd2.setCursor(0,0);  lcd2.print("    PROGRESS    ");
    for (c = 0; c <= 48; c += 1) { //2.5 detik*48 = 120 detik
(2menit)
        updatesensor();
        cres();

        r1rg=r1rg+r1;  r1rg=r1rg/2;
        r2rg=r2rg+r2;  r2rg=r2rg/2;
        r3rg=r3rg+r3;  r3rg=r3rg/2;
        r4rg=r4rg+r4;  r4rg=r4rg/2;
        r5rg=r5rg+r5;  r5rg=r5rg/2;
        a = c*100/48;
        lcd2.setCursor(0,1);  lcd2.print("                "); lcd2.print(a);
        lcd2.print("%                ");
        delay(2500);
    }
    Serial.print(r1rg); Serial.print(" "); Serial.print(r2rg);
    Serial.print(" ");
    Serial.print(r3rg); Serial.print(" "); Serial.print(r4rg);
    Serial.print(" ");
    Serial.print(r5rg); Serial.print(" ");
    lcd.setCursor(0,1);  lcd.print("Rg DATA COMPLETE");
    lcd2.noBacklight();
    lcd2.setCursor(0,0);  lcd2.print("                ");
    lcd2.setCursor(0,1);  lcd2.print("                ");
    a=0;
}
//-----
//-----//
//-----HEATER-----
//-----//

void kontrolheater(){
    if (DHT.temperature <= 35) {
        lcd2.setCursor(0,0);  lcd2.print(" T chamber ");
        lcd2.print(DHT.temperature);
        lcd2.setCursor(0,1);  lcd2.print("    HEATER ON    "); //heater
on
        digitalWrite(heater, HIGH);
        delay(1000);
        for (c = 0; c <= 5 ; c += 1) {
            updatesensor();
            lcd2.setCursor(0,0);  lcd2.print(" T chamber ");
            lcd2.print(DHT.temperature);
            if (c == 3) {c=0;}
            if (DHT.temperature >= 35) {c=6;}
        }
    }
}

```

```

    }
//-----
//-----//
//-----DATA NN ONLINE-----
//-----//

void NNon(){
    lcd.setCursor(0,0); lcd.print("    UJI DAGING    ");
    lcd.setCursor(0,1); lcd.print("    LANGSUNG    ");
// heater on, tunggu 2 menit / suhu diatas 30
// read adc -> convert to res
// sensitivitas
// kalkulasi nn
kontrolheater();
updatesensor();
// ----- 2
digitalWrite(heater, HIGH);
cresro();
    lcd.setCursor(0,0); lcd.print("    SUNTIKKAN GAS    ");
    lcd.setCursor(0,1); lcd.print("    KEDALAM CHAMBER");
    for (c = 0; c <= 52 ; c += 1) {
        updatepb();
        if (buttonState2 == HIGH) {
            c = 52;
        }
        if (c == 50) {
            c = 0;
        }
        delay(50);
    }
cresrg();
lcd2.backlight();
sensitivitas();
NN();

    lcd.setCursor(0,0);lcd.print(" HASIL CALC NN    ");
//if( Keluaran[0],DEC >= Keluaran[1],DEC){
// lcd.setCursor(0,1); lcd.print("    DAGING SEGAR    ");
//}
//else if( Keluaran[0],DEC <= Keluaran[1],DEC){
// lcd.setCursor(0,1); lcd.print("DAGING TDK SEGAR");
//}
//else {
// lcd.setCursor(0,1); lcd.print("    INVALID DATA    ");
//};
    }

//-----
//-----//
//-----tampil ke serial monitor-----
//-----//

void sensitivitas(){
    lcd.setCursor(0,0); lcd.print("    SENSITIVITAS    ");
    lcd.setCursor(0,1); lcd.print("                    ");
    sensi[0]=r1rg/r1ro; sensi[1]=r2rg/r2ro; sensi[2]=r3rg/r3ro;
    sensi[3]=r4rg/r4ro; sensi[4]=r5rg/r5ro;
    Serial.print(sensi[0]); Serial.print(" ");
    Serial.print(sensi[1]); Serial.print(" ");
    Serial.print(sensi[2]); Serial.print(" ");
    Serial.print(sensi[3]); Serial.print(" ");

```

```

    Serial.print(sensi[4]); Serial.print(" ");

