



**ANALISA SISTEM MONITORING *KWh* METER PADA RUMAH
MENGUNAKAN WIRELESS DENGAN PROTOKOL TCP DAN UDP**

SKRIPSI

Oleh

Antony Yacob H.

NIM 111910201089

PROGRAM STUDI STRATA I TEKNIK ELEKTRO

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS JEMBER

2018



**ANALISA SISTEM MONITORING *KWh* METER PADA RUMAH
MENGUNAKAN WIRELESS DENGAN PROTOKOL TCP DAN UDP**

SKRIPSI

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

Oleh

**Antony Yacob H.
NIM 111910201089**

**PROGRAM STUDI STRATA I TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2018**

PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT yang telah memberikan limpahan rahmat, taufiq, serta hidayah yang sangat luar biasa kepada penulis sehingga skripsi ini dapat selesai dengan baik. Tidak lupa sholawat serta salam kita haturkan kepada Nabi Besar Muhammad SAW yang menunjukkan kita sebagai manusia menuju jalan yang terang benderang dengan kehidupan yang lebih baik. Skripsi ini merupakan karya yang tidak pernah ternilai dan terlupakan bagi penulis yang selain sebagai syarat menyelesaikan program studi juga untuk kemajuan umat manusia agar lebih baik. Oleh karenanya karya ini ingin saya persembahkan untuk:

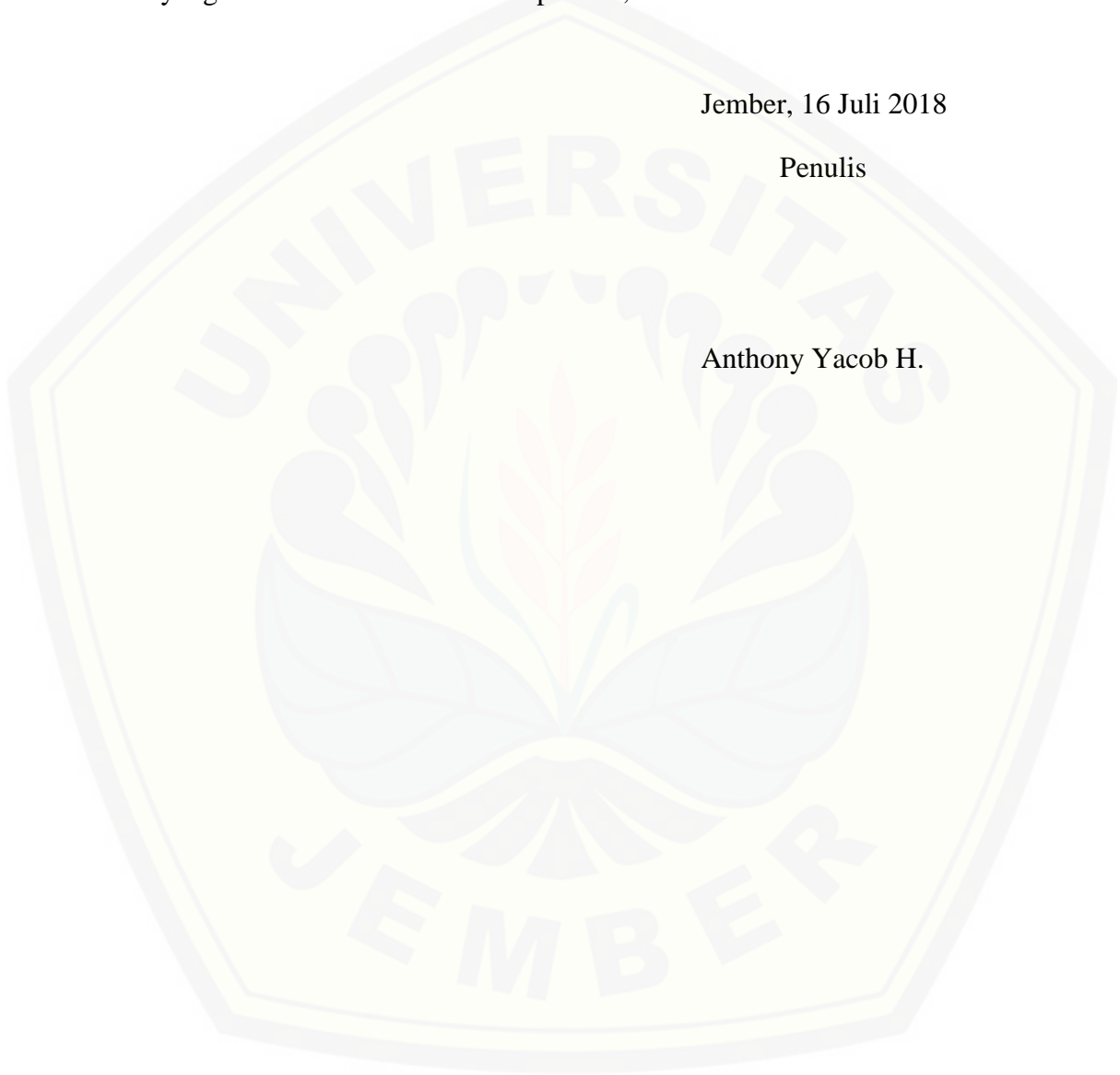
1. Allah SWT, karena perlindungan, pertolongan, dan ridho-Nya penulis mampu menyelesaikan tugas akhir ini dengan baik serta Nabi Besar Muhammad SAW;
2. Keluarga terutama kepada ibu dan ayah, terima kasih dukungan, bantuan, serta doa restunya hingga selesainya studi ini;
3. Kerabat dan sanak keluarga, dan semua keluargaku yang telah memberikan dukungan, motivasi, dan doa;
4. Dosen pembimbing skripsi, Bapak Dodi Setiabudi, S.T., M.T. selaku DPU dan Bapak Catur Suko Sarwono, S.T., M.Si selaku DPA yang bersedia meluangkan waktu dan pikirannya guna memberikan bimbingan dan arahan demi terselesainya skripsi ini;
5. Dosen penguji 1, Bapak Sumardi, S.T., M.T. dan Dosen penguji 2, Bapak Widya Cahyadi, S.T., M.T. yang telah meluangkan banyak waktu dan pikiran guna memberikan pengarahan demi kemajuan dan terselesainya penulisan skripsi ini dengan baik;
6. Semua Dosen Jurusan Teknik Elektro Fakultas Teknik Universitas Jember yang telah memberikan ilmu dan membimbing selama kurang lebih empat tahun ini. Penulis sampaikan banyak terima kasih atas semua ilmu, didikan, dan pengalaman yang sangat luar biasa;

7. Teman-teman elektro yang telah berjuang bersama di almamater tercinta, pengalaman mencari ilmu bersama kalian adalah hal yang tidak akan terlupakan. Aku bangga menjadi bagian dari kalian;
8. Semua pihak yang telah membantu dalam kelancaran pembuatan skripsi ini yang tidak bisa disebutkan satu persatu;

Jember, 16 Juli 2018

Penulis

Anthony Yacob H.



MOTTO

“Sesungguhnya Allah tidak akan mengubah nasib suatu kaum hingga mereka mengubah diri mereka sendiri”
(*Q.S Ar-ra'd ayat 11*)

“Allah tidak akan membebani seseorang itu melainkan sesuai dengan kesanggupannya”
(*Q.S Al-Baqarah ayat 286*)

“Bermimpilah seakan kau akan hidup selamanya. Hiduplah seakan kau akan mati hari ini.”
(James Dean)

“So begins a new age of knowledge”
(Invoker)

“Mottoku adalah setiap kau melakukan kesalahan, lakukan dua kali.”
(Anthony Yacob H)

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Antony Yacob H.

NIM : 111910201089

menyatakan dengan sesungguhnya bahwa karya tulis yang berjudul “**Analisa Sistem Monitoring Kwh Meter Pada Rumah Menggunakan Wireless Dengan Protokol Tcp Dan Udp**” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan subtransi disebutkan sumbernya, dan belum pernah diajukan pada institusi mana pun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung Tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 16 juli 2018
Yang menyatakan,

Antony Yacob H.
NIM 111910201089

SKRIPSI

**ANALISA SISTEM MONITORING *KWh* METER PADA RUMAH
MENGUNAKAN WIRELESS DENGAN PROTOKOL TCP DAN UDP**

SKRIPSI

Oleh :

Antony Yacob H.

111910201089

Pembimbing

Dosen Pembimbing Utama

: Dodi Setiabudi, S.T., M.T.

Dosen Pembimbing Anggota

: Catur Suko Sarwono, S.T., M.Si

PENGESAHAN

Skripsi berjudul “**Analisa Sistem Monitoring Kwh Meter Pada Rumah Menggunakan Wireless Dengan Protokol Tcp Dan Udp**” karya Antony Yacob Haindarto telah diuji dan disahkan oleh Program Studi S-1 Teknik Elektro, Fakultas Teknik, Universitas Jember dan dinyatakan lulus pada:

Hari, tanggal : Jumat, 18 Mei 2018

Tempat : Fakultas Teknik Universitas Jember

Tim Penguji:

Pembimbing Utama,

Pembimbing Anggota,

Dodi Setiabudi, S.T., M.T.
NIP 198405312008121004

Catur Suko Sarwono, S.T., M.Si
NIP 196801191997021001

Penguji I,

Penguji II,

Sumardi, S.T., M.T.
NIP 196701131998021001

Widya Cahyadi, S.T., M.T.
NIP 198511102014041001

Mengesahkan,
Dekan Fakultas Teknik

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

Antony Yacob H.

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

Abstrak

Analisa Sistem Monitoring Kwh Meter Pada Rumah Menggunakan Wireless Dengan Protokol Tcp Dan Udp. Penelitian ini bertujuan untuk mengintegrasikan alat sistem monitoring kWh meter pada rumah menggunakan wireless dalam bentuk RUPiah. Semakin berkembangnya jaman semakin banyak pula kebutuhan – kebutuhan manusia yang diperlukan salah satunya energy listrik. Banyaknya aktivitas masyarakat dan perusahaan – perusahaan yang bertambah sangat tergantung terhadap ketersediaan sumber energy listrik saat ini. Oleh karena itu upaya untuk mensejahterahkan masyarakat dan juga meningkatkan perekonomian saat ini banyak muncul ide – ide yang bermanfaat salah satunya penghematan energy listrik. *Wireless* adalah salah satu teknik komunikasi untuk menyampaikan informasi dengan menggunakan gelombang. Protokol IEEE 802.15.4 merupakan standar protokol komunikasi untuk jaringan terbatas, kecepatan rendah dan konsumsi daya sedikit. Penelitian ini dimulai dari mengkalibrasi sensor arus dan sensor tegangan sebagai pendeteksi nilai arus dan tegangan. kemudian data dikirimkan ke arduino untuk diproses yang nantinya akan ditampilkan pada lcd. Kemudian data akan dikirimkan ke PC sebagai server. Pada penelitian ini didapatkan rata – rata error persen yang didapat pada tegangan sebesar 1.4 % dan arus sebesar 1.3 %. Tegangan *input* pada pengukuran 1 sampai 5 memiliki rata-rata sebesar 224 *volt*, sementara tegangan keluaran saat mendeteksi sebesar 2.51 *volt*. Sementara *delay* yang dibutuhkan untuk mengirimkan notifikasi dalam satu jaringan ke klien sebesar 1.365 *second*.

Kata Kunci: *Packet Loss, Wireless, Delay, Protocol TCP dan UDP, Arduino Uno.*

Antony Yacob H.

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

Abstract

Analyze Monitoring System Kwh Meter At Home Using Wireless With Tcp And Udp Protocol. This study aims to integrate kWh meter monitoring system tool at home using wireless in the form of Rupiah. Increasingly the era of more and more people needed one of them electrical energy. The large number of community activities and growing companies are highly dependent on the availability of current electrical energy sources. Therefore, efforts to succeed the community and also improve the current economy many ideas emerge that one of them is energy saving electricity. Wireless is one of the communication techniques to convey information by using wave. The IEEE 802.15.4 protocol is a standard communications protocol for limited network, low speed and low power consumption. This research starts from calibrating current sensors and voltage sensors as a detector of current and voltage values. then the data is sent to arduino for processing which will be displayed on lcd. Then the data will be sent to PC as server. In this study, the average percent error obtained at a voltage of 1.4% and current of 1.3%. The input voltage in the measurement 1 to 5 has an average of 224 volts, while the output voltage at detecting is 2.51 volts. While the delay required to send notification in one network to the client amounted to 1365 second.

Keywords: Packet Loss, Wireless, Delay, TCP and UDP Protocol, Arduino Uno.

RINGKASAN

Analisa Sistem Monitoring Kwh Meter Pada Rumah Menggunakan Wireless Dengan Protokol Tcp Dan Udp; Antony Yacob H.; 111910201089; 2018; 85 halaman; Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Web server memiliki arti sebagai perangkat lunak yang memberikan atau melayani permintaan dari klien. Didalam web server memiliki database yang bertujuan untuk menyimpan segala informasi yang dibutuhkan oleh perangkat lunak lain atau klien. Pada penelitian ini menggunakan *motioneye* karya Ccrisan yang tersedia di distro linux maupun raspbian sebagai *web server*.

Dalam TCP/IP Transport Layer digunakan 2 macam protokol, yakni TCP dan UDP. TCP (Transmission Control Protocol) adalah protokol yang bertugas untuk membentuk koneksi antar node. Sifat TCP adalah connection-oriented. TCP baru akan membuat koneksi jika kedua belah pihak telah setuju. Karenanya, TCP dianggap reliable (dapat diandalkan). Berbeda dengan UDP (User Datagram Protocol) yang connectionless, transmisi data yang berbasis UDP akan langsung mengirimkan paket karena tidak ada kesepakatan dulu antar node yang bertransmisi.

Masing-masing protokol memiliki karakteristik tertentu dan mendukung protokol-protokol pada layer di atasnya. Misalnya TCP mendukung HTTP dan FTP, sementara UDP mendukung DNS dan TFTP. Perbedaan antara kedua protokol tersebut ada pada reliabilitasnya. Untuk menjalankan tugasnya baik TCP dan UDP menambahkan header pada data yang akan dikirim. Isi header antara kedua protokol tersebut berbeda, sesuai dengan karakteristik masing-masing protokol. Header yang dipasang oleh kedua protokol tersebut dapat identifikasi dan dianalisis dengan menggunakan network analyzer tool, salah satunya adalah Wireshark.

Arduino Uno merupakan alat elektronik yang mempunyai sifat *open source* yang mempunyai bahasa pemrograman yaitu bahasa C dan dapat membaca suatu file yang telah terdeteksi oleh sensor, didalam arduino itu sendiri memiliki 14 pin

input dimana 6 pin input tersebut digunakan sebagai output PWM dan 6 pin input analog. Untuk menjalankan arduino tersebut, kita hanya menghubungkan board arduino ke dalam laptop atau dapat juga menggunakan sebuah baterai.

Dalam proses pengiriman data, sensor dan Arduino disini sebagai Tx (*Transmitter*) dimana nilai dari sensor arus dan tegangan akan membaca nilai arus dan tegangan dan diproses didalam Arduino yang nantinya akan dikirimkan ke server sebagai Rx (*Received*). Transmitter yaitu rangkaian alat yang akan mengambil nilai besaran daya dari sensor dan diolah didalam Arduino.

Dalam proses pengamatan, klien harus melakukan konfigurasi terhadap local server menggunakan *IP address* yang sudah ditentukan didalam program arduino agar klien dapat mengakses web dimanapun dan kapanpun asalkan klien terhubung dengan internet.

Pada preses kalibrasi sensor arus dilakukan sebanyak 5 kali uji coba didapatkan hasil pengujian pertama, percobaan pertama nilai *error* % dari arus bernilai 1.2% sedangkan pada sensor tegangan bernilai *error* 1.3 %. Pada percobaan kedua nilai *error* % dari sensor arus bernilai 1.1% sedangkan pada sensor tegangan bernilai *error* 1.4 %. Pada percobaan ketiga nilai *error* % dari sensor arus bernilai 2.4% sedangkan pada sensor tegangan bernilai *error* 2.2 %. Pada percobaan keempat nilai *error* % dari sensor arus bernilai 1.3% sedangkan pada sensor tegangan bernilai *error* 1.2 %. Pada percobaan kelima nilai *error* % dari sensor arus bernilai 1.2% sedangkan pada sensor tegangan bernilai *error* 2.6 %. Setelah dilakukanya pengambilan data pada pengujian sensor sebanyak 5 kali dalam durasi waktu tertentu maka didapatkan rata-rata *error* sensor arus sebesar 1.3%, dan nilai rata-rata *error* sensor tegangan sebesar 1.4%.

Pada pengujian sensor pengambilan data dilakukan 1 jam sekali dalam waktu 6 jam. Pada percobaan pertama daya yang didapat sebesar 0.095 KWh dengan harga sebesar 44.65 rupiah dan status terkirim. Pada percobaan kedua daya yang didapat sebesar 0.190 KWh dengan harga sebesar 93.75 rupiah dan status terkirim. Pada percobaan ketiga daya yang didapat sebesar 0.285 KWh

dengan harga sebesar 139.75 rupiah dan status terkirim. Pada percobaan keempat daya yang didapat sebesar 0.385 KWh dengan harga sebesar 188.43 rupiah dan status terkirim. Pada percobaan kelima daya yang didapat sebesar 0.470 KWh dengan harga sebesar 232.57 rupiah dan status terkirim. Pada percobaan keenam daya yang didapat sebesar 0.575 KWh dengan harga sebesar 284.116 rupiah dan status terkirim. Dari hasil pengukuran daya yang dilakukan didapatkan rata-rata yang digunakan sebesar 0.090 KWh perjam.

Hasil pengujian *delay* pada proses monitoring secara realtime dari *server* hingga ke *klien* melalui *browser* menunjukkan bahwa rata-rata *delay* yang didapat sebesar 1.365 detik. Dari data pertama daya sebesar 0.095 KWh didapat *delay* sebesar 1.368 detik. Dari data kedua daya sebesar 0.190 KWh didapat *delay* sebesar 1.358 detik. Dari data ketiga daya sebesar 0.285 KWh didapat *delay* sebesar 1.372 detik. Dari data keempat daya sebesar 0.385 KWh didapat *delay* sebesar 1.366 detik. Dari data kelima daya sebesar 0.470 KWh didapat *delay* sebesar 1.367 detik. Dari data keenam daya sebesar 0.575 KWh didapat *delay* sebesar 1.365 detik. Kestabilan sinyal modem pada penelitian ini berpengaruh terhadap nilai *delay*.

