



TUGAS AKHIR - TE 141599

**IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN
RENDAH MENGGUNAKAN HARMONIA SECARA REAL
TIME BERBASIS FAST FOURIER TRANSFORM DAN
NEURAL NETWORK**

Naufal Rasyiq
07111340000135

Dosen Pembimbing
Dr. Ir. Margo Pujiantara, MT.
Vita Lystianingrum Budiharto Putri, ST, M.Sc.,Ph.D

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

Load Identification At Low Voltage Using Harmonic In Real Time Based Fast Fourier Transform and Neural Network

Naufal Rasyiq
07111340000135

Advisors
Dr. Ir. Margo Pujiantara, MT.
Vita Lystianingrum Budiharto Putri, ST, M.Sc.,Ph.D

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2018

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Identifikasi Beban Pada Sistem Tegangan Rendah Menggunakan Harmonisa Secara *Real Time* Berbasis *Fast Fourier Transform* dan *Neural Network***" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2018

Naufal Rasyiq
NRP. 07111340000135

Halaman ini sengaja dikosongkan

**IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN
RENDAH MENGGUNAKAN HARMONIA SECARA
*REAL TIME BERBASIS FAST FOURIER
TRANSFORM DAN NEURAL NETWORK***

TUGAS AKHIR

Diajukan untuk Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Teknik Sistem Tenaga
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember

Dosen Pembimbing I



Dr. Ir. Margo Pujiantara, MT
NIP. 196603181990101001

Menyetujui:

Dosen Pembimbing II


18/7/18

Vita Lystianingtum B. P., ST., M.Sc., Ph.D.
NIP. 198208292006042001



\Halaman ini sengaja dikosongkan

**IDENTIFIKASI BEBAN PADA SISTEM TEGANGAN RENDAH
MENGGUNAKAN HARMONISA SECARA *REAL TIME*
BERBASIS *FAST FOURIER TRANSFORM* DAN *NEURAL
NETWORK***

Nama : Naufal Rasyiq
Pembimbing I : Dr. Ir. Margo Pujiantara, MT
Pembimbing II : Vita Lystianingrum Budiharto Putri, ST, M.Sc., Ph.D

ABSTRAK

Semakin meningkatnya penggunaan beban non linear menimbulkan masalah pada sistem tenaga listrik. Beban non linear berpengaruh negatif terhadap sistem tenaga listrik seperti memperpendek usia peralatan dan mempercepat kerusakan peralatan listrik. Bagaimanapun, KWh meter yang terpasang pada pelanggan listrik PLN hanya menampilkan konsumsi energi listrik.

Pada Tugas Akhir ini, dilakukan identifikasi jenis beban menggunakan nilai harmonik arus yang dihasilkan oleh peralatan rumah tangga menggunakan metode *backpropagation neural network*. Proses pengumpulan data beban yang mengandung nilai harmonik dengan proses FFT (*Fast Fourier Transform*) menggunakan *Microcontroller ARM STM32F7* yang dihubungkan dengan 5 jenis beban dipasang secara pararel. Backpropagation neural network akan dilatih untuk menyerap informasi untuk mengidentifikasi jenis beban/perangkat yang sedang digunakan dari nilai harmonik arusnya.

Hasil dari tahap pelatihan ini akan menghasilkan source code yang akan dimasukkan kedalam micro yang sudah terpasang di alat smart meter yang berguna di masa yang akan datang untuk keperluan pemantauan kualitas daya dan bisa mengetahui informasi daya total yang dihasilkan oleh alat rumah tangga tersebut.

Kata kunci: *Fast Fourier Transform, Arus Harmonisa, Neural Network, Backpropagation*

Halaman ini sengaja dikosongkan

LOAD IDENTIFICATION AT LOW VOLTAGE USING HARMONIC IN REAL TIME BASED FAST FOURIER TRANSFORM AND NEURAL NETWORK

Name : Naufal Rasyiq
1st Advisor : Dr. Ir. Margo Pujiantara, MT
2nd Advisor : Vita Lystianingrum Budiharto Putri, ST, M.Sc., Ph.D

ABSTRACT

The increasing use of non-linear loads creates problems with the power system. Non-linear loads negatively affect electrical power systems such as shortening the life of equipment and accelerating electrical equipment damage. However, the KWh meter installed on PLN's electricity customers only displays electrical energy consumption.

In this Final Project, identification of load types using harmonic current values generated by household appliances using backpropagation neural network method. The process of collecting load data containing harmonic values with the FFT (Fast Fourier Transform) process using the ARM STM32F7 Microcontroller which is connected to the 5 types of loads installed in parallel. Backpropagation neural network will be trained to absorb information to identify the type of load / device being used from its harmonic current value.

The results of this training phase will generate source code that will be incorporated into the micro that has been installed in the smart meter tool that is useful in the future for the purpose of monitoring the quality of power and can know the total power information generated by the household appliance.

Keywords: *Fast Fourier Transform, Harmonic Current, Neural Network, Backpropagation*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur Saya panjatkan kepada Allah SWT yang selalu melimpahkan rahmat dan karunia-Nya sehingga Saya dapat menyelesaikan Tugas Akhir ini dengan baik dan tepat waktu.

Tugas akhir yang berjudul "**Identifikasi Beban Pada Sistem Tegangan Rendah Menggunakan Harmonisa Secara Real Time Berbasis Fast Fourier Transform Dan Neural Network**" ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan sarjana pada bidang studi Teknik Sistem Tenaga, departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember, Surabaya.

Pelaksanaan dan penyelesaian tugas akhir ini tidak lepas dari bantuan berbagai pihak. Untuk itu Saya mengucapkan terima kasih sebesar-besarnya kepada:

1. Kedua orangtua yang selalu memberikan dukungan moral dan materiil. Terkhusus mama, terimakasih atas nasehat dan semangat yang diberikan.
2. Bapak Dr. Ir. Margo Pujiyantara, MT. dan Ibu Vita Lystianingrum Budiharto Putri, ST, M.Sc., Ph.D selaku dosen pembimbing atas segala ilmu dan waktu yang telah diberikan untuk membimbing saya hingga terselesaiannya tugas akhir ini.
3. Bapak Dr. Ir. Margo Pujiyantara, MT. selaku kepala Laboratorium Tegangan Tinggi atas peminjaman fasilitas yang ada di laboratorium untuk mendukung terselesaiannya Tugas Akhir ini.
4. Mas Dimas Okky, sebagai mentor saya dalam penggerjaan tugas akhir ini.
5. Teman – teman angkatan e53, badminton ITS, dan pihak – pihak lainnya yang membantu penggerjaan TA ini yang tidak dapat Saya sebutkan satu-persatu.

Akhir kata, penulis berharap tugas akhir ini dapat bermanfaat bagi siapapun yang ingin mendalami ilmu mengenai identifikasi beban ini. Semoga bermanfaat.

Surabaya, Juli 2018

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

HALAMAN JUDUL

LEMBAR PERNYATAAN

LEMBAR PENGESAHAN

ABSTRAK	i
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi

BAB I. PENDAHULUAN

1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan	2
1.4. Batasan Masalah	2
1.5. Metodologi	2
1.6. Sistematika Penulisan	3
1.7. Manfaat	3

BAB II. TEORI PENUNJANG

2.1. Harmonisa	5
2.1.1. Dampak Harmonisa	5
2.1.2. Standard Distorsi Harmonisa	5
2.2. Transformasi Fourier	7
2.2.1. Algoritma Diskrit Fourier Transform	7
2.2.2. Algortima Fast Fourier Transform	9
2.3. Neural Network	10
2.3.1 <i>Backpropagation</i>	11
2.3.1.1 Algoritma <i>Backpropagation</i>	12
2.4. Microcontroller STM32F7	14

BAB III. METODE PENELITIAN

3.1. Sistem Identifikasi Beban	17
3.2. Perencanaan.....	18
3.2.1. Pengambilan Data	18
3.2.2. Pemodelan.....	21

3.2.3. Identifikasi/Klasifikasi	22
3.2.4. Informasi beban listrik yang terpasang	22
BAB IV. HASIL PERCOBAAN DAN ANALISIS DATA	
4.1. Aarsitektur <i>Neural Netwok</i>	25
4.1.1. Input Layer.....	25
4.1.2. Hidden Layer	25
4.1.3. Output Layer (Target)	25
4.2. Parameter Pengujian Neural Network	25
4.3. Tahap Pembelajaran	26
4.3.1. Variasi Jumlah Input Harmonisa.....	29
4.3.2. Variasi Jumlah <i>Neuron Hidden Layer</i>	32
4.3.3. Variasi Jumlah <i>Epochs</i>	36
4.4. Tahap Pengujian	40
4.4.1. Variasi Jumlah Input	40
4.4.2. Variasi Jumlah <i>Neuron Hidden Layer</i>	41
4.4.3. Variasi Jumlah <i>Epochs</i>	43
4.5. Hasil Uji Coba Alat Identifikasi Beban Secara <i>Real Time</i>	47
BAB V. PENUTUP	
5.1. Kesimpulan	51
5.2. Saran	51
DAFTAR PUSTAKA	53
RIWAYAT HIDUP PENULIS	55

DAFTAR GAMBAR

Gambar 2.1.	Arsitektur <i>Backpropagation</i>	12
Gambar 2.2.	Microcontroller ST32F7	15
Gambar 3.1	Desain sistem identifikasi beban	17
Gambar 3.2	Tahap 1: pengambilan data	18
Gambar 3.3	Contoh hasil pengambilan harmonisa beban	20
Gambar 3.4	Tahap 2 : Pemodelan	21
Gambar 3.5	Tahap 3 : Identifikasi/Klasifikasi	22
Gambar 3.6	Tahap 4 : Informasi beban listrik sedang terpasang ...	23
Gambar 4.1	Gelombang dan <i>spectrum</i> harmonisa arus pada beban AC	26
Gambar 4.2	Gelombang dan <i>spectrum</i> harmonisa arus pada beban kipas angin	26
Gambar 4.3	Gelombang dan <i>spectrum</i> harmonisa arus pada beban lampu	26
Gambar 4.4	Gelombang dan <i>spectrum</i> harmonisa arus pada beban laptop	27
Gambar 4.5	Gelombang dan <i>spectrum</i> harmonisa arus pada beban TV	27
Gambar 4.6	Gelombang dan <i>spectrum</i> harmonisa arus pada beban AC dan lampu	27
Gambar 4.7	Gelombang dan <i>spectrum</i> harmonisa arus pada beban AC dan laptop	28
Gambar 4.8	Gelombang dan <i>spectrum</i> harmonisa arus pada beban AC dan TV.....	28
Gambar 4.9	Gelombang dan <i>spectrum</i> harmonisa arus pada beban kipas angin dan AC	28
Gambar 4.10	Tampilan awal alat	47
Gambar 4.11	Tampilan menu <i>Load Measure</i>	48
Gambar 4.12	Tampilan informasi beban yang sedang beroperasi (TV)	49
Gambar 4.13	Tampilan informasi beban yang sedang beroperasi (TV dan Lampu Hemat Energi)	49

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 2.1	Batas distorsi arus (dalam % I_L) untuk sistem distribusi (level tegangan 120-69.000 V) IEEE std. 519-1992.....	6
Tabel 2.2	Batas distorsi arus (dalam % I_L) untuk sistem subtransmisi (level tegangan 69.001-161.000 V) IEEE std. 519-1992.....	7
Tabel 2.3	Batas distorsi arus (dalam % I_L) untuk sistem transmisi (level tegangan >161.000 V) IEEE std. 519-1992.....	7
Tabel 3.1	Kombinasi Beban.....	19
Tabel 4.1	Hasil pembelajaran dengan Jumlah Input Harmonisa	3
Tabel 4.2	Hasil Pembelajaran dengan Jumlah Input Harmonisa	4
Tabel 4.3	Hasil Pembelajaran dengan Jumlah Input Harmonisa	5
Tabel 4.4	Hasil Pembelajaran dengan Jumlah <i>Neuron</i> masing-masing <i>Layer</i> 5-10-1	33
Tabel 4.5	Hasil Pembelajaran dengan Jumlah <i>Neuron</i> masing-masing <i>Layer</i> 10-15-1	34
Tabel 4.6	Hasil Pembelajaran dengan Jumlah <i>Neuron</i> masing-masing <i>Layer</i> 15-20-1	35
Tabel 4.7	Hasil Pembelajaran dengan Jumlah <i>epochs</i> 100	36
Tabel 4.8	Hasil Pembelajaran dengan Jumlah <i>epochs</i> 250	38
Tabel 4.9	Hasil Pembelajaran dengan Jumlah <i>epochs</i> 500	39
Tabel 4.10	Hasil Pengujian Identifikasi Beban dengan Variasi Jumlah Input Harmonisa.....	40
Tabel 4.11	Hasil Pengujian Identifikasi Beban dengan Variasi Jumlah <i>Neuron</i> masing-masing <i>Layer</i>	42
Tabel 4.12	Hasil Pengujian Identifikasi Beban dengan Jumlah <i>Epochs</i> dan 3 angka ganjil pertama harmonisa.....	43
Tabel 4.13	Hasil Pengujian Identifikasi Beban dengan Jumlah <i>Epochs</i> dan 4 angka ganjil pertama harmonisa.....	45
Tabel 4.14	Hasil Pengujian Identifikasi Beban dengan Jumlah <i>Epochs</i> dan 5 angka ganjil pertama harmonisa.....	46

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Jumlah penggunaan beban pada rumah tangga, industri dan komersial yang meningkat sangat pesat diikuti oleh penggunaan peralatan – peralatan elektronik yang semakin banyak. Peralatan elektronik seperti komputer, televisi, lampu LHE, konverter merupakan beban – beban non linier yang menghasilkan sumber harmonisa. Harmonisa penting diperhatikan karena mempengaruhi keamanan dan kualitas daya pada sistem tenaga listrik seperti penurunan kualitas daya, terdapat arus pada fase netral, meningkatkan temperatur pada kabel, meningkatkan rugi – rugi daya, salah ukur pada KWH-meter, dan menurunnya *life time* peralatan. Harmonisa memiliki karakteristik yang berbeda pada beban non linier. Oleh karena itu, harmonisa dapat digunakan untuk identifikasi beban dalam sistem tenaga listrik [1-4].

