

Universitas Indonesia



Rancang Bangun Simulasi Pengendali AMR  
(*Automatic Meter Reading*) Berbasis TCP/IP

Skripsi

Fajar Trapsilo  
0806365816

Fakultas Teknik  
Departemen Teknik Elektro  
2009

## HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.**

**Nama : Fajar Trapsilo**

**NPM : 0806365816**

**Tanda Tangan :**

**Tanggal : 7 Juli 2010**

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh:

Nama : Fajar Trapsilo  
NPM : 0806365816  
Program Studi : Teknik Elektro  
Judul : Rancang Bangun Simulasi Pengendali  
AMR berbasis TCP/IP

**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia**

### DEWAN PENGUJI

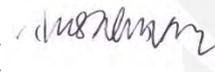
Pembimbing : Aji Nur Widyanto ST. MT.

()

Penguji : Dr. Ir. Anak Agung Putri Ratna M.Eng

()

Penguji : Muhammad Salman ST., MIT

()

Ditetapkan di : Depok

Tanggal : 7 Juli 2010

## UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Bapak Aji Nur Widyanto ST. MT. Dan Ibu Dr. Ir. Anak Agung Putri Ratna M.Eng, selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
- (2) pihak X Company yang telah banyak membantu dalam usaha memperoleh data yang saya perlukan;
- (3) orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral;
- (4) dan sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu pengetahuan.

Depok, 7 Juli 2010

Penulis

## **HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPERLUAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Fajar Trapsilo  
NPM : 0806365816  
Program Studi : Elektro  
Departemen : Teknik Elektro  
Fakultas : Teknik  
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul :

**Simulasi Pengendali AMR berbasis TCP/IP**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada Tanggal : 7 Juli 2010

Yang menyatakan

Fajar Trapsilo

## Abstrak

Nama : Fajar Trapsilo  
Program Studi : Teknik Elektro  
Judul : Rancang Bangun Simulasi Pengendali AMR (*Automatic Meter Reading*) Berbasis TCP/IP

Sejak adanya internet, komunikasi antara satu komputer dengan komputer lainnya diseluruh dunia menjadi sangat mudah. Hampir semua orang didunia telah memanfaatkan internet untuk keperluan bisnis, pendidikan, rumah tangga, hingga hiburan. Contoh aplikasi internet yang biasa digunakan antara lain *email, chatting, video conference, elearning*, hingga *game online*. Pada skripsi ini, internet dengan protocol TCP/IP dimanfaatkan sebagai sarana pengiriman data dari dan ke simulasi AMR. Data dari simulasi AMR berupa counter simulasi nilai *kilo watt hour* (KWH) yang dibuat dalam program mikro controller AVR, sedangkan data ke simulasi AMR berupa data serial yang diterima mikro controller AVR sebagai input untuk mengendalikan relay on atau off. Pengendalian relay merupakan simulasi untuk memutuskan aliran listrik dari AMR atau tidak. Keseluruhan proses simulasi pengendali AMR secara otomatis diatur oleh *windows application* yang dibuat dengan bahasa pemrograman Vb.net.

kunci:  
AMR, AVR, Vb.Net, internet, TCP/IP

## **Abstract**

Name : Fajar Trapsilo  
Study Programe : Electrical Engineering  
Title : Design Simulation of AMR (Automatic Meter Reading)  
Control Using TCP/IP Base

Since the Internet, communication between one computer to other computers around the world becomes very easy. Almost everyone in the world have been utilizing the Internet for business, education, household appliances, to entertainment. Examples of commonly used Internet applications such as email, chat, video conferencing, elearning, until the game online. In this script the Internet protocol TCP / IP is used as a means of sending data to and from AMR simulation. AMR simulation data from a simulated counter value kilo watt hour (KWH) made in the program AVR micro-controller, while the AMR simulation data into a serial data received AVR micro-controller as an input for controlling the relay on or off. Control relay is simulated to cutting power from AMR. AMR controlling the whole process simulation will be automatically set by the windows application created with Vb.net programming language.

Keywords:

AMR, AVR, Vb.Net, internet, TCP/IP

## DAFTAR ISI

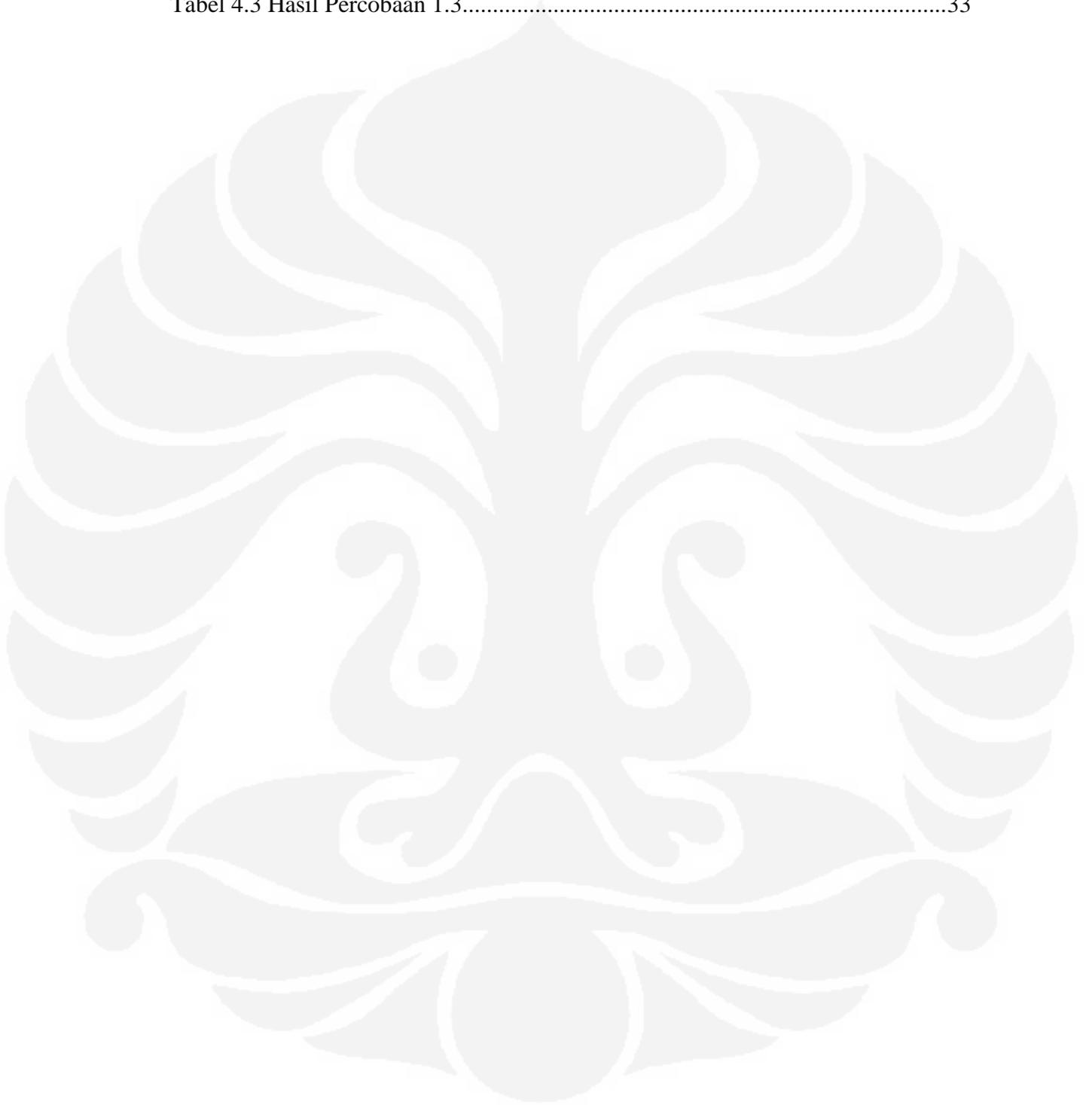
HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN ORISINALITAS.....	ii
HALAMAN PENGESAHAN.....	iii
UCAPAN TERIMA KASIH.....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPERLUAN AKADEMIS.....	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	x
1. PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	2
1.3 Batasan Masalah.....	2
1.4 Metode Penulisan.....	2
2. TEORI PENUNJANG.....	3
2.1 AMR (Automatic Meter Reading).....	3
2.2 TCP/IP.....	6
2.2.1 Dasar Arsitektur TCP/IP.....	6
2.2.2 Komponen Fisik dalam Jaringan TCP/IP.....	9
2.3 Database.....	9
2.4 Visual Basic.Net.....	11
2.5 Sistem Minimum AVR.....	11
2.6 Relay.....	12
3. PERANCANGAN.....	14
3.1 Perancangan Keseluruhan Sistem.....	14
3.2 Perancangan Algoritma Sistem AMR Bridge.....	15
3.3 Perancangan Algoritma Sistem dengan BASCOM AVR.....	28
3.4 Perancangan Rangkaian Kendali Relay.....	29
4. HASIL DAN ANALISA.....	30
4.1 Hasil Tampilan Sistem AMR Bridge.....	30
4.2 Hasil Percobaan Koneksi.....	31
4.3 Hasil Percobaan Sistem AMR Bridge.....	31
4.4 Hasil Percobaan Akurasi Data.....	32
4.5 Analisa.....	33
5. KESIMPULAN.....	34
DAFTAR PUSTAKA.....	35