}
//-----
//-----//
//-----tampil ke serial monitor-----
//-----//

void serial(){
    lcd.setCursor(0,0); lcd.print(sensorValue1); lcd.print(" ");
    lcd.setCursor(8,0); lcd.print(sensorValue2); lcd.print("
");
    lcd.setCursor(0,1); lcd.print(sensorValue3); lcd.print("
");
    lcd.setCursor(6,1); lcd.print(sensorValue4); lcd.print("
");
    lcd.setCursor(12,1); lcd.print(sensorValue5); lcd.print("
");
    lcd2.setCursor(12,0); lcd2.print(vbat); lcd2.print("V ");
    lcd2.setCursor(0,1); lcd2.print("H ");
    lcd2.print(DHT.humidity, 1);
    lcd2.setCursor(7,1); lcd2.print("T ");
    lcd2.print(DHT.temperature, 1); lcd2.print(" "); lcd2.print(ia);
    lcd2.print(ja);
    lcd2.setCursor(0,0);
    lcd2.print(ka); lcd2.print(" "); lcd2.print(la); lcd2.print("
"); lcd2.print(menit);
    lcd2.print(" ");
}

//-----
//-----//
//-----isi data ke SDCARD data online-----
//-----//

void savesdon(){
//    File dataFile = SD.open("DataOnLine.txt", FILE_WRITE);
//    dataFile.println("");
//    dataFile.print(sensi[0]); dataFile.print(" , ");
//    dataFile.print(sensorValue2); dataFile.print(" , ");
//    dataFile.print(sensorValue3); dataFile.print(" , ");
//    dataFile.print(sensorValue4); dataFile.print(" , ");
//    dataFile.print(sensorValue5); dataFile.print(" , ");
//    dataFile.print(DHT.humidity, 1); dataFile.print(" , ");
//    dataFile.print(DHT.temperature, 1); dataFile.print(" ,
");
//    dataFile.print(vbat);
//    dataFile.close();
}

//-----
//-----//
//-----kalkulasi NN-----
//-----//

void NN(){

    lcd.setCursor(0,0); lcd.print("    WAIT DATA    ");
    lcd.setCursor(0,1);

```



```

//masukan
Serial.println (" Input ");

lcd2.setCursor(0,0); lcd2.print ("      loading      ");
lcd2.setCursor(0,1); lcd2.print ("                      ");
for( i = 0 ; i < InputNodes ; i++ ) {
    Masukan[i] = sensi[i] / maxi;
    Serial.print ( sensi[i], DEC);
    Serial.print (" ");
    Serial.print ( Masukan[i], DEC);
    Serial.print (" ");
//      dataFile.print(Masukan[i], DEC);      dataFile.print("
, ");
    delay(10);
}
Serial.println ();

/*****
*
* Compute hidden layer activations
*****/
/
for( j = 0 ; j < HiddenNodes ; j++ ) {
    sum2=0;
    for( i = 0 ; i < HiddenNodes ; i++ ) {          //nilai sum
        sum2 += Masukan[i] * BobotInHid[i][j] ;
        delay(10);
    }
    Hidden[j] = 1.0/(1.0 + exp(-(sum2+Bias[j]))); //nilai
sigmoid
}

/*****
*
* Compute output layer
*****/
/
for( k = 0 ; k < OutputNodes ; k++ ) {
    sum2=0;
    for( j = 0 ; j < HiddenNodes ; j++ ) {          //nilai
sum
        sum2 += Hidden[j] * BobotHidOut[j][k] ;
        delay(10);
    }
    Keluaran[k] = 1.0/(1.0 + exp(-(sum2 + Bias[k+8]))); //nilai
sigmoid
}

//luaran
lcd2.print("");
Serial.println (" Output ");
lcd2.setCursor(0,1);
lcd2.print ("                      "); //clear lcd
lcd2.setCursor(0,0);
lcd2.print ("                      "); //clear lcd
lcd2.setCursor(0,0);
for( i = 0 ; i < OutputNodes ; i++ ) {
    Serial.print (Keluaran[i],DEC);

```

```

Serial.print (" ");

lcd2.print(" "); lcd2.print(Keluaran[i],DEC);
lcd2.setCursor(0,1);
//          dataFile.print(sensi[i]);      dataFile.print(" ,
");
}

Serial.println ();
if(ja == 2){
    if( Keluaran[0] >= Keluaran[1]){
        lcd.setCursor(0,1); lcd.print("  DAGING SEGAR
");
    }
    else {
        lcd.setCursor(0,1); lcd.print("DAGING TDK
SEGAR");
    }
}
//reset input, hidden dan luaran
sum2=0;
//for( i = 0 ; i < InputNodes ; i++ ) {
//    sensi[i]=0;
//    Masukan[i]=0;
//}
//    for( j = 0 ; j < HiddenNodes ; j++ ) {
//        Hidden[j]=0;
//    }
//    for( k = 0 ; k < OutputNodes ; k++ ) {
//        Keluaran[k] =0;
//    }
//}
//-----//
//-----DATA NN OFFLINE-----//
void NNoff(){