Pada standar IEEE 519.1992 (*Recommended Practices and Requirements for Harmonic Control in Electrical Power System*) sudah ditetapkan bahwa susunan batasan level yang diterima dari harmonisa dalam sistem tenaga listrik. Standar ini menjamin bahwa pelanggan daya listrik memiliki peran penting dalam menjaga level harmonik yang tinggi [5].

Metode analisis harmonisa yang sering digunakan oleh alat ukur yang terdapat dipasaran adalah *fast fourier transform* (FFT). Metode FFT sering digunakan dalam alat pengukuran harmonisa karena komputasi lebih cepat dan mudah dalam implementasi.

Alat ukur yang terdapat dipasaran pada umumnya hanya untuk mendekripsi harmonisa tetapi tidak bisa digunakan untuk identifikasi beban yang terpasang. Oleh karena itu, dalam Tugas Akhir ini mengembangkan prototipe yang dapat mengidentifikasi beban menggunakan harmonisa. Metode yang digunakan untuk mendapatkan harmonisa adalah FFT.

Identifikasi beban dalam penelitian tugas akhir ini dikembangkan dalam bentuk *prototype* yang terdiri dari *microkontroller*, sensor arus, sensor tegangan dan display yang mampu menganalisis harmonisa dan mengidentifikasi pemakaian beban. Model Artificial Neural Network (ANN) yang digunakan dalam identifikasi beban adalah backpropagation neural network. Nilai amplitudo harmonik dari orde ke 1 sampai orde ke

15 digunakan *backpropagation neural network* untuk identifikasi beban.[6-7].

Hasil dari penelitian pada tugas akhir ini diharapkan prototipe dapat bekerja secara *real time* dan memiliki akurasi yang baik dalam pendekripsi harmonisa dan identifikasi beban.

1.2 Perumusan Masalah

Dari latar belakang yang sudah dipaparkan, permasalahan yang dibahas dalam usulan tugas akhir ini adalah :

1. Bagaimana cara melakukan proses transformasi dari sinyal daerah waktu ke daerah frekuensi menggunakan metode *Fast Fourier Transform* ?
2. Bagaimana cara melakukan identifikasi pemakaian beban menggunakan metode neural network ?
3. Bagaimana cara membuat *prototype* identifikasi pemakaian beban secara *real time* ?

1.3 Tujuan

Sesuai dengan latar belakang permasalahan yang telah diuraikan, maka tujuan tugas akhir ini adalah mengembangkan *prototype* identifikasi beban menggunakan metode *Fast Fourier Transform* dan neural network secara *real time*.

1.4 Batasan Masalah

Batasan masalah yang berlaku untuk tugas akhir ini antara lain:

1. Beban yang dipakai merupakan beban nonlinear peralatan rumah tangga sistem tegangan rendah.
2. Beban nonlinear yang dipakai adalah AC, TV, Kipas Angin, Laptop ASUS, dan Lampu Hemat Energi.
3. Pengambilan data beban dan harmonisa arus menggunakan microcontroller ST32F7
4. Komputasi backpropagation neural network menggunakan software Matlab 2014b

1.5 Metodologi

Pada tahap pertama studi literatur, mempelajari dan membaca referensi - referensi terkait yang menunjang pengerjaan tugas akhir. Setelah itu, menganalisis harmonisa melalui simulasi dan *prototype*

menggunakan metode *Fast Fourier Transform* dan di uji menggunakan *power quality analyzer*, apabila akurasi yang didapatkan tidak baik maka dilakukan perbaikan. Tahap ketiga, yaitu pembuatan neural network dari data yang sudah didapatkan. Input *neural network* adalah harmonisa sedangkan outputnya adalah jenis beban. Apabila dari pembuatan program *neural network* berhasil maka bisa dilanjutkan ke tahap selanjutnya yaitu pembuatan *prototype*. Setelah *prototype* selesai, maka perlu pengujian dan analisis agar alat tersebut untuk menunjukkan performa dan akurasi *prototype* yang dibuat. Jika dalam *prototype* tersebut terdapat masalah maka kembali pada proses pertama yaitu studi literatur untuk mengevaluasi kesalahan pada alat. Jika pada pengujian dan analisis berhasil maka bisa dilanjutkan ke tahap terakhir yaitu pembuatan buku.

1.6 Sistematika Penulisan

Sistematika penulisan dalam tugas akhir ini terdiri dari lima bab yang berisi tentang:

1. **BAB I : Pendahuluan**
Berisi latar belakang pembuatan alat tugas akhir, permasalahan, pembatasan masalah, maksud dan tujuan tugas akhir, metodologi, sistematika penulisan serta manfaat tugas akhir.
2. **BAB II : Teori Penunjang**
Bab ini secara garis besar membahas tentang harmonisa, dan Neural Network.
3. **BAB III : Pengumpuan dan Pengolahan Data**
Bab ini berisi tentang pengolahan data yang diperoleh yaitu data harmonik arus sebagai *input* data *training* dan *testing*.
4. **BAB IV : Simulasi dan validasi**
Bab ini membahas tentang hasil training dan identifikasi jenis beban melalui harmonik arus menggunakan metode *backpropagation neural network*
5. **BAB V : Kesimpulan dan Saran**
Berisikan tentang kesimpulan atas hasil yang diperoleh pada tugas akhir ini. Serta berisi saran-saran untuk penelitian dan pengembangan lebih lanjut.

1.7 Manfaat Tugas Akhir

Manfaat yang dapat diperoleh dari tugas akhir ini adalah sebagai berikut:

1. Dapat mengetahui apa itu harmonisa arus, serta penyebab harmonisa arus dan akibat yang ditimbulkannya.
2. Dapat memahami algoritma *backpropagation neural network* untuk diterapkan pada sistem tenaga secara umum dan identifikasi jenis beban berdasarkan arus harmonik secara *Real Time*.

BAB II

TEORI PENUNJANG

2.1 Harmonisa

Harmonisa adalah gangguan yang terjadi pada sistem distribusi tenaga listrik akibat terjadinya distorsi gelombang arus dan tegangan. Harmonisa bisa diartikan terbentuk dari gelombang - gelombang dengan frekuensi berbeda yang merupakan perkalian bilangan bulat dengan frekuensi dasarnya. Bilangan bulat menunjukkan urutan frekuensi. Contohnya adalah frekuensi dasar suatu sistem tenaga listrik adalah 50 Hz maka harmonisa ketiganya adalah gelombang dengan frekuensi sebesar 150 Hz, harmonisa kelima adalah gelombang dengan frekuensi sebesar 250 Hz dan seterusnya. Gelombang-gelombang ini kemudian bercampur pada gelombang murni / aslinya sehingga terbentuk gelombang yang terdistorsi yang merupakan hasil penjumlahan antara gelombang murni sesaat dengan gelombang harmonisanya [1].

Pada sistem tenaga listrik, penggunaan peralatan elektronika daya dan beban nonlinear merupakan penyebab utama dari masalah distorsi gelombang oleh harmonisa. Umumnya distorsi gelombang arus dan tegangan tidak terjadi pada saluran transmisi, pada saluran transmisi presentasi distorsi gelombang dapat terjaga hingga kurang dari 1 % [8]. Akan tetapi pada jaringan distribusi, penggunaan beban nonlinear untuk komersial maupun industri semakin meningkat sehingga meningkat pula distorsi gelombang arus dan tegangan oleh harmonisa. Dalam kondisi tersebut nilai THD (*Total Harmonic Distortion*) yang tinggi dapat berbahaya bagi sistem tenaga listrik [9].

2.1.1 Dampak Harmonisa

Harmonik disebabkan oleh beban – beban non linier yang terpasang di sistem tenaga listrik. Peralatan elektronik yang termasuk beban non linier merupakan kontribusi utama dari harmonik dalam sistem tenaga listrik. Peralatan ini biasanya dimodelkan sebagai sumber arus yang menginjekkan harmonisa arus ke dalam sistem tenaga. Beberapa contoh dari beban non linier adalah komputer, televisi dan lampu hemat energi.

2.1.2 Standard Distorsi Hamonisa

Indikator distorsi harmonisa yang umum digunakan adalah THD (*Total Harmonic Distortion*) yang berelasi dengan gelombang tegangan.

THD didefinisikan sebagai prosentase *root mean square* (r.m.s.) dari nilai urutan harmonisa dengan nilai *root mean square* (r.m.s.) *fundamental* dan biasanya nilai berupa persen sesuai persamaan berikut :

$$THDv = \frac{\sqrt{\sum_{n=2}^N v_n^2}}{V_1} \times 100\% \quad (2.1)$$

Keterangan :

V_n adalah besarnya tegangan harmonisa orde n

N adalah orde harmonisa tertinggi yang dianalisa

V_1 adalah tegangan fasa r.m.s. pada frekuensi dasar

$$THDi = \frac{\sqrt{\sum_{n=2}^N i_n^2}}{i_1} \times 100\% \quad (2.2)$$

Keterangan :

i_n adalah besarnya arus harmonisa orde n

N adalah orde harmonisa tertinggi yang dianalisa

i_1 adalah arus fasa pada frekuensi dasar

Indeks ini digunakan untuk mengukur penyimpangan bentuk gelombang pada satu periode akibat adanya distorsi. Selain menggunakan THD_i dan THD_v untuk mengukur harmonisa juga bisa menggunakan TDD yaitu perbandingan nilai rms antara komponen arus harmonisa dengan arus beban maksimum.

Tabel 2.1 Batas distorsi arus (dalam % I_L) untuk sistem distribusi (level tegangan 120-69.000 V) IEEE std. 519-1992

I_L	<11	$11 \leq h < 17$	$17 \leq h < 23$	$23 \leq h < 35$	$35 \geq h$	TDD
<20	4.0	2.0	1.5	0.6	0.3	5.0
20-50	7.0	3.5	2.5	1.0	0.5	8.0
50-100	10.0	4.5	4.0	1.5	0.7	12.0
100-1000	12.0	5.5	5.0	2.0	1.0	15.0
>1000	15.0	7.0	6.0	2.5	1.4	20.0

Tabel 2.2 Batas distorsi arus (dalam % I_L) untuk sistem sub transmisi (level tegangan 69.001-161.000 V) IEEE std. 519-1992

I_L	<11	$11 \leq h < 17$	$17 \leq h < 23$	$23 \leq h < 35$	$35 \geq h$	TDD
<20	2.0	2.0	0.75	0.30	0.15	2.5
20-50	3.5	3.5	1.25	0.50	0.25	4.0
50-100	5.0	4.5	2.00	0.75	0.35	6.0
100-1000	6.0	5.5	2.50	1.00	0.50	7.5
>1000	7.5	7.0	3.00	1.25	0.70	10.0

Tabel 2.3 Batas distorsi arus (dalam % I_L) untuk sistem transmisi (level tegangan >161.000 V) IEEE std. 519-1992

I_L	<11	$11 \leq h < 17$	$17 \leq h < 23$	$23 \leq h < 35$	$35 \geq h$	TDD
<50	2.0	1.0	0.75	0.30	0.15	2.50
≥ 50	3.0	1.5	1.15	0.45	0.22	3.75

2.2 Transformasi Fourier

Transformasi Fourier berasal dari nama Joseph Fourier, merupakan suatu perhitungan transformasi dari domain waktu ke dalam domain frekuensi. Transformasi Fourier dapat ditulis sebagai berikut :

$$\mathcal{F}(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (2.3)$$

Transformasi Fourier ini digunakan untuk sinyal waktu kontinyu. Keluaran dari Transformasi Fourier ini merupakan sebuah sinyal yang di plot pada daerah frekuensi.