## DAFTAR GAMBAR

Gambar 2.1 Automatic Meter Reading (AMR) .....	5
Gambar 2.2 Blok Diagram 3 Komponen Primer Sistem AMR .....	5
Gambar 2.3 Layer TCP/IP.....	7
Gambar 2.4 Pergerakan Data dalam Layer TCP/IP .....	9
Gambar 2.5 Sistem Minimum AVR .....	12
Gambar 2.6 Relay SPDT .....	13
Gambar 3.1 Blok Diagram Keseluruhan Sistem.....	14
Gambar 3.2 Flow Chart Algoritma Keseluruhan Sistem AMR Bridge .....	16
Gambar 3.3 Flow Chart Algoritma Sub Select Nilai Kwh Pelanggan x pada DB.....	17
Gambar 3.4 Flow Chart Algoritma Sub Ambil Data dari Mikro Kontroller .....	17
Gambar 3.5 Flow Chart Algoritma Sub Pecah Data.....	18
Gambar 3.6 Flow Chart Algoritma Sub Update Nilai Kwh Pelanggan x pada DB ...	19
Gambar 3.7 Flow Chart Algoritma Sub Select Status Pelanggan x pada DB.....	20
Gambar 3.8 Flow Chart Algoritma Sub Cek Status Pelanggan x .....	20
Gambar 3.9 Flow Chart Algoritma Sub Cek Tanggal.....	21
Gambar 3.10 Flow Chart Algoritma Sub Cek Validasi .....	22
Gambar 3.11 Flow Chart Algoritma Sub Cek Status Pembayaran 1 .....	23
Gambar 3.12 Flow Chart Algoritma Sub Cek Status Pembayaran 2 .....	24
Gambar 3.13 Flow Chart Algoritma Sub Update Status Pelanggan x pada DB Sama dengan Off.....	25
Gambar 3.14 Flow Chart Algoritma Sub Update Status Pelanggan x pada DB Sama dengan On .....	26
Gambar 3.15 Flow Chart Algoritma Sub Kirim Status = On ke Mikro Kontroller ...	27
Gambar 3.16 Flow Chart Algoritma Sub Kirim Status = Off ke Mikro Kontroller ..	27
Gambar 3.17 Flow Chart Algoritma Sistem dengan BASCOM AVR .....	28
Gambar 3.18 Rangkaian Kendali Relay.....	29
Gambar 4.1 Tampilan Sistem AMR Bridge.....	30

## DAFTAR TABEL

Tabel 4.1 Hasil Percobaan 1.1.....	31
Tabel 4.2 Hasil Percobaan 1.2.....	32
Tabel 4.3 Hasil Percobaan 1.3.....	33



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

AMR (Automatic Meter Reading) merupakan pengukur daya listrik berbasis digital yang sudah dilengkapi dengan pengendali elektronik dan antarmuka komunikasi. Pengendali elektronik dapat memproses data dari sensor arus dan tegangan listrik menjadi data-data digital yang berisi informasi nilai arus listrik, tegangan listrik, daya kompleks, daya aktif, daya reaktif, dan lain-lain. Data-data digital tersebut kemudian dapat dikirim melalui antarmuka komunikasi. Antarmuka komunikasi ini dapat berupa modem GSM, modul TCP/IP, dan lain-lain sesuai dengan jaringan komunikasi yang akan digunakan.

Di Indonesia, AMR baru diterapkan untuk pelanggan-pelanggan besar seperti perusahaan dan pabrik-pabrik besar. Sedangkan di daerah perumahan, sebagian besar masih menggunakan kwh (Kilo Watt Hour) meter analog dan sebagian lagi ada yang sudah menggunakan kwh meter digital yang dilengkapi kartu Prabayar. Kwh meter analog atau digital tersebut belum dilengkapi fasilitas canggih seperti AMR. Untuk setiap periode evaluasi data kwh dari seluruh pelanggan listrik di daerah perumahan, perusahaan listrik harus mengerahkan banyak petugas listrik untuk mencatat data kwh dari semua pelanggannya tersebut.

Dengan menggunakan AMR, suatu perusahaan penyedia layanan listrik dapat memantau penggunaan daya listrik dan status pembayaran rekening listrik dari setiap rumah pelanggannya melalui website perusahaan yang diberi nama *AMR-online*. Seorang admin dari perusahaan tersebut cukup duduk di depan komputer yang terhubung dengan internet, membuka website *AMR-online*, memeriksa database pelanggan, dan menentukan pelanggan mana yang harus diputus aliran listriknya. Pemeriksaan database dari tiap-tiap pelanggan berguna untuk mengetahui status pelanggan. Misalnya, apabila pelanggan  $x$  belum melunasi pembayaran rekening listriknya, maka admin akan mengubah database status pelanggan  $x$  menjadi *off* dari website *AMR-online*-nya untuk memutus aliran listrik pelanggan tersebut. AMR di setiap rumah pelanggan dilengkapi dengan

switch relay yang terhubung dengan pengendali elektronik dalam AMR. Hal ini yang memungkinkan admin dari perusahaan penyedia layanan listrik dapat memutus aliran listrik pelanggan melalui websitenya.

Namun sistem yang telah dijelaskan diatas terlalu tergantung pada admin. Admin bisa saja kurang teliti dan tidak selalu ada didepan komputer admin. Hal tersebut bisa mengakibatkan error pada sistem. Misalkan saja ada pelanggan yang belum membayar tagihan listrik tetapi AMR dirumahnya tidak dimatikan. Untuk mengatasi hal tersebut maka pada skripsi ini akan dibuat simulasi pengendali AMR berbasis TCP/IP.

## **1.2 Tujuan**

Tujuan dari skripsi ini adalah untuk mensimulasikan suatu sistem pengendali AMR berbasis TCP/IP yang dapat mempermudah perusahaan penyedia listrik dalam memutus aliran listrik pelanggan yang belum membayar tagihan listrik.

## **1.3 Batasan Masalah**

Penulisan skripsi ini menggunakan data yang diperoleh dari hasil pengujian sistem yang terdiri dari relay, mikro kontroller, dan komputer 1 (untuk menjalankan aplikasi vb.net) yang dihubungkan dengan komputer 2 (sebagai server untuk menjalankan web aplikasi dan database) melalui jaringan lokal.

## **1.4 Metode Penulisan**

Skripsi ini disusun dalam lima bab. Bab pertama merupakan pendahuluan yang berisi latar belakang, tujuan penelitian, pembatasan masalah dan metode penulisan. Bab kedua membahas tentang teori dasar AMR, TCP/IP, database, Vb.Net, mikro kontroller dan relay. Bab ketiga merupakan perancangan dari simulasi pengendali AMR berbasis TCP/IP. Bab keempat merupakan hasil pengujian dan analisa dari hasil pengujian. Bab kelima merupakan kesimpulan dari simulasi yang telah dilakukan.

## BAB II

### KONSEP AMR DAN TCP/IP

#### 2.1 AMR (Automatic Meter Reading) <sup>[1,2]</sup>

AMR (*Automatic Meter Reading*) adalah suatu alat berbasis digital yang dapat mencatat penggunaan daya listrik, debit air, atau gas secara lengkap dan mentransfer data ke database pusat. Transfer data dapat menggunakan jaringan telepon (kabel atau nirkabel), frekuensi radio (RF), atau powerline transmisi. Salah satu bentuk fisik dari AMR dapat dilihat pada Gambar 2.1.