Hasil pengujian *throughput* pada proses monitoring secara realtime dari *server* hingga ke *klien* melalui *browser* menunjukkan bahwa rata-rata *throughput* yang didapat sebesar 0.3 KB/sec. Dari data pertama daya sebesar 0.095 KWh didapat *throughput* sebesar 0.3 KB/sec. Dari data kedua daya sebesar 0.190 KWh didapat *throughput* sebesar 0.3 KB/sec. Dari data ketiga daya sebesar 0.285 KWh didapat *throughput* sebesar 0.3 KB/sec. Dari data keempat daya sebesar 0.385 KWh didapat *throughput* sebesar 0.3 KB/sec. Dari data kelima daya sebesar 0.470 KWh didapat *throughput* sebesar 0.3 KB/sec. Dari data keenam daya sebesar 0.575 KWh didapat *throughput* sebesar 0.3 KB/sec. Nilai *throughput* berbanding lurus dengan jumlah data yang dikirimkan.

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Skripsi yang berjudul “**Analisa Sistem Monitoring Kwh Meter Pada Rumah Menggunakan Wireless Dengan Protokol Tcp Dan Udp**”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Penyusunan Skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Allah Subhanahu Wata’ala yang telah memberikan segala nikmat yang tak terhingga. Terima kasih atas ridho dan kehendak-Mu sehingga hamba-Mu ini dapat menyelesaikan tugas akhir ini.
2. Drs. Moh. Hasan, M.Sc., Ph.d selaku Rektor Universitas Jember.
3. Ibu Dr. Ir. Entin Hidayah, M.U.M., selaku dekan Fakultas Teknik Universitas Jember.
4. Bapak Dodi Setiabudi, S.T., M.T. selaku Dosen Pembimbing Utama dan Bapak Catur Suko Sarwono, S.T., M.Si selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini.
5. Bapak Moch. Gozali, S.T., M.T. selaku Dosen Pembimbing Akademik yang telah membimbing selama penulis menjadi mahasiswa.
6. Keluarga terutama kepada ibu dan ayah, terima kasih dukungan, bantuan, serta doa restunya hingga selesainya studi ini.
7. Seluruh Bapak dan Ibu dosen Teknik Elektro beserta staf karyawan di lingkungan Fakultas Teknik Universitas Jember.
8. Teman-Teman ELEKTRO 11 yang tidak bisa saya sebutkan satu persatu, terima kasih atas dukungan dan bantuannya selama proses penyusunan Skripsi ini.

9. Seluruh keluarga besar MAHADIPA dan seluruh organisasi pencinta alam di Indonesia.
10. Anggela Irene Tea Putri S. yang selalu memberikan dorongan, doa dan semangatnya.
11. Komunitas Scooteris Jember JSC, Komunitas Scooteris Banyuwangi BSC dan seluruh komunitas vespa seluruh indonesia, terima kasih atas dukungan dalam pengerjaan skripsi ini.
12. Sahabat-sahabat saya yang selalu memberikan motivasi dan semangatnya.
13. Pihak-pihak lain yang tidak bisa saya sebutkan satu persatu, terima kasih atas dukungan dan motivasi kalian dalam penyusunan Skripsi ini.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap semoga skripsi ini dapat bermanfaat.

Jember, 16 Juli 2018

Penulis

DAFTAR ISI

| | |
|--|-----------|
| HALAMAN JUDUL | i |
| HALAMAN PERSEMBAHAN | iii |
| HALAMAN MOTTO | v |
| HALAMAN PERNYATAAN | vi |
| HALAMAN PEMBIMBINGAN | vii |
| HALAMAN PENGESAHAN | viii |
| ABSTRAK | ix |
| ABSTRACT | x |
| RINGKASAN | xi |
| PRAKATA | xiv |
| DAFTAR ISI | xvi |
| DAFTAR GAMBAR | xviii |
| DAFTAR TABEL | xx |
| BAB 1. PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 3 |
| 1.3 Batasan Masalah | 3 |
| 1.4 Tujuan | 3 |
| 1.5 Manfaat | 3 |
| BAB 2. TINJAUAN PUSTAKA | 4 |
| 2.1 Jurnal Penelitian yang Relevan | 4 |
| 2.2 Arduino | 5 |
| 2.3 Sensor Arus | 6 |
| 2.4 Web Server | 7 |
| 2.5 Sensor Tegangan | 8 |
| 2.6 Faktor Daya | 8 |
| 2.7 LCD (<i>Liquid Crystal Display</i>) | 9 |
| 2.8 Wireless | 11 |

| | |
|---|-----------|
| 2.9 Paket Loss | 14 |
| 2.10 Model Jaringan Komputer | 15 |
| 2.10.1 Jaringan <i>Local Area Network</i> (LAN) | 15 |
| 2.10.2 Jaringan <i>Metropolitan Area Network</i> (MAN)..... | 16 |
| 2.10.3 <i>Wide Area Network</i> (WAN) | 17 |
| 2.11 <i>Network Analyzer</i> | 17 |
| 2.12 Delay..... | 18 |
| 2.13 Jitter | 19 |
| 2.14 Troughput..... | 20 |
| 2.15 Smartsniff | 20 |
| 2.16 Protokol TCP dan UDP..... | 21 |
| 2.17 VPN (<i>Virtual Private Network</i> | 28 |
| BAB 3. METODE PENELITIAN..... | 32 |
| 3.1 Tempat dan Waktu Penelitian | 32 |
| 3.2 Hardware | 32 |
| 3.2.1 Arduino UNO | 32 |
| 3.2.2 Spesifikasi Sensor Arus | 32 |
| 3.3 Software | 32 |
| 3.3.1 Smartsniff | 32 |
| 3.3.2 Arduino IDE | 34 |
| 3.4 Prosedur penelitian | 34 |
| 3.5 Realisasi Pembuatan Alat..... | 35 |
| 3.6 Perancangan Alat..... | 36 |
| 3.6.1 Perancangan Sistem..... | 36 |
| 3.6.2 Perancangan Hardware..... | 36 |
| 3.6.3 Perancangan Software..... | 38 |
| 3.6.4 Perancangan Sensor..... | 40 |
| 3.7 <i>Flowchart</i> | 42 |
| 3.8 Konfigurasi VPN | 44 |
| BAB 4. HASIL DAN PEMBAHASAN | 48 |
| 4.1 Pengujian Sensor | 48 |

| | |
|--|-----------|
| 4.2 Pengujian Web | 50 |
| 4.3 Pengujian QOS Protokol TCP | 55 |
| 4.3.1 <i>Delay</i> | 56 |
| 4.3.2 <i>Throughput</i> | 57 |
| 4.4 Pengujian QOS Protokol UDP | 59 |
| 4.4.1 Pengujian Delay Server Singapura | 59 |
| 4.4.2 Pengujian Throughput Sever Singapura | 61 |
| 4.4.3 Pengujian Delay Server Belanda | 62 |
| 4.4.4 Pengujian Throughput Sever Belanda..... | 64 |
| 4.5 Pengujian LCD | 66 |
| BAB 5. PENUTUP..... | 68 |
| 5.1 Kesimpulan | 68 |
| 5.2 Saran | 69 |
| DAFTAR PUSTAKA | 70 |
| LAMPIRAN | |

DAFTAR GAMBAR

| | |
|---|----|
| 2.1 Arduino Uno | 6 |
| 2.2 Rangkaian Sensor Tegangan | 8 |
| 2.3 Gelombang Sinus | 8 |
| 2.4 LCD (<i>Liquid Cristal Display</i>) | 10 |
| 2.5 Jaringan LAN | 16 |
| 2.6 <i>Jaringan MAN</i> | 16 |
| 2.7 Jaringan WAN | 17 |
| 2.8 <i>Smartsniff</i> | 21 |
| 3.1 Tampilan Smartsniff | 23 |
| 3.2 Tampilan Arduino IDE | 24 |
| 3.3 Blok Diagram | 26 |
| 3.4 Perancangan Hardware | 27 |
| 3.5 Tampilan Arduino | 29 |
| 3.6 Tampilan Pengaturan Rupiah | 29 |
| 3.7 Tampilan Sensor Arus | 30 |
| 3.8 Pengaturan Sensor | 31 |
| 3.9 Flowchart Pengambilan Data | 32 |
| 3.10 Flowchart Pengiriman Data | 33 |
| 4.1 Sensor Arus SCTO13.000 100A/50mAh | 35 |
| 4.2 Pengukuran DT-266 | 36 |
| 4.3 Tampilan alat | 38 |
| 4.4 Tampilan menggunakan laptop | 38 |
| 4.5 Tampilan menggunakan laptop | 39 |
| 4.6 Tampilan menggunakan android | 39 |
| 4.7 Tampilan arduino | 40 |
| 4.8 Tampilan Arduino saat terhubung di Web | 40 |
| 4.9 Tampilan Arduino saat tidak terhubung di Web | 41 |
| 4.10 <i>Capture</i> dengan <i>software</i> smartsniff | 42 |
| 4.11 Tampilan LCD saat sensor tidak mengukur nilai daya | 46 |

| | |
|---|----|
| 4.12 Tampilan LCD saat sensor mengukur nilai arus dan tegangan..... | 46 |
| 4.13 Tampilan awal LCD saat alat diaktifkan..... | 46 |



DAFTAR TABEL

| | |
|---|----|
| 2.1 Tabel Standarisasi <i>Wireless</i> | 12 |
| 2.2 Standar-Standar IEEE 802.11 | 13 |
| 2.3 Klasifikasi QoS parameter <i>packet loss</i> | 14 |
| 2.4 Standarisasi ITU-T <i>Delay</i> | 19 |
| 2.5 Klasifikasi QoS parameter <i>end-to-end delay</i> | 19 |
| 2.6 Klasifikasi QoS parameter <i>jitter</i> | 20 |
| 2.7 Klasifikasi QoS parameter <i>throughput</i> | 20 |
| 3.1 Koneksi Pin LCD Ke I2C | 27 |
| 3.2 Koneksi Pin Arduino Ke LCD | 28 |
| 3.3 Konfigurasi Pin | 30 |
| 3.4 Pengaturan Sensor | 31 |
| 4.1 Hasil Pengukuran sensor arus dan tegangan | 35 |
| 4.2 Hasil Pengukuran Daya..... | 37 |
| 4.3 Perhitungan rata-rata <i>delay</i> menggunakan Ms. Excel..... | 43 |
| 4.4 Perhitungan rata-rata Throughput menggunakan Ms. Excel..... | 44 |

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Semakin berkembangnya jaman semakin banyak pula kebutuhan – kebutuhan manusia yang diperlukan salah satunya energy listrik. Banyaknya aktivitas masyarakat dan perusahaan – perusahaan yang bertambah sangat tergantung terhadap ketersediaan sumber energy listrik saat ini. Oleh karena itu upaya untuk mensejahterahkan masyarakat dan juga meningkatkan perekonomian saat ini banyak muncul ide – ide yang bermanfaat salah satunya penghematan energy listrik.

Karena peran strategisnya, ketersediaan energy listrik saat ini sangat berpengaruh dalam kemajuan perekonomian nasional maka dari itu dibutuhkan ketersediaan energy listrik dalam jumlah banyak dan berkualitas baik. Akan tetapi, semakin berkembangnya jaman, pertumbuhan penduduk yang semakin tahun semakin meningkat, kemajuan teknologi yang semakin hari semakin pesat, menyebabkan permintaan energy listrik semakin hari semakin meningkat. Banyak pembangunan pembangkit listrik yang terkendala dikarenakan krisis ekonomi yang sering terjadi di Negara berkembang ini maka dari itu jumlah ketersediaan energy listrik saat ini masih belum mencukupi untuk permintaan energy listrik yang semakin harinya semakin meningkat. Disamping itu, peran pemerintah untuk berinvestasi terhadap ketenagalistrikan terutama dalam pembangunan pembangkit listrik masih terbatas. Dalam hal tersebut menyebabkan jumlah pasokan energy listrik tidak dapat mengimbangi jumlah permintaan energy listrik yang semakin hari semakin meningkat ini, sehingga menyebabkan kekurangan energy listrik untuk beberapa daerah yang tidak dapat dihindari. Salah satu solusi yang dapat dilakukan untuk mengatasi kondisi tersebut dari sisi konsumen adalah dengan melakukan penghematan pemanfaatan energi listrik. Penghematan konsumsi energi listrik harus dilakukan dari berbagai sektor, baik dari sektor industri, instansi ataupun masyarakat umum.

Penghematan konsumsi energi listrik pada sisi konsumen secara teknis dapat dilakukan dengan melakukan *monitoring* pemanfaatan energi listrik di tiap-tiap titik beban listrik. Data hasil *monitoring* tersebut digunakan sebagai dasar untuk

melakukan pengaturan atau kontrol dalam pemanfaatan energi listrik. Oleh karena itu diperlukan suatu teknologi penghematan konsumsi energy listrik yang dapat melakukan aktivitas pemantauan (*monitoring*) yang cerdas dan mudah untuk diaplikasikan.

Kwh meter adalah alat ukur yang digunakan untuk mengukur daya yang digunakan oleh seseorang dalam suatu rumah atau perusahaan untuk melakukan transaksi antara pln dan pelanggan. Dengan adanya sistem AMR (*Automatic Meter Reading*) dapat memberikan beberapa kemudahan untuk mengukur parameter – parameter energy listrik secara langsung dan akurat. Untuk mengoptimasikan sistem AMR ini membutuhkan pemilihan provider yang tepat. Meter elektronik adalah suatu alat ukur besaran – besaran listrik yang memiliki kemampuan untuk mengukur energi aktif (Kwh), energi reaktif (kVARh), energi semu (kva) dan besaran – besaran arus (Ampere), tegangan (volt), Faktor daya (Cos Phi), Frekuensi (Hz) dan lain – lain.

Saat ini perkembangan teknologi semakin pesat dan piranti elektronik yang seiring dengan perkembangan protocol komunikasi membawa kita dalam suatu sensor/alat deteksi yang murah dan mempunyai jangkauan yang sangat luas. Kemampuan sesnsor yang lebih baik dipengaruhi oleh kemajuan dibidang desain, material, perancangan jaringan komunikasi yang berdampak positif.

Agar dapat meningkatkan keefektifan dalam monitoring energy listrik, diciptakan sebuah alat yang dapat melakukan pengukuran dan monitoring energy listrik dengan menggunakan Wireless dimana komponen yang akan digunakan salah satunya yaitu modul Wireless yang akan mengirimkan data. Pada proyek akhir ini, parameter-parameter yang dideteksi adalah tegangan, arus, daya dan kualitas pengiriman data. Diharapkan dengan dibuatnya alat ini dapat memonitor penggunaan energi listrik ditiap titik-titik beban sebagai upaya untuk melakukan penghematan pemanfaatan energi listrik.

1.2 Rumusan Masalah

Dari latar belakang dapat dirumuskan beberapa rumusan masalah sebagai berikut:

1. Bagaimana merancang alat sistem monitoring kWh meter pada rumah menggunakan sistem telemetri?
2. Bagaimana mengintegrasikan alat sistem monitoring kWh meter pada rumah menggunakan wireless dalam bentuk Rupiah?

1.3 Batasan Masalah

Batasan masalah digunakan untuk membatasi masalah – masalah diluar konsep dari penelitian ini. Batasan masalah tersebut ialah:

1. Tidak membahas alat secara elektronika, tapi lebih kearah telekomunikasi.
2. Membahas tentang kualitas pengiriman data.
3. Monitoring atau *output* data ditampilkan di Web.

1.4 Tujuan

Tujuan dari penelitian ini dapat dirumuskan sebagai berikut:

1. Dapat merancang sistem monitoring konsumsi energi listrik berbasis Wireless.
2. Dapat mengetahui kualitas pengiriman data (QoS) pada sistem telemetrinya.

1.5 Manfaat

Manfaat yang diperoleh dari penelitian ini adalah:

1. Dapat digunakan sebagai media untuk memperoleh informasi mengenai penggunaan konsumsi energi dalam rumah.
2. Memberikan manfaat bagi pengembangan ilmu pengetahuan dan teknologi khususnya tentang sistem telemetri.

BAB 2. TINJAUAN PUSTAKA

Pada bab 2 kali ini, membahas tentang publikasi resmi yang berhubungan dengan konsep Alat Sistem Monitoring Konsumsi Energi Pada rumah menggunakan Arduino berbasis Wireless dan mencakup beberapa aspek dan factor yang berhubungan dengan penelitian kali ini. Berikut ini adalah beberapa teori dan konsep yang bersangkutan dengan penelitian yang akan dibahas kali ini.

2.1 Jurnal Penelitian yang Relevan

Pada bab 2.1 ini, membahas berbagai jurnal yang berhubungan dengan konsep Alat Sistem Monitoring Konsumsi Energi Pada rumah menggunakan Arduino berbasis Wireless dan mencakup beberapa aspek dan factor yang berhubungan dengan penelitian kali ini.

Mengacu pada latar belakang peneliti menjadikan beberapa jurnal berikut sebagai acuan penulisan tugas akhir ini. Yang pertama adalah jurnal dari Temy Nusa 2015 dengan judul *Sistem Monitoring Konsumsi Energi Listrik Secara Real Time Berbasis Mikrokontroler*. Pada jurnal ini permasalahan penulisan tersebut adalah dimana setiap tahunnya konsumsi energy listrik semakin meningkat dan tidak mencukupinya pasokan cadangan sumber energy listrik yang akhirnya membuat energy listrik menjadi sesuatu yang sangat mahal. Supaya dapat melakukan penghematan konsumsi energy listrik dirumah maka kita memerlukan suatu perangkat yang dapat memonitoring suatu beban yang sering digunakan dengan memanfaatkan mikrokontroler. Peneliti mengambil jurnal ini sebagai acuan untuk objek. Dimana terdapat kesamaan jenis perangkat serta sistem operasi yang digunakan.

Pada jurnal kedua yaitu Hestry Angraini, Yeffry Handoko Putra 2015. *Sistem Monitoring Energi Listrik Menggunakan Mikrokontroler Berbasis Web*. Pada jurnal ini permasalahan utama adalah tidak adanya sistem monitoring yang dapat terhubung dengan user secara langsung. Oleh karena itu, dibuat sebuah sistem untuk memonitoring pemakaian listrik tiap-tiap ruangan dengan jaringan internet.

Peneliti mengambil jurnal ini sebagai jurnal acuan dalam tahapan pengukuran dan analisis data pada penelitian ini.

Dan pada jurnal yang ketiga Riswandi 2014. *Perancangan Alat Monitoring Arus Kwh Meter Tiga Phasa Dengan Memanfaatkan Mikrokontroler Arduino Dan Sms Gateway Berbasis Web*. Pada jurnal ini permasalahan terdapat pada pengukuran kWh (Kilo Watt Hour) meter tiga phasa sering kali terjadi kesalahan, salah satu penyebabnya adalah hilangnya salah satu phasa arus pada kWh meter tersebut. Jika hilangnya salah satu phasa arus tidak segera dilakukan pengecekan dan perbaikan, maka kerugian perusahaan listrik semakin besar. Hal ini dikarenakan perhitungan di kWh meter tidak sesuai dengan pemakaian pelanggan. Dari permasalahan tersebut penulis memiliki gagasan untuk membuat alat monitoring arus kWh meter tiga phasa dengan menggunakan mikrokontroler arduino berbasis sms gateway dan web. Penulis mengambil jurnal ini sebagai acuan untuk melakukan analisis terhadap kualitas performansi.