    lcd.setCursor(0,0); lcd.print("  UJI DAGING  ");
    lcd.setCursor(0,1); lcd.print("  TIDAK LANGSUNG");
    lcd.setCursor(0,1);
        for (a = 0; a <= 16; a += 1) {
            lcd.print("#");
            delay(200);
        }
    lcd.setCursor(0,0); lcd.print(" Load data Input");
    lcd.setCursor(0,1); lcd.print(" ");
    char val;
    for (c = 1; c <= 22; c += 1) {
        d=1;
        val = c;
        if (val==1) {
            sensi[0] = 0.995; sensi[1] = 1.081; sensi[2] = 1.003; sensi[3] =
1.001; sensi[4] = 1.005;
        }
        else if (val==2) {
            sensi[0] = 0.99; sensi[1] = 1.12; sensi[2] = 1.01; sensi[3] =
0.97; sensi[4] = 1.01;

```

```

}
else if (val==3) {
    sensi[0] = 0.961; sensi[1] = 1.067; sensi[2] = 0.976; sensi[3] =
0.99; sensi[4] = 0.964;
}
else if (val==4) {
    sensi[0] = 0.997; sensi[1] = 1.119; sensi[2] = 1.009; sensi[3] =
1.025; sensi[4] = 1.021;
}
else if (val==5) {
    sensi[0] = 0.989; sensi[1] = 1.092; sensi[2] = 0.995; sensi[3] =
0.96; sensi[4] = 1.005;
}
else if (val==6) {
    sensi[0] = 0.992; sensi[1] = 1.085; sensi[2] = 0.995; sensi[3] =
0.999; sensi[4] = 1.001;
}
else if (val==7) {
    sensi[0] = 0.997; sensi[1] = 1.086; sensi[2] = 0.996; sensi[3] =
1.013; sensi[4] = 1.025;
}
else if (val==8) {
    sensi[0] = 1.001; sensi[1] = 1.088; sensi[2] = 1.004; sensi[3] =
1.017; sensi[4] = 1.022;
}
else if (val==9) {
    sensi[0] = 1.001; sensi[1] = 1.089; sensi[2] = 1.013; sensi[3] =
0.991; sensi[4] = 0.998;
}
else if (val==10) { //a 10
    sensi[0] = 1.006; sensi[1] = 1.034; sensi[2] = 1.025; sensi[3] =
1.01; sensi[4] = 0.999;
}
else if (val==11) { //b 11
    sensi[0] = 1.012; sensi[1] = 1.068; sensi[2] = 1.015; sensi[3] =
1.008; sensi[4] = 1.012;
}
else if (val==12) { //c 12
    sensi[0] = 2.92; sensi[1] = 1.15; sensi[2] = 2.83; sensi[3] =
2.46; sensi[4] = 3.70;
}
else if (val==13) { //d 13
    sensi[0] = 1.108; sensi[1] = 1.114; sensi[2] = 1.12; sensi[3] =
0.999; sensi[4] = 1.117;
}
else if (val==14) { //e 14
    sensi[0] = 1.374; sensi[1] = 0.996; sensi[2] = 1.266; sensi[3] =
1.228; sensi[4] = 1.417;
}
else if (val==15) { //f 15
    // sensi[0] = 2.143; sensi[1] = 0.980; sensi[2] = 1.940; sensi[3]
= 1.760; sensi[4] = 2.510;
    sensi[0] = 2.143; sensi[1] = 0.9; sensi[2] = 1.940; sensi[3] =
1.760; sensi[4] = 2.510;
}
else if (val==16) { //g 16
    sensi[0] = 2.279; sensi[1] = 1.298; sensi[2] = 2.294; sensi[3] =
2.24; sensi[4] = 2.639;
}
}

```