AMR mempunyai 3 komponen utama, yaitu *meter interface module*, *communication systems*, *central office systems equipment*. Meter interface module mempunyai 4 bagian utama, yaitu *power supply*, *meter sensor*, *controlling electronic*, dan *communication interface*. Power supply berfungsi sebagai sumber energi untuk sistem AMR. Meter sensor berfungsi untuk mengukur arus dan tegangan listrik. Controlling electronic dapat berupa mikro controller yang berfungsi untuk mengolah data dari meter sensor menjadi data daya dan lain-lain dalam bentuk digital serta mengendalikan communication interface untuk mengirim data-data tersebut. Communication interface dapat berupa modem ADSL, modem GSM, modem IC ADE8165, modul TCP/IP dan lain-lain sesuai dengan jaringan komunikasi yang digunakan.

Berikutnya, komponen kedua dari AMR, yaitu *communication systems*. Communication systems berfungsi untuk mengirim data dari AMR ke kantor perusahaan listrik melalui media komunikasi tertentu. Media komunikasi yang digunakan dapat berupa jaringan kabel telepon, powerline carrier (plc), radio frekwensi (RF), atau cable television. Berikutnya, komponen ketiga dari sistem AMR, yaitu *central office systems equipment*. Central office systems equipment mempunyai 3 bagian utama, yaitu *receivers data*, komputer server, dan komputer host. Receivers data dapat berupa modem ADSL, modem GSM, modem IC ADE8165, dan lain-lain sesuai dengan media komunikasi yang digunakan untuk mengirim data. Receivers data, terhubung dengan komputer server dan berfungsi untuk menerima data dari AMR. komputer server merupakan komputer dengan

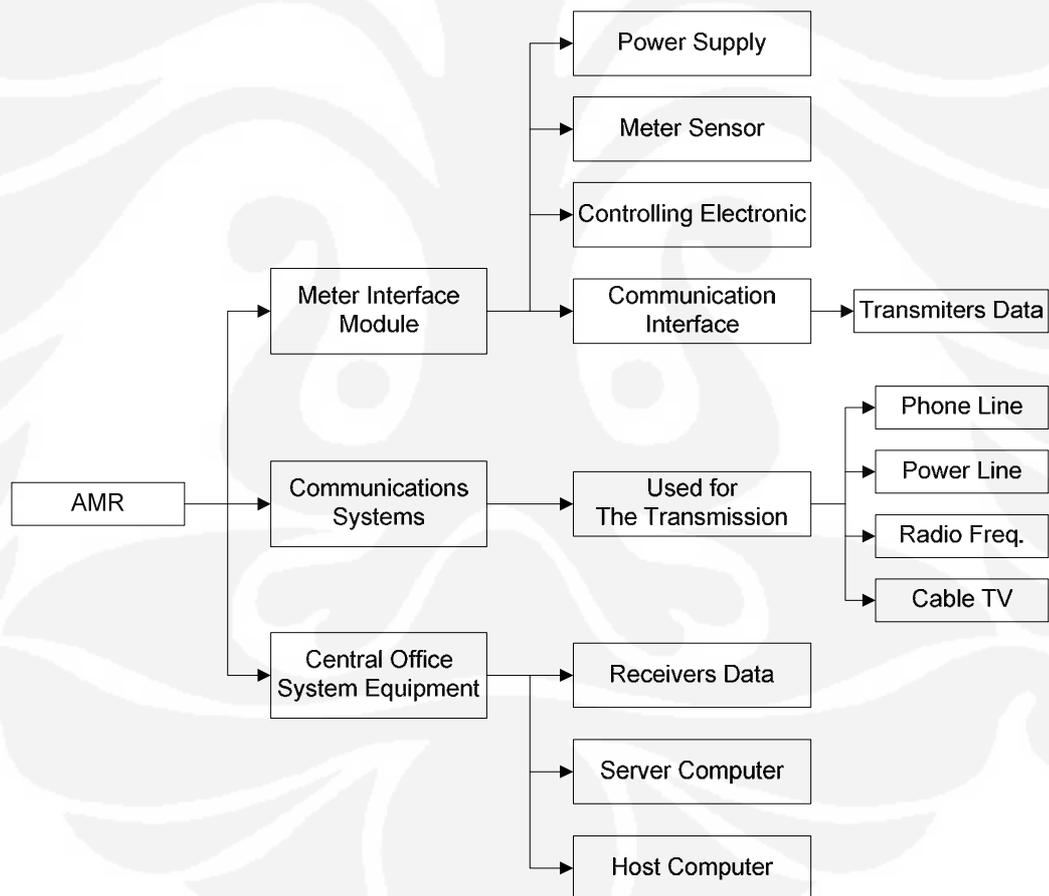
kemampuan diatas rata-rata komputer biasa yang dilengkapi dengan *operating system* khusus server. Komputer server berfungsi untuk menjalankan aplikasi web dan database serta melayani permintaan dari komputer host untuk mengakses aplikasi web dan database tersebut. Komputer host merupakan komputer biasa yang digunakan oleh admin dari perusahaan listrik untuk mengakses aplikasi web AMR dan database pelanggan dari perusahaan listrik tersebut. Untuk lebih jelasnya dapat dilihat pada Gambar 2.2.

Dalam pengoperasiannya, sistem AMR melakukan pembacaan energi listrik dengan cara menurunkan terlebih dahulu tegangan listrik dari 40 KV menjadi 220 V menggunakan current transformer, kemudian arus dan tegangan listrik dibaca oleh sensor arus dan tegangan listrik. Salah satu sensor arus dan tegangan yang digunakan dalam AMR adalah ADE7757. Data dari sensor-sensor tersebut kemudian masuk ke dalam mikro kontroller untuk diproses menjadi data nilai arus, tegangan, daya kompleks, daya aktif, daya reaktif, dan lain-lain. Setelah itu, data-data tersebut ditampilkan pada LCD AMR. Selain itu, mikro kontroller juga mengendalikan *communication interface* untuk mengirimkan data-data tersebut ke database perusahaan listrik melalui media komunikasi tertentu.

AMR merupakan salah satu solusi untuk perusahaan listrik dalam memonitor penggunaan daya listrik dari pelanggannya. Dengan menggunakan AMR, perusahaan listrik tidak perlu mengerahkan banyak petugas listrik untuk mencatat data daya dari seluruh pelanggan listrik di setiap periode evaluasi data daya listrik yang telah digunakan pelanggannya. Data daya listrik dari setiap pelanggan akan secara otomatis terkirim ke dalam database perusahaan listrik sesuai periode pengiriman yang telah ditetapkan. Selain itu, dari sisi pelanggan listrik, sistem AMR mempermudah pelanggan untuk melihat tagihan listriknya setiap bulan. Cukup dengan mengakses *website* AMR yang telah disiapkan oleh perusahaan listrik, lalu memasukkan password, maka pelanggan tersebut sudah dapat melihat total daya listrik yang digunakan dan tagihan listriknya.



Gambar 2.1. Automatic Meter Reading (AMR) <sup>[1]</sup>



Gambar 2.2. Blok Diagram 3 Komponen Primer AMR <sup>[2]</sup>

## 2.2 TCP/IP<sup>[6,8]</sup>

TCP/IP (*Transmission Control Protocol/ Internet Protocol*) adalah sekelompok protocol yang mengatur komunikasi data komputer di internet. Komputer-komputer yang terhubung ke internet berkomunikasi dengan protokol ini. Karena menggunakan bahasa yang sama, yaitu protokol TCP/IP, perbedaan jenis komputer dan sistem operasi tidak menjadi masalah. Jadi, jika sebuah komputer menggunakan protokol TCP/IP dan terhubung langsung ke internet, maka komputer tersebut dapat berhubungan dengan komputer di belahan dunia mana pun yang juga terhubung ke internet.

### 2.2.1 Dasar Arsitektur TCP/IP

Pada dasarnya, komunikasi data merupakan proses mengirimkan data dari satu komputer ke komputer yang lain. Untuk dapat mengirimkan data, pada komputer harus ditambahkan alat khusus, yang dikenal sebagai *network interface* (interface jaringan). Jenis interface jaringan ini bermacam-macam, bergantung pada media fisik yang digunakan untuk mentransfer data tersebut.

Dalam proses pengiriman data ini ada beberapa proses yang harus dipecahkan. Pertama, data harus dapat dikirimkan ke komputer yang tepat, sesuai tujuannya. Hal ini akan menjadi rumit bila komputer tujuan transfer data ini tidak berada pada jaringan local, melainkan di tempat yang jauh. Jika lokasi komputer yang saling berkomunikasi “jauh” (secara jaringan) maka terdapat kemungkinan data rusak atau hilang. Karenanya, perlu ada mekanisme yang mencegah rusaknya data ini. Hal lain yang perlu diperhatikan adalah, pada komputer tujuan transfer data mungkin terdapat lebih dari satu aplikasi yang menunggu datangnya data. Data yang dikirim harus sampai ke aplikasi yang tepat, pada komputer yang tepat, tanpa kesalahan.