2.2 Arduino

Arduino Uno merupakan alat elektronik yang mempunyai sifat *open source* yang mempunyai bahasa pemrograman yaitu bahasa C dan dapat membaca suatu file yang telah terdeteksi oleh sensor, didalam arduino itu sendiri memiliki 14 pin input dimana 6 pin input tersebut digunakan sebagai output PWM dan 6 pin input analog. Untuk menjalankan arduino tersebut, kita hanya menghubungkan board arduino ke dalam laptop atau dapat juga menggunakan sebuah baterai.

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan satu daya eksternal. Eksternal (non-USB) dapat di ambil baik berasal dari AC ke adaptor DC atau baterai. Untuk menghubungkan adaptor ini dapat dilakukan dengan cara menancapkan plug jack ke konektor power. Arduino mempunyai bahasa pemrograman yaitu bahasa C. arduino membutuhkan daya sebesar 7 – 12V agar dapat stabil saat digunakan dan dijalankan, jika regulator tegangan dapat merusak board uno jika dalam kondisi regulator tegangan panas. Karena sifat open source dari arduino ini dapat memberikan beberapa keuntungan tersendiri untuk kita saat memakai board ini, karena kita tidak hanya bergantung dengan satu merek akan

tetapi kita dapat menggunakan beberapa merek yang terdapat dipasaran. Arduino memiliki kelebihan untuk dapat berkomunikasi terhadap computer, arduino dan bahkan dengan mikrokontroler lainnya.



Gambar 2.1 Arduino Uno

2.3 Sensor Arus

Sensor adalah sebuah alat yang sering digunakan untuk mengukur sebuah besaran listrik yang nantinya dapat dihitung dengan menggunakan rangkaian elektronik tertentu. Didalam sebuah rangkaian elektronik memiliki tegangan, arus dan hambatan. Ampere meter merupakan alat elektronik yang digunakan untuk membaca arus listrik yang mengalir pada rangkaian. Arus listrik yang mengalir pada suatu konduktor menimbulkan medan magnet. Oleh sebab itu arus listrik dapat diukur dengan besarnya medan magnet. Medan magnet dipengaruhi oleh beberapa faktor antara lain:

1. Arah medan magnet yang terbentuk.
2. Besaran arus listrik.
3. Jarak antara medan magnet dan titik pengukuran.

Medan magnet adalah suatu medan yang terjadi karena adanya arus listrik yang memunculkan gaya di muatan listrik. Medan magnet terbentuk karena terjadi sebuah putaran dari satu partikel yang dipengaruhi oleh dirinya sendiri seperti arus listrik. Sebuah medan magnet bias dibidang medan vektor, karena berhubungan dengan setiap titik dalam sebuah ruang vektor yang dapat berubah – ubah dalam setiap waktu. Arah dari medan ini adalah seimbang dengan arah jarum kompas

yang diletakkan di dalam medan tersebut. Untuk mengukur kuat arus dapat dilakukan dengan menghubungkan alat menjadi rangkaian seri. Kemampuan alat ukur akan semakin meningkat selama kemajuan teknologi semakin berkembang. Semakin berkembangnya kemajuan teknologi digital ini penggunaan alat elektronik semakin mudah dan dari segi bentuk juga menjadi lebih efisien untuk digunakan dan juga dari segi ekonomi membuat alat elektronik ini bias didapatkan oleh semua kalangan. Semakin pesat kemajuan teknologi ini menyebabkan penelitian - penelitian dapat dilakukan dengan lebih baik dan cepat. Alat ukur dapat tersusun atas bagian digital dan analog. Didalam alat ukur memiliki tiga bagian utama, yaitu sensor, penampil data dan pengelola data. Alat ukur dengan tampilan digital lebih mudah untuk digunakan karena dapat membaca nilai/besaran arus lebih tepat dan teliti. Dengan cara digital mengelolah data menjadi lebih mudah meskipun ada beberapa yang tidak dapat di kelola menggunakan digitan akan tetapi menggunakan sistem analog. Alat yang sering digunakan untuk mengukur suatu besaran arus listrik yaitu sensor arus.

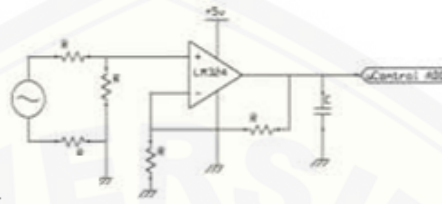
2.4 Web Server

Web merupakan sebuah sistem komunikasi yang sering digunakan sebagai sarana informasi yang menggunakan internet. Banyak penelitian – penelitian yang di publikasikan didalam web dan juga bisnis – bisnis online yang menggunakan web sebagai sarana pemasaran karena jangkauan dari web itu sendiri sangat lah luas dan mudah untuk diakses.

Dalam web memiliki kesederhanan dalam segi konsep. Pada sistem operasinya sendiri web melakukan pertukaran informasi antara klien dan server. Web server memiliki arti sebagai perangkat lunak yang memberika atau melayani permintaan dari klien. Didalam web server memiliki database yang bertujuan untuk menyimpan segala informasi yang dibutuhkan oleh perangkat lunak lain atau klien. Pada penelitian ini menggunakan *motioneye* karya Ccrisan yang tersedia di distro linux maupun raspbian sebagai *web server*. Didalam *motioneye* menggunakan *server* tornado sebagai python *web framework* yang memberikan optimalisasi koneksi akses ke *server* secara bersamaan dalam jumlah banyak (Abdul Kadir, 2005).

2.5 Sensor Tegangan

Ada 3 parameter pokok yang dimiliki pada proses pengukuran daya antara lain yaitu arus, tegangan dan faktor daya. Dimana parameter - parameter turunan dapat dikalkulasi berdasarkan parameter pokok tersebut.



Gambar 2.2 Rangkaian Sensor Tegangan.

Gambar 2.2 merupakan rangkaian dimana digunakan untuk pengukuran besar tegangan. Tahap pertama yang harus dilakukan yaitu dengan melakukan pembagian tegangan. Hal yang harus dilakukan pertama – tama menurunkan tegangan AC 220V hingga input op-amp dapat menerima sinyal tegangan tersebut. Kemudian sinyal dikuatkan dengan menggunakan sinyal sinus pada pembagi tegangan, kemudian dengan catu daya non simetris akan didapatkan pemotongan sinyal keluaran.



Gambar 2.3 Gelombang Sinus

Ketika catu daya negatif op amp dihubungkan pada ground maka potongan sinyal keluaran yang akan terjadi pada tegangan negatif. Efek dari perpotongan sinyal tersebut seperti *halfwave rectifier*.

2.6 Faktor Daya

Perbandingan daya sesungguhnya dengan daya semu sering juga disebut $\cos \theta$ atau factor daya. Setiap alat elektronik yang sering digunakan dalam keseharian kita baik itu dirumah maupun di tempat kerja pasti membutuhkan energy listrik

untuk dapat digunakan. Daya adalah energy yang memerlukan tiap satuan waktu atau detik. Tingkat efisiensi dari daya listrik yang dihasilkan dipengaruhi oleh beberapa factor salah satunya yaitu nilai factor daya yang terdapat pada perangkat sumber tegangan listrik. Semakin tinggi nilai faktor daya dari perangkat penghasil sumber tegangan listrik berbanding lurus dengan kualitas perangkat tersebut

Arus bolak balik memiliki nilai besaran daya yang berbeda antara teori dengan hasil sesungguhnya dari hasil pembacaan alat ukur. Perbandingan antara daya sesungguhnya dan daya semu disebut faktor daya yang dinyatakan dalam persamaan :

$$\text{Cos} = \frac{P_{ss}}{P_{sm}} \dots\dots\dots (1)$$

Nilai daya sesungguhnya yang terdapat pada rangkaian arus AC dapat dinyatakan berbanding lurus dengan hasil perkalian daya hasil perkalian antara tegangan dan kuat arus dengan faktor daya yang secara umum dapat dituliskan :

$$P = V I \cos \dots\dots\dots (2)$$

di mana :

P = daya sesungguhnya (watt)

V = tegangan efektif (Volt)

I = kuat arus efektif (A)

$\cos \theta = \underline{\text{faktor daya}}$

2.7 LCD (*Liquid Crystal Display*)

Liquid crystal display (LCD) adalah salah satu komponen elektronika yang dapat menampilkan suatu data dari suatu rangkaian listrik, baik karakter, huruf.. LCD (*Liquid Cristal Display*) memiliki fungsi sebagai penampil data baik karakter, huruf, angka ataupun grafik dalam bentuk digital.

LCD memiliki 2 lapisan yaitu lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan elektroda pada kaca belakang. Ketika lcd diberikan dengan medan listrik (tegangan), elektroda dari segmen akan disesuaikan oleh molekul organik.



Gambar 2.4 LCD

Dalam modul LCD (Liquid Cristal Display) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD (Liquid Cristal Display). Mikrokontroler pada suatu LCD (*Liquid Cristal Display*) memiliki sebuah memori dan register. Beberapa memori yang dimiliki mikrokontroler internal LCD adalah :

1. *Display Data Random Access Memory* (DDRAM) merupakan memori tempat menampilkan karakter tersebut .
2. *Character Generator Random Access Memory* (CGRAM) merupakan memori yang berfungsi untuk menggambarkan pola sebuah karakter dan dapat mengubah – ubah bentuk suatu karakter sesuai keinginan,.
3. *Character Generator Read Only Memory* (CGROM) merupakan memori yang hanya dapat untuk membaca suatu karakter dikarenakan pola tersebut adalah karakter dasar yang sudah dipatenkan oleh pembuat LCD (Liquid Cristal Display) tersebut, sehingga pengguna tidak dapat merubah karakter dasar yang ada dalam CGROM dan tinggal mengambilnya sesuai alamat memorinya.

Suatu LCD memiliki bebrapa register kontrol yaitu:

1. Register perintah merupakan register untuk memerintahkan mikrokontroler ke dalam lcd saat proses penulisan data agar data dapat dibaca saat melakukan pengukuran.
2. Register data merupakan register untuk membaca dan menuliskan data. Data yang dituliskan pada register akan ditempatkan ke dalam DDRAM sesuai alamat yang dituju.

2.8 Wireless

Teknologi *wireless* merupakan komunikasi antara dua perangkat elektronik atau lebih tanpa menggunakan kabel atau nirkabel. Untuk dapat berkomunikasi teknologi *wireless* ini menggunakan transmisi frekuensi radio sebagai alat untuk mengirimkan data. Teknologi *wireless* memiliki beberapa sistem kompleks seperti *Wireless Local Area Network (WLAN)* dan telepon selular hingga peralatan sederhana seperti *headphone wireless*, *microphone wireless* dan peralatan lain yang tidak memproses atau menyimpan informasi. *Wireless Local Area Network (WLAN)* adalah jaringan komunikasi antara komputer yang satu dengan komputer lainnya dengan mempergunakan sedikit kabel. Untuk berkomunikasi jaringan komputer tersebut menggunakan gelombang radio sebagai media transmisi datanya. Informasi data ditransfer dari satu komputer ke komputer lain menggunakan gelombang radio. *WLAN* sering disebut sebagai Jaringan Nirkabel atau Jaringan *Wireless*. Berikut ini adalah standarisasi frekuensi pada sebuah jaringan wireless:

Tabel 2.1 Tabel Standarisasi *Wireless*
(Sumber: *Hero ShvrootKit, 2012*)

| Standard | Frequencies | Spectrum Type |
|------------------------------|--|--|
| UMTS over W-CDMA | 850 MHz, 1.9, 1.9/2.1, and 1.7/2.1 GHz | Licensed (Cellular/PCS/3G/AWS) |
| UMTS-TDD | 450, 850 MHz, 1.9, 2, 2.5, and 3.5 GHz ^[4] 2 GHz | Licensed (Cellular, 3G TDD, BRS/IMT-ext, FWA) Unlicensed (see note) |
| CDMA2000 (inc. EV-DO, 1xRTT) | 450, 850, 900 MHz 1.7, 1.8, 1.9, and 2.1 GHz | Licensed (Cellular/PCS/3G/AWS) |
| EDGE/GPRS | 850 MHz 900 MHz 1.8 GHz 1.9 GHz | Licensed (Cellular/PCS/PCN) |
| iBurst | 1.8, 1.9 and 2.1 GHz | Licensed |
| Flash-OFDM | 450 and 870 MHz | Licensed |
| 802.16e | 2.3, 2.5, 3.5, 3.7 and 5.8 GHz | Licensed |
| 802.11a | 5.25, 5.6 and 5.8 GHz | Unlicensed 802.11a and ISM |
| 802.11b/g/n | 2.4 GHz | Unlicensed ISM |
| Bluetooth | 2.4 GHz | Unlicensed ISM |
| Wibree | 2.4 GHz | Unlicensed ISM |
| ZigBee | 868 MHz, 915 MHz, 2.4 GHz | Unlicensed ISM |
| Wireless USB, UWB | 3.1 to 10.6 GHz | Unlicensed Ultrawideband |
| EnOcean* | 868.3 MHz | Unlicensed ISM |

Standarisasi yang digunakan pada *Wireless LAN*, sebagai berikut :

a. **802.11a**

Menggunakan frekuensi 5 GHz dengan kecepatan transfer data hingga 150 Mbps dengan jangkauan 50 km dengan menggunakan perangkat wireless tertentu.

b. **802.11b**

Menggunakan frekuensi 2.4 GHz dengan kecepatan transfer data hingga 11 Mbps.

c. **802.11g**

Menggunakan frekuensi 2.4 GHz dengan kecepatan transfer data hingga 108 Mbps.

d. **802.11n**

Pada standarisasi 802.11n ini dikhususkan untuk jaringan WLAN dengan kecepatan data 450 Mbps dengan menggunakan teknologi 3T3R.

Adapun pengertian lainnya adalah sekumpulan standar yang digunakan untuk Jaringan Lokal Nirkabel (*Wireless Local Area Networks – WLAN*) yang didasari pada spesifikasi IEEE 802.11. Standarisasi menurut IEE 802.11 adalah sebagai berikut:

Tabel 2.2 standar-standar IEEE 802.11

| Standar | Waktu Dikeluarkan | Ruang Lingkup |
|--------------|-------------------|--|
| IEEE 802.11 | 1997 | -kontrol akses medium (MAC): satu lapisan MAC bersama untuk semua aplikasi WLAN -lapisan fisik : infra-merah pada laju 1 dan 2 Mbps -lapisan fisik : FHSS 2,4 GHz pada 1 dan 2 Mbps -lapisan fisik : DSSS 2,4 Ghz pada 1 dan 2 Mbps |
| IEEE 802.11a | 1999 | Lapisan fisik : OFDM 5 Ghz pada laju 6-54 Mbps |
| IEEE 802.11b | 1999 | Lapisan fisik : DSSS 2,4 Ghz pada 5,5 dan 11 Mbps |
| IEEE 802.11c | 2003 | Operasi <i>bridging</i> pada lapisan MAC 802.11 |
| IEEE 802.11d | 2001 | Lapisan fisik : perluasan Operasi WLAN 802.11 ke wilayah-wilayah hukum baru (negara-negara selain AS) |
| IEEE 802.11e | Masih berlanjut | MAC : penyempurnaan untuk kualitas layanan (QOS) dan penyempurnaan mekanisme-mekanisme keamanan |
| IEEE 802.11f | Masih berlanut | Praktik-praktik yang direkomendasikan untuk interoperabilitas titik akses multi-vendor |
| IEEE 802.11g | 2003 | Lapisan fisik : perluasan 802.11b untuk laju data > 20 Mbps |
| IEEE 802.11h | Masih berlanjut | Fisik /MAC : penyempurnaan IEEE 802.11a untuk menambahkan kemampuan pemilihan kanal <i>indoor</i> dan <i>outdoor</i> dan perbaikan manajemen spektrum dan layanan transmisi |
| IEEE 802.11i | Masih berlanjut | MAC : penyempurnaan mekanisme-mekanisme otentikasi dan keamanan data |
| IEEE 802.11j | Masih berlanjut | Fisik : penyempurnaan IEEE 802.11a untuk menyesuaikan kriteria pengguna-pengguna di Jepang |

| | | |
|--------------|-----------------|---|
| IEEE 802.11k | Masih berlanjut | Penyempurnaan mekanisme pengukuran kanal radio dengan penambahan antarmuka pengukuran kinerja kanal radio bagi lapisan-lapisan atas |
| IEEE 802.11m | Masih berlanjut | Perbaikan untuk standarisasi IEEE 802.11 tahun 1999, dengan sejumlah revisi teknis dan redaksional |
| IEEE 802.11n | 2008 | Fisik/MAC : penyempurnaan untuk mencapai <i>throughput</i> yang lebih tinggi |

(sumber : Stalling, 2007)

2.9 Paket Loss

Packet loss merupakan suatu parameter *QOS (Quality of Service)* yang menunjukkan suatu paket yang hilang dalam sebuah jaringan komunikasi antara perangkat satu dengan perangkat lain. Setiap komunikasi sebuah jaringan memiliki *buffer* yang digunakan untuk menerima data. *Buffer* sendiri tidak dapat menerima paket baru jika terjadi suatu antrian yang banyak atau overload pada jaringan.

Berikut tabel kategori QoS berdasarkan standarisasi TIPHON pada parameter *packet loss*:

Tabel 2.3 Klasifikasi QoS parameter *packet loss*

| Kategori <i>Packet Loss</i> | <i>Paket Loss</i> | Indeks |
|-----------------------------|-------------------|--------|
| Sangat Baik | 0 % | 4 |
| Baik | 3 % | 3 |
| Sedang | 15 % | 2 |
| Buruk | 25 % | 1 |

(Sumber : TIPHON, 1999)

Penyebab terjadinya *packet loss*, yaitu:

1. Penyebab terjadinya kongesti dikarenakan terjadinya overload pada jaringan.
2. Node yang bekerja melebihi kapasitas buffer.
3. Memory yang terbatas pada node.
4. Kontrol terhadap jaringan untuk memastikan bahwa jumlah trafik yang mengalir sesuai dengan besarnya bandwidth. Jika besarnya trafik yang mengalir didalam jaringan melebihi dari kapasitas

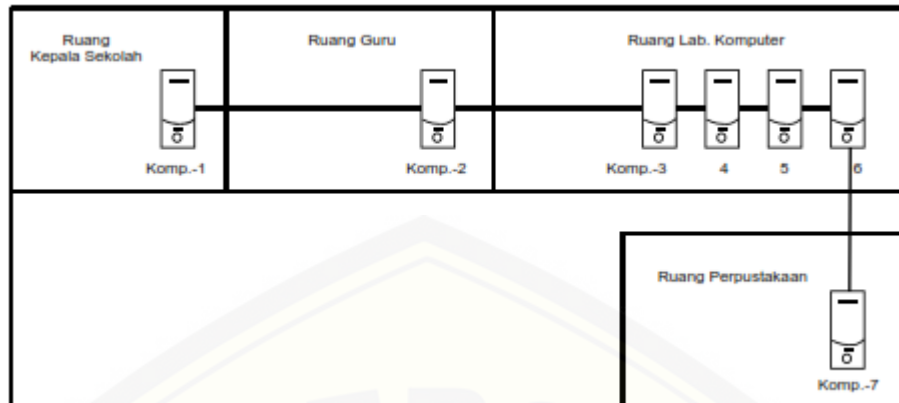
bandwidth yang ada maka policing control akan membuang kelebihan trafik yang ada.