2.2.1 Algoritma Diskrit Fourier Transform

DFT (*Discrete Fourier Transform*) merupakan Transformasi Fourier pada daerah waktu diskrit. Transformasi ini digunakan untuk pengolahan sinyal digital yang menghasilkan jumlah data sebanyak n data hasil pencuplikan. Persamaan dari DFT dapat ditulis sebagai berikut :

$$X[m] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi\frac{mn}{N}} \quad (2.4)$$

Untuk :

$$\begin{aligned} m &= 0, 1, 2, 3, \dots, \\ n &= 0, 1, 2, 3, \dots, N-1 \end{aligned}$$

Pada umumnya $X[m]$ merupakan bentuk dari bilangan kompleks, sehingga dapat di ekspresikan sebagai berikut :

$$X[m] = \operatorname{Re}\{X[m]\} + \operatorname{Im}\{X[m]\} \quad (2.5)$$

Dengan menggunakan entitas euler, maka bentuk eksponensial dapat ditulis sebagai :

$$e^{-j2\pi\frac{mn}{N}} = \cos\frac{2\pi mn}{N} - j\sin\frac{2\pi mn}{N} \quad (2.6)$$

Persamaan DFT diatas dapat dituliskan kembali sebagai berikut :

$$\begin{aligned} X[m] &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi\frac{mn}{N}} \\ &= \sum_{n=0}^{N-1} x[n] \cos\frac{2\pi mn}{N} - j \sum_{n=0}^{N-1} x[n] \sin\frac{2\pi mn}{N} \end{aligned} \quad (2.7)$$

Untuk :

$$m = 0, 1, 2, 3, \dots,$$

$$n = 0, 1, 2, 3, \dots, N-1$$

Untuk $n=0$, maka nilai dari $X[m]=x[0]$. Kemudian bagian nyata $\operatorname{Re}\{X[m]\}$ dapat dihitung dari :

$$\operatorname{Re}\{X[m]\} = x[0] + \sum_{n=0}^{N-1} x[n] \cos\frac{2\pi mn}{N} \quad (2.8)$$

Untuk :

$$m = 0, 1, 2, 3, \dots,$$

$$n = 0, 1, 2, 3, \dots, N-1$$

dan untuk bagian imaginer $\operatorname{Im}\{X[m]\}$ dapat dihitung dari :

$$\operatorname{Im}\{X[m]\} = - \sum_{n=0}^{N-1} x[n] \sin\frac{2\pi mn}{N} \quad (2.9)$$

Untuk :

$$m = 0, 1, 2, 3, \dots,$$

$$n = 0, 1, 2, 3, \dots, N-1$$

karena bentuk dari $e^{-\frac{j2\pi}{N}}$ adalah vector yang selalu berotasi dari jangkauan $0 \leq \theta \leq 2\pi$, dibagi kedalam $360/N$ bagian yang sama. Oleh karena itu dapat didefinisikan ulang dengan W_N , dimana:

$$W_N = e^{-\frac{j2\pi}{N}} \quad (2.10)$$

Sehingga persamaan DFT sekarang menjadi :

$$X[m] = \sum_{n=0}^{N-1} x[n] W_N^{mn} \quad (2.11)$$

Untuk :

$$m = 0, 1, 2, 3, \dots$$

$$n = 0, 1, 2, 3, \dots, N-1$$

2.2.2 Algoritma Fast Fourier Transform

Fast Fourier Transform (FFT) merupakan sebuah metode untuk melakukan transformasi fourier dengan jumlah perhitungan yang lebih sedikit dibandingkan dengan DFT. FFT memiliki 2 jenis yaitu *decimation in time* dan *decimation in frequency*. Decimation in time merupakan algoritma FFT yang digunakan untuk mendapatkan sinyal pada daerah frekuensi dari daerah waktu, sedangkan untuk decimation in frequency lebih dikenal sebagai IFFT, yaitu mendapatkan sinyal pada daerah waktu dari daerah frekuensi atau kebalikan dari FFT. Algoritma tersebut dikembangkan berdasarkan metode dekomposisi matriks dimana W_N difaktorkan menjadi L matriks yang lebih kecil.

$$[W_N] = [W_0]. [W_1]. \dots . [W_L] \quad (2.12)$$

Dimana L dipilih dari $L = \log_2 N$ atau $N = 2^L$

Fungsi DFT dibagi menjadi 2, yaitu fungsi genap dan fungsi ganjil. Dimana fungsi genap adalah

$$F[m]_{genap} = \sum_{n=0}^{\frac{N}{2-l}} x[2n] W_N^{2nm} \quad (2.13)$$

Untuk :

$$\begin{aligned}m &= 0, 1, 2, 3, \dots, \\n &= 0, 1, 2, 3, \dots, N-1\end{aligned}$$

Fungsi Ganjil adalah

$$F[m]_{ganjil} = \sum_{n=0}^{\frac{N}{2-1}} x[2n+1]W_N^{(2n+1)m} \quad (2.14)$$

$$F[m]_{ganjil} = W_N^m \sum_{n=0}^{\frac{N}{2-1}} x[2n+1]W_N^{2nm} \quad (2.15)$$

Untuk :

$$\begin{aligned}m &= 0, 1, 2, 3, \dots, \\n &= 0, 1, 2, 3, \dots, N-1\end{aligned}$$

Dari persamaan diatas maka dapat disusun ulang sebagai berikut :

$$F[m] = F[m]_{genap} + F[m]_{ganjil} \quad (2.15)$$

$$F[m] = F[m]_{genap} + W_N^m F[m]_{ganjil} \quad (2.16)$$

Dengan menggunakan algoritma FFT, maka pengerjaan operasi yang dilakukan untuk menghitung spectrum frekuensi dapat direduksi secara algoritmik. Sehingga semakin banyak sample yang dianalisa, maka perbandingan operasi DFT dan FFT akan semakin kecil.

2.3 Neural Network

Jaringan syaraf tiruan (JST) atau yang biasa disebut *Artificial Neural Network* (ANN) atau *Neural Network* (NN) saja, merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf pada makhluk hidup. Neural network berupa suatu model sederhana dari suatu syaraf nyata dalam otak manusia seperti suatu unit threshold yang biner.

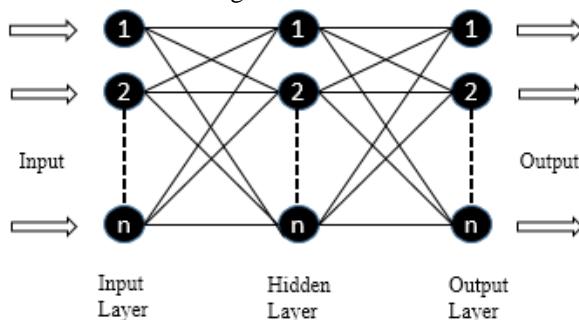
Neural network merupakan sebuah mesin pembelajaran yang dibangun dari sejumlah elemen pemrosesan sederhana yang disebut neuron atau node. Setiap neuron dihubungkan dengan neuron yang lain dengan hubungan komunikasi langsung melalui pola hubungan yang disebut arsitektur jaringan. Bobot-bobot pada koneksi mewakili besarnya informasi yang digunakan jaringan. Metode yang digunakan untuk menentukan bobot koneksi tersebut dinamakan dengan algoritma pembelajaran. Setiap neuron mempunyai tingkat aktivasi yang

merupakan fungsi dari input yang masuk padanya. Aktivasi yang dikirim suatu neuron ke neuron lain berupa sinyal dan hanya dapat mengirim sekali dalam satu waktu, meskipun sinyal tersebut disebarluaskan pada beberapa neuron yang lain.

2.3.1 Backpropagation

Salah satu metode yang digunakan dalam JST adalah *Backpropagation*. *Backpropagation* adalah algoritma pembelajaran untuk memperkecil tingkat *error* dengan cara menyesuaikan bobotnya berdasarkan perbedaan *output* dan target yang diinginkan. *Backpropagation* juga merupakan sebuah metode sistematis untuk pelatihan multilayer JST karena *Backpropagation* memiliki tiga *layer* dalam proses pelatihannya, yaitu *input layer*, *hidden layer* dan *output layer*, dimana *backpropagation* ini merupakan perkembangan dari single *layer network* (Jaringan Layar Tunggal) yang memiliki dua *layer*, yaitu *input layer* dan *output layer*. Dengan adanya *hidden layer* pada *backpropagation* dapat menyebabkan tingkat *error* pada *backpropagation* lebih kecil dibanding tingkat *error* pada single *layer network*. karena *hidden layer* pada *backpropagation* berfungsi sebagai tempat untuk meng-*update* dan menyesuaikan bobot, sehingga didapatkan nilai bobot yang baru yang bisa diarahkan mendekati dengan target *output* yang diinginkan.

Arsitektur algoritma *backpropagation* terdiri dari tiga *layer*, yaitu *input layer*, *hidden layer* dan *output layer*. Pada *input layer* tidak terjadi proses komputasi, namun terjadi pengiriman sinyal *input X* ke *hidden layer*. Pada *hidden* dan *output layer* terjadi proses komputasi terhadap bobot dan bias dan dihitung pula besarnya *output* dari *hidden* dan *output layer* tersebut berdasarkan fungsi aktivasi tertentu.



Gambar 2.1 Arsitektur *Backpropagation*

2.3.1.1 Algoritma Backpropagation

Algoritma backpropagation merupakan algoritma untuk memperkecil error dengan menyesuaikan bobot berdasarkan input dan target yang diinginkan.[11]

Algoritma pembelajaran NN :

1. Inisiasi bobot
 2. Proses feed forward dan backpropagation
- Komputasi maju (feed forward) digunakan sebagai algoritma untuk menghitung nilai aktivasi yang ada pada semua neuron baik yang ada di lapis tersembunyi atau hidden layer taupun lapis keluaran atau output layer.

Algoritma proses feed forward :

Unit input (X_i , $i=1,2,\dots,n$)

Menerima input X_i

Mengirimkannya ke semua unit layer diatasnya (Hidden layer)

Unit Hidden (Z_j , $j=1,2,\dots,n$)

Menghitung semua sinyal input dengan bobotnya :

$$z_inj = v_{oj} + \sum_{i=1}^n x_i v_{ij} \quad (2.17)$$

Menghitung nilai aktivasi setiap unit hidden sebagai output unit hidden :

$$zj = f(z_inj) \quad (2.18)$$

bila yang digunakan adalah fungsi sigmoid maka bentuk fungsi tersebut adalah:

$$Z_j = \frac{1}{1+\exp(-z_inj)} \quad (2.19)$$

Mengirim nilai aktivasi sebagai input untuk unit output. Unit Output (Y_k , $k=1,2,\dots,n$)

$$y_{in_k} = w_{0k} + \sum_{j=1}^p Z_j w_{jk} \quad (2.20)$$

Menghitung nilai aktivasi setiap unit output sebagai output jaringan.

$$y_k = f(y_{in_k}) \quad (2.21)$$

- *Backpropagation* merupakan sebuah algoritma yang berfungsi untuk melakukan perhitungan balik dari *neuron* keluaran agar memiliki nilai bobot yang sesuai dalam jaringan *neural network*. Dengan komputasi balik ini nilai error atau kesalahan bisa dikurangi dengan cukup baik.

Algoritma proses *backpropagation* :

Unit Output (Y_k , $k=1,2,\dots,m$)

Menerima pola target yang bersesuaian dengan pola input

Menghitung informasi error :

$$\delta_k = (t_k - y_k)f'(y_{in_k}) \quad (2.22)$$

Menghitung besarnya koreksi bobot unit output :

$$f(y_{in_k}) = \frac{1}{1+e^{-y_{in_k}}} \quad (2.23)$$

Menghitung besarnya koreksi bias output :

$$\Delta w_{0k} = \alpha \delta_k \quad (2.24)$$

Mengirimkan δ_k ke unit-unit yang ada pada layer di bawahnya, yaitu ke hidden layer.

Unit Hidden (Z_j , $j=1,2,\dots,p$)

Menghitung semua koreksi error :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.23)$$

Menghitung nilai aktivasi koreksi error :

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.24)$$

Menghitung koreksi bobot unit hidden :

$$\Delta v_{0j} = \alpha \delta_j x_i \quad (2.25)$$

Menghitung koreksi error bias unit hidden :

$$\Delta v_{0j} = \alpha \delta_j \quad (2.26)$$

Update bobot dan bias

Unit Output (Y_k , $k = 1, 2, \dots, m$)

Mengupdate bobot dan biasnya ($j = 0, \dots, p$) :

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \quad (2.27)$$

Unit lapis tersembunyi (Z_j , $j = 1..p$) melakukan *update* / perbaikan bias dan bobotnya :

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \quad (2.28)$$

3. Stopping

Terdapat dua kondisi stopping pada algoritma backpropagation ini, yaitu:

1. Error < Error maksimum

Error adalah perbedaan yang terjadi antara output terhadap target yang diinginkan. Proses ANN akan berhenti jika besarnya error yang terjadi telah bernilai lebih kecil dari nilai error maksimum yang telah ditetapkan. Besarnya nilai error dihitung dengan menggunakan fungsi error kuadratis.

2. Fungsi error tersebut merupakan bagian dari algoritma backpropagation yang menggunakan gradient descent yaitu dengan cara menuruni lembah permukaan error untuk mencapai nilai error yang convergen.

3. Epoch > Epoch maksimum

Epoch adalah suatu langkah yang dilakukan dalam pembelajaran pada ANN. Jika besarnya epoch lebih besar dari besarnya epoch maksimum yang telah ditetapkan, maka proses pembelajaran akan berhenti.

Kedua kondisi stopping di atas digunakan dengan logika OR. Jadi kondisi stopping adak terjadi jika besarnya Error < Error maksimum atau Epoch > Epoch maksimum.