```

    else if (val==17) { //h 17
        sensi[0] = 3.589; sensi[1] = 1.109; sensi[2] = 3.419; sensi[3] =
2.922; sensi[4] = 5.091;
    }
    else if (val==18) { //i 18
        sensi[0] = 4.198; sensi[1] = 1.152; sensi[2] = 4.102; sensi[3] =
3.336; sensi[4] = 5.625;
    }
    else if (val==19) { //j 19
        sensi[0] = 3.031; sensi[1] = 1.161; sensi[2] = 2.978; sensi[3] =
2.597; sensi[4] = 3.788;
    }
    else if (val==20) { //k 20
        sensi[0] = 3.48; sensi[1] = 1.07; sensi[2] = 3.43; sensi[3] =
2.86; sensi[4] = 4.31;
    }
    else if (val==21) { //l 21
        sensi[0] = 2.26; sensi[1] = 1.51; sensi[2] = 2.207; sensi[3] =
2.135; sensi[4] = 2.789;
    }
    else if (val==22) { //m 22
        sensi[0] = 3.93; sensi[1] = 1.14; sensi[2] = 3.90; sensi[3] =
3.11; sensi[4] = 5.14;
    }
    // -----data uji segar-----
    -----//
    else if (val==23) { //m
        sensi[0] = 0.702839407; sensi[1] = 0.88890896; sensi[2] =
0.959802318; sensi[3] = 0.899047367; sensi[4] = 0.987455068;
    }
    else if (val==24) { //n
        sensi[0] = 1.145625678; sensi[1] = 1.119375457; sensi[2] =
1.002090354; sensi[3] = 1.140697643; sensi[4] = 1.01452403;
    }
    else if (val==254) { //o
        sensi[0] = 1.019807489; sensi[1] = 1.108160731; sensi[2] =
0.996963351; sensi[3] = 1.121426149; sensi[4] = 1.005300521;
    }
    // -----data uji tidak
segar-----//
    else if (val==26) { //p
        sensi[0] = 1.137050283; sensi[1] = 1.03709916; sensi[2] =
1.006642131; sensi[3] = 1.058695014; sensi[4] = 1.025453744;
    }
    else if (val==27) { //q
        sensi[0] = 1.217713189; sensi[1] = 1.576189945; sensi[2] =
1.606527094; sensi[3] = 1.642779849; sensi[4] = 1.311457964;
    }
    else if (val==28) { //r
        sensi[0] = 1.202653492; sensi[1] = 1.62285434; sensi[2] =
1.778264882; sensi[3] = 1.631823461; sensi[4] = 1.277608188;
    };

    NN();

    lcd.setCursor(0,1); lcd.print ("    DATA KE "); lcd.print (c);
    updatepb();
    if (buttonState == HIGH)    ia++;    if (buttonState2 == HIGH)
ja++;

```



```

if (ja == 2) {
lcd2.setCursor(0,0); lcd2.print(" cancel by user ");
lcd2.setCursor(0,1); lcd2.print(" ");
delay(2000);
homescreen(); ja = 0; ia = 0; c = 22;
}
delay(1500);
}
}

//-----
//-----//
//----- thanks to -----
//-----//

void about(){
lcd.setCursor(0,0); lcd.print(" ");
lcd.setCursor(0,1); lcd.print(" ");
for (c = 16; c >= 0; c -= 1) {
lcd.setCursor(c,0); lcd.print("ALLAH SUBHANAHU ");
lcd.setCursor(c,1); lcd.print(" WA TAALA ");
delay(200);
}
delay(2000);

for (c = 16; c >= 0; c -= 1) {
lcd.setCursor(c,0); lcd.print("AYAH DAN ALM IBU");
lcd.setCursor(c,1); lcd.print(" ");
delay(200);
}
delay(2000);
lcd2.backlight();
for (c = 16; c >= 0; c -= 1) {
lcd.setCursor(c,0); lcd.print(" BUDI GUNAWAN ");
lcd.setCursor(c,1); lcd.print(" ST.,MT. ");
lcd2.setCursor(c,0); lcd2.print(" ARIEF SUDARMAJI");
lcd2.setCursor(c,1); lcd2.print(" ST.,MT.,PhD. ");
delay(200);
}
delay(2000);
for (c = 16; c >= 0; c -= 1) {
lcd.setCursor(c,0); lcd.print(" SOLEKHAN ");
lcd.setCursor(c,1); lcd.print(" ST.,MT. ");
lcd2.setCursor(c,0); lcd2.print(" DAN ");
lcd2.setCursor(c,1); lcd2.print(" DOSEN UMK ");
delay(200);
}
delay(2000);
for (c = 16; c >= 0; c -= 1) {
lcd.setCursor(c,0); lcd.print(" TEMAN TEMAN ");
lcd.setCursor(c,1); lcd.print(" UMK & UNSOED ");
lcd2.noBacklight();
lcd2.setCursor(c,0); lcd2.print(" ");
lcd2.setCursor(c,1); lcd2.print(" ");
delay(200);
}
delay(2000);
}
}

```

BIODATA PENULIS



Nama : Rachmawan Wijaya
Tempat & Tgl Lahir : Kudus, 18 September 1996
Alamat : Mlatinorowito gg1a kudus
Email : rachmawan.wijaya96@gmail.com
No HP : +62 89 885 227 36

Riwayat pendidikan

- SD 1 MLATILOR KUDUS 2002 - 2008
- SMP 2 BAE KUDUS 2008 - 2011
- SMK WISUDHA KARYA KUDUS TEKNIK MEKATRONIKA 2011 - 2014