2.10 Model Jaringan Komputer

Banyak jaringan-jaringan yang dapat kita temui sehari-hari. Mulai dari jaring nelayan, raket, laba-laba dan sebagainya. Pada komputer terdapat pula beragam jenis jaringan komputer yang pembagiannya didasarkan pada besar kecilnya cakupan jaringan yang menghubungkan antar komputer. Jenis-jenis jaringan tersebut yaitu :

2.10.1 Model Jaringan *Local Area Network* (LAN)

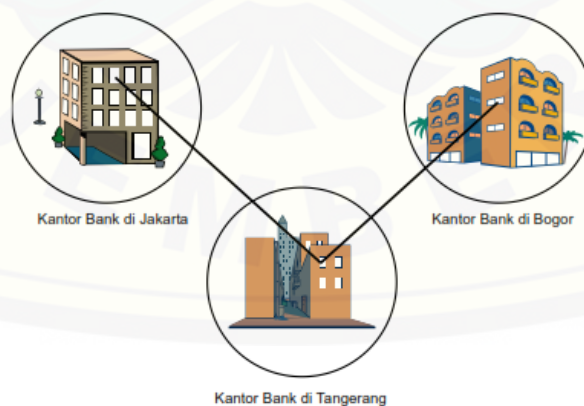
Local Area Network biasa disingkat LAN adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung, atau sebuah sekolah, dan biasanya tidak jauh dari 1 km persegi, secara garis besar LAN terdapat terdapat dua tipe jaringan LAN yaitu jaringan peer to peer dan jaringan client server. Peer to peer artinya adalah setiap komputer yang terhubung ke dalam jaringan dapat bertindak sebagai komputer pengguna (workstation) maupun komputer penyedia layanan (server). Sedangkan pada jaringan clien server hanya ada satu kompuer yang bertindak sebagai server dan yang lain sebagai client. Beberapa model konfigurasi LAN, satu komputer biasanya dijadikan sebuah file server untuk menyimpan perangkat lunak yang mengatur aktifasi jaringan, ataupun sebagai perangkat lunak yang dapat digunakan oleh komputer-komputer yang terhubung ke dalam network. Biasanya kemampuan workstation di bawah file server dan mempunyai aplikasi lain di dalam harddisknya selain aplikasi untuk jaringan. Kebanyakan LAN menggunakan media kabel untuk menghubungkan satu komputer dengan komputer lainnya (Soemarwanto, 2008).



Gambar 2.5 Gambar jaringan LAN
(sumber : Soemarwanto, 2008)

2.10.2 Jaringan *Metropolitan Area Network* (MAN)

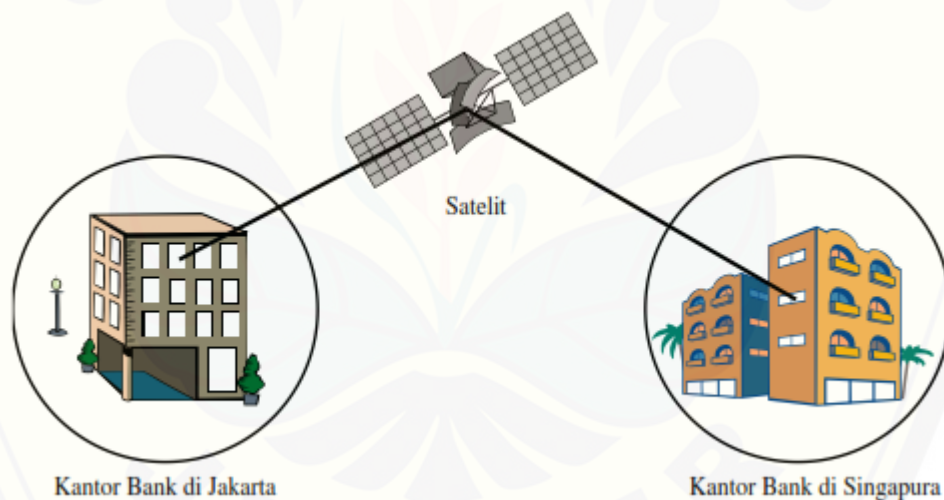
Metropolitan Area Network (MAN) biasanya meliputi area yang lebih besar dari LAN, misalnya antar satu wilayah dalam satu provinsi. Dalam hal ini jaringan menghubungkan beberapa buah jaringan-jaringan kecil ke dalam lingkungan area yang lebih besar, sebagai contoh yaitu : jaringan bank dimana beberapa kantor cabang sebuah bank di dalam sebuah kota besar dihubungkan antara satu dengan yang lainnya. Misalnya bank mandiri yang ada di seluruh wilayah jakarta-bogor-depok-tangerang-bekasi (Soemarwanto, 2008).



Gambar 2.6 Gambar jaringan MAN
(sumber : Soemarwanto, 2008)

2.10.3 Wide Area Network (WAN)

Wide Area Network (WAN) adalah jaringan yang lingkungannya sudah menggunakan sarana satelit atau kabel bawah laut sebagai contoh keseluruhan jaringan BANK MANDIRI yang ada di Indonesia ataupun yang ada di Negara-negara lain. WAN mencakup daerah geografis yang luas, seringkali mencakup sebuah Negara atau benua. Menggunakan sarana WAN, sebuah bank yang ada di Jakarta bisa menghubungi kantor cabangnya yang berada di Singapura, hanya dalam beberapa menit. Biasanya WAN agak rumit dan sangat kompleks, menggunakan banyak sarana untuk menghubungkan antara LAN dan WAN ke dalam komunikasi global seperti internet. Tetapi bagaimapun juga LAN, MAN dan WAN tidak banyak berbeda dalam beberapa hal, hanya lingkup areanya saja yang berbeda (Soemarwanto, 2008).



Gambar 2.7 Gambar jaringan WAN
(sumber : Soemarwanto, 2008)

2.11 Network Analyzer

Analisa jaringan yang juga dikenal sebagai *protocol analysis* merupakan cara pengambilan data dalam dunia telekomunikasi data dan jaringan yang biasanya dilakukan untuk memastikan bagaimana peralatan-peralatan berkomunikasi dan menentukan kesehatan dari sebuah jaringan. Analisa biasa dilakukan terhadap data *traffic* sebuah jaringan yang berhasil ditangkap dengan

menggunakan *software* sejenis tcpdump, sniffer, whireshak dan lain-lain. Sebuah penyaring tangkapan juga dikenal sebagai *pre-filter* bisa mengurangi jumlah *traffic* yang ditangkap ke dalam *trace buffer* (tempat meletakkan paket-paket). Jika penyaing tangkapan tidak diterapkan maka semua *traffic* jaringan mengalir ke dalam *trace buffer*.

2.12 Delay

Delay merupakan salah satu parameter QOS (*Quality of Service*) yang mengukur waktu tunda suatu paket yang dikirim dari satu titik ke titik lain saat melakukan transmisi data. Beberapa golongan yang terdapat pada *delay* sebagai berikut:

a. *Packet Delay*

Delay yang disebabkan saat proses pembentukan paket IP dari informasi *user*. *Delay* ini hanya terjadi sekali saja, yaitu di *source* informasi.

b. *Queuing Delay*

Delay ini disebabkan oleh waktu proses yang diperlukan oleh *router* *micro second*.

c. *Delay Propagasi*

Proses perjalanan informasi selama di dalam media transmisi, misalnya SDH, *coax* atau tembaga yang dapat menyebabkan propagasi.

Berikut ini adalah tabel parameter kualitas sebuah jaringan dilihat dari besarnya *delay* menurut ITU-T:

Tabel 2.4 Standarisasi ITU-T *Delay*

| Kategori <i>Delay</i> | Besar <i>Delay</i> (ms) |
|-----------------------|-------------------------|
| Sangat Bagus | < 150 ms |
| Bagus | 150 – 300 ms |
| Sedang | 300 – 450 ms |
| Jelek | > 450 ms |

Untuk menghitung besaran nilai *delay* dapat digunakan rumus:

$$delay = paket_{terima} - paket_{kirim} \dots\dots\dots(2.1)$$

Berikut tabel kategori QoS berdasarkan standarisasi TIPHON pada parameter *end-to-end delay*

Tabel 2.5 Klasifikasi QoS parameter *end-to-end delay*

| Kategori <i>delay</i> | Besar <i>delay</i> (ms) | Indeks |
|-----------------------|-------------------------|--------|
| Sangat Baik | < 150 ms | 4 |
| Baik | 150 ms s/d 300 ms | 3 |
| Sedang | 300 s/d 450 ms | 2 |
| Buruk | > 450 ms | 1 |

(Sumber : TIPHON, 1999)

2.13 Jitter

Paket yang dikirim dari sumber ke tujuan akan mengalami perbedaan delay pada setiap pengiriman. Sebuah variasi paket *delay* setiap pengirimannya tidak dapat diprediksi. Variasi pengiriman delay terkenal dengan sebutan *jitter*. Nilai *jitter* pada sebuah jaringan bisa bernilai nol (Thethi, 2010). Perhitungan *jitter* dapat dilakukan dengan cara menghitung selisih antara *delay* ke n dengan delay ke n+1. Hasil selisih tersebut mendapatkan hasil yang mutlak tanpa minus.

Berikut tabel kategori QoS berdasarkan standarisasi TIPHON pada parameter *jitter*:

Tabel 2.6 Klasifikasi QoS parameter *jitter*

| Kategori <i>Jitter</i> | <i>Jitter</i> | Indeks |
|------------------------|--------------------|--------|
| Sangat Baik | 0 ms | 4 |
| Baik | 0 ms s/d 75 ms | 3 |
| Sedang | 75 ms s/d 125 ms | 2 |
| Buruk | >125 ms s/d 225 ms | 1 |

(Sumber : Lubis, 2014)

2.14 *Throughput*

Throughput yaitu kecepatan (*rate*) transfer data efektif yang diukur dalam *bit per second* (bps) atau menyesuaikan. Pengukuran *throughput* adalah dengan membagi jumlah paket data yang terkirim dengan total waktu/durasi yang dibutuhkan dalam mengirimkan paket data tersebut (Lubis, 2104).

Berikut tabel kategori QoS berdasarkan standarisasi TIPHON pada parameter *Throughput*:

Tabel 2.7 Klasifikasi QoS parameter *throughput*

| Kategori <i>Throughput</i> | <i>Throughput</i> | Indeks |
|----------------------------|-------------------|--------|
| Sangat Baik | 100 % | 4 |
| Baik | 75 % | 3 |
| Sedang | 5 % | 2 |
| Buruk | > 25 % | 1 |

(Sumber : Lubis, 2014)

2.15 *Smartsniff*

Smartsniff adalah sebuah tool yang bisa digunakan untuk memonitoring sebuah jaringan dan sekaligus mengcapture TCP/IP yang melewati jaringan tersebut dan *smartsniff* ini hanya bias digunakan untuk windows dan dibawah ini adalah gambaran dari proses monitoring menggunakan *smasrtniff*.

| Index | Protocol | Local Address | Remote Address | Local Port | Remote Port | Local Host | Remote Host | Service Name | Packets | Data Size | Total Size | Data Speed |
|-------|----------|-----------------|-----------------|------------|-------------|------------|-----------------------------------|--------------|---------|---------------|---------------|--------------|
| 0 1 | UDP | 192.168.224.130 | 192.168.224.2 | 1027 | 53 | | | domain | 26 | 2,314 Bytes | 3,181 Bytes | 0.1 KB/Sec |
| 0 2 | TCP | 192.168.224.130 | 203.190.242.69 | | | | dekl.com | | 124 | 3,960 Bytes | 7,580 Bytes | 0.1 KB/Sec |
| 0 3 | UDP | 192.168.224.1 | 192.168.224.255 | 137 | 137 | | | netbios-ns | 12 | 600 Bytes | 1,614 Bytes | 0.0 KB/Sec |
| 0 4 | UDP | 192.168.224.1 | 224.0.0.252 | 54083 | 5395 | | | | 1 | 24 Bytes | 104 Bytes | |
| 0 5 | UDP | 192.168.224.1 | 224.0.0.252 | 53092 | 5395 | | | | 1 | 24 Bytes | 104 Bytes | |
| 0 6 | TCP | 192.168.224.130 | 173.191.117.22 | 1143 | 80 | | gmail.com | http | 5 | 1,577 Bytes | 2,182 Bytes | 3.7 KB/Sec |
| 0 7 | TCP | 192.168.224.130 | 74.125.130.17 | 1144 | 443 | | googlemail.l.google.com | https | 35 | 12,763 Bytes | 14,391 Bytes | 0.3 KB/Sec |
| 0 8 | TCP | 192.168.224.130 | 114.4.42.91 | 1145 | 80 | | clients.l.google.com | http | 25 | 11,620 Bytes | 13,426 Bytes | 0.7 KB/Sec |
| 0 9 | TCP | 192.168.224.130 | 74.125.68.94 | 1146 | 443 | | accounts.l.google.com | https | 19 | 9,494 Bytes | 10,666 Bytes | 2.2 KB/Sec |
| 0 10 | TCP | 192.168.224.130 | 74.125.130.18 | 1147 | 443 | | googlemail.l.google.com | https | 413 | 431,208 Bytes | 441,944 Bytes | 13.0 KB/Sec |
| 0 11 | UDP | 192.168.224.130 | 192.168.224.2 | 1044 | 53 | | | domain | 30 | 2,570 Bytes | 3,476 Bytes | 0.1 KB/Sec |
| 0 12 | TCP | 192.168.224.130 | 74.125.130.18 | 1149 | 443 | | googlemail.l.google.com | https | 236 | 246,204 Bytes | 256,122 Bytes | 8.2 KB/Sec |
| 0 13 | TCP | 192.168.224.130 | 74.125.130.18 | 1148 | 443 | | googlemail.l.google.com | https | 206 | 216,961 Bytes | 225,619 Bytes | 7.4 KB/Sec |
| 0 14 | TCP | 192.168.224.130 | 74.125.130.95 | 1153 | 443 | | googlemaps.l.google.com | https | 19 | 7,803 Bytes | 8,703 Bytes | 0.3 KB/Sec |
| 0 15 | TCP | 192.168.224.130 | 114.4.42.95 | 1154 | 443 | | clients.l.google.com | https | 19 | 10,912 Bytes | 10,979 Bytes | 0.4 KB/Sec |
| 0 16 | TCP | 192.168.224.130 | 114.4.42.95 | 1155 | 443 | | clients.l.google.com | https | 27 | 18,607 Bytes | 19,894 Bytes | 0.7 KB/Sec |
| 0 17 | TCP | 192.168.224.130 | 114.4.42.95 | 1156 | 443 | | clients.l.google.com | https | 20 | 11,470 Bytes | 12,477 Bytes | 0.4 KB/Sec |
| 0 18 | TCP | 192.168.224.130 | 114.4.42.123 | 1160 | 443 | | clients.l.google.com | https | 107 | 100,598 Bytes | 105,076 Bytes | 3.8 KB/Sec |
| 0 19 | TCP | 192.168.224.130 | 114.4.42.123 | 1161 | 443 | | clients.l.google.com | https | 85 | 78,693 Bytes | 82,513 Bytes | 3.0 KB/Sec |
| 0 20 | TCP | 192.168.224.130 | 74.125.68.132 | 1164 | 443 | | googlehosted.l.google.com | https | 34 | 29,570 Bytes | 31,150 Bytes | 9.0 KB/Sec |
| 0 21 | TCP | 192.168.224.130 | 114.4.42.95 | 1157 | 443 | | clients.l.google.com | https | 18 | 13,637 Bytes | 14,776 Bytes | 106.5 KB/Sec |
| 0 22 | TCP | 192.168.224.130 | 114.4.42.101 | 1163 | 443 | | clients.l.google.com | https | 17 | 8,872 Bytes | 9,760 Bytes | 32.6 KB/Sec |
| 0 23 | TCP | 192.168.224.130 | 74.125.130.18 | 1159 | 443 | | googlemail.l.google.com | https | 15 | 12,674 Bytes | 13,692 Bytes | 131.7 KB/Sec |
| 0 24 | TCP | 192.168.224.130 | 74.125.130.18 | 1152 | 443 | | googlemail.l.google.com | https | 18 | 15,327 Bytes | 16,165 Bytes | 104.8 KB/Sec |
| 0 25 | TCP | 192.168.224.130 | 74.125.130.18 | 1151 | 443 | | googlemail.l.google.com | https | 24 | 26,294 Bytes | 27,872 Bytes | 71.3 KB/Sec |
| 0 26 | TCP | 192.168.224.130 | 74.125.200.97 | 1159 | 443 | | ssl-google-analytics.l.google.com | https | 33 | 23,228 Bytes | 24,765 Bytes | 29.7 KB/Sec |
| 0 27 | TCP | 192.168.224.130 | 74.125.130.18 | 1162 | 443 | | clients.l.google.com | https | 98 | 94,287 Bytes | 99,167 Bytes | 1.4 KB/Sec |

Gambar 2.8 Gambar *smartsniff*

2.16 Protokol TCP dan UDP

Transport Layer Protocol adalah lapisan keempat dari model referensi OSI dan jantung dari hierarki protokol secara keseluruhan karena protokol ini berfungsi untuk menyediakan data transport dari komputer asal atau sumber menuju ke komputer tujuan, yang tidak bergantung pada jaringan fisik atau jaringan-jaringan yang digunakan. Tanpa adanya Transport Layer Protocol ini maka seluruh konsep protocol yang menggunakan layer menjadi tidak ada fungsinya.

Transport layer protocol inilah yang juga mengatur koneksi dari suatu komputer pengirim ke komputer penerima serta juga yang membangun koneksi logic antara host pengirim dengan client penerima dalam sebuah jaringan. Layer ini juga mengatur dan mengaplikasikan layanan transport yang efektif antar jaringan untuk layer-layer di atasnya.

Dalam TCP/IP Transport Layer digunakan 2 macam protokol, yakni TCP dan UDP. TCP (Transmission Control Protocol) adalah protokol yang bertugas untuk membentuk koneksi antar node. Sifat TCP adalah connection-oriented. TCP baru akan membuat koneksi jika kedua belah pihak telah setuju. Karenanya, TCP dianggap reliable (dapat diandalkan). Berbeda dengan UDP (User Datagram Protocol) yang connectionless, transmisi data yang berbasis UDP akan langsung mengirimkan paket karena tidak ada kesepakatan dulu antar node yang bertransmisi.