2.4 Microcontroller STM32F7

MCU STM32F7 merupakan anggota ST STM32 MCU dengan tingkat kinerja tertinggi dan menggabungkan inti Cortex-M7 dengan

periferal canggih untuk memberikan peluang yang tak tertandingi untuk meningkatkan kinerja aplikasi, menambah fungsi baru, memperpanjang masa pakai baterai dan memastikan keamanan, serta meminimalkan penggunaan komponen eksternal untuk menghemat biaya dan ukuran.[12]

ARM Cortex-M7 memiliki keunggulan yang lebih *core* sebelumnya seperti *Cortex-M4*, seri MCU STM32F7 memiliki kemampuan DSP (*Digital Signal Processor*) dua kali lipat, membuat MCU ideal untuk aplikasi yang membutuhkan audio berkecepatan tinggi atau multi-channel, video, nirkabel, pengenalan gerak atau kontrol motor. Selain itu, arsitekturnya sangat fleksibel, seri ini memiliki matriks bus *AXI* dan *multi-AHB* untuk interkoneksi inti, periferal dan memori. Pemilihan periferal canggih mencakup fitur-fitur seperti rel listrik khusus untuk *OTG USB*, yang memungkinkan konektivitas *USB* untuk terus beroperasi sementara sisa chip ini didukung pada 1.8V untuk menghemat daya. Dukungan jam ganda pada sebagian besar periferal juga memungkinkan kecepatan CPU dikurangi untuk meminimalkan konsumsi daya, sementara menjaga frekuensi clock tidak berubah pada periferal komunikasi.



Gambar 2.2 Microcontroller ST332F7

Sumber : <http://www.connectingindustry.com/electronics/rs-components-announces-availability-of-new-stmicroelectronics-stm32f7-microcontroller-.aspx>

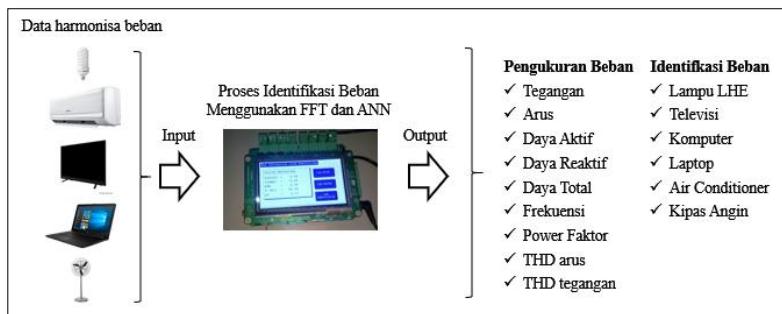
Halaman ini sengaja dikosongkan

BAB III

METODE PENILITIAN

3.1 Sistem Identifikasi Beban

Pada penelitian ini, desain sistem identifikasi beban yang akan dibuat dapat digambarkan sebagai berikut:



Gambar 3.1 Desain sistem identifikasi beban

Sebelum melakukan proses identifikasi beban diperlukan data beban non-linear yang banyak digunakan dalam rumah tangga yang mengandung harmonika.

Selanjutnya, dari data tersebut digunakan untuk pembelajaran terhadap jenis beban menggunakan ANN (*Artificial Neural Network*). Berikutnya menentukan desain jaringan aksitektur Neural Network. Pada tahap ini, dilakukan penentuan pola masukan dan keluaran. Pola masukan menggunakan data harmonika orde ke 1 sampai ke 7. Sedangkan, pola keluaran adalah jenis beban. Pola masukan dan keluaran digunakan untuk keperluan kegiatan pelatihan dan pengujian pada jaringan saraf tiruan. Proses pelatihan menggunakan Matlab. Hasil proses pelatihan didapatkan model Neural Network yang akan menghasilkan source code untuk Microcontroller ARM ST32F7.

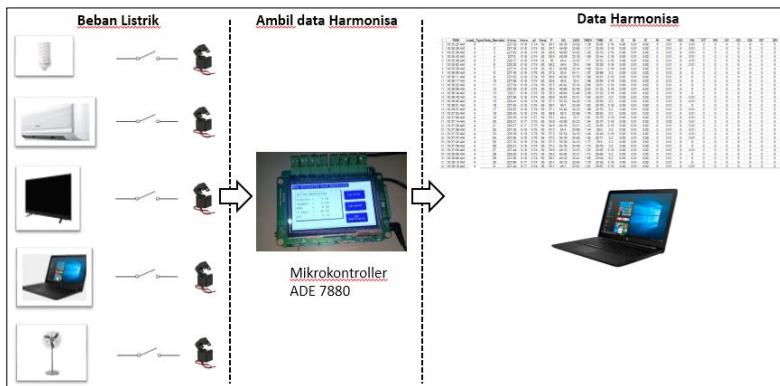
Tahap terakhir yaitu pengujian terhadap model Neural Network dan source code yang sudah didapat pada proses sebelumnya. Tujuan dari pengujian ini adalah untuk mengidentifikasi beban tersebut apakah berstatus “ON atau OFF”.

3.2 Perencanaan

Pada tahap perencanaan ini terdiri dari empat tahapan, antara lain :

- Pengambilan data
- Pemodelan
- Identifikasi/Klasifikasi
- Informasi beban listrik yang terpasang

3.2.1 Pengambilan Data



Gambar 3.2 Tahap 1: pengambilan data

Pada tahap pengambilan data ini dilakukan pengambilan data harmonisa pada 5 jenis beban rumah tangga yaitu Lampu, AC (*Air Conditioner*), TV, Laptop ASUS, dan kipas angin. Proses pengambilan data ini menggunakan microcontroller ARM ST32F7 yang terhubung langsung dengan computer melalui kabel serial yang di dalamnya sudah diprogram untuk mencatat data pengukuran berupa harmonisa arus dari orde 1 sampai harmonisa ke-29, THD (*Total Harmonic Distortion*), frekuensi, daya, tegangan, arus dan power factor.

Dalam melakukan pengambilan data ini dipakai lima peralatan yang telah disebutkan diatas dan diproses dalam 31 kombinasi beban tersebut, baik itu peralatan itu sendiri dan beberapa peralatan yang dipakai secara bersamaan. Jika peralatan terhubung dengan sumber listrik, maka akan mempunyai nilai target 1, jika tidak terhubung dengan sumber listrik maka nilainya adalah 0 seperti yang tertera pada tabel 3.1.

Tabel 3.1 Kombinasi Beban

No	AC	Kipas Angin	TV	Laptop	Lampu Hemat Energi
1.	0	0	0	0	1
2.	0	0	0	1	0
3.	0	0	0	1	1
4.	0	0	1	0	0
5.	0	0	1	0	1
6.	0	0	1	1	0
7.	0	0	1	1	1
8.	0	1	0	0	0
9.	0	1	0	0	1
10.	0	1	0	1	0
11.	0	1	0	1	1
12.	0	1	1	0	0
13.	0	1	1	0	1
14.	0	1	1	1	0
15.	0	1	1	1	1
16.	1	0	0	0	0
17.	1	0	0	0	1
18.	1	0	0	1	0
19.	1	0	0	1	1
20.	1	0	1	0	0
21.	1	0	1	0	1
22.	1	0	1	1	0
23.	1	0	1	1	1
24.	1	1	0	0	0
25.	1	1	0	0	1
26.	1	1	0	1	0
27.	1	1	0	1	1
28.	1	1	1	0	0
29.	1	1	1	0	1
30.	1	1	1	1	0
31.	1	1	1	1	1

Target jenis beban :

1. Lampu Hemat Energi
2. Laptop
3. Laptop – Lampu Hemat Energi
4. TV
5. TV – Lampu Hemat Energi
6. TV – Laptop
7. TV – Laptop – Lampu Hemat Energi

8. Kipas Angin
9. Kipas Angin – Lampu Hemat Energi
10. Kipas Angin – Laptop
11. Kipas Angin – Laptop – Lampu Hemat Energi
12. Kipas Angin – TV
13. Kipas Angin – TV – Lampu Hemat Energi
14. Kipas Angin – TV – Laptop
15. Kipas Angin – TV – Laptop – Lampu Hemat Energi
16. AC
17. AC – Lampu Hemat Energi
18. AC – Laptop
19. AC – Laptop – Lampu Hemat Energi
20. AC – TV
21. AC – TV – Lampu Hemat Energi
22. AC – TV – Laptop
23. AC – TV – Laptop – Lampu Hemat Energi
24. AC – Kipas Angin
25. AC – Kipas Angin – Lampu Hemat Energi
26. AC – Kipas Angin – Laptop
27. AC – Kipas Angin – Laptop – Lampu Hemat Energi
28. AC – Kipas Angin – TV
29. AC – Kipas Angin – TV – Lampu Hemat Energi
30. AC – Kipas Angin – TV – Laptop
31. AC – Kipas Angin – TV – Laptop – Lampu Hemat Energi

Dalam proses pengambilan data ini dilakukan sebanyak 100 kali pengambilan untuk setiap jenis beban.

Data_Number	I1	I3	I5	I7	I9	I11	I13	I15	I17	I19	I21	I23	I25	I27	I29	Target
1	0.1148	0.0608	0.0382	0.0260	0.0165	0.0000	0.0009	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1
2	0.1158	0.0801	0.0382	0.0273	0.0158	0.0005	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1
3	0.1133	0.0788	0.0383	0.0256	0.0166	0.0000	0.0004	0.0003	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1
4	0.1118	0.0787	0.0370	0.0258	0.0162	0.0008	0.0000	0.0004	0.0000	0.0005	0.0000	0.0000	0.0000	0.0000	0.0000	1
5	0.1132	0.0786	0.0388	0.0254	0.0174	0.0017	0.0011	0.0000	0.0000	0.0014	0.0000	0.0000	0.0000	0.0000	0.0000	1
6	0.1120	0.0786	0.0385	0.0246	0.0184	0.0011	0.0000	0.0009	0.0005	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	1
7	0.1121	0.0783	0.0373	0.0253	0.0171	0.0002	0.0010	0.0027	0.0000	0.0003	0.0000	0.0000	0.0000	0.0000	0.0000	1
8	0.1118	0.0790	0.0387	0.0257	0.0177	0.0003	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1
9	0.1117	0.0783	0.0384	0.0259	0.0181	0.0016	0.0005	0.0003	0.0000	0.0009	0.0000	0.0000	0.0000	0.0000	0.0000	1
10	0.1112	0.0795	0.0381	0.0270	0.0178	0.0004	0.0008	0.0017	0.0000	0.0021	0.0000	0.0000	0.0000	0.0000	0.0000	1

Gambar 3.3 Contoh hasil pengambilan harmonisa beban

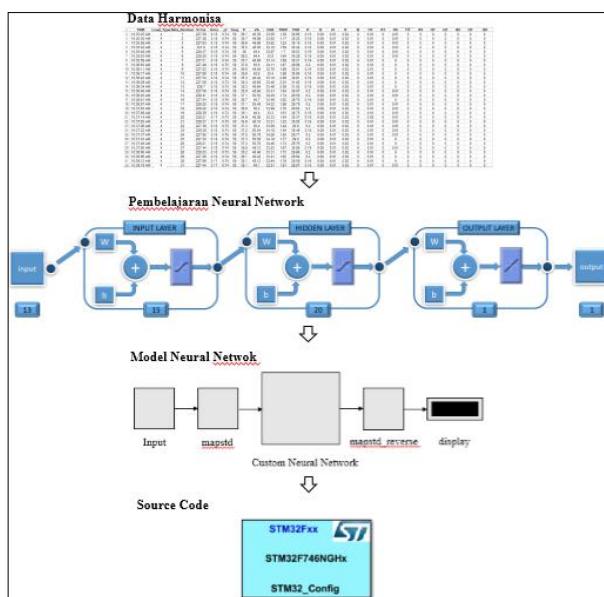
Pada tabel 3.3 ini menunjukkan hasil pengambilan harmonisa beban pada target beban 1 sebanyak 100 data.

3.2.2 Pemodelan

Setelah mendapatkan data harmonisa dari peralatan yang digunakan dalam penelitian ini, proses selanjutnya adalah tahap pembelajaran terhadap data harmonisa.

Pada tahap pembelajaran ini menggunakan *script* simulasi *neural network*. Parameter yang digunakan dalam proses pembelajaran ini adalah variasi jumlah input harmonisa 3, 4, 5, 6, 7 ; variasi jumlah *epochs* 100, 250, 500, 750, 1000 dan variasi jumlah *neuron hidden layer* 5-10-1, 10-15-1, 15-20-1, 20-25-1, 25-30-1. Tujuan dari proses ini yaitu untuk mempelajari pola dari data harmonisa untuk mencapai target. Proses ini menggunakan software MATLAB 2014b.

Proses pemodelan berikutnya menggunakan Simulink pada MATLAB, dengan menggunakan block diagram generator Simulink Neural Network, block diagram ini terdiri dari *input*, *mapstd*, *custom neural network*, *mapstd_reverse* dan *display*.