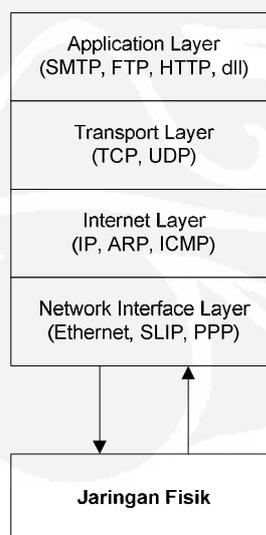
Cara alamiah untuk menghadapi setiap masalah yang rumit ialah memecahkan masalah tersebut menjadi bagian yang lebih kecil. Dalam memecahkan masalah transfer data diatas, para ahli jaringan komputer pun melakukan hal yang sama. Untuk setiap problem komunikasi data, diciptakan

solusi khusus berupa aturan-aturan untuk menangani problem tersebut. Untuk menangani semua masalah komunikasi data, keseluruhan aturan ini harus bekerja sama satu dengan yang lainnya. Sekumpulan aturan untuk mengatur proses pengiriman data ini disebut sebagai protokol komunikasi data. Protokol ini diimplementasikan dalam bentuk program komputer (*software*) yang terdapat pada komputer dan peralatan komunikasi data lainnya.

TCP/IP adalah sekumpulan protokol yang didesain untuk melakukan fungsi-fungsi komunikasi data pada *Wide Area Network* (WAN). TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. Berkat prinsip ini tugas masing-masing protokol menjadi jelas dan sederhana. Protokol yang satu tidak perlu mengetahui cara kerja protokol yang lain, selama masih dapat saling mengirim dan menerima data.

Berkat menggunakan prinsip ini, TCP/IP menjadi protokol komunikasi data yang fleksibel. Protokol TCP/IP dapat diterapkan dengan mudah di setiap jenis komputer dan interface jaringan, karena sebagian besar isi kumpulan protokol ini tidak spesifik terhadap satu komputer atau peralatan jaringan tertentu. Agar TCP/IP dapat berjalan diatas interface jaringan tertentu, hanya perlu dilakukan perubahan pada protokol yang berhubungan dengan interface jaringan saja.

Sekumpulan protokol TCP/IP ini dimodelkan dengan empat layer TCP/IP, sebagaimana terlihat pada Gambar 2.3.



Gambar 2.3 Layer TCP/IP <sup>[8]</sup>

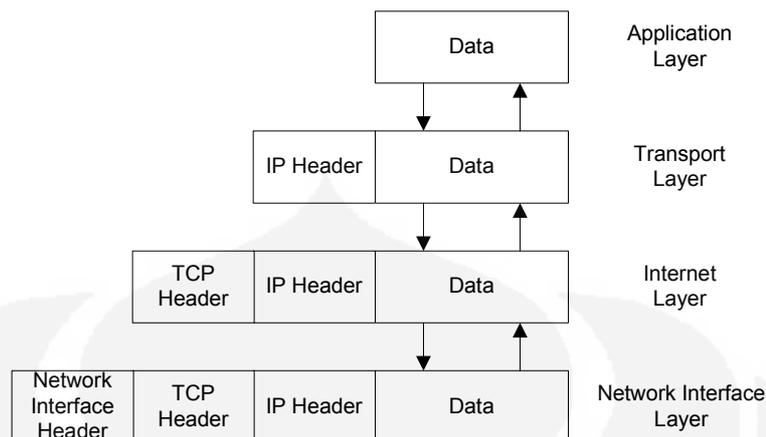
Dalam TCP/IP, terjadi penyampaian data dari protokol yang berada di satu layer ke protokol yang berada di layer yang lain seperti yang terlihat pada Gambar 2.4. Semua protokol memperlakukan semua informasi yang diterimanya dari protokol lain sebagai data. Jika satu protokol menerima data dari protokol lain di layer atasnya, protokol tersebut akan menambahkan informasi tambahan ke data tersebut. Informasi ini memiliki fungsi yang sesuai dengan fungsi protokol tersebut. Setelah itu, data ini diteruskan lagi pada layer dibawahnya.

Hal yang sebaliknya terjadi jika suatu protokol menerima data dari protokol lain yang berada pada layer dibawahnya. Jika data ini dianggap valid, protokol akan melepas informasi tambahan tersebut, untuk kemudian meneruskannya ke protokol lain yang berada pada layer diatasnya.

Lapisan/ layer terbawah, yaitu Network Interface Layer, bertanggung jawab mengirim dan menerima data ke dan dari media fisik. Media fisiknya dapat berupa kabel, serat optic, atau gelombang radio. Karena tugasnya ini, protokol pada layer ini harus mampu menerjemahkan sinyal listrik menjadi data digital yang dimengerti komputer, yang berasal dari peralatan lain yang sejenis.

Lapisan protokol berikutnya adalah Internet Layer. Protokol pada layer ini bertanggung jawab dalam proses pengiriman paket ke alamat yang tepat. Pada layer ini terdapat tiga macam protokol, yaitu IP, ARP, dan ICMP. IP (*internet protocol*) berfungsi untuk menyampaikan paket data ke alamat yang tepat. ARP (*Address Resolution Protocol*) adalah protokol yang digunakan untuk menemukan alamat hardware dari host/ komputer yang terletak pada *network* yang sama. Sedangkan ICMP (*Internet Control Message Protocol*) adalah protokol yang bertugas mengirimkan pesan-pesan kesalahan dan kondisi lain yang memerlukan perhatian khusus. Pesan/ paket ICMP dikirim jika terjadi masalah pada internet layer dan layer diatasnya (TCP/ UDP).

Lapisan berikutnya adalah Transport Layer, berisi protokol yang bertanggung jawab untuk mengadakan komunikasi antara dua host/ komputer. Dalam transport layer ini terdapat dua protokol. Kedua protokol tersebut adalah TCP (*Transmission Control Protocol*) dan UDP (*User Datagram Protocol*). Terakhir, layer teratas adalah Application Layer. Pada layer inilah terletak semua aplikasi yang menggunakan protokol TCP/IP ini.



Gambar 2.4 Pergerakan data dalam layer TCP/IP [8]

### 2.2.2 Komponen Fisik dalam Jaringan TCP/IP

Komputer dengan protokol TCP/IP dapat terhubung ke komputer lain dan jaringan lain karena bantuan peralatan jaringan komputer. Pada komputer itu sendiri, ditambahkan alat yang disebut *network interface*. Network interface ini bisa berupa *card Ethernet* atau *modem*. Card Ethernet terhubung ke komputer lain via kabel RG-58 atau ke hub Ethernet via kabel UTP. Modem terhubung ke jaringan melalui kabel telepon atau gelombang radio. Diluar peralatan yang disebutkan ini, masih diperlukan peralatan lain untuk membentuk jaringan komputer. Peralatan ini disebut sebagai peralatan penghubung jaringan. Peralatan penghubung jaringan secara umum dibagi dalam beberapa kategori, yaitu *repeater*, *bridge*, dan *router*.

### 2.3 Database [10]

Basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi.

Istilah "basis data" berawal dari ilmu komputer. Meskipun kemudian artinya semakin luas, memasukkan hal-hal di luar bidang elektronika, artikel ini mengenai basis data komputer. Catatan yang mirip dengan basis data sebenarnya sudah ada sebelum revolusi industri yaitu dalam bentuk buku besar, kuitansi dan kumpulan data yang berhubungan dengan bisnis.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Skema menggambarkan obyek yang diwakili suatu basis data, dan hubungan di antara obyek tersebut. Ada banyak cara untuk mengorganisasi skema, atau memodelkan struktur basis data: ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah layanan mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika). Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel.

Istilah *basis data* mengacu pada koleksi dari data-data yang saling berhubungan, dan perangkat lunaknya seharusnya mengacu sebagai sistem manajemen basis data (*database management system/DBMS*). Jika konteksnya sudah jelas, banyak administrator dan programmer menggunakan istilah basis data untuk kedua arti tersebut.

Perangkat lunak basis data yang banyak digunakan dalam pemrograman antara lain: DB2, Microsoft SQL Server, Microsoft Access, Oracle, MySQL, Force, QuickSilver, dan lain-lain.