Masing-masing protokol memiliki karakteristik tertentu dan mendukung protokol-protokol pada layer di atasnya. Misalnya TCP mendukung HTTP dan FTP, sementara UDP mendukung DNS dan TFTP. Perbedaan antara kedua protokol tersebut ada pada reliabilitasnya. Untuk menjalankan tugasnya baik TCP dan UDP menambahkan header pada data yang akan dikirim. Isi header antara kedua protokol tersebut berbeda, sesuai dengan karakteristik masing-masing protokol. Header yang dipasang oleh kedua protokol tersebut dapat identifikasi dan dianalisis dengan menggunakan network analyzer tool, salah satunya adalah Wireshark.

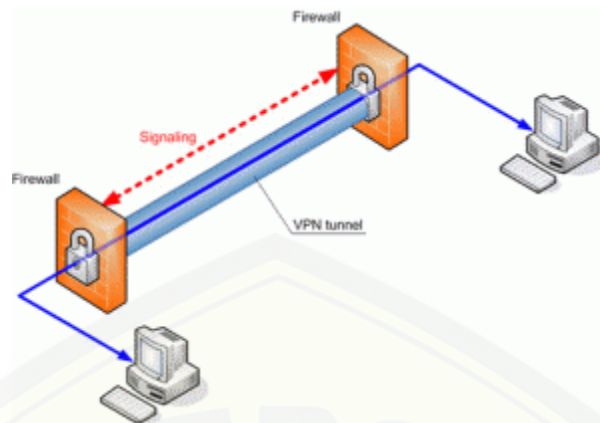
Simpelnya dapat dianalogikan yaitu pada TCP, sebelum computer A mengirim data pada computer B, computer A terlebih dahulu akan mengenali alamat logic dan fisiknya, setelah itu computer A akan memberikan kode pada computer B saat pengiriman paket karena menunggu respon dari computer B terlebih dahulu, computer A akan mengkalkulasi rute yang akan ditempuh sekaligus menentukan rute cadangan bilamana terjadi kegagalan pengiriman. Berbeda pada UDP, computer A akan langsung mengirim paket tanpa pemberitahuan pada computer B terlebih dahulu tanpa menentukan rute khusus untuk antisipasi sehingga paket akan langsung dikirim sehingga UDP mempunyai kelebihan dalam respon kecepatan, sedangkan TCP dalam kehandalan.

Pada kenyataannya, TCP digunakan untuk transmisi yang sifatnya kritical dan tentu saja, butuh kehandalan. Sedangkan UDP digunakan untuk komunikasi antar proses yang tidak begitu memerlukan kehandalan, justru membutuhkan kecepatan respon.

Aplikasi – aplikasi yang sering digunakan untuk protokol TCP dan UDP:

1. UDP

- a. Tunelling VPN



VPN merupakan singkatan dari ***Virtual Private Network***, yaitu sebuah koneksi private melalui jaringan publik (dalam hal ini internet). Disini ada 2 kata yang dapat digaris bawah yaitu:

1. ***Virtual network***, yang berarti jaringan yang terjadi hanya bersifat virtual. Tidak ada koneksi jaringan secara riil antara dua titik yang akan berhubungan.
2. ***Private***, jaringan yang terbentuk *bersifat private* dimana tidak semua orang bisa mengaksesnya. Data yang dikirimkan *terenkripsi* sehingga tetap rahasia meskipun melalui jaringan publik.

Dengan VPN ini kita seolah-olah membuat jaringan didalam jaringan atau biasa disebut *tunnel* (terowongan). Adapun pengertian ***Tunneling*** itu sendiri adalah suatu cara membuat jalur privat dengan menggunakan infrastruktur pihak ketiga. VPN menggunakan salah satu dari tiga teknologi tunneling yang ada yaitu: *PPTP*, *L2TP* dan standar terbaru, *Internet Protocol Security* (biasa disingkat menjadi *IPSec*). VPN merupakan perpaduan antara teknologi tunneling dan enkripsi.

b. Media Streaming

Video streaming adalah istilah yang sering kita gunakan saat melihat video diinternet melalui browser dimana kita tidak perlu *men-download* file video tersebut untuk dapat memutarinya. Istilah ini tersebut terdiri dari dua suku kata yaitu video dan streaming, secara istilah video berarti teknologi untuk menangkap,

merekam, memproses, mentransmisikan dan menata ulang gambar bergerak, sedangkan streaming berarti proses penghantaran data dalam aliran berkelanjutan dan tetap yang memungkinkan pengguna mengakses dan menggunakan file sebelum data dihantar sepenuhnya.



c. Games (Update Rutin)

Maksudnya adalah game yang selalu mengupdate contentnya. Contohnya adalah DOTA 2, Audition dll.



2. TCP

a. Web

Website adalah sekumpulan halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet.



b. SSH, FTP, TELNET

FTP (File Transfer Protocol) adalah sebuah protocol internet yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pentransferan berkas (file) computer antar mesin-mesin dalam sebuah internetwork.

SSH adalah protocol jaringan yang memungkinkan pertukaran data secara aman antara dua komputer.

TELNET (Tele Network) adalah suatu terminal yang dapat digunakan untuk mengakses resource yang ada di computer Server, adapun cara untuk mengakses computer server tersebut kita dapat menggunakan beberapa tools yang sudah disediakan missal (putty, winscp, winftp, dll).



c. SMTP, Sending Mail

Komunikasi dengan *e-mail* mungkin saat ini merupakan salah satu aplikasi yang paling luas dipakai pada internet. Ada beberapa protokol yang dapat digunakan untuk melayani transfer *e-mail*, tetapi yang paling umum digunakan adalah *Simplemail-transfer-protocol* (SMTP).



d. IMAP, POP, Receiving mail



Adapun perbedaan yang penting antara TCP dan UDP itu adalah :

1. UDP adalah “*datagram oriented*”, TCP adalah “*session-oriented*”. Datagram adalah paket informasi self-contained; UDP berhubungan dengan datagram atau paket individu yang dikirim dari client ke server, atau sebaliknya.
2. UDP adalah *connectionless*. Client tidak membangun koneksi ke server sebelum mengirim data – client hanya mengirim data secara langsung.
3. UDP “tidak andal” dalam pengertian jaringan formal :
 1. Paket dapat hilang. UDP tidak dapat mendeteksinya.
 2. Program aplikasi – client atau server – (sebagai kebalikan TCP/IP stack sendiri) harus mendeteksi paket yang hilang dan menangani transmisi ulang, dan lain-lain. Aplikasi sering menunggu hingga timeout habis, dan kemudian mencoba lagi.
 3. Paket dapat mengalami kerusakan. Paket UDP berisi checksum semua data dalam paket. Checksum ini memungkinkan UDP mendeteksi kapan suatu paket mengalami kerusakan. Jika hal ini terjadi, maka paket tersebut dikeluarkan, dan sebagaimana biasa aplikasi-lah yang harus mendeteksi hal ini dan melakukan transmisi ulang sepenuhnya.
 4. Operasi checksum ini dapat dihentikan, dan beberapa aplikasi melakukannya untuk alasan unjuk kerja. Akan tetapi hal ini dapat berarti paket yang rusak tidak terdeteksi atau layer aplikasi harus melakukan pemeriksaan integritas data sendiri, hal ini merupakan false economy (penghematan finansial yang sebenarnya menuju pada pengeluaran yang lebih besar)
 5. Karena UDP adalah datagram-oriented dan pada level protokol setiap paket berdiri sendiri, maka UDP tidak memiliki konsep paket sesuai urutan, yang selanjutnya berarti tidak memerlukan nomor urut pada paket tersebut.
 6. Sejak pertama kali dikembangkan, TCP telah dilengkapi dengan mekanisme yang sangat canggih untuk mengendalikan kecepatan

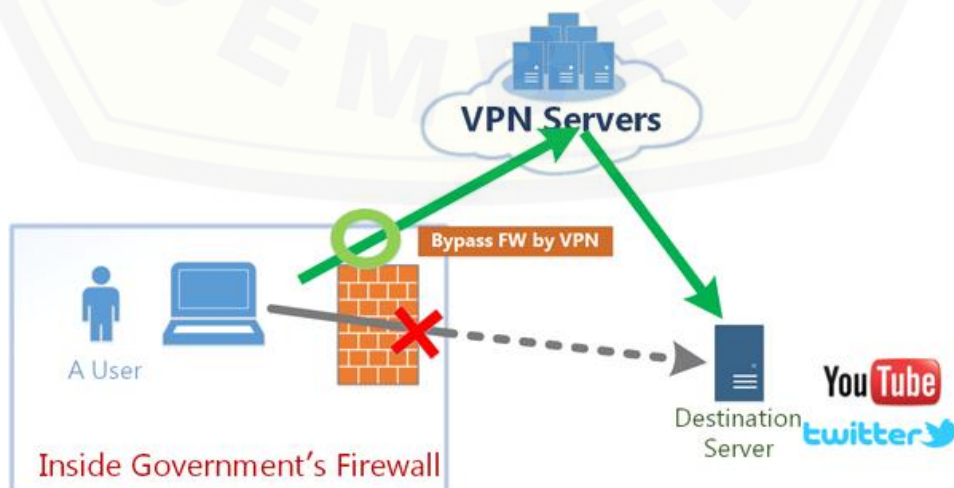
aliran dalam koneksinya, untuk menghindari kemacetan dan kehilangan paket yang berlebihan. Karena UDP hanya mengirim paket tunggal, yang berdiri sendiri, maka UDP tidak memerlukan mekanisme kontrol yang rumit. Hal ini membuat UDP lebih mudah dan lebih kecil (dalam baris data dan memori) untuk diimplementasikan, tetapi juga membuatnya tidak cocok untuk sejumlah besar data.

Selain itu ada beberapa kemiripan antara UDP dan TCP antara lain:

1. UDP adalah protokol transport: UDP hanya berhubungan dengan komunikasi antara dua end point (misalnya aplikasi client pada mesin Anda, dan aplikasi server pada mesin remote). Intermediate router tidak berhubungan dengan data UDP dalam paket yang dikirimkannya – router hanya beroperasi pada layer IP atau network lower-down.
2. UDP menggunakan port untuk membedakan antara traffic dari banyak aplikasi UDP pada mesin yang sama, dan untuk mengirim paket yang tepat ke aplikasi yang sesuai (ini disebut demultiplexing). UDP dan port-nya menyediakan interface antara program aplikasi dan layer networking IP.

2.17 VPN (*Virtual Private Network*)

Layanan VPN didukung oleh beberapa protokol komunikasi data, yang mana tiap protokol tersebut memiliki konsep keamanan yang berbeda-beda. Mari kita mengenal lebih jauh konsep dan jenis tiap VPN.



1. Point-to-Point Transfer Protocol (PPTP)

PPTP, adalah protokol yang paling populer dan didukung oleh banyak perangkat, yang paling mudah untuk menginstal, dan overhead setidaknya untuk digunakan. Kelemahan untuk menggunakan PPTP adalah bahwa ia menggunakan kunci enkripsi yang lemah (128 bit) sehingga tidak boleh digunakan dengan transfer data sensitif.

2. Layer 2 Transfer Protocol (L2TP)

L2TP dengan sendirinya tidak jauh berbeda dari PPTP karena mengandalkan titik ke titik protokol untuk menghubungkan. L2TP tidak aman dengan sendirinya dan sering dipasangkan dengan metode enkripsi luar protokol seperti IPSec dan 3DES. Menambahkan enkripsi ke protokol ini memberikan overhead yang lebih tinggi dibandingkan dengan protokol lain.

3. Internet Protocol Security (IPSec)

Internet Protocol Security – IPSec adalah dipercaya enkripsi dan tunneling protokol yang menggunakan enkripsi pada lalu lintas IP melalui terowongan yang diberikan. Kerugian untuk IPSec mungkin memakan waktu instalasi klien.

4. Internet Key Exchange (IKEv2)

Dalam komputasi, Internet Key Exchange (IKE, kadang-kadang IKEv1 atau IKEv2, tergantung pada versi) adalah protokol yang digunakan untuk mendirikan sebuah asosiasi keamanan (SA) di IPsec protokol. IKE dibangun berdasarkan protokol Oakley dan ISAKMP.

5. Routing Protocols (MPLS/BGP)

Multi-Protocol Label Switching (MPLS) tidak cocok digunakan oleh pengguna biasa sebaliknya sangat ideal untuk perusahaan atau bisnis dengan

koneksi situs ke situs (site-to-site). Semua konfigurasi dilakukan di tingkat jaringan sehingga pengguna akhir akan melihat koneksi jaringan yang nyata saja.

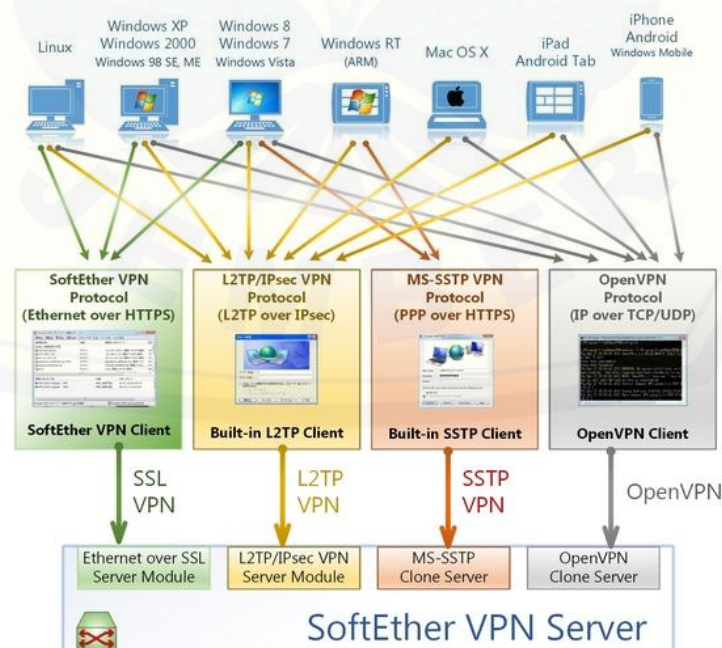
6. Secure Socket Tunneling Protocol (SSTP)

Bentuk terowongan VPN yang menyediakan mekanisme untuk mengangkut PPP atau L2TP. Lalu lintas melalui SSL / TLS saluran. SSL / TLS memberikan keamanan transportasi tingkat dengan kunci-negosiasi, enkripsi dan integritas lalu lintas memeriksa.

7. SSL-VPN

Bentuk VPN yang dapat digunakan dengan browser Web standar. Berbeda dengan VPN tradisional Internet Protocol Security (IPsec), SSL VPN tidak memerlukan instalasi perangkat lunak klien khusus di komputer pengguna akhir.

Seluruh fungsi-fungsi diatas dapat ditemui pada Softether VPN (Sebuah program VPN yang mendukung SSL-VPN, L2TP/IPsec, OpenVPN dan protokol-protokol SSTP. Lihat gambar dibawah.



SoftEther VPN tidak hanya server VPN alternatif untuk produk VPN yang ada (OpenVPN, IPsec dan MS-SSTP). SoftEther VPN juga memiliki protokol SSL-VPN asli kuat untuk menembus segala jenis firewall. Ultra-dioptimalkan SSL-VPN Protokol SoftEther VPN memiliki throughput yang sangat cepat, latency rendah dan ketahanan firewall.



BAB 3. METODE PENELITIAN

Pada bab ini akan dipaparkan tentang metode penelitian yang nantinya sebagai tata cara atau aturan dalam penelitian agar sistem penelitian berjalan secara teratur dan mendapatkan hasil yang diinginkan.

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilaksanakan di Laboratorium Telekomunikasi dan Terapan, Fakultas Teknik Universitas Jember. Dengan lama waktu pengerjaan selama 8 bulan mulai bulan Oktober 2015 hingga bulan Mei 2016.

3.2 Hardware

3.2.1 Arduino UNO

Arduino Uno merupakan alat elektronik yang mempunyai sifat *open source* yang mempunyai bahasa pemrograman yaitu bahasa C dan dapat membaca suatu file yang telah terdeteksi oleh sensor, didalam arduino itu sendiri memiliki 14 pin input dimana 6 pin input tersebut digunakan sebagai output PWM dan 6 pin input analog. Untuk menjalankan arduino tersebut, kita hanya menghubungkan board arduino ke dalam laptop atau dapat juga menggunakan sebuah baterai.

3.2.2 Spesifikasi Sensor Arus

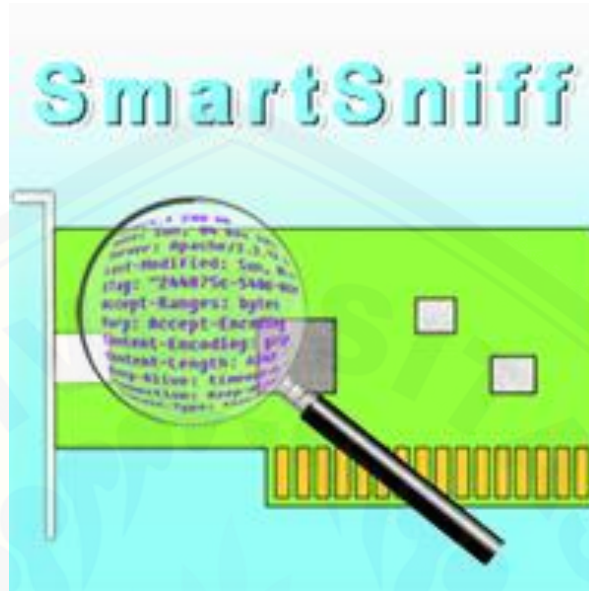
1. Spesifikasi Input Current: 0 ~ 100A
2. Keluaran AC : 0 ~ 50mA
3. Resistance Grade: Grade B
4. Temperatur Kerja: -25 ° C ~ + 70 ° C
5. Kekuatan Dielektrik (antara shell dan output): 1000V AC / 1min 5mA

3.3 Software

3.3.1 Smartniff

Smartniff adalah sebuah tool yang bisa digunakan untuk memonitoring sebuah jaringan dan sekaligus mengcapture TPC/IP yang melewati sebuah jaringan yang sama dan smartsniff ini hanya bias digunakan untuk windows dan dibawah ini adalah gambar dari smartsniff. Kita dapat melihat TCP / IP

percakapan dalam modus Ascii (untuk protokol berbasis teks, seperti HTTP, SMTP, POP3 dan FTP.) Atau sebagai hex dump.