Gambar 3.4 Tahap 2 : Pemodelan

Block diagram *input* berfungsi sebagai masukan data harmonisa yang akan diuji. *Mapstd* merupakan proses normalisasi rata-rata dan standard deviasi dari proses pembelajaran, fungsinya untuk menormalkan input

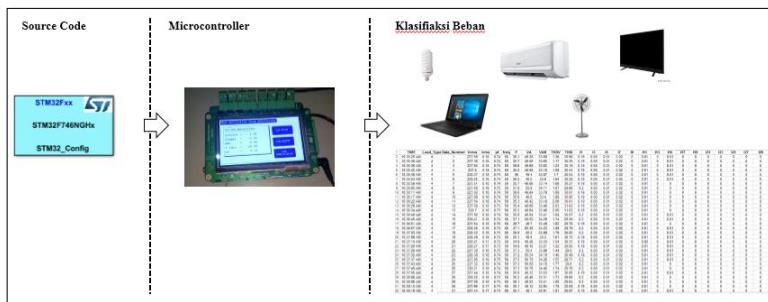
dan target sehingga memiliki nilai rata-rata 0 dan nilai standard deviasi 1. Sedangkan *mapstd_reverse* merupakan proses denormalisasi yaitu mengembalikan nilai yang sudah dinormalisasi ke nilai yang sebenarnya. Block diagram display untuk menunjukkan hasil dari pengujian dari model *neural netwok*.

Hasil dari pemodelan tersebut akan muncul di Block display. Tujuan dari proses pemodelan ini akan menghasilkan source code yang akan dimasukkan kedalam microcontroller ARM ST332F7.

3.2.3 Identifikasi/Klasifikasi

Tahap ke-3 yaitu melakukan identifikasi/klasifikasi jenis beban dengan memasukkan data harmonisa yang sudah didapatkan dalam tahap pertama.

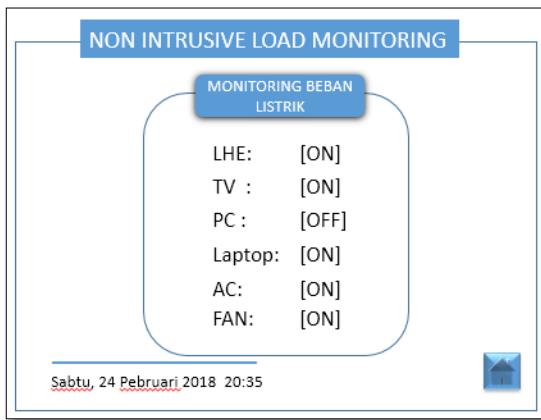
Sebelum melakukan pengujian identifikasi benamkan source code yang sudah didapatkan dari tahap sebelumnya kedalam microcontroller ARM ST332F7 menggunakan software Keil Vision.



Gambar 3.5 Tahap 3 : Identifikasi/Klasifikasi

3.2.4 Informasi beban listrik yang terpasang

Pada tahap terakhir kali ini hasil dari identifikasi jenis beban tersebut ditampilkan pada LCD yang sudah terpasang di microcontroller. Tampilan pada LCD tersebut merupakan menampillkan jenis beban mana yang terpasang.



Gambar 3.6 Tahap 4 : Informasi beban listrik yang terpasang

Halaman ini sengaja dikosongkan

BAB IV

HASIL PERCOBAAN DAN ANALISIS DATA

Dalam bab ini akan dibahas tentang hasil percobaan dan analisis data. Selain itu, parameter aksitektur *neural network*, hasil pembelajaran dan hasil pengujian *neural network* dibahas dalam bab ini untuk mengevaluasi metode yang diusulkan.

4.1 Arsitektur Neural Network

Sedikitnya *backpropagation neural network* mempunyai 3 lapisan *neuron*, yaitu satu lapis masukan atau *input layer*, satu lapis tersembunyi atau *hidden layer*, dan satu lapis keluaran atau *output layer*.

4.1.1 Input Layer

Input layer pada penelitian ini mempunyai peran sebagai jumlah kriteria yang dipakai sebagai pembeda antara kelas yang satu dengan kelas yang lain. Jumlah *node* yang digunakan pada *input layer* kali ini adalah 7 node, yaitu 7 harmonik ganjil pertama dari peralatan yang terpasang.

4.1.2 Hidden Layer

Pada *hidden layer* ini tidak ada teori yang menentukan jumlah *node*. Artinya kita bebas dalam menentukan jumlah *node* pada *hidden layer*. Untuk mendapatkan hasil yang memiliki *error* yang sedikit kita harus mencoba beberapa variasi jumlah *node* pada *hidden layer* pada saat proses pelatihan. Pada penelitian ini memakai 3 *hidden layer*, yaitu *tansig*, *logsig*, *purelin*.

4.1.3 Output Layer (Target)

Output Layer ditentukan berdasarkan jumlah kombinasi peralatan yang akan diuji. Pada kasus ini memakai 5 peralatan, jadi ada 32 kombinasi. Daftar target dapat dilihat pada **Tabel 3.1**

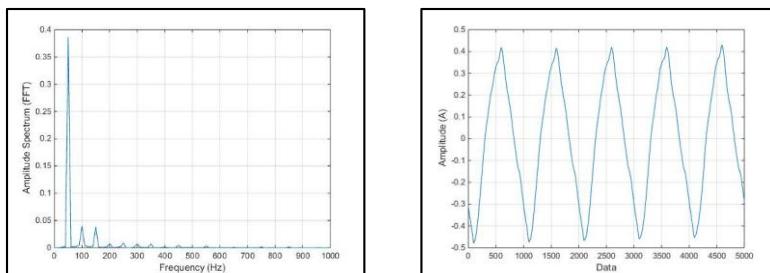
4.2 Parameter Pembelajaran dan Pengujian Neural Network

Untuk menentukan keberhasilan dari proses pengujian pada algoritma *neural network* dipilih parameter yang tepat. Parameter tersebut yaitu menurut jumlah *Eoops*, jumlah *Input* dan jumlah *Neuron* masing-masing *Layer*.

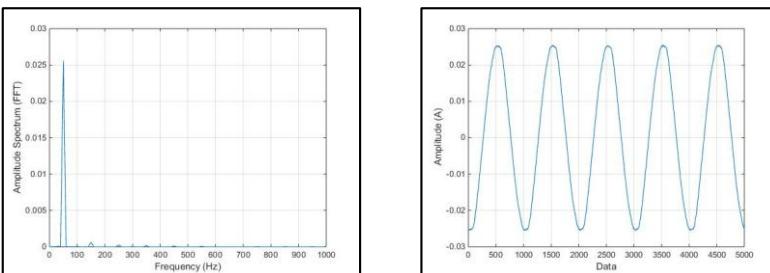
4.3 Tahap Pembelajaran

Pada tahap ini dilakukan tahap pembelajaran terhadap data harmonisa beban peralatan yang digunakan untuk penelitian ini. Tujuan dari tahap ini untuk mempelajari pola data harmonisa yang dihasilkan oleh beban peralatan agar mencapai target beban yang sudah ditentukan.

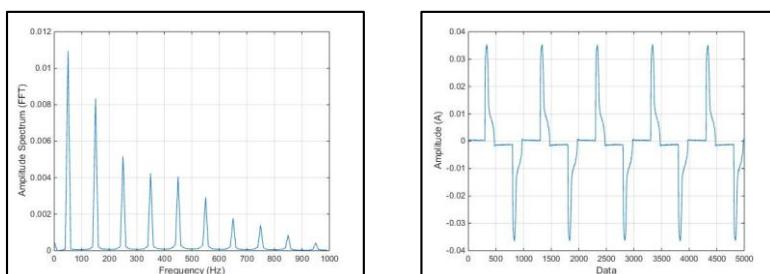
Berikut merupakan hasil rekam pengambilan data gelombang dan *spectrum* harmonisa yang dihasilkan oleh beban-beban yang akan digunakan.



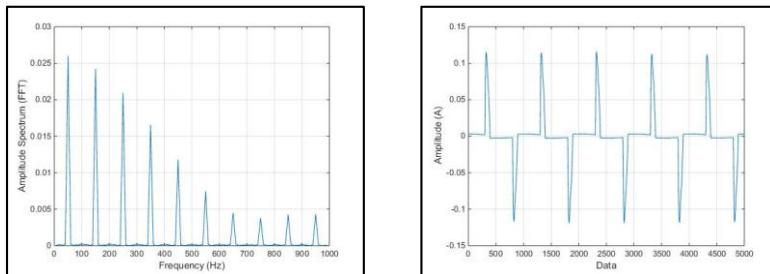
Gambar 4.1 Gelombang dan *spectrum* harmonisa arus pada beban AC



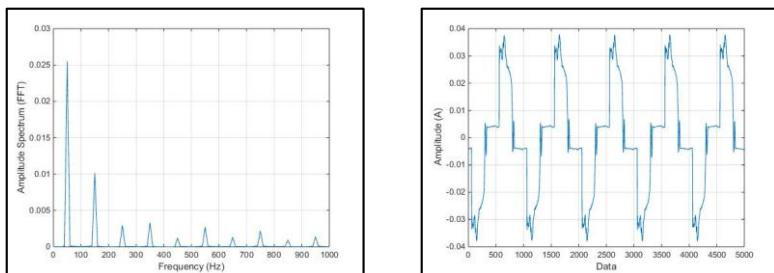
Gambar 4.2 Gelombang dan *spectrum* harmonisa arus pada beban fan



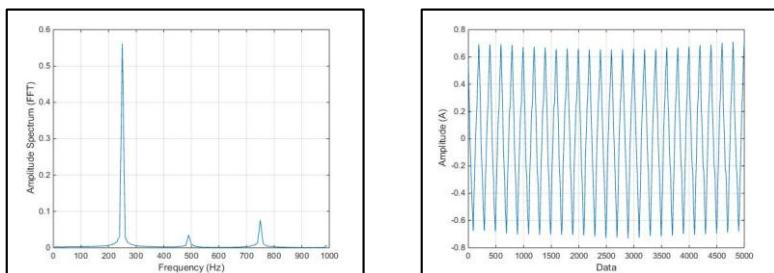
Gambar 4.3 Gelombang dan *spectrum* harmonisa arus pada beban lampu



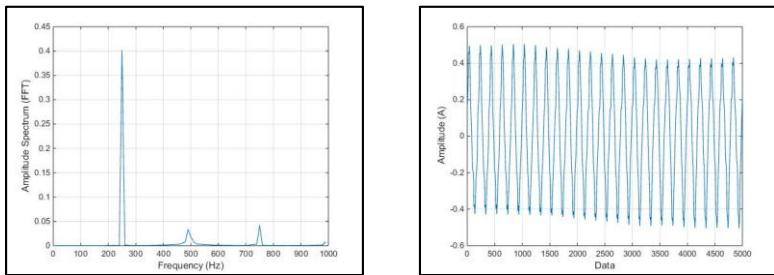
Gambar 4.4 Gelombang dan *spectrum* harmonisa arus pada beban laptop



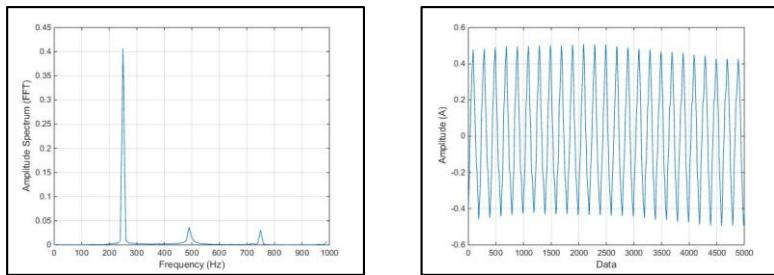
Gambar 4.5 Gelombang dan *spectrum* harmonisa arus pada beban TV



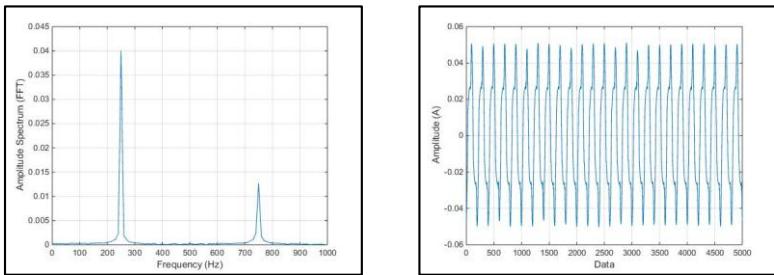
Gambar 4.6 Gelombang dan *spectrum* harmonisa arus pada beban AC dan lampu



Gambar 4.7 Gelombang dan *spectrum* harmonisa arus pada beban AC dan laptop



Gambar 4.8 Gelombang dan *spectrum* harmonisa arus pada beban AC dan TV



Gambar 4.9 Gelombang dan *spectrum* harmonisa arus pada beban fan dan AC

4.3.1 Tahap Pembelajaran dengan Variasi Jumlah Input Harmonisa

Pada tahap pembelajaran kali ini proses pembelajaran menggunakan variasi jumlah *input* harmonisa dengan jumlah *epochs* 1000 dan *neuron* masing-masing *layer* 25-30-1. Dengan menghitung rata-rata *error*, sebagai berikut :

$$error = abs \left(\frac{angka\ taget\ beban - angka\ hasil\ pembelajaran}{angka\ target} \right) \times 100\% \quad (4.1)$$

$$rata - rata\ error = \frac{jumlah\ hasil\ error}{100} \quad (4.2)$$

Berikut hasil dari pengujian identifikasi beban dengan jumlah input harmonisa 3, *neuron* masing-masing *layer* 25-30-1, dan jumlah *epochs* 1000.