## 2.4 Visual Basic .Net <sup>[9]</sup>

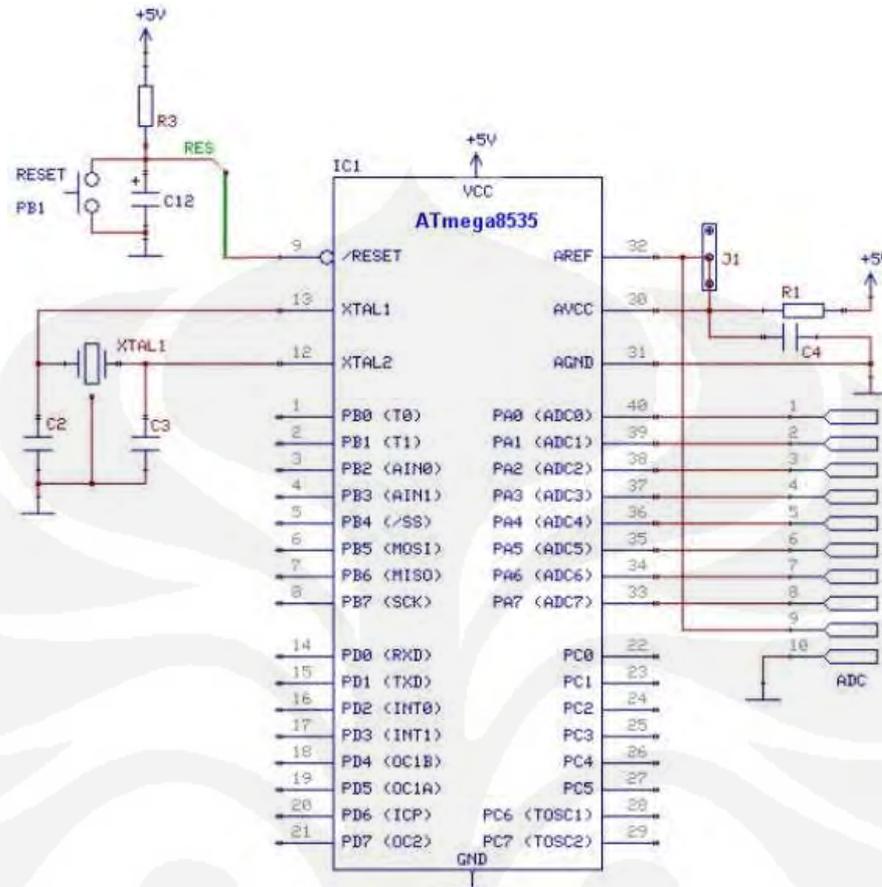
Visual Basic .NET (atau VB.NET) merupakan salah satu bahasa pemrograman yang bisa digunakan untuk membangun aplikasi-aplikasi .NET di platform Microsoft .NET. Tidak seperti generasi sebelumnya, Visual Basic versi 6.0 ke bawah yang lebih difokuskan untuk pengembangan aplikasi desktop, Visual Basic .NET memungkinkan untuk membangun bermacam aplikasi, baik desktop maupun aplikasi web.

## 2.5 Sistem Minimum AVR <sup>[10]</sup>

Sistem minimum mikrokontroler adalah rangkaian elektronik minimum yang diperlukan untuk mengoperasikan IC mikrokontroler AVR. Sistem minimum ini kemudian bisa dihubungkan dengan rangkaian lain untuk menjalankan fungsi tertentu. Di keluarga mikrokontroler AVR, seri ATmega8535 adalah seri yang banyak digunakan.

Untuk membuat sistem minimum atmel AVR 8535 pada Gambar 2.5 diperlukan beberapa komponen utama yaitu, IC ATMEGA 8535, XTAL 4 MHz atau 8 MHz (XTAL1), 3 kapasitor kertas yaitu dua 22 pF (C2 dan C3) serta 100 nF (C4), 1 kapasitor elektrolit 4.7 uF (C12) 2 resistor yaitu 100 ohm (R1) dan 10 Kohm (R3), 1 tombol reset pushbutton (PB1). Selain itu tentunya diperlukan power supply yang bisa memberikan tegangan 5V DC.

Setelah IC ATMEGA 8535 diaktifkan oleh system minimum AVR, IC tersebut harus diisi program yang sesuai untuk aplikasi yang diinginkan. Pemrograman ini dapat dilakukan dengan menggunakan bahasa C, visual basic, BASCOM (Basic Compiler), dan lain-lain. Sebelum diisi ke dalam IC ATMEGA 8535, program tersebut harus dihimpun (compile) dalam bentuk bahasa mesin.

Gambar 2.5 Sistem Minimum AVR <sup>[10]</sup>

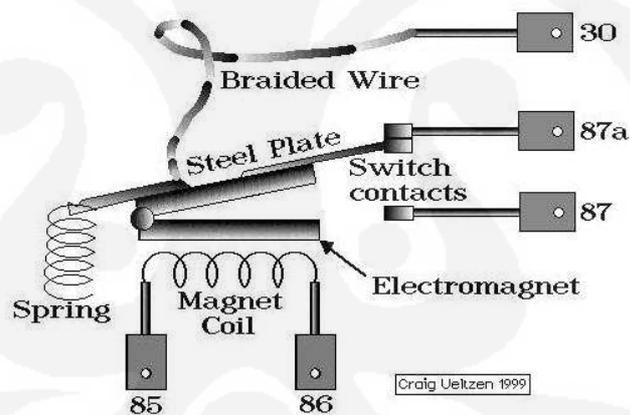
## 2.6 Relay <sup>[10]</sup>

Relay merupakan switch mekanik-elektrik dengan isolasi baik. Relay disebut switch mekanik-elektrik karena prinsip kerjanya sama seperti switch mekanik biasa, tetapi dikendalikan secara elektrik melalui solenoid/ coil. Solenoid yang diberi beda potensial menjadi bersifat magnet, sehingga dapat menarik armature dari bagian normally close (NC)/ 87a kontak ke bagian normally open (NO)/ 87 kontak. Namun saat solenoid tidak diberi beda potensial, armature akan ditarik kembali oleh spring ke bagian NC kontak. Untuk lebih jelasnya dapat dilihat pada Gambar 2.6.

Relay yang telah dijelaskan diatas merupakan jenis relay SPDT (*Single Pole Double Throw*) atau relay yang mempunyai satu kutup (common) yang dapat dilontarkan kedua bagian kontak, yaitu ke bagian NO atau NC. Selain itu, masih ada banyak jenis relay di toko komponen elektronik. Biasanya, perbedaan jenis-

jenis relay dapat dilihat dari jumlah *pole* dan *throw*-nya, batas tegangan dan arus listrik DC atau AC yang dapat dialirkan pada bagian NO / NC relay serta batas tegangan untuk bagian solenoid.

Berdasarkan jumlah pole dan throw-nya, ada 4 jenis relay yang umum dipasarkan di toko elektronik, yaitu SPST (Single Pole Single Throw), SPDT (Single Pole Double Throw), DPST (Double Pole Single Throw), dan DPDT (Double Pole Double Throw). Bila dilihat dari batasan arus dan tegangan yang dapat dialirkan pada bagian solenoid dan NO / NC relay, tentunya ada banyak sekali jenis relay dengan berbagai variasi batasan arus dan tegangannya. Contoh, pada bodi relay SPDT tertera DC 5V, NO: 10A 24VDC, NC: 6A 24VDC. Data tersebut berarti, bagian solenoid relay dapat dapat diberi tegangan DC 5V, batas arus dan tegangan untuk bagian NO adalah 10 ampere dan 24 volt DC, dan batas arus dan tegangan untuk bagian NC adalah 6 ampere dan 24 volt DC.



Gambar 2.6 Relay SPDT <sup>[10]</sup>

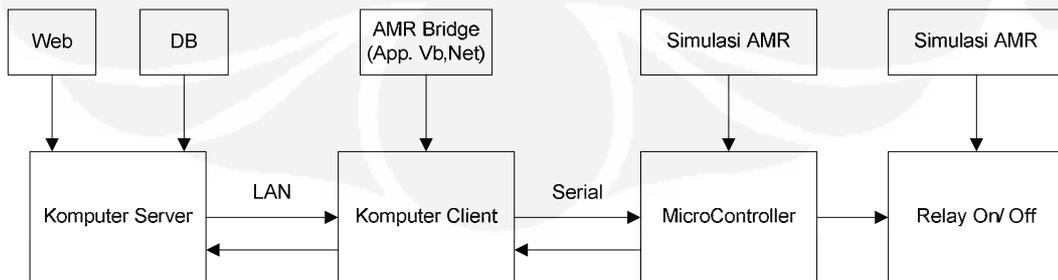
# BAB III

## PERANCANGAN SISTEM SIMULASI PENGENDALI AMR BERBASIS TCP/IP

### 3.1 Perancangan Keseluruhan System

Komputer server yang menjalankan aplikasi web dan DB (database) terhubung dengan komputer client (dalam hal ini, disimulasikan sebagai komputer yang terhubung dalam AMR) melalui jaringan LAN. Komputer client dapat memperbaharui (*Update*) database dalam komputer server secara otomatis berdasarkan data Kwh yang diterima dari simulasi AMR (mikro controller AVR) secara serial setiap detik. Komputer client juga dapat mencari tahu status pelanggan listrik dari database dalam komputer server setiap detik dan mengirimkannya ke mikro controller secara serial.