Gambar 3.1 Tampilan Smartsniff

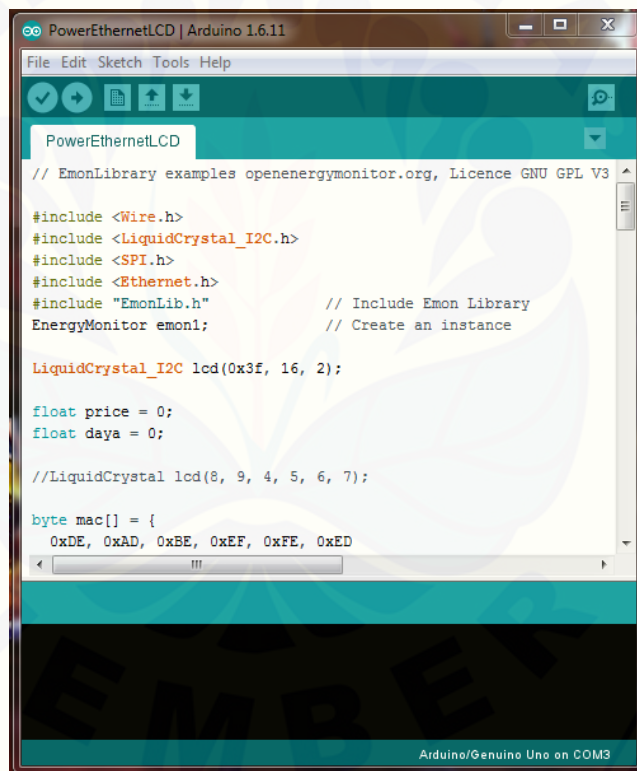
Metode untuk menangkap TCP / IP paket pada smartsniff memiliki 3 metode yaitu:

1. *Raw sockets* direkomendasikan hanya untuk windows XP atau lebih besar tanpa menginstall driver. Dalam metode ini memiliki kelebihan dan kekurangan
2. *WinPcap Capture Driver*: Dapat digunakan pada semua sistem operasi windows untuk menangkap TCP / IP paket. Diantara lain yaitu Windows 98/ME/NT/2000/XP/2003/Vista untuk menggunakannya, tetapi sebelumnya kita harus mendownload dan menginstal Driver WinPcapnya terlebih dahulu dari situs Web agar dapat digunakan. Metode ini lebih sering digunakan untuk menangkap TCP / paket IP dengan SmartSniff, dan bekerja lebih baik dibandingkan dengan metode *Raw Sockets*.
3. *Microsoft Network Monitor Driver* hanya dapat digunakan untuk Windows 2000/XP/2003. Microsoft menyediakan driver bebas di bawah

Windows 2000/XP/2003 yang dapat digunakan oleh SmartSniff, tapi driver ini tidak diinstal secara default.

3.3.2 Arduino IDE

Arduino IDE adalah software dari arduino uno yang menghasilkan format file berupa hex yang dapat didownload pada arduino atau mikrokontroler lain. Dalam software arduino ini kita memprogram suatu perang elektronik yang akan digunakan untuk mengukur sebuah daya sebelum memasuki ke dalam board arduino. Kita dapat merubah atau mengedit suatu file yang akan digunakan sebelum pengukuran dilakukan.



```
PowerEthernetLCD | Arduino 1.6.11
File Edit Sketch Tools Help
PowerEthernetLCD
// EmonLibrary examples openenergymonitor.org, Licence GNU GPL V3

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <Ethernet.h>
#include "EmonLib.h" // Include Emon Library
EnergyMonitor emon1; // Create an instance

LiquidCrystal_I2C lcd(0x3F, 16, 2);

float price = 0;
float daya = 0;

//LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};

Arduino/Genuino Uno on COM3
```

Gambar 3.2 Tampilan Arduino IDE

3.4 Prosedur penelitian

Metode penelitian yang akan dilakukan secara bertahap akan diuraikan langkah-langkah dalam penelitian sebagai berikut:

- a. Perumusan Masalah: Merumuskan masalah dari beberapa hal yang berkaitan dengan difokuskan untuk sistem monitoring konsumsi energi pada rumah menggunakan arduino berbasis Wireless.
- b. Studi literature terhadap objek dan penelitian
Mengumpulkan dan mempelajari literature atau landasan teori yang berkaitan dengan menguji dan menganalisa mengenai sistem monitoring, konsumsi energi dan parameter pengujian.
- c. Perancangan alat
Tahap perancangan yang pertama dilakukan adalah melengkapi semua alat dan bahan yang diperlukan. Selanjutnya membuat diagram blok sistem secara keseluruhan, setelah membuat diagram blok selanjutnya melakukan perancangan alat yang meliputi perangkat keras dan perangkat lunak sistem.
- d. Pengambilan Data
Data yang digunakan adalah data dari hasil pengujian keandalan alat dan keandalan sistem yang meliputi proses pengiriman data menggunakan Wireless dan Arduino
- e. Analisis Sistem
Menganalisa parameter-parameter yang harus diamati ketika melakukan pengambilan data serta menganalisa data yang telah diperoleh dari sistem yang berjalan.
- f. Pengambilan kesimpulan dan saran
Pengambilan kesimpulan dari semua hasil analisis data yang telah didapat berdasarkan dasar teori dan pengujian.

3.5 Realisasi Pembuatan Alat

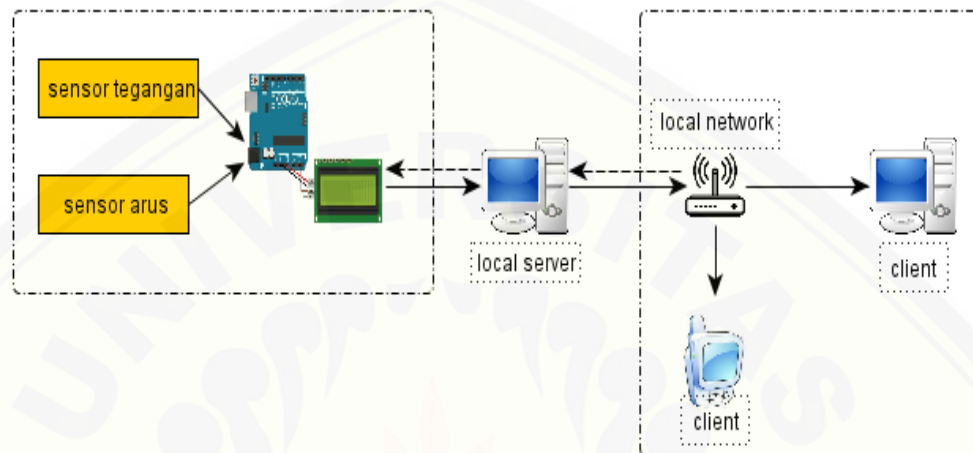
Pembuatan sistem monitoring konsumsi energi pada rumah menggunakan arduino berbasis Wireless meliputi tahapan-tahapan sebagai berikut:

1. Pembuatan mekanik sistem berdasarkan pada perancangan.
2. Pembuatan *sketch* arduino sesuai dengan diagram alir yang telah direncanakan.
3. Pembuatan akses pengiriman data melalui wireless

3.6 Perancangan Alat

3.6.1 Perancangan Sistem

Gambaran umum perancangan sistem monitoring konsumsi energi pada rumah menggunakan arduino berbasis Wireless dijelaskan sebagai berikut:



Gambar 3.3 Blok diagram

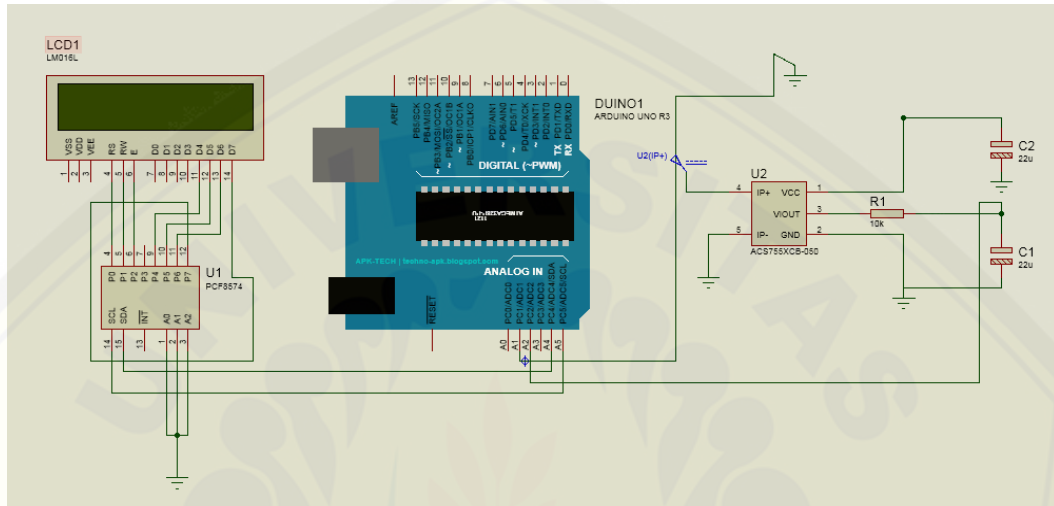
Pada gambar 3.3 dijelaskan bahwa pertama sensor arus SCTO13.000 100A/50mAh dan sensor tegangan akan mengambil data dari sebuah sumber tegangan dan sumber arus atau titik pengamatan yang kemudian data tersebut akan dikirim kedalam arduino. Nilai arus dan tegangan yang telah terbaca oleh arduino akan diubah menjadi besaran daya dan kemudian akan ditampilkan ke dalam LCD untuk melihat apakah nilai daya yang ditampilkan tepat atau tidak.

Dalam proses pengiriman data, sensor dan Arduino disini sebagai Tx (*Transmitter*) dimana nilai dari sensor arus dan tegangan akan membaca nilai arus dan tegangan dan diproses didalam Arduino yang nantinya akan dikirimkan ke server sebagai Rx (*Received*). Transmitter yaitu rangkaian alat yang akan mengambil nilai besaran daya dari sensor dan diolah didalam Arduino.

Dalam proses pengamatan, klien harus melakukan konfigurasi terhadap local server menggunakan *IP address* yang sudah ditentukan didalam program arduino agar klien dapat mengakses web dimanapun dan kapanpun asalkan klien terhubung dengan internet.

3.6.2 Perancangan Hardware

Pada bab kali ini akan di jelaskan perancangan hardware pada sistem monitoring energy listrik pada rumah menggunakan wireless dan hardware – hardware yang digunakan pada saat pembuatan alat dan koneksi pin arduino yang digunakan. Gambaran perancangan hardware dijelaskan sebagai berikut:



Gambar 3.4 Perancangan hardware

Pada gambar 3.4 dimana hardware yang digunakan untuk membuat alat sistem monitoring energy listrik yang digunakan adalah LCD (*Liquid Crystal Display*), I2C (*Inter Integrated Circuit*), Arduino dan Sensor Arus. Pin yang digunakan untuk dapat menghubungkan i2c dan lcd sebagai berikut:

Tabel 3.1 Koneksi pin LCD ke I2C

| No | Pin I2C (<i>Inter Integrated Circuit</i>) | Pin LCD (<i>Liquid Crystal Display</i>) |
|----|--|--|
| 1 | P0 | RS |
| 2 | P1 | RW |
| 3 | P2 | E |
| 4 | P4 | D4 |
| 5 | P5 | D5 |
| 6 | P6 | D6 |
| 7 | P7 | D7 |

Pada tabel diatas dimana pin P0 pada I2C terhubung pada pin RS pada LCD, pin P1 pada I2c terhubung pada pin RW pada LCD, pin P2 pada I2c terhubung pada pin E pada LCD, pin P4 pada I2c terhubung pada pin D4 pada LCD, pin P5 pada I2c terhubung pada pin D5 pada LCD, pin P6 pada I2c terhubung pada pin D6 pada LCD, pin P7 pada I2c terhubung pada pin D7 pada LCD,

Ketika LCD dan I2C terhubung maka selanjutnya menghubungkan atau mengkoneksikan antara LCD dan Arduino yang akan di jelaskan pada tabel berikut:

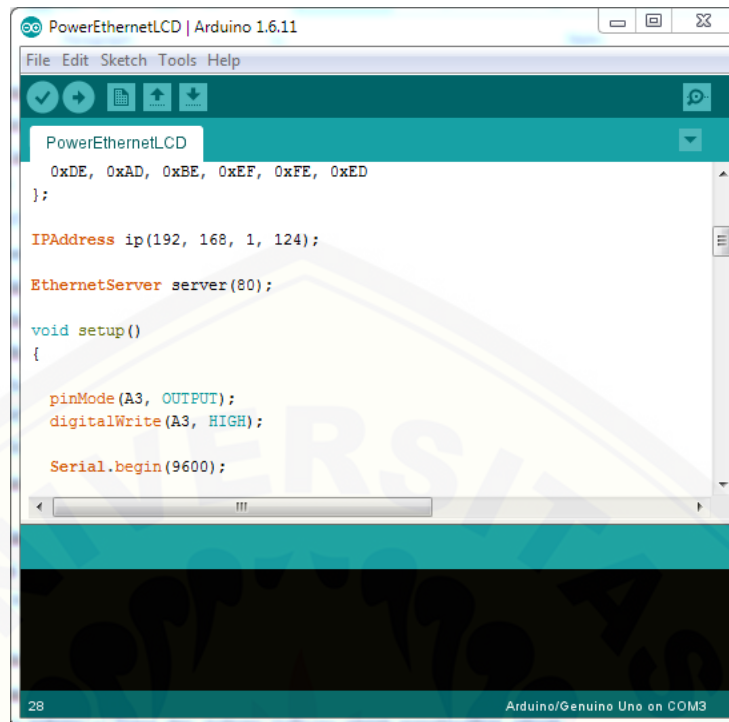
Tabel 3.2 Koneksi pin arduino ke LCD

| No | Arduino | Pin I2C (<i>Inter Integrated Circuit</i>) |
|----|---------|--|
| 1 | A4 | SDA |
| 2 | A5 | SCL |
| 3 | A1 | GND |
| 4 | GND | A0 |
| 5 | GND | A1 |
| 6 | GND | A2 |

Pada tabel diatas dimana pin A4 pada arduino terhubung pada pin SDA I2c, pin A5 pada arduino terhubung pada pin SCL I2c, pin A1 pada arduino terhubung terhadap GND, dan pin A0, A1 dan A2 pada I2c terhubung terhadap GND. Dengan demikian arduino dapat terhubung dengan LCD, dimana nantinya nilai yang terbaca oleh arduino dapat ditampilkan kedalam LCD.

3.6.3 Perancangan Software

Untuk melakukan konfigurasi jaringan supaya Arduino dapat terhubung ke jaringan internet secara *wireless*, kita harus memasukkan *IPAddress* ip(192, 168, 1, 124); dan *EthernetServer* server(80); untuk pemogramannya sendiri akan dijelaskan pada gambar berikut ini:



Gambar 3.5 Tampilan Arduino

Setelah tersimpan selanjutnya mengatur nilai harga per KWh agar nantinya dapat muncul dalam web maupun lcd. Harga per KWh mengikuti harga terbaru dari PLN.

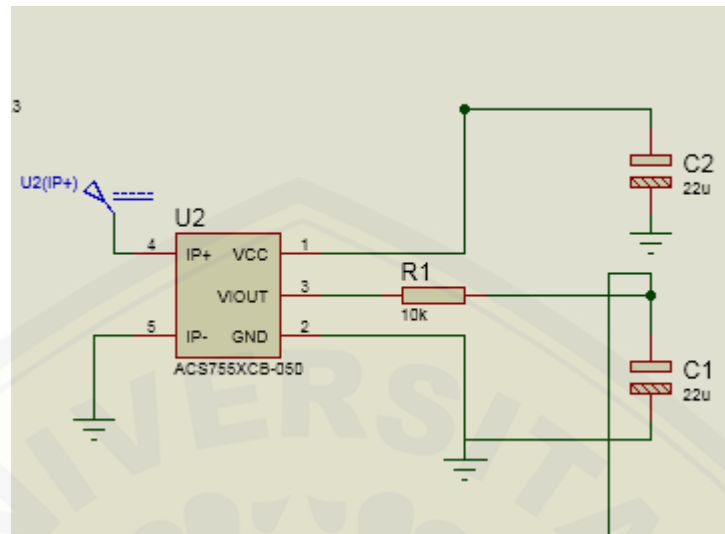
```
client.print("Harga : ");
client.print("Rp ");
client.print(price * 500); //Rubah Harga
client.println("<br />");

lcd.setCursor(0, 1);
lcd.print("Rp: ");
lcd.print(price * 500); //Rubah Harga
```

Gambar 3.6 Tampilan Pengaturan Rupiah

Setelah melakukan konfigurasi jaringan agar arduino dapat terhubung maka arduino sekarang memiliki ip 192, 168, 1, 124. Dan web akan memunculkan nilai daya dan rupiah yang dapat diakses oleh client melalui pc maupun hp ketika terhubung dengan internet.