Tabel 4.1 Hasil Pembelajaran dengan Jumlah Input Harmonisa 3.

epoch	Neuron tiap Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	3	1	1,92
1000	25-30-1	3	2	4,40
1000	25-30-1	3	3	1,78
1000	25-30-1	3	4	1,25
1000	25-30-1	3	5	1,49
1000	25-30-1	3	6	1,31
1000	25-30-1	3	7	0,71
1000	25-30-1	3	8	0,34
1000	25-30-1	3	9	0,83
1000	25-30-1	3	10	0,72
1000	25-30-1	3	11	0,56
1000	25-30-1	3	12	0,33
1000	25-30-1	3	13	0,54
1000	25-30-1	3	14	0,51
1000	25-30-1	3	15	0,37
1000	25-30-1	3	16	5,01
1000	25-30-1	3	17	4,71
1000	25-30-1	3	18	4,60
1000	25-30-1	3	19	2,87
1000	25-30-1	3	20	6,98
1000	25-30-1	3	21	2,45
1000	25-30-1	3	22	3,89
1000	25-30-1	3	23	1,15

epoch	Neuron tiap Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	3	24	3
1000	25-30-1	3	25	4,46
1000	25-30-1	3	26	3,22
1000	25-30-1	3	27	3,06
1000	25-30-1	3	28	4,96
1000	25-30-1	3	29	1,64
1000	25-30-1	3	30	1,29
1000	25-30-1	3	31	0,69

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.1 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-12 dengan rata-rata **error 0,33%**.

Berikut hasil dari pengujian identifikasi beban dengan jumlah input harmonisa 4, *neuron* masing-masing *layer* 25-30-1, dan jumlah *epochs* 1000.

Tabel 4.2 Hasil Pembelajaran dengan Jumlah Input Harmonisa 4.

epoch	Hidden Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	4	1	4,01
1000	25-30-1	4	2	2,79
1000	25-30-1	4	3	0,87
1000	25-30-1	4	4	0,88
1000	25-30-1	4	5	0,74
1000	25-30-1	4	6	0,97
1000	25-30-1	4	7	0,61
1000	25-30-1	4	8	0,1
1000	25-30-1	4	9	0,73
1000	25-30-1	4	10	0,23
1000	25-30-1	4	11	0,57
1000	25-30-1	4	12	0,57
1000	25-30-1	4	13	0,63
1000	25-30-1	4	14	0,7
1000	25-30-1	4	15	0,14
1000	25-30-1	4	16	0,9
1000	25-30-1	4	17	2,38
1000	25-30-1	4	18	0,85

epoch	Hidden Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	4	19	1,91
1000	25-30-1	4	20	3,19
1000	25-30-1	4	21	1,95
1000	25-30-1	4	22	2,69
1000	25-30-1	4	23	1,6
1000	25-30-1	4	24	0,49
1000	25-30-1	4	25	2,34
1000	25-30-1	4	26	1,11
1000	25-30-1	4	27	2,23
1000	25-30-1	4	28	2,95
1000	25-30-1	4	29	1,71
1000	25-30-1	4	30	1,44
1000	25-30-1	4	31	1,02

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.2 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-8 dengan rata-rata *error 0,1%*.

Berikut hasil dari pengujian identifikasi beban dengan jumlah input harmonisa 4, *neuron* masing-masing *layer* 25-30-1, dan jumlah *epochs* 1000.

Tabel 4.3 Hasil Pembelajaran dengan Jumlah Input Harmonisa 5.

epoch	Hidden Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	5	1	5,58
1000	25-30-1	5	2	2,55
1000	25-30-1	5	3	1,75
1000	25-30-1	5	4	0,51
1000	25-30-1	5	5	0,58
1000	25-30-1	5	6	0,46
1000	25-30-1	5	7	0,83
1000	25-30-1	5	8	0,01
1000	25-30-1	5	9	0,83
1000	25-30-1	5	10	0,66
1000	25-30-1	5	11	0,51
1000	25-30-1	5	12	0,26
1000	25-30-1	5	13	0,56

epoch	Hidden Layer	Jumlah Input Harmonisa	Target Beban	Rata-rata Error (%)
1000	25-30-1	5	14	0,17
1000	25-30-1	5	15	0,3
1000	25-30-1	5	16	1,6
1000	25-30-1	5	17	2,1
1000	25-30-1	5	18	1,64
1000	25-30-1	5	19	1,76
1000	25-30-1	5	20	2,14
1000	25-30-1	5	21	1,29
1000	25-30-1	5	22	1,48
1000	25-30-1	5	23	1,12
1000	25-30-1	5	24	0,98
1000	25-30-1	5	25	1,37
1000	25-30-1	5	26	0,95
1000	25-30-1	5	27	1,2
1000	25-30-1	5	28	1,61
1000	25-30-1	5	29	0,81
1000	25-30-1	5	30	0,91
1000	25-30-1	5	31	0,97

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.3 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-8 dengan rata-rata *error 0,01%*.

Pada hasil pembelajaran dengan variasi jumlah input, nilai dengan error terkecil yang didapatkan adalah pada jumlah input harmonisa sebanyak 5. Hal ini menunjukkan bahwa semakin banyak input harmonisa maka semakin jelas karakteristik beban sehingga neural network akan semakin mudah mengenali dan semakin baik akurasi hasil pembelajarannya.

4.3.2 Tahap Pembelajaran dengan Variasi Jumlah *Neuron masing-masing layer*

Pada tahap pembelajaran kali ini proses pembelajaran menggunakan variasi jumlah *neuron* masing-masing *layer* dengan jumlah *epochs* 1000 dan jumlah input harmonisa 5. Berikut hasil dari pengujian identifikasi beban dengan variasi *neuron hidden layer*.

Berikut hasil dari pengujian identifikasi beban dengan variasi *neuron* masing-masing *layer* 5-10-1, input harmonisa 5, dan jumlah *epochs* 1000.

Tabel 4.4 Hasil Pembelajaran dengan Jumlah *Neuron* masing-masing *Layer* 5-10-1.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
1000	5	5-10-1	1	5,03
1000	5	5-10-1	2	14,4
1000	5	5-10-1	3	6,01
1000	5	5-10-1	4	3,46
1000	5	5-10-1	5	3,87
1000	5	5-10-1	6	11,95
1000	5	5-10-1	7	3,66
1000	5	5-10-1	8	0,07
1000	5	5-10-1	9	3,39
1000	5	5-10-1	10	1,72
1000	5	5-10-1	11	1,96
1000	5	5-10-1	12	0,76
1000	5	5-10-1	13	2,62
1000	5	5-10-1	14	1,39
1000	5	5-10-1	15	0,97
1000	5	5-10-1	16	7,86
1000	5	5-10-1	17	4,72
1000	5	5-10-1	18	7,91
1000	5	5-10-1	19	6,68
1000	5	5-10-1	20	7,37
1000	5	5-10-1	21	4,83
1000	5	5-10-1	22	7,22
1000	5	5-10-1	23	3,13
1000	5	5-10-1	24	4,28
1000	5	5-10-1	25	4,75
1000	5	5-10-1	26	5,62
1000	5	5-10-1	27	3,97
1000	5	5-10-1	28	7,71
1000	5	5-10-1	29	4,04
1000	5	5-10-1	30	2,35
1000	5	5-10-1	31	2,47

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.4 diatas tahap pembelajaran yang menunjukkan rata-rata nilai *error* terkecil adalah pada jenis beban ke-8 dengan rata-rata *error* **0,07%**.

Berikut hasil dari pengujian identifikasi beban dengan variasi *neuron* masing-masing *layer* 10-15-1, input harmonisa 5, dan jumlah *epochs* 1000

Tabel 4.5 Hasil Pembelajaran dengan Jumlah *Neuron* masing-masing *Layer* 10-15-1.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
1000	5	10-15-1	1	2,71
1000	5	10-15-1	2	2,63
1000	5	10-15-1	3	1,79
1000	5	10-15-1	4	0,76
1000	5	10-15-1	5	2,22
1000	5	10-15-1	6	4,06
1000	5	10-15-1	7	1,34
1000	5	10-15-1	8	1,12
1000	5	10-15-1	9	2,43
1000	5	10-15-1	10	0,53
1000	5	10-15-1	11	2,22
1000	5	10-15-1	12	0,51
1000	5	10-15-1	13	1,23
1000	5	10-15-1	14	0,6
1000	5	10-15-1	15	0,54
1000	5	10-15-1	16	6,17
1000	5	10-15-1	17	3,59
1000	5	10-15-1	18	5,9
1000	5	10-15-1	19	2,12
1000	5	10-15-1	20	4,36
1000	5	10-15-1	21	1,89
1000	5	10-15-1	22	4,51
1000	5	10-15-1	23	3,19
1000	5	10-15-1	24	3,68
1000	5	10-15-1	25	3,04
1000	5	10-15-1	26	4,63
1000	5	10-15-1	27	1,57

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
1000	5	10-15-1	28	5,51
1000	5	10-15-1	29	1,9
1000	5	10-15-1	30	1,84
1000	5	10-15-1	31	1,61

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.5 diatas tahap pembelajaran yang menunjukkan rata-rata nilai *error* terkecil adalah pada jenis beban ke-12 dengan rata-rata *error* **0,51%**.

Berikut hasil dari pengujian identifikasi beban dengan variasi *neuron* masing-masing *layer* 15-20-1, input harmonisa 5, dan jumlah *epochs* 1000

Tabel 4.6 Hasil Pembelajaran dengan Jumlah *Neuron* masing-masing *Layer* 15-20-1.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
1000	5	15-20-1	1	7,88
1000	5	15-20-1	2	4,85
1000	5	15-20-1	3	6,56
1000	5	15-20-1	4	4,89
1000	5	15-20-1	5	2
1000	5	15-20-1	6	2,14
1000	5	15-20-1	7	1,18
1000	5	15-20-1	8	0,95
1000	5	15-20-1	9	1,57
1000	5	15-20-1	10	0,74
1000	5	15-20-1	11	1,03
1000	5	15-20-1	12	0,98
1000	5	15-20-1	13	0,83
1000	5	15-20-1	14	0,76
1000	5	15-20-1	15	0,7
1000	5	15-20-1	16	5,27
1000	5	15-20-1	17	4,9
1000	5	15-20-1	18	5,71
1000	5	15-20-1	19	1,67
1000	5	15-20-1	20	3,34
1000	5	15-20-1	21	3,11

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
1000	5	15-20-1	22	3,69
1000	5	15-20-1	23	3,02
1000	5	15-20-1	24	2,82
1000	5	15-20-1	25	4,15
1000	5	15-20-1	26	3,56
1000	5	15-20-1	27	2,25
1000	5	15-20-1	28	3,54
1000	5	15-20-1	29	2,09
1000	5	15-20-1	30	1,44
1000	5	15-20-1	31	1,66

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.6 diatas tahap pembelajaran yang menunjukkan rata-rata nilai *error* terkecil adalah pada jenis beban ke-15 dengan rata-rata *error* **0,7%**.

Pada hasil pembelajaran dengan variasi jumlah neuron masing-masing layer, nilai dengan error terkecil yang didapatkan adalah pada jumlah neuron masing-masing layer sebanyak 15-10-1. Hal ini menunjukkan bahwa semakin banyak banyak bobot dan bias yang dihasilkan, sehingga *neural netwok* dapat mengenali pola harmonisa lebih akurat.

4.3.3 Tahap Pembelajaran dengan Variasi Jumlah Epochs

Pada tahap pembelajaran kali ini proses pembelajaran menggunakan variasi jumlah *epochs* dengan input harmonisa 7 dan menggunakan *neuron* masing-masing *layer* 25-30-1.

Berikut hasil dari pengujian identifikasi beban dengan variasi jumlah *epochs* 100, input harmonisa 7, dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.7 Hasil Pembelajaran dengan Jumlah *epochs* 100.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
100	7	25-30-1	1	11,7
100	7	25-30-1	2	10,26
100	7	25-30-1	3	5,89
100	7	25-30-1	4	4,08
100	7	25-30-1	5	3,62

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
100	7	25-30-1	6	3,02
100	7	25-30-1	7	2,09
100	7	25-30-1	8	1,95
100	7	25-30-1	9	0,97
100	7	25-30-1	10	1,63
100	7	25-30-1	11	1,67
100	7	25-30-1	12	1,18
100	7	25-30-1	13	1,4
100	7	25-30-1	14	1,67
100	7	25-30-1	15	1,54
100	7	25-30-1	16	2,09
100	7	25-30-1	17	2,42
100	7	25-30-1	18	2,98
100	7	25-30-1	19	1,47
100	7	25-30-1	20	2,47
100	7	25-30-1	21	1,47
100	7	25-30-1	22	1,93
100	7	25-30-1	23	1,09
100	7	25-30-1	24	1,67
100	7	25-30-1	25	2,17
100	7	25-30-1	26	2,25
100	7	25-30-1	27	0,6
100	7	25-30-1	28	2,37
100	7	25-30-1	29	1,23
100	7	25-30-1	30	0,65
100	7	25-30-1	31	0,85

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.7 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-27 dengan rata-rata **error 0,27%**.