Pada tanggal yang telah ditetapkan, aplikasi AMR bridge dalam komputer client akan mengirim status terakhir dari pelanggan listrik secara serial ke mikro controller. Jika status terakhir dari pelanggan listrik adalah hutang, maka mikro controller akan mengirim signal ke relay (disimulasikan terhubung pada kabel listrik sebelum circuit breaker AMR) untuk pindah ke posisi off. Dengan demikian, pelanggan listrik tersebut tidak akan mendapat aliran listrik. Sebaliknya, jika status terakhir dari pelanggan listrik adalah lunas, mikro controller akan mengirim signal ke relay untuk tetap on, sehingga pelanggan listrik tersebut akan tetap mendapat aliran listrik. Untuk lebih jelasnya dapat dilihat pada Gambar 3.1.



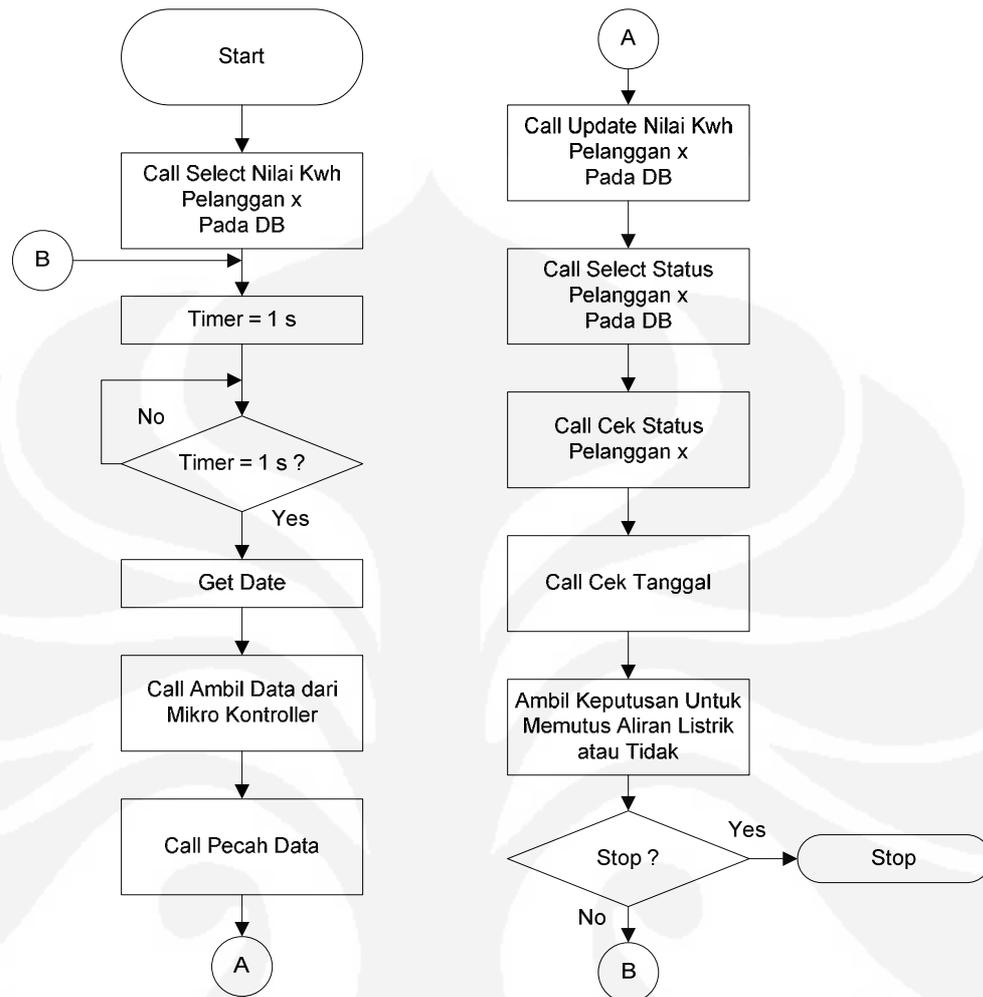
Gambar 3.1 Blok Diagram Keseluruhan Sistem

### 3.2 Perancangan Algoritma Sistem AMR Bridge

Secara garis besar, sistem AMR bridge ini berfungsi untuk menerima data serial dari simulasi AMR, memasukkan data tersebut ke database di komputer server, melakukan pengecekan tanggal untuk memberi peringatan dan memutuskan aliran listrik pelanggan yang belum membayar tagihan listrik pada tanggal yang telah ditetapkan, serta mengambil data status terakhir dari pelanggan listrik dalam database komputer server dan mengirimkan data tersebut secara serial ke mikro controller.

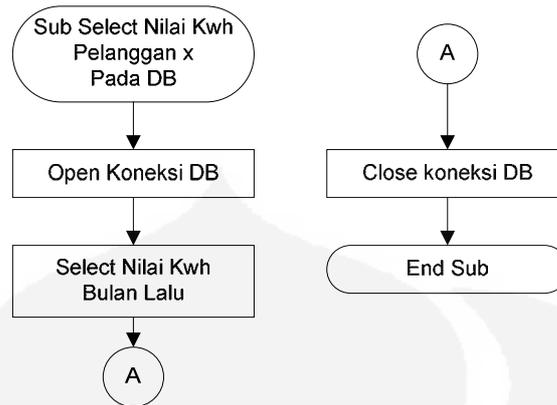
Perancangan algoritma sistem AMR bridge yang dibuat dengan bahasa pemrograman vb.net, secara garis besar dapat digambarkan dengan flow chart algoritma keseluruhan sistem AMR bridge yang terbagi dalam sembilan tahapan. Pada tahap pertama, sistem AMR bridge akan memanggil sub program select nilai kwh pelanggan x pada database. Pada tahap kedua, ketika sistem di start, timer (T) dideklarasikan 1 detik, kemudian dilakukan pengecekan apakah T sudah sama dengan 1 detik ?. Jika timer sudah sama dengan 1 detik, maka akan dilanjutkan dengan perintah tahap ketiga untuk mengambil tanggal dari komputer. Jika timer belum sama dengan 1 detik, maka alur algoritma pemrograman akan kembali ke perintah untuk mengecek apakah T sudah sama dengan 1 detik hingga syarat tersebut terpenuhi.

Setelah itu, dilanjutkan dengan perintah tahap keempat untuk memanggil sub program ambil data dari mikro controller. Kemudian dilanjutkan dengan perintah tahap kelima untuk memecah data dari mikro controller. Lalu dilanjutkan dengan perintah tahap keenam untuk memanggil sub program update nilai kwh pelanggan x pada database. Kemudian dilanjutkan dengan perintah tahap ketujuh untuk memanggil sub program select status pelanggan x pada database. Lalu dilanjutkan dengan perintah tahap kedelapan untuk memanggil sub program cek status pelanggan x. Kemudian dilanjutkan dengan perintah tahap kesembilan untuk memanggil sub program cek tanggal. Setelah itu, dilakukan pengambilan keputusan untuk memutuskan aliran listrik atau tidak. Untuk lebih jelasnya dapat dilihat pada Gambar 3.2.



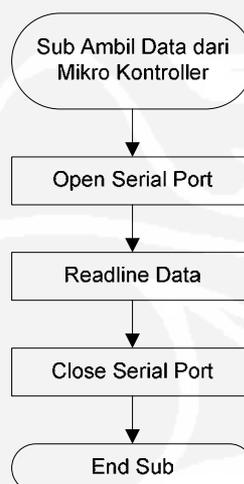
Gambar 3.2 Flow Chart Algoritma Keseluruhan Sistem AMR Bridge

Pada tahap pertama dipanggil sub program select nilai kwh pelanggan x pada database. Sub program tersebut diawali dengan perintah untuk membuka koneksi ke database. Lalu dilanjutkan dengan perintah untuk mengambil nilai kwh pelanggan x yang telah disimpan dalam database pada bulan lalu. Hal tersebut bertujuan untuk mengetahui nilai kwh yang terpakai pada bulan berikutnya. Kemudian dilanjutkan dengan perintah untuk menutup kembali koneksi dengan database. Setelah itu diakhiri dengan perintah untuk keluar dari sub. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.3. Tahap kedua dan ketiga telah dijelaskan sebelumnya pada Gambar 3.2.