3.6.4 Perancangan Sensor



Gambar 3.7 Sensor arus

Untuk mendeteksi arus yang mengalir dalam suatu beban ini menggunakan sensor arus SCTO13.000 100A/50mAh. Sensor arus SCTO13.000 100A/50mAh ini berfungsi untuk mendeteksi besar arus yang mengalir dalam suatu beban. Sensor ini dapat mendeteksi besar arus hingga 30A dan sinyal arus ini dapat dibaca melalui analog IO port Arduino. Konfigurasi Pin dari sensor arus ini dapat dilihat pada tabel berikut:

Tabel 3.3 Konfigurasi pin

| Nomor | Nama | Keterangan |
|---------|---------------|--|
| 1 dan 2 | IP+ | Pin mendeteksi arus |
| 3 dan 4 | IP- | Pin mendeteksi arus |
| 5 | GND | Pin <i>Ground</i> |
| 6 | <i>Filter</i> | Pin untuk kapasitor eksternal yang digunakan menentukan <i>bandwidth</i> |
| 7 | Vout | Arus keluaran yang dihitung |
| 8 | VCC | Tegangan Power supply 5 V |

Untuk dapat melakukan pengukuran yang maksimal selanjutnya mengatur sensor agar nilai yang nantinya didapat memiliki sedikit error persen terhadap nilai sebenarnya. Untuk pemrogramannya sendiri dijelaskan pada gambar berikut:

```

void setup()
{
  Serial.begin(9600);
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("Bismillah");
  delay(3000);
  emon1.voltage(2, 115.40, 1.7); // Voltage: input pin, calibration, phase_shift
  emon1.current(1, 111.1);      // Current: input pin, calibration.
}

```

Gambar 3.8 Pengaturan sensor

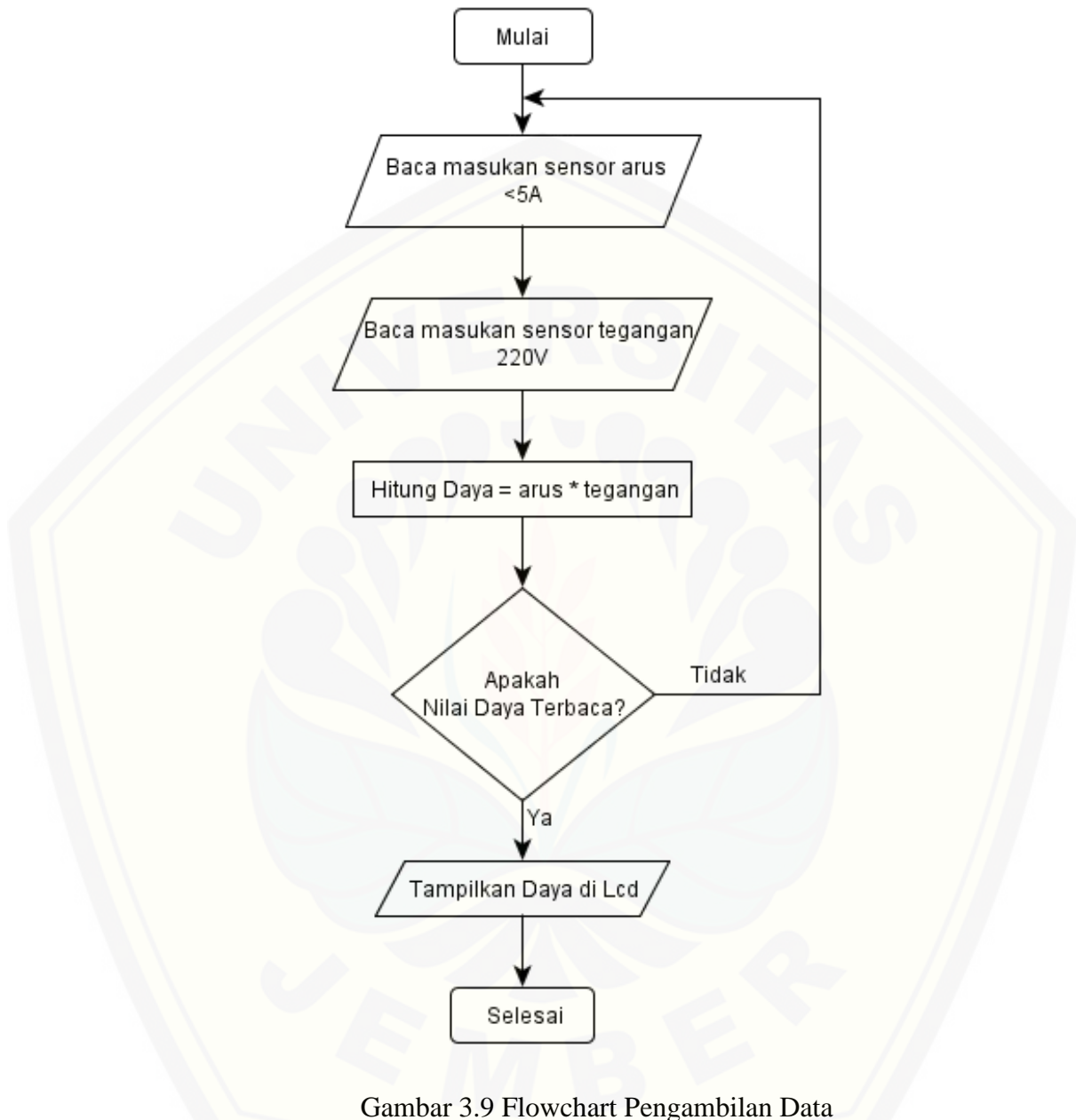
Pada gambar 3.8 dijelaskan dimana pada baris *emon1.voltage(2, 115.40, 1.7)*; digunakan untuk mengatur nilai tegangan dan pada baris *emon1.current(1, 111.1)*; digunakan untuk mengatur nilai arus. Hasil dari pengaturan sensor dapat dilihat pada tabel dibawah ini:

Tabel 3.4 Pengaturan Sensor

| Hasil ukur dengan Multimeter | | Hasil ukur dengan sistem | |
|------------------------------|----------------|--------------------------|----------------|
| Arus (I) | Tegangan (V) | Arus (I) | Tegangan (V) |
| 0.15 | 223 | 0.13 | 220 |
| 0.13 | 224 | 0.12 | 220 |
| 0.14 | 226 | 0.16 | 220 |
| 0.15 | 223 | 0.14 | 220 |
| 0.16 | 226 | 0.14 | 220 |

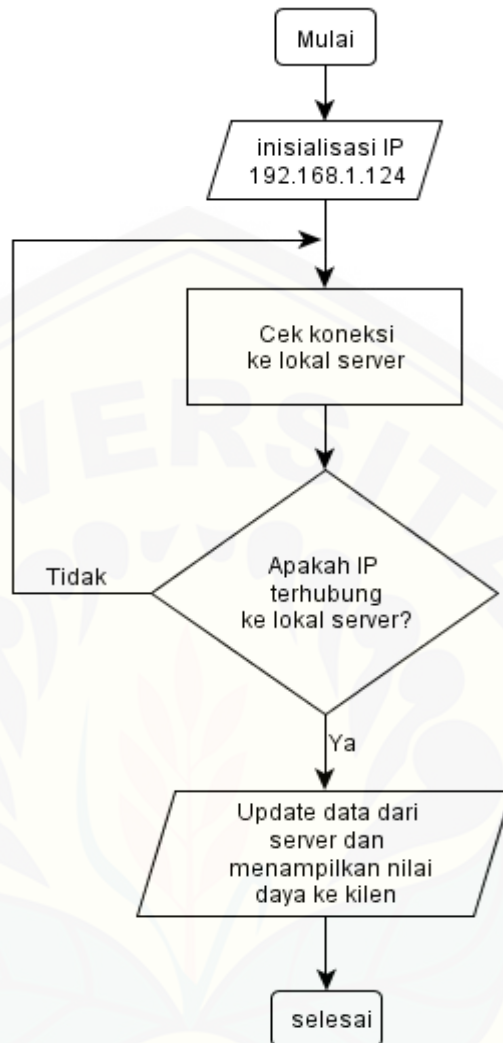
Pada tabel 3.4 diatas didapatkan hasil ukur dengan multimeter dan hasil ukur dengan sistem didapatkan nilai terbesar untuk arus dan tegangan dengan menggunakan multimeter sebesar 0.16 A dan 226 V, sedangkan untuk nilai arus dan tegangan dengan menggunakan system didapatkan sebesar 0.16 A dan 220 V.

3.7 Flowchart



Gambar 3.9 Flowchart Pengambilan Data

Pada saat system diaktifkan, sensor arus dan sensor tegangan akan mengambil data dari suatu objek. Setelah sensor arus dan sensor tegangan mengambil data, selanjutnya data akan dikirimkan ke pada arduino untuk membaca data yang telah diambil oleh sensor arus dan sensor tegangan. Dari Arduino data tersebut akan diubah menjadi nilai daya dan ditampilkan ke dalam Lcd sebagai keluarannya.



Gambar 3.10 Flowchart nilai daya dari klien

Pada saat sistem diaktifkan sensor arus dan tegangan akan mengambil nilai arus dan tegangan yang telah diubah menjadi nilai daya saat terbaca arduino. Nilai daya akan ditampilkan kedalam web agar dapat terbaca oleh klien. Selanjutnya data tersebut dapat diakses menggunakan *web browser* oleh klien melalui alamat IP Address 192.168.1.124 yang didapatkan dari local server dengan menggunakan protocol TCP.

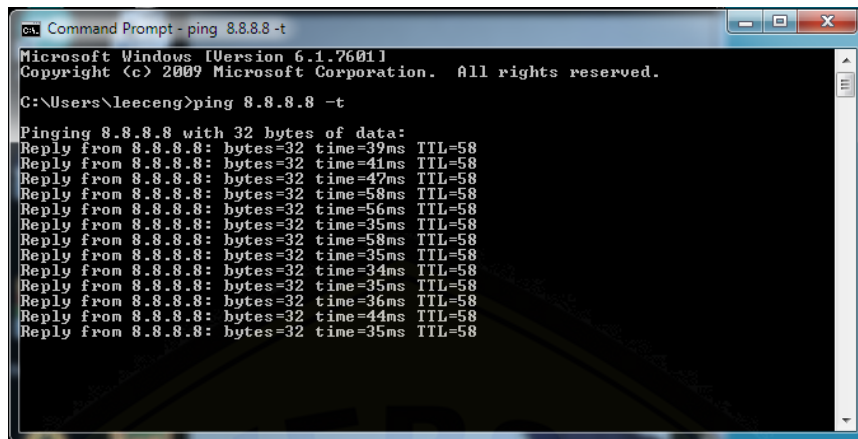
3.8 Konfigurasi VPN

Pada bab ini dimana akan dijelaskan proses pengukuran menggunakan protokol UDP. Untuk melakukan proses pengukuran menggunakan protokol UDP kali ini menggunakan sistem VPN dan software yang digunakan yaitu Hide.me VPN.

Untuk dapat mengakses software Hide.me hal yang harus dilakukan adalah membuat akun Hide.me yang nantinya akan digunakan untuk menjalankan software Hide.me. Di dalam software Hide.me sendiri memiliki berbagai lokasi yang dapat digunakan sebagai server yaitu Singapura, Belanda, dan Kanada dan memiliki beberapa protokol VPN yaitu IKEv2, OpenVPN, SoftetherVPN, SSTP, dan PPTP. Selanjutnya *login* dengan *username* dan *password* yang telah dibuat sebelumnya dan memilih server yang nantinya akan digunakan saat pengukuran. Untuk setiap server memiliki alamat IP yang berbeda – beda seperti pada server Singapura memiliki IP 43.249.38.69, untuk server Belanda memiliki IP 109.201.137.53 dan untuk server Kanada memiliki IP 67.215.13.253.

Untuk mempermudah dapat dilakukan registrasi menggunakan akun Gmail. Pada menu *setting – protocol* terdapat beberapa protokol yang dapat digunakan akan tetapi lebih baik menggunakan protokol yang telah direkomendasikan, selanjutnya setelah selesai memilih protokol dan server yang digunakan aktifkan software Hide.me.

Selanjutnya untuk mengetahui apakah software Hide.me berjalan dengan baik dengan cara Ping IP dengan *command prompt* (*C:\Users\leecheng>ping 8.8.8.8 -t*).

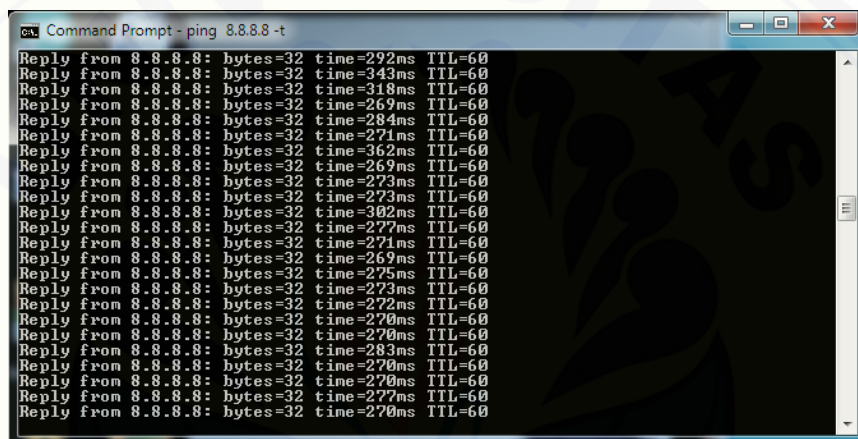


```
Command Prompt - ping 8.8.8.8 -t
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\leeceng>ping 8.8.8.8 -t

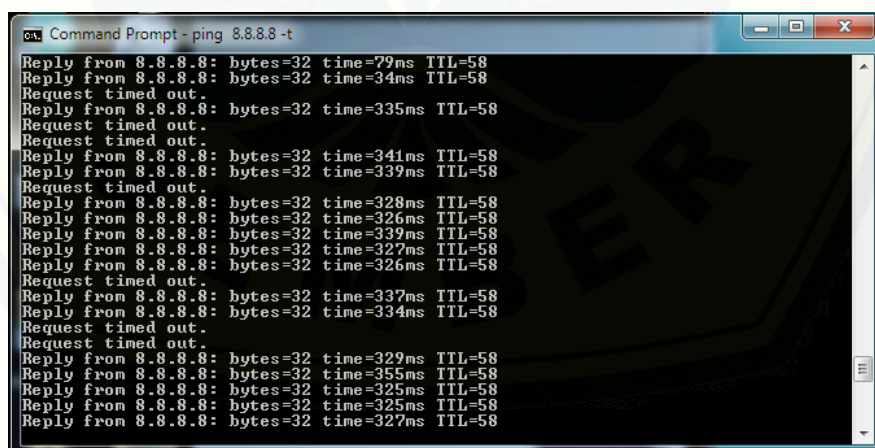
Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=39ms TTL=58
Reply from 8.8.8.8: bytes=32 time=41ms TTL=58
Reply from 8.8.8.8: bytes=32 time=47ms TTL=58
Reply from 8.8.8.8: bytes=32 time=58ms TTL=58
Reply from 8.8.8.8: bytes=32 time=56ms TTL=58
Reply from 8.8.8.8: bytes=32 time=35ms TTL=58
Reply from 8.8.8.8: bytes=32 time=58ms TTL=58
Reply from 8.8.8.8: bytes=32 time=35ms TTL=58
Reply from 8.8.8.8: bytes=32 time=34ms TTL=58
Reply from 8.8.8.8: bytes=32 time=35ms TTL=58
Reply from 8.8.8.8: bytes=32 time=36ms TTL=58
Reply from 8.8.8.8: bytes=32 time=44ms TTL=58
Reply from 8.8.8.8: bytes=32 time=35ms TTL=58
```

Gambar 3.15 Tampilan Ping tanpa VPN



```
Command Prompt - ping 8.8.8.8 -t
Reply from 8.8.8.8: bytes=32 time=292ms TTL=60
Reply from 8.8.8.8: bytes=32 time=343ms TTL=60
Reply from 8.8.8.8: bytes=32 time=318ms TTL=60
Reply from 8.8.8.8: bytes=32 time=269ms TTL=60
Reply from 8.8.8.8: bytes=32 time=284ms TTL=60
Reply from 8.8.8.8: bytes=32 time=271ms TTL=60
Reply from 8.8.8.8: bytes=32 time=362ms TTL=60
Reply from 8.8.8.8: bytes=32 time=269ms TTL=60
Reply from 8.8.8.8: bytes=32 time=273ms TTL=60
Reply from 8.8.8.8: bytes=32 time=273ms TTL=60
Reply from 8.8.8.8: bytes=32 time=302ms TTL=60
Reply from 8.8.8.8: bytes=32 time=277ms TTL=60
Reply from 8.8.8.8: bytes=32 time=271ms TTL=60
Reply from 8.8.8.8: bytes=32 time=269ms TTL=60
Reply from 8.8.8.8: bytes=32 time=275ms TTL=60
Reply from 8.8.8.8: bytes=32 time=273ms TTL=60
Reply from 8.8.8.8: bytes=32 time=272ms TTL=60
Reply from 8.8.8.8: bytes=32 time=270ms TTL=60
Reply from 8.8.8.8: bytes=32 time=270ms TTL=60
Reply from 8.8.8.8: bytes=32 time=283ms TTL=60
Reply from 8.8.8.8: bytes=32 time=270ms TTL=60
Reply from 8.8.8.8: bytes=32 time=270ms TTL=60
Reply from 8.8.8.8: bytes=32 time=277ms TTL=60
Reply from 8.8.8.8: bytes=32 time=270ms TTL=60
```

Gambar 3.16 Tampilan Ping menggunakan Server Belanda



```
Command Prompt - ping 8.8.8.8 -t
Reply from 8.8.8.8: bytes=32 time=79ms TTL=58
Reply from 8.8.8.8: bytes=32 time=34ms TTL=58
Request timed out.
Reply from 8.8.8.8: bytes=32 time=335ms TTL=58
Request timed out.
Request timed out.
Reply from 8.8.8.8: bytes=32 time=341ms TTL=58
Reply from 8.8.8.8: bytes=32 time=339ms TTL=58
Request timed out.
Reply from 8.8.8.8: bytes=32 time=328ms TTL=58
Reply from 8.8.8.8: bytes=32 time=326ms TTL=58
Reply from 8.8.8.8: bytes=32 time=339ms TTL=58
Reply from 8.8.8.8: bytes=32 time=327ms TTL=58
Reply from 8.8.8.8: bytes=32 time=326ms TTL=58
Request timed out.
Reply from 8.8.8.8: bytes=32 time=337ms TTL=58
Reply from 8.8.8.8: bytes=32 time=334ms TTL=58
Request timed out.
Request timed out.
Reply from 8.8.8.8: bytes=32 time=329ms TTL=58
Reply from 8.8.8.8: bytes=32 time=355ms TTL=58
Reply from 8.8.8.8: bytes=32 time=325ms TTL=58
Reply from 8.8.8.8: bytes=32 time=325ms TTL=58
Reply from 8.8.8.8: bytes=32 time=327ms TTL=58
```

Gambar 3.17 Tampilan Ping menggunakan Server Kanada

Dari ketiga server yang digunakan terlihat jelas perbedaan saat jaringan menggunakan VPN dan jaringan tanpa menggunakan VPN. Ping saat tidak menggunakan VPN terlihat lebih normal sedangkan Ping saat menggunakan VPN

terlihat lebih besar dan semakin jauh server yang digunakan semakin besar juga Ping yang akan didapat.