Berikut hasil dari pengujian identifikasi beban dengan variasi jumlah *epochs* 250, input harmonisa 7, dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.8 Hasil Pembelajaran dengan Jumlah *epochs* 250.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
250	7	25-30-1	1	5,69
250	7	25-30-1	2	5,03
250	7	25-30-1	3	2,69
250	7	25-30-1	4	0,29
250	7	25-30-1	5	1,16
250	7	25-30-1	6	2
250	7	25-30-1	7	1,16
250	7	25-30-1	8	0,14
250	7	25-30-1	9	0,93
250	7	25-30-1	10	0,63
250	7	25-30-1	11	0,49
250	7	25-30-1	12	0,84
250	7	25-30-1	13	0,84
250	7	25-30-1	14	0,57
250	7	25-30-1	15	0,51
250	7	25-30-1	16	1,48
250	7	25-30-1	17	1,19
250	7	25-30-1	18	1,55
250	7	25-30-1	19	1,29
250	7	25-30-1	20	1,28
250	7	25-30-1	21	0,74
250	7	25-30-1	22	1,18
250	7	25-30-1	23	0,9
250	7	25-30-1	24	0,82
250	7	25-30-1	25	0,95
250	7	25-30-1	26	0,84
250	7	25-30-1	27	0,74
250	7	25-30-1	28	0,97
250	7	25-30-1	29	0,58
250	7	25-30-1	30	0,8
250	7	25-30-1	31	0,45

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.8 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-8 dengan rata-rata **error 0,14%**.

Berikut hasil dari pengujian identifikasi beban dengan variasi jumlah *epochs* 500, input harmonisa 7, dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.9 Hasil Pembelajaran dengan Jumlah *epochs* 500.

epoch	Jumlah Input Harmonisa	Jumlah Neuron Hidden Layer	Target Beban	Rata-rata Error (%)
500	7	25-30-1	1	4,1
500	7	25-30-1	2	2,21
500	7	25-30-1	3	1,98
500	7	25-30-1	4	0,73
500	7	25-30-1	5	1,01
500	7	25-30-1	6	1,5
500	7	25-30-1	7	1,52
500	7	25-30-1	8	0,03
500	7	25-30-1	9	0,7
500	7	25-30-1	10	0,36
500	7	25-30-1	11	0,66
500	7	25-30-1	12	0,44
500	7	25-30-1	13	0,38
500	7	25-30-1	14	0,25
500	7	25-30-1	15	0,49
500	7	25-30-1	16	1,17
500	7	25-30-1	17	0,97
500	7	25-30-1	18	1,38
500	7	25-30-1	19	1,26
500	7	25-30-1	20	0,93
500	7	25-30-1	21	0,88
500	7	25-30-1	22	1,07
500	7	25-30-1	23	0,51
500	7	25-30-1	24	0,65
500	7	25-30-1	25	0,73
500	7	25-30-1	26	0,57
500	7	25-30-1	27	0,82
500	7	25-30-1	28	0,57
500	7	25-30-1	29	0,56
500	7	25-30-1	30	0,47
500	7	25-30-1	31	0,63

Dari hasil pembelajaran yang dapat dilihat dari Tabel 4.9 diatas tahap pembelajaran yang menunjukkan rata-rata nilai error terkecil adalah pada jenis beban ke-8 dengan rata-rata **error 0,03%**.

Pada hasil pembelajaran dengan variasi jumlah *epochs*, nilai dengan error terkecil yang didapatkan adalah pada jumlah *epochs* sebanyak 500. Hal ini menunjukkan bahwa semakin banyak banyak *epochs* maka *neuronal network* melakukan lebih banyak pengulangan dalam mempelajari pola harmonisa.

4.4 Tahap Pengujian

Setelah memperoleh bobot / *weight* pada tahap pembelajaran, pada tahap pengujian kali ini akan mendapatkan *weight* baru dari model *backpropagation neural network*. Data *inputan* yang akan digunakan dalam tahap ini merupakan data harmonisa arus beban yang digunakan pada tahap pembelajaran.

4.4.1 Tahap Pengujian dengan Variasi Jumlah Input Harmonisa

Pada tahap pengujian kali ini proses identifikasi berdasarkan variasi jumlah input harmonisa 3,4,5,6 dan 7 serta menggunakan jumlah epochs 1000 dan neuron masing-masing layer 25-30-1.

Berikut adalah tabel 4.10 merupakan hasil dari pengujian identifikasi beban dengan variasi jumlah input harmonisa.

Tabel 4.10 Hasil Pengujian Identifikasi Beban dengan Variasi Jumlah Input Harmonisa.

Jenis Beban	(% Error dari Jumlah Input)				
	3	4	5	6	7
1	8,76	4,09	3,06	0,97	1,02
2	7,2	4,35	1,95	2,65	2,25
3	3,86	3,13	2,43	1,2	0,66
4	1,2	0,67	0,52	0,15	0,05
5	2,16	1,8	1,6	0,64	0,06
6	2,86	1,18	0,61	0,28	0,03
7	1,11	0,57	0,47	0,35	0,25
8	0,27	0,03	0,02	0,01	0
9	1,43	1,28	0,65	0,22	0,07
10	1,4	1,4	1,37	1,2	0,13
11	2,54	1,54	1	0,54	0,09

Jenis Beban	(\%) Error dari Jumlah Input				
	3	4	5	6	7
12	1,08	0,75	0,41	0,16	0,08
13	3,92	0,38	0,3	0,3	0,07
14	2,5	1	0,14	0,07	0
15	1,26	0,93	0,93	0,33	0,06
16	1	0,68	0,5	0,5	0
17	4,41	0,47	0,17	0,05	0,06
18	6,61	2,27	1,38	1,16	0,83
19	2,89	1,84	0,68	0,42	0,73
20	4,3	1	0,95	0,55	0,25
21	2,33	1,85	1,57	0,09	0,04
22	6,13	2	1,9	0,72	0,27
23	1,86	1	0,91	0,65	0,47
24	2,54	2,5	0,91	0,66	0,04
25	5,08	1,48	0,92	0,52	0
26	8,38	1,88	1,61	0,8	0,46
27	17,25	0,85	0,77	0,55	0,07
28	12	4,39	1,1	0,82	0,75
29	0,51	0,62	0,34	0,24	0,1
30	3,4	2,53	1,3	0,86	0,13
31	2,48	1,77	0,67	0,19	0,09
Rata-rata	3,96	1,62	1,03	0,62	0,3

Dari hasil pengujian yang dapat dilihat dari Tabel 4.10 diatas tahap pengujian yang menunjukkan rata-rata nilai error terkecil adalah pada percobaan jumlah input 7 angka ganjil pertama harmonisa dengan rata-rata **error 0,3%**. Pada pengujian ini semakin banyak jumlah input harmonisa maka error akan semakin kecil, karena neural network bisa mempelajari pola harmonisa yang lebih banyak dan lebih kompleks.

4.4.2 Tahap Pengujian dengan Variasi Jumlah *Neuron* masing-masing *Layer*

Pada tahap pengujian kali ini proses identifikasi berdasarkan variasi jumlah *neuron* masing-masing *layer* serta menggunakan jumlah *epochs* 1000 dan jumlah input harmonisa 5 angka ganjil pertama.

Berikut adalah tabel 4.11 merupakan hasil dari pengujian identifikasi beban dengan variasi jumlah *neuron* masing-masing *layer*.

Tabel 4.11 Hasil Pengujian Identifikasi Beban dengan Variasi Jumlah Neuron masing-masing Layer.

Jenis Beban	(\%) Error dari Neuron Hidden Layer				
	5-10-1	10-15-1	15-20-1	20-25-1	25-30-1
1	5,3	2,82	2,8	2,6	2,5
2	4,65	4	3,6	3,2	0,45
3	18,4	4,93	1,9	0,66	0,13
4	4,8	1,25	0,77	0,7	0,52
5	13,7	6,12	1,16	1,08	0,64
6	13,25	5,51	2,66	0,83	0,36
7	2,31	1,25	1,07	0,94	0,55
8	0,1	0,1	0,07	0,01	0,01
9	6,33	3,77	3,42	2,72	1,43
10	0,7	0,3	0,2	0,13	0,1
11	14,64	5,45	3,18	1,81	0,54
12	2,33	1,91	0,41	0,16	0,16
13	2,31	1,23	1,23	1,15	1
14	0,64	0,5	0,42	0,35	0
15	0,87	0,86	0,06	0,06	0
16	12,88	1	0,81	0,37	0,31
17	2,65	2,58	1,88	0,17	0,05
18	17,89	17,61	11,22	3,88	1,38
19	4,84	3,36	1,52	0,68	0,36
20	2,7	2,6	1,4	0,95	0,6
21	4,33	2,66	0,9	0,19	0,04
22	4,82	4,54	2,13	0,72	0,5
23	3,57	2,73	1,69	1,17	1
24	2,29	2,12	0,5	0,37	0,04
25	7,4	4,2	2,6	1,48	1,12
26	4,54	3,88	1,92	1,61	1,42
27	4,85	2,81	0,55	0,4	0,33
28	3,89	3,1	1,1	0,71	0,33
29	4,07	0,62	0,58	0,31	0,06
30	5,7	4,5	3,93	2,5	0,13
31	10,81	4,19	4,16	3,29	0,67
Rata-rata	6,05	3,31	1,934	1,139	0,543

Dari hasil pengujian yang dapat dilihat dari Tabel 4.11 diatas tahap pengujian yang menunjukkan rata-rata nilai error terkecil adalah pada percobaan jumlah *neuron* masing-masing *layer* 25-30-1 dengan rata-rata *error* **0,543%**. Pengujian ini sesuai karena semakin banyak jumlah *neuron* masing-masing *layer* maka semakin banyak bobot dan biasnya, sehingga mengenali pola harmonisa lebih akurat.

4.4.3 Tahap Pengujian dengan Variasi Jumlah *Epochs*

Pada tahap pengujian kali ini proses identifikasi berdasarkan variasi jumlah *Epochs* serta menggunakan *neuron* masing-masing *layer* 25-30-1 dan jumlah input harmonisa 3, 4, 5 angka ganjil pertama.

Berikut adalah hasil dari pengujian identifikasi beban dengan variasi jumlah *epochs* dengan input harmonisa 3 dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.12 Hasil Pengujian Identifikasi Beban dengan Jumlah *Epochs* dan 3 angka ganjil pertama harmonisa.

Jenis Beban	(\%) Error dari Jumlah <i>Epochs</i>				
	100	250	500	750	1000
1	25.6	10.36	6.4	5.9	1.17
2	14.9	10.9	3.3	0.95	0.1
3	5.03	1.86	1.36	0.3	0.26
4	3.87	2.2	1.62	0.92	0.32
5	8.38	2.66	2.66	2.08	1.8
6	8.66	4.86	3.4	2.86	2.66
7	1.55	1.17	0.55	0.35	0.2
8	6.66	0.32	0.27	0.06	0.05
9	1.97	1.17	1.51	0.75	0.65
10	3.5	2.37	1.2	1	0.18
11	3.63	2.54	2.36	1.54	0.45
12	2.5	0.08	0.83	0.16	0
13	11.92	3.84	2.84	1.92	1.07
14	1.92	0.5	1	0.42	0.21
15	1.53	1.4	0.93	0.8	0.13
16	4.81	4.06	1.43	0.56	0
17	1.29	4.41	3.88	2.52	0.05
18	15.72	8.66	6.61	3.83	2.11
19	3.31	1.84	1.52	1.36	0.63

Jenis Beban	(\%) Error dari Jumlah Epochs				
	100	250	500	750	1000
20	12.95	8.75	8.15	4.3	1.9
21	2.66	2.33	1.33	0.8	0.14
22	9.81	6.77	6.13	3.36	0.63
23	1.86	0.78	0.34	0.34	0.13
24	4.58	3.45	2.54	1.41	0.79
25	7.08	4.92	4.08	0.52	0.2
26	17.76	15	11.3	8.73	8.38
27	18.33	13.55	16.92	15.55	13.62
28	18.42	12	11.89	3.25	1.57
29	6.75	2.79	0.51	0.37	0.37
30	4.3	2.56	2.53	2.16	1.4
31	16.45	9.67	9.35	2.48	2.35
Rata-rata	7.996	4.769	3.833	2.312	1.406

Dari hasil pengujian yang dapat dilihat dari Tabel 4.12 diatas tahap pengujian yang menunjukkan rata-rata nilai error terkecil adalah pada percobaan jumlah *epoch*s 1000 dengan rata-rata *error* **1.406%**. Pada pengujian ini semakin banyak jumlah *epoch*s maka semakin sedikit errornya karena dilakukan lebih banyak iterasi dalam satu kali proses pembelajaran.

Berikut adalah hasil dari pengujian identifikasi beban dengan variasi jumlah *epoch*s dengan input harmonisa 4 dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.13 Hasil Pengujian Identifikasi Beban dengan Jumlah Epochs dan 4 angka ganjil pertama harmonisa.