Gambar 3.3 Flow Chart Algoritma Sub Select Nilai Kwh Pelanggan x pada DB

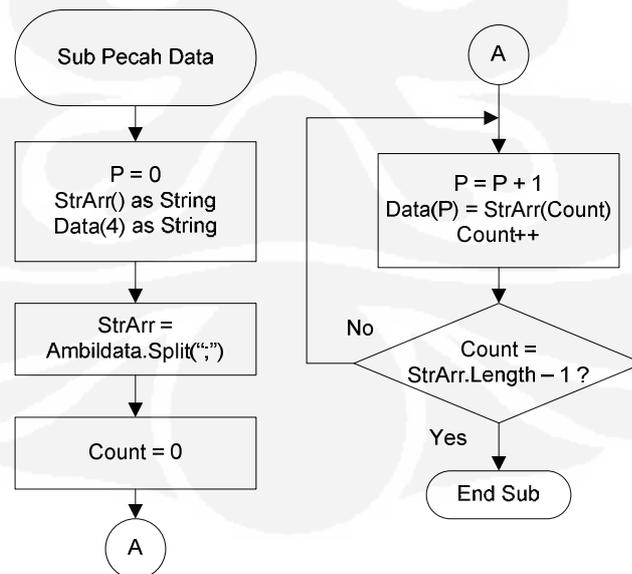
Pada tahap keempat dipanggil sub program ambil data dari mikro controller. Sub program ambil data dari mikro controller diawali dengan perintah untuk membuka gerbang komunikasi serial. Kemudian dilanjutkan dengan perintah untuk membaca data serial dari mikro controller. Data serial tersebut merupakan data nama, status, dan nilai kwh pelanggan x yang digabungkan dengan pemisah (*split*) berupa karakter “ ; “. Kemudian dilanjutkan dengan perintah untuk menutup kembali gerbang komunikasi serial. Setelah itu, dilanjutkan dengan perintah akhir untuk keluar dari sub (*end sub*). Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.4.



Gambar 3.4 Flow Chart Algoritma Sub Ambil Data dari Mikro Kontroller

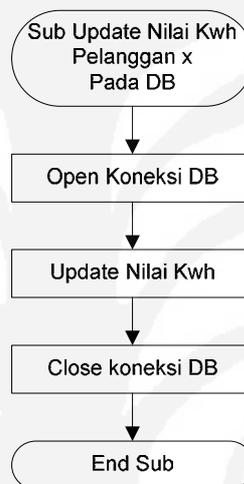
Pada tahap kelima, dipanggil sub program pecah data. Sub program ini diawali dengan perintah untuk mendeklarasikan variabel. Variabel  $P = 0$  sebagai variabel untuk memberi nilai urutan array pada  $Data(P)$ . Variabel  $StrArr(counter)$  dan  $Data(P)$  sebagai variabel array untuk menyimpan sementara data yang diterima dan dipecah. Setelah itu dilanjutkan dengan perintah untuk memecah data yang dipisahkan oleh karakter “ ; ” dan menyimpan data-data tersebut dalam variabel  $StrArr(counter)$ . Kemudian dilanjutkan dengan perintah untuk mendeklarasikan nilai variabel  $counter = 0$  sebagai variabel untuk memberi nilai urutan array pada  $StrArr(counter)$ .

Lalu dilanjutkan dengan perintah untuk menyimpan data-data yang telah dipecah dalam variabel  $StrArr(counter)$  kedalam variabel  $Data(P)$ . Hal tersebut dilakukan dengan cara memutar program untuk menambah nilai  $counter$ , nilai  $P$ , dan menyimpan  $Data(P) = StrArr(counter)$ . Program diputar sebanyak  $counter = StrArr.Length - 1$  (banyak data yang dipisahkan oleh karakter “ ; “). Jika nilai  $counter$  telah sama dengan banyak data yang dipisah oleh karakter “ ; “, maka sub program pecah akan diakhiri dengan perintah *End Sub*. Tujuan dari sub program pecah data adalah memisahkan data-data dari mikro kontroller yang masih digabung dengan format pemisah berupa karakter “ ; ”. Untuk lebih jelasnya, dapat dilihat pada Gambar 3.5.



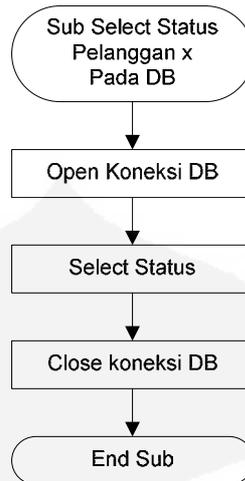
Gambar 3.5 Flow Chart Algoritma Sub Pecah Data

Pada tahap keenam, dipanggil sub program *update* nilai kwh pelanggan x pada database. Sub program ini diawali dengan perintah untuk membuka koneksi ke database. Kemudian dilanjutkan dengan perintah untuk memperbaharui (*update*) nilai kwh pelanggan x. Kemudian dilanjutkan dengan perintah untuk menutup kembali koneksi dengan database. Setelah itu, dilanjutkan dengan perintah akhir untuk keluar dari sub (*end sub*). Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.6.



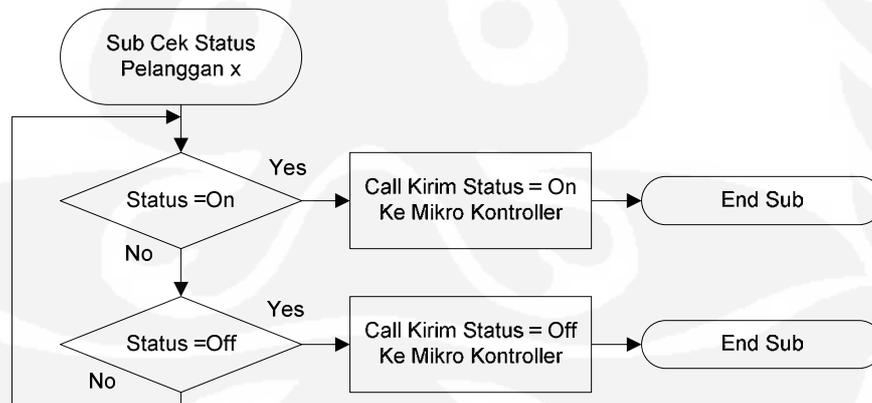
Gambar 3.6 Flow Chart Algoritma Sub Update Nilai Kwh Pelanggan x pada DB

Pada tahap ketujuh, dipanggil sub program *select* status pelanggan x pada database. Sub program ini diawali dengan perintah untuk membuka koneksi ke database. Kemudian dilanjutkan dengan perintah untuk memilih (*select*) status pelanggan x. Kemudian dilanjutkan dengan perintah untuk menutup kembali koneksi dengan database. Setelah itu, dilanjutkan dengan perintah akhir untuk keluar dari sub. Sub program ini bertujuan untuk mengambil status pelanggan x dari dalam database. Status pelanggan x dalam database dapat berisi “on” atau “off”. Pada tanggal 26, database status pelanggan x dapat berubah secara otomatis sesuai dengan status pembayarannya. Selain tanggal 26, database status pelanggan x dapat diubah secara manual oleh admin perusahaan penyedia listrik melalui website AMR online. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.7.