Selanjutnya agar dapat melihat kinerja dari VPN itu sendiri dapat dilakukan menggunakan *software wireshark* dengan salah satu server yang nantinya akan digunakan seperti gambar berikut:

| No. | Time | Source | Destination | Protocol | Length | Time delta from previous displayed frame | Info |
|-----|----------|----------------|-------------|----------|--------|--|----------------------|
| 583 | 4.685137 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.009221000 | ESP (SPI=0x8c20d52a) |
| 584 | 4.697539 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.012402000 | ESP (SPI=0x8c20d52a) |
| 587 | 4.702967 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.003922000 | ESP (SPI=0x8c20d52a) |
| 588 | 4.703773 | 109.201.137.41 | 192.168.1.3 | ESP | 126 | 0.000060000 | ESP (SPI=0x8c20d52a) |
| 590 | 4.712486 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.007449000 | ESP (SPI=0x8c20d52a) |
| 595 | 4.721548 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.002215000 | ESP (SPI=0x8c20d52a) |
| 597 | 4.729965 | 109.201.137.41 | 192.168.1.3 | ESP | 1454 | 0.000631000 | ESP (SPI=0x8c20d52a) |
| 599 | 4.742973 | 109.201.137.41 | 192.168.1.3 | ESP | 158 | 0.012807000 | ESP (SPI=0x8c20d52a) |
| 610 | 4.964306 | 109.201.137.41 | 192.168.1.3 | ESP | 174 | 0.011918000 | ESP (SPI=0x8c20d52a) |
| 637 | 5.033982 | 109.201.137.41 | 192.168.1.3 | ESP | 142 | 0.014007000 | ESP (SPI=0x8c20d52a) |
| 618 | 5.034969 | 109.201.137.41 | 192.168.1.3 | ESP | 126 | 0.000087000 | ESP (SPI=0x8c20d52a) |
| 625 | 5.186966 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.024859000 | ESP (SPI=0x8c20d52a) |
| 626 | 5.196987 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.010021000 | ESP (SPI=0x8c20d52a) |
| 629 | 5.208696 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.009596000 | ESP (SPI=0x8c20d52a) |
| 630 | 5.213568 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.004872000 | ESP (SPI=0x8c20d52a) |
| 633 | 5.222632 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.002814000 | ESP (SPI=0x8c20d52a) |
| 634 | 5.231345 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.000713000 | ESP (SPI=0x8c20d52a) |
| 637 | 5.239994 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.001645000 | ESP (SPI=0x8c20d52a) |
| 638 | 5.249162 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.009168000 | ESP (SPI=0x8c20d52a) |
| 640 | 5.258001 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.008502000 | ESP (SPI=0x8c20d52a) |
| 643 | 5.267984 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.005784000 | ESP (SPI=0x8c20d52a) |
| 645 | 5.275988 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.007785000 | ESP (SPI=0x8c20d52a) |
| 646 | 5.284996 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.009008000 | ESP (SPI=0x8c20d52a) |
| 648 | 5.285983 | 109.201.137.41 | 192.168.1.3 | ESP | 222 | 0.000779000 | ESP (SPI=0x8c20d52a) |
| 650 | 5.294988 | 109.201.137.41 | 192.168.1.3 | ESP | 1486 | 0.008225000 | ESP (SPI=0x8c20d52a) |

Gambar 3.18 Tampilan *Wireshark*

Dapat dilihat bahwa sistem dapat mengirimkan data dari server dengan IP 109.201.137.53 ke klien dengan alamat IP 192.168.1.3 dengan baik. Protokol yang digunakan yaitu protokol *ESP (Encapsulating Security Payload)*. Protokol *ESP (Encapsulating Security Payload)* ini sendiri melakukan enkapsulasi serta enkripsi terhadap data pengguna untuk meningkatkan kerahasiaan data. *ESP* juga dapat memiliki skema autentikasi dan perlindungan dari beberapa serangan dan dapat digunakan sendirian atau bersamaan dengan *authentication header*. Seperti halnya *AH (Authentication Header)*, informasi mengenai *ESP* juga dimasukkan ke dalam *Header* paket IP yang dikirimkan.

```

* Frame 393: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface 0
  Interface id: 0 (\Device\NPF_{69119983-ADC9-4360-8DB9-D6CC7A6DC11A})
  Encapsulation type: Ethernet (1)
  Arrival Time: Feb 25, 2018 17:25:30.086518000 SE Asia Standard Time
  [Time shift for this packet: 0.000000000 seconds]
  Epoch Time: 1519554330.086518000 seconds
  [Time delta from previous captured frame: 0.000235000 seconds]
  [Time delta from previous displayed frame: 0.000235000 seconds]
  [Time since reference or first frame: 3.350931000 seconds]
  Frame Number: 393
  Frame Length: 126 bytes (1008 bits)
  Capture Length: 126 bytes (1008 bits)
  [Frame is marked: False]
  [Frame is ignored: False]
  [Protocols in frame: eth:ethertype:ip:udp:udpencap:esp]
  [Coloring Rule Name: UDP]
  [Coloring Rule String: udp]

```

Gambar 3.19 Tampilan *Frame Wireshark*

Untuk waktu paket saat dikirimkan dan diterima didapatkan *Time since reference or first frame: 3.350931000 seconds* dan untuk data yang terlihat *Time delta from previous captured frame: 0.000235000 seconds* dan untuk data yang telah terbaca *Time delta from previous displayed frame: 0.000235000 seconds*.

```
Internet Protocol Version 4, Src: 192.168.1.3, Dst: 109.201.137.41
  0100 .... = Version: 4
  ... 0101 = Header Length: 20 bytes
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 112
    Identification: 0x046c (1132)
  ▸ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
  ▸ Header checksum: 0xfe72 [validation disabled]
    Source: 192.168.1.3
    Destination: 109.201.137.41
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
```

Gambar 3.20 *Protocol UDP*

Untuk *internet protocol version 4, Src: 192.168.1.3, Dst: 109.201.137.41*. *Header Length* sebesar 20 bytes. Jenis protokol yang digunakan yaitu protokol UDP.

Tabel 3.5 Perbedaan Protokol TCP dan UDP

| No | TCP | UDP |
|----|--|--|
| 1. | Beroperasi berdasarkan konsep koneksi. | Tidak berdasarkan konsep koneksi, jadi harus membuat kode sendiri. |
| 2. | Jaminan pengiriman - penerimaan data akan reliable dan teratur. | Tidak ada jaminan bahwa pengiriman dan penerimaan data akan reliable dan teratur, sehingga paket data mungkin dapat kurang, terduplikat, atau bahkan tidak sampai sama sekali. |
| 3. | Secara otomatis memecah data ke dalam paket-paket. | Pemecahan ke dalam paket-paket dan proses pengirimannya dilakukan secara manual. |
| 4. | Tidak akan mengirimkan data terlalu cepat sehingga memberikan jaminan koneksi internet dapat menanganinya. | Harus membuat kepastian mengenai proses transfer data agar tidak terlalu cepat sehingga internet masih dapat menanganinya. |
| 5. | Mudah untuk digunakan, transfer paket data seperti menulis dan membaca file. | Jika paket ada yang hilang, perlu dipikirkan di mana letak kesalahan yang terjadi dan mengirim ulang data yang diperlukan. |

BAB 5. PENUTUP

5.1 Kesimpulan

Dari hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut :

1. Penggunaan *wireless* untuk memonitoring daya pada rumah mampu menghasilkan nilai daya dan rupiah. Hasil monitoring tersebut dapat diakses melalui *web* dengan URL <http://192.168.1.124> dalam jaringan yang sama.
2. Tegangan *input* pada pengukuran 1 sampai 5 memiliki rata-rata sebesar 224 *volt*, sementara tegangan keluaran saat mendeteksi sebesar 2.51 *volt*. Sementara *delay* yang dibutuhkan untuk mengirimkan notifikasi dalam satu jaringan ke klien sebesar 1.365 *second*.
3. Rata – rata error persen yang didapat pada tegangan sebesar 1.4 % dan arus sebesar 1.3 %.
4. Web tidak akan berjalan atau memunculkan nilai daya dan rupiah jika alat tidak terhubung oleh modem/internet.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan terdapat beberapa kekurangan atau kendala, berikut ini merupakan saran untuk pengembangan lebih lanjut :

1. Menambahkan database server agar nantinya nilai daya dan rupiah dapat disimpan dan dipanggil kembali oleh user.
2. Sebaiknya menggunakan mikrokontroler yang lebih efektif.



Daftar Pustaka

Temy Nusa. 2015, Sistem monitoring konsumsi energi listrik secara real time berbasis mikrokontroler. Teknik elektro FT. UNSRAT, Manado.

Hestry Angraini, Yeffry Handoko P. 2015, Sistem monitoring energi listrik menggunakan mikrokontroler berbasis web. Teknik komputer Unikom, Bandung.

Chaunghai Peng Kun Qian "Development and application of zigbee-based building energy monitoring and control system",the scientific world journal volume 2014.

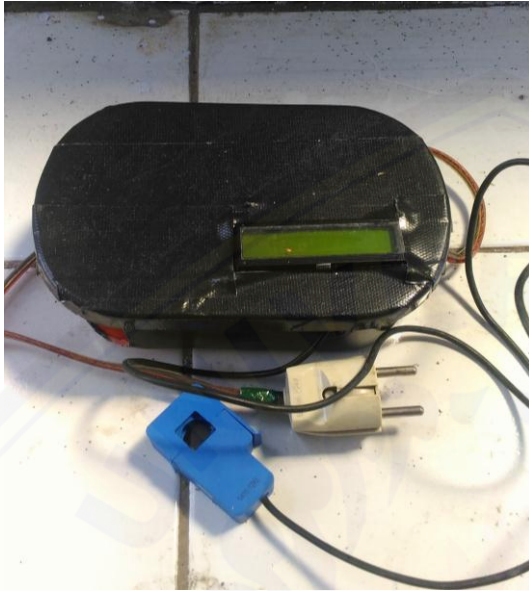
Sakti Wibisana, Boromeus, 2008. Analisis perbandingan pembacaan kWh meter analog dengan kWh meter digital pada ketidakseimbangan beban. Universitas Indonesia, Depok.

Sunanda, Wahri dan Irwan Dinata. "Penerapan Perangkat Wireless Monitoring Energi Listrik Berbasis Arduino dan Internet". Jurnal Amplifier, ISSN: 2089-2020, Volume 04 Nomor 02, November 2014: 21-23.

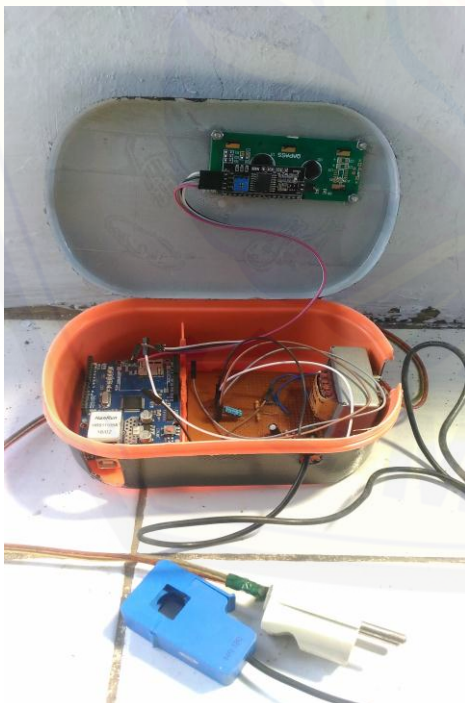
Riswandi, Perancangan alat monitoring arus KWH (Kilo Watt Hours) Meter Tiga Fasa dengan Memanfaatkan Mikrokontroler Arduino dan Sms Gateway Berbasis Web. Progam studi teknik informatika, STMIK Cikarang.

LAMPIRAN

A. Dokumentasi Foto dan Alat



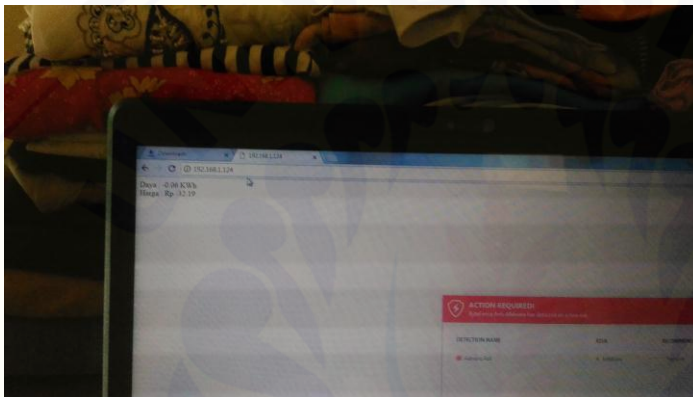
Gambar alat keseluruhan



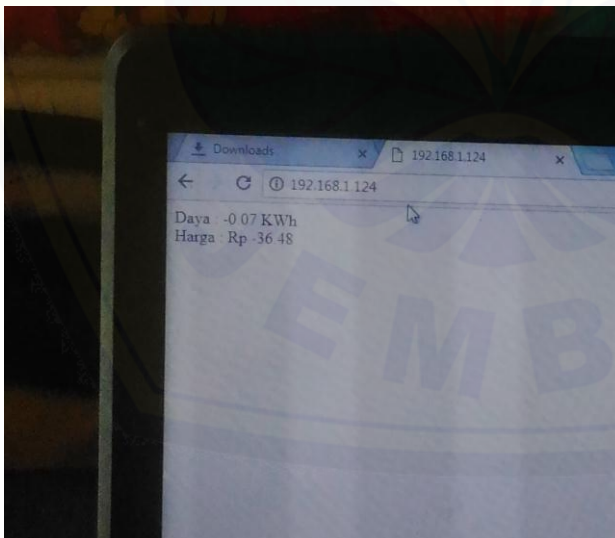
Gambar rangkaian



Gambar tampilan LCD



Gambar tampilan web clien



Gambar tampilan web server



Gambar tampilan multimeter arus



Gambar tampilan multimeter Arus



Gambar tampilan multimeter tegangan



B. Listing program notifikasi web

```
// EmonLibrary examples openenergymonitor.org, Licence
GNU GPL V3

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <Ethernet.h>
#include "EmonLib.h"           // Include Emon
Library
EnergyMonitor emon1;         // Create an instance

LiquidCrystal_I2C lcd(0x3f, 16, 2);

float price = 0;
float daya = 0;

//LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};

IPAddress ip(192, 168, 1, 124);

EthernetServer server(80);

void setup()
{
  pinMode(A3, OUTPUT);
  digitalWrite(A3, HIGH);

  Serial.begin(9600);

  lcd.begin();

  lcd.backlight();
  lcd.print("System On");

  while (!Serial) {
  }

  Ethernet.begin(mac, ip);
  server.begin();
  Serial.print("server is at ");
```

```
Serial.println(Ethernet.localIP());

//lcd.begin(16, 2);
//lcd.print("Bismillah");
//delay(3000);

emon1.voltage(2, 234.26, 1.7); // Voltage: input
pin, calibration, phase shift
emon1.current(1, 111.1); // Current: input pin,
calibration.
}

void loop()
{
EthernetClient client = server.available();
if (client) {
Serial.println("new client");
boolean currentLineIsBlank = true;
while (client.connected()) {
if (client.available()) {
char c = client.read();
Serial.write(c);
if (c == '\n' && currentLineIsBlank) {
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("Connection: close");
client.println("Refresh: 5");
client.println();

emon1.calcVI(20, 2000); // Calculate
all. No.of half wavelengths (crossings), time-out
emon1.serialprint(); // Print out
all variables (realpower, apparent power, Vrms, Irms,
power factor)

float realPower = emon1.realPower;
//extract Real Power into variable
float apparentPower = emon1.apparentPower;
//extract Apparent Power into variable
float powerFactor = emon1.powerFactor;
//extract Power Factor into Variable
float supplyVoltage = emon1.Vrms;
//extract Vrms into Variable
float Irms = emon1.Irms;
//extract Irms into Variable
```

```
    lcd.clear();
    lcd.print("VA: ");
    lcd.print(realPower / 100000);
    lcd.print(" KWh");

    client.print("Daya : ");

    daya = daya + (realPower / 100000);

    client.print(daya);
    client.print(" kWh");
    client.println("<br />");

    price = price + (realPower / 100000);

    client.print("Harga : ");
    client.print("Rp ");
    client.print(price * 500); //Rubah Harga
    client.println("<br />");

    lcd.setCursor(0, 1);
    lcd.print("Rp: ");
    lcd.print(price * 500); //Rubah Harga

    delay(1000);

    break;
}

}

delay(1);
client.stop();
Serial.println("client disconnected");
}
```

```
}
```

C. Listing program notifikasi sensor

```
// EmonLibrary examples openenergymonitor.org, Licence
GNU GPL V3

#include <LiquidCrystal.h>

#include "EmonLib.h"           // Include Emon
Library

EnergyMonitor emon1;         // Create an instance

// initialize the library with the numbers of the
interface pins
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

void setup()
{
  Serial.begin(9600);
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("Bismillah");
  delay(3000);
  emon1.voltage(2, 125.40, 1.7); // Voltage: input
pin, calibration, phase_shift
  emon1.current(1, 111.1);      // Current: input pin,
calibration.
}

void loop()
{
  emon1.calcVI(20,2000);       // Calculate all.
No.of half wavelengths (crossings), time-out
```



```
    emon1.serialprint();           // Print out all
variables (realpower, apparent power, Vrms, Irms, power
factor)

    float realPower      = emon1.realPower;
//extract Real Power into variable

    float apparentPower  = emon1.apparentPower;
//extract Apparent Power into variable

    float powerFactor    = emon1.powerFactor;
//extract Power Factor into Variable

    float supplyVoltage  = emon1.Vrms;
//extract Vrms into Variable

    float Irms           = emon1.Irms;
//extract Irms into Variable

lcd.clear();
lcd.print("daya:");
lcd.setCursor(6,1);
lcd.print("KWh");
lcd.setCursor(0,1);
lcd.print(realPower/1000);
delay(1000);
}
```

D. Linstening notifikasi wireshark

Frame 2: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits) on interface 0

Interface id: 0 (\Device\NPF_{691199B3-ADC9-4360-8DB9-D6CC7A6DC11A})

Encapsulation type: Ethernet (1)

Arrival Time: Feb 25, 2018 17:16:20.999182000 SE Asia Standard Time

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1519553780.999182000 seconds

[Time delta from previous captured frame:
0.010707000 seconds]

[Time delta from previous displayed frame:
0.010707000 seconds]

[Time since reference or first frame: 0.010707000
seconds]

Frame Number: 2

Frame Length: 142 bytes (1136 bits)

Capture Length: 142 bytes (1136 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame:
eth:ethertype:ip:udp:udpcap:esp]

[Coloring Rule Name: UDP]

[Coloring Rule String: udp]

Ethernet II, Src: Azurewav_68:b8:2b
(6c:71:d9:68:b8:2b), Dst: HuaweiTe_2b:00:f6
(9c:c1:72:2b:00:f6)

Destination: HuaweiTe_2b:00:f6 (9c:c1:72:2b:00:f6)

Address: HuaweiTe_2b:00:f6 (9c:c1:72:2b:00:f6)

.... ..0. = LG bit:
Globally unique address (factory default)

.... ...0 = IG bit:
Individual address (unicast)

Source: Azurewav_68:b8:2b (6c:71:d9:68:b8:2b)

Address: Azurewav_68:b8:2b (6c:71:d9:68:b8:2b)

.... ..0. = LG bit:
Globally unique address (factory default)

.... ...0 = IG bit:
Individual address (unicast)

Type: IPv4 (0x0800)

Internet Protocol Version 4, Src: 192.168.1.3, Dst:
43.249.38.69

0100 = Version: 4

.... 0101 = Header Length: 20 bytes

Differentiated Services Field: 0x00 (DSCP: CS0,
ECN: Not-ECT)

0000 00.. = Differentiated Services Codepoint:
Default (0)

.... ..00 = Explicit Congestion Notification:
Not ECN-Capable Transport (0)

Total Length: 128

Identification: 0x25c9 (9673)

Flags: 0x00

0... .. = Reserved bit: Not set

.0.. .. = Don't fragment: Not set

..0. = More fragments: Not set

Fragment offset: 0

Time to live: 255

Protocol: UDP (17)

Header checksum: 0x81ba [validation disabled]

[Good: False]

[Bad: False]

Source: 192.168.1.3

Destination: 43.249.38.69

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

User Datagram Protocol, Src Port: 4500 (4500), Dst
Port: 4500 (4500)

Source Port: 4500

Destination Port: 4500

Length: 108

Checksum: 0x0000 (none)

[Good Checksum: False]

[Bad Checksum: False]

[Stream index: 0]

UDP Encapsulation of IPsec Packets

Encapsulating Security Payload

ESP SPI: 0xc268c60c (3261646348)

ESP Sequence: 1712