Jenis Beban	(\%) Error dari Jumlah Epochs				
	100	250	500	750	1000
1	10,3	3,58	2,37	2	0,09
2	11	7,3	5,85	0,7	0,3
3	9,43	4,83	3,9	3,83	2,56
4	2,27	1,62	0,67	0,12	0,07
5	5,36	5,3	2,16	1,6	0,94
6	3,26	2,11	0,61	0,45	0,06
7	1,37	1,22	0,75	0,35	0,28
8	0,35	0,35	0,21	0,03	0

Jenis Beban	(\%) Error dari Jumlah Epochs				
	100	250	500	750	1000
9	10,88	0,53	0,51	0,38	0,2
10	1,62	1,4	0,7	0,7	0,3
11	7,45	2,9	2,27	1,54	0,63
12	9,75	7,75	6,25	5,66	3,5
13	2	1,84	1,07	0,23	0,07
14	2,5	0,35	0,14	0,14	0,07
15	2,46	2,4	0,53	0,33	0,06
16	3,18	1,75	0,5	0,37	0
17	1,82	1,23	1	0,47	0,11
18	19,11	13,05	3,38	1,16	0,11
19	2,89	2,89	2,52	2,36	0,36
20	2,95	1,8	1,75	1	0,55
21	4,57	1,8	1,8	1,14	0,09
22	3,5	2	0,81	0,68	0,22
23	1,13	0,95	0,65	0,65	0,13
24	2,5	2,5	1,7	0,54	0,37
25	6,48	5,08	4,6	2,32	1,04
26	6,38	3,07	1,15	0,96	0,46
27	1,48	1,14	0,85	0,85	0,11
28	4,39	3,17	3,14	1,28	0,17
29	1	0,55	0,27	0,24	0,1
30	10,93	6,9	4,43	3,4	2,2
31	16,58	6,22	1,77	0,9	0,77
Rata-rata	5,45	3,151	1,884	1,176	0,516

Dari hasil pengujian yang dapat dilihat dari Tabel 4.13 diatas tahap pengujian yang menunjukkan rata-rata nilai error terkecil adalah pada percobaan jumlah *epoch*s 1000 dengan rata-rata *error* **0,516%**. Pada pengujian ini semakin banyak jumlah *epoch*s maka semakin sedikit errornya karena dilakukan lebih banyak iterasi dalam satu kali proses pembelajaran.

Berikut adalah hasil dari pengujian identifikasi beban dengan variasi jumlah *epoch*s dengan input harmonisa 5 dan *neuron* masing-masing *layer* 25-30-1.

Tabel 4.14 Hasil Pengujian Identifikasi Beban dengan Jumlah *Epochs* dan 5 angka ganjil pertama harmonisa.

Jenis Beban	(%) Error dari Jumlah <i>Epochs</i>				
	100	250	500	750	1000
1	16,91	15,1	4,7	3,9	2,5
2	9	7,6	5,25	2,85	1,95
3	9,16	6,1	2,86	1,9	1,06
4	4,35	2,4	1,87	1,2	0,52
5	6,42	4,06	1,44	0,96	0,64
6	9,25	5,86	5,41	4,88	0,36
7	11,5	3,28	1,85	0,47	0,01
8	0,12	0,12	0,12	0,12	0,07
9	2,92	1,43	1,16	0,43	0,4
10	1,56	1,01	0,46	0,13	0,12
11	1,63	1	0,9	0,63	0,54
12	3,91	1,25	0,41	0,25	0,16
13	2,53	1,61	1	1	0,69
14	1,07	0,78	0,35	0,28	0
15	3,13	0,4	0,33	0,06	0
16	5	3,43	2,25	1	0,62
17	4,17	1,17	0,64	0,64	0,05
18	8	1,83	1,38	0,72	0,72
19	3,68	0,68	0,68	0,57	0,57
20	5,8	3,5	3,5	0,95	0,35
21	1,33	0,71	0,61	0,33	0,04
22	3,86	2,13	0,72	0,27	0,09
23	1,78	1	0,91	0,21	0,17
24	1,87	1,83	1,33	0,41	0,41
25	4,24	3,72	3,2	2,24	1,48
26	1,61	1,03	0,88	0,46	0,42
27	2,7	2,4	0,62	0,59	0,55
28	2,53	1,17	1,1	0,71	0,07
29	3,5	1,37	0,62	0,41	0,27
30	7	5,16	4,2	1,8	0,13
31	2,06	1,12	0,67	0,25	0,22
Rata-rata	4,605	2,721	1,659	0,99	0,481

Dari hasil pengujian yang dapat dilihat dari Tabel 4.14 diatas tahap pengujian yang menunjukkan rata-rata nilai *error* terkecil adalah pada percobaan jumlah *epochs* 1000 dengan rata-rata *error* **0,481%**. Pada pengujian ini semakin banyak jumlah *epochs* maka semakin sedikit errornya karena dilakukan lebih banyak iterasi dalam satu kali proses pembelajaran.

4.5 Hasil Uji Coba Alat Identifikasi Beban Secara *Real Time*



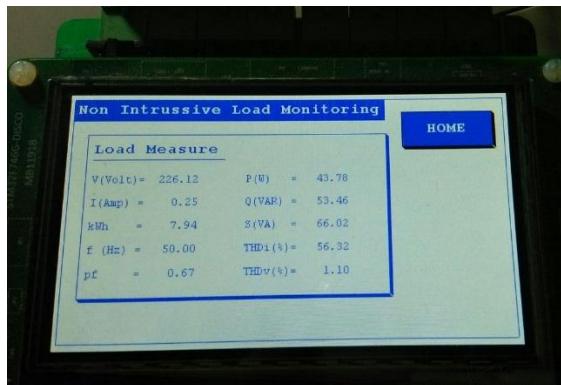
Gambar 4.10 Tampilan awal alat

Pada subbab ini akan menampilkan hasil unjuk kerja dari alat identifikasi beban secara *real time* ditegangan rendah yang telah dirancang dalam *microcontroller*. Alat yang ini diprogram agar dapat mengambil data harmonisa beban, mengetahui informasi pengukuran (daya, frekuensi, arus, tegangan, power factor) beban yang sedang terpakai dan mengidentifikasi beban yang sedang beroperasi. Gambar 4.1 merupakan tampilan awal dari alat tersebut.

Pada pilihan menu *Load Record* merupakan dimana proses pengambilan data harmonisa beban dilakukan. Hasil data tersebut akan langsung diterima oleh computer. Harmonisa tersebut akan digunakan

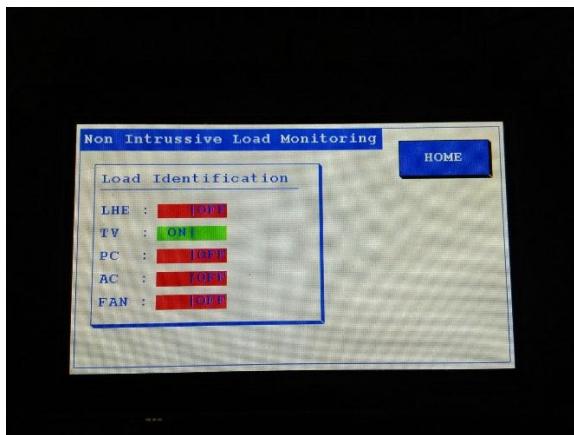
dalam tahap pembelajaran *neural network* dan sebagai input dalam tahap pengujian identifikasi beban.

Untuk melihat informasi pengukuran beban, seperti daya, tegangan, arus, frekuensi, dsb, maka hanya perlu memilih pada menu Load Measure yang menampilkan informasi pengukuran seperti pada gambar 4.2.



Gambar 4.11 Tampilan menu *Load Measure*

Setelah data harmonisa telah melalui tahap-tahap yang sudah dibahas pada subbab sebelumnya, maka dapat dilakukan pengujian terhadap alat ini untuk melakukan identifikasi beban yang sedang beroperasi. Pengguna hanya perlu memasang beban yang akan diuji yang sudah tersambung pararel dengan alat ini. Gambar 4.3 dan 4.4 adalah contoh tampilan *Load Identification* untuk mengetahui beban mana yang sedang beroperasi secara *real time*.



Gambar 4.12 Tampilan informasi beban yang sedang beroperasi (TV)



Gambar 4.13 Tampilan informasi beban yang sedang beroperasi (TV dan Lampu Hemat Energi)

Halaman ini sengaja dikosongkan

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil pembelajaran, pengujian dan analisis data pada penelitian tugas akhir ini, dapat disimpulkan beberapa hal sebagai berikut :

1. Dari hasil pengujian berdasarkan variasi *input* harmonisa yang menghasilkan *error* paling kecil yaitu dengan 7 input harmonisa. Karena *neural network* bisa mempelajari pola harmonisa yang lebih banyak dan lebih kompleks.
2. Dari hasil pengujian berdasarkan variasi jumlah *neuron hidden layer* yang menghasilkan error paling kecil yaitu menggunakan jumlah *hidden layer* sebanyak 25-30-1. Karena semakin banyak jumlah *neuron hidden layer* maka semakin banyak bobot dan bias yang dihasilkan, sehingga *neural netwok* dapat mengenali pola harmonisa lebih akurat.
3. Dari hasil pengujian berdasarkan variasi jumlah *epochs* yang menghasilkan error paling kecil yaitu menggunakan jumlah *epochs* sebanyak 1000. Karena dengan jumlah epochs yang banyak maka *neural network* melakukan lebih banyak pengulangan dalam mempelajari pola harmonisa.

5.2 Saran

Untuk lebih memperbaiki dan menyempurnakan kinerja dari tugas akhir yang kami buat ini ini, maka perlu disarankan :

- a. Sebaiknya peralatan yang akan diujikan mempunyai nilai harmonisa arus yang besarnya signifikan. Agar proses identifikasi beban dengan *backpropagation neural network* semakin menekan jumlah *error*.
- b. Untuk penelitian selanjutnya disarankan untuk memperbanyak kombinasi beban yang akan diujikan.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Arrillaga, J., Watson, N.R., “*Power System Harmonics Second Edition*”, John Wiley & Sons Ltd, England, Ch. 1, 2, 5, 2003
- [2] Mishra, S., Bhende, C.N., Panigrahi, B.K., “Detection and Classification of Power Quality Disturbances Using S-Transform and Probabilistic Neural Network”, IEEE Trans. Power Del., Vol. 23, No.1, Jan. 2008
- [3] Santoso, S., Hofmann, P., “Power Quality Assessment Via Wavelet Transform Analysis”, IEEE Trans. Power Del., Vol. 11, No.2, Apr. 1996
- [4] Santoso, S., Powers, E.J., Grady, W.M., “Power Quality Disturbance Data Compression Using Wavelet Transform Methods”, IEEE Trans. Power Del., Vol. 12, No. 3, Jul. 1997
- [5] IEEE Standard 519-1992, “Recommended Practices and Requirements for Harmonic Control in Electrical Power Systems”, *Institute of Electrical and Electronics Engineers*, 1993
- [6] D. Srinivasan, W. S. Ng, and A. C. Liew, “Neural network-based signature recognition for harmonic source identification,” *IEEE Trans. Power Del.*, vol. 21, no. 01, pp. 398–405, 2006.
- [7] Fernandes, R.A.S., Oleskovics, M., “Load Profile Identification Interface for consumer online monitoring purposes in smart grids”, *IEEE Trans. Industrial Inf.*, vol.9, no.3, August 2013
- [8] Dugan, Roger C., “*Electrical Power Systems Quality*”. McGraw-Hill, Second Edition, 2004.
- [9] Baggini, Angelo. “*Handbook of Power Quality*”. John Wiley & Sons Ltd. Inggris. 2008.
- [10] Rodriguez, M.V., Tronsoco, R.J.R., Perez, A.G., Lieberman, D.G., Rios, R.A.O., “Reconfigurable Instrument for Neural Network Based Power Quality Monitoring in 3 phase Power Systems”, IET Gener. Transm. Distrib., Vol.7, Iss.12, pp. 1298-1207, 2013
- [11] “*BACKPROPAGATION*”. www.novikaginanto.wordpress.com
- [12] “RS Components announces availability of new STMicroelectronics STM32F7 microcontroller”. www.connectingindustry.com

Halaman ini sengaja dikosongkan

RIWAYAT HIDUP PENULIS



Naufal Rasyiq, adalah anak kedua dari dua bersaudara. Lahir pada tanggal 25 Agustus 1995 di Surabaya, Jawa Timur. Penulis telah menyelesaikan pendidikan di SD AL FALAH Surabaya, SMP AL FALAH Deltasari, dan SMA Muhammadiyah 2 Surabaya. Penulis melalui jalur SBMPTN kemudian diterima sebagai salah satu mahasiswa di jurusan S1 Teknik Elektro, Institut Teknologi Sepuluh Nopember, Surabaya. Selama masa kuliah, penulis aktif di beberapa kegiatan kepanitiaan event tingkat jurusan maupun tingkat institut, seminar, dan mengikuti beberapa pelatihan yang diadakan oleh laboratorium PSSL. Penulis dapat dihubungi via email naufal5421@gmail.com.

Halaman ini sengaja dikosongkan