Gambar 3.7 Flow Chart Algoritma Sub Select Status Pelanggan x pada DB

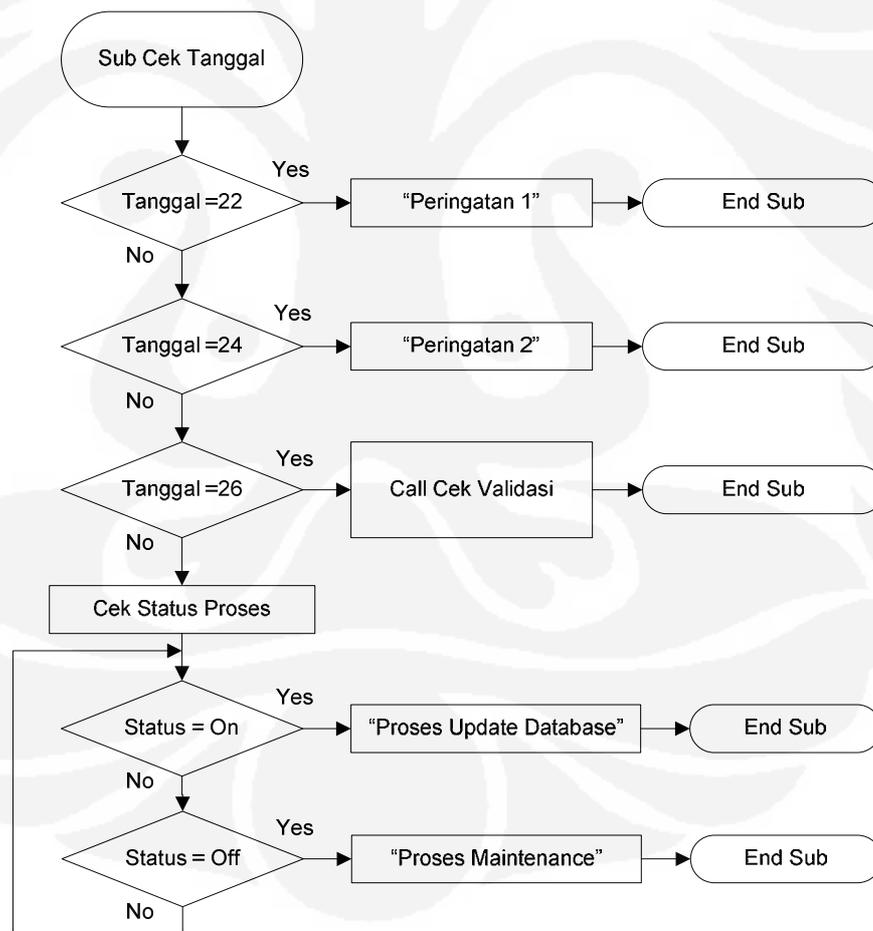
Pada tahap kedelapan, dipanggil sub program cek status pelanggan x. Pada sub program ini hanya ada perintah untuk mengecek status dan memanggil sub program kirim status ke mikro controller (dapat dilihat pada Gambar 3.15 dan Gambar 3.16). Untuk lebih jelasnya, flow chart sub program cek status pelanggan x ini dapat dilihat pada Gambar 3.8.



Gambar 3.8 Flow Chart Algoritma Sub Cek Status Pelanggan x

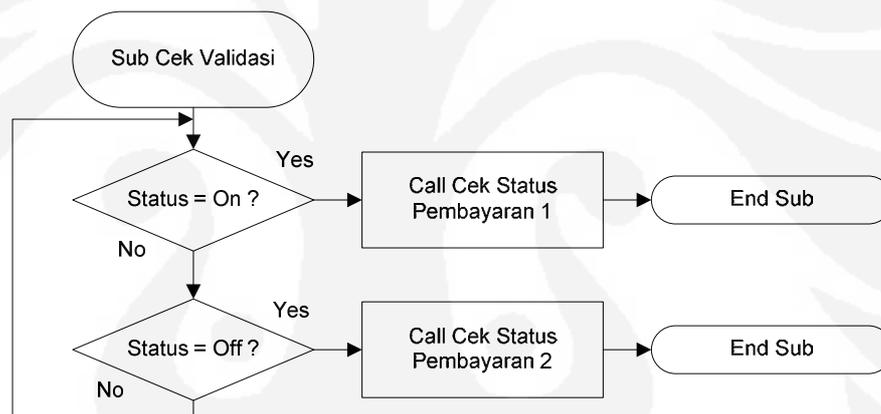
Pada tahap kesembilan, dipanggil sub program cek tanggal. Sub program ini diawali dengan perintah untuk mengecek apakah sudah tanggal 22 ?. Jika sudah

tanggal 22, maka akan muncul “peringatan 1” pada text box status proses sistem AMR bridge, lalu dilanjutkan dengan perintah akhir untuk keluar dari sub. Jika bukan tanggal 22, maka dilanjutkan dengan perintah untuk mengecek apakah sudah tanggal 24 ?. Bila sudah tanggal 24, maka akan muncul “peringatan 2” pada text box status proses sistem AMR bridge, lalu dilanjutkan dengan perintah akhir untuk keluar dari sub. Bila bukan tanggal 24, maka dilanjutkan dengan perintah untuk mengecek apakah sudah tanggal 26 ?. Jika sudah tanggal 26, maka dilanjutkan dengan perintah untuk memanggil sub program cek validasi, lalu dilanjutkan dengan perintah akhir untuk keluar dari sub. Jika bukan tanggal 26, maka akan muncul “proses update database” pada text box status proses sistem AMR bridge. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.9.



Gambar 3.9 Flow Chart Algoritma Sub Cek Tanggal

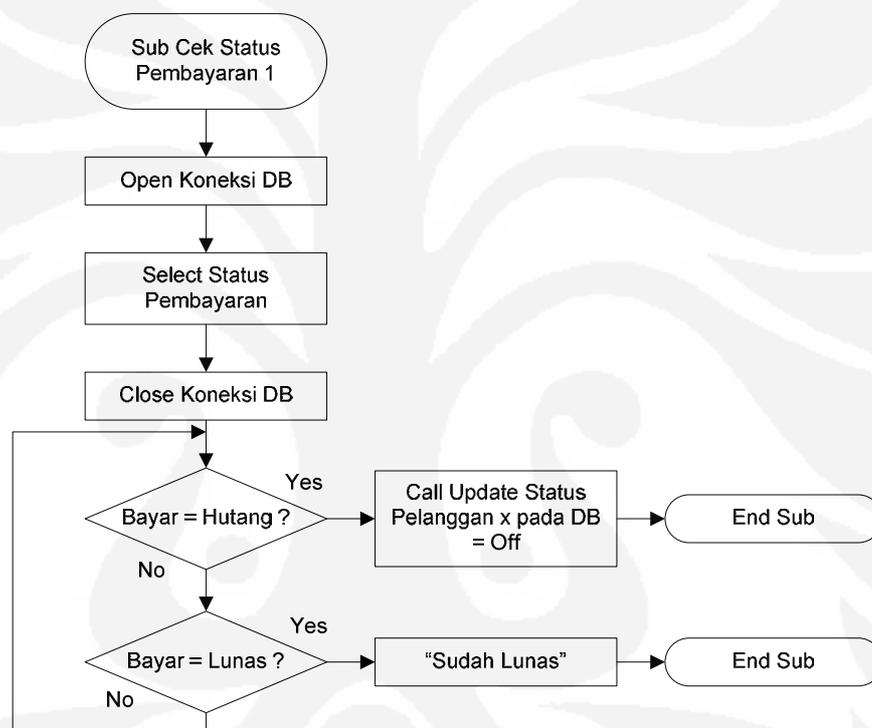
Berikutnya, sub program cek validasi yang dipanggil pada sub program cek tanggal diatas, diawali dengan perintah untuk mengecek apakah status pelanggan x sama dengan on ?. Jika statusnya sama dengan on, maka akan dilanjutkan dengan perintah untuk memanggil sub program cek status pembayaran 1, lalu diakhiri dengan perintah untuk keluar dari sub. Jika statusnya tidak sama dengan on, maka akan dilanjutkan dengan perintah untuk mengecek apakah status pelanggan x sama dengan off ?. Bila statusnya sama dengan off, maka akan dilanjutkan dengan perintah untuk memanggil sub program cek status pembayaran 2, lalu diakhiri dengan perintah untuk keluar dari sub. Bila statusnya tidak sama dengan off, maka alur algoritma akan kembali lagi pada perintah untuk mengecek apakah status pelanggan x sama dengan on ?. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.10.



Gambar 3.10 Flow Chart Algoritma Sub Cek Validasi

Sub program cek status pembayaran 1 yang dipanggil pada sub program cek validasi diatas, diawali dengan perintah untuk membuka koneksi ke database, memilih (select) status pembayaran dari pelanggan x, lalu menutup kembali koneksi dengan database. Kemudian dilanjutkan dengan perintah untuk mengecek apakah status pembayaran tersebut sama dengan hutang ?. Jika status pembayarannya sama dengan hutang, maka akan dilanjutkan dengan perintah untuk memanggil sub program update status pelanggan x pada database = off (dapat dilihat pada Gambar 3.13), lalu diakhiri dengan perintah untuk keluar dari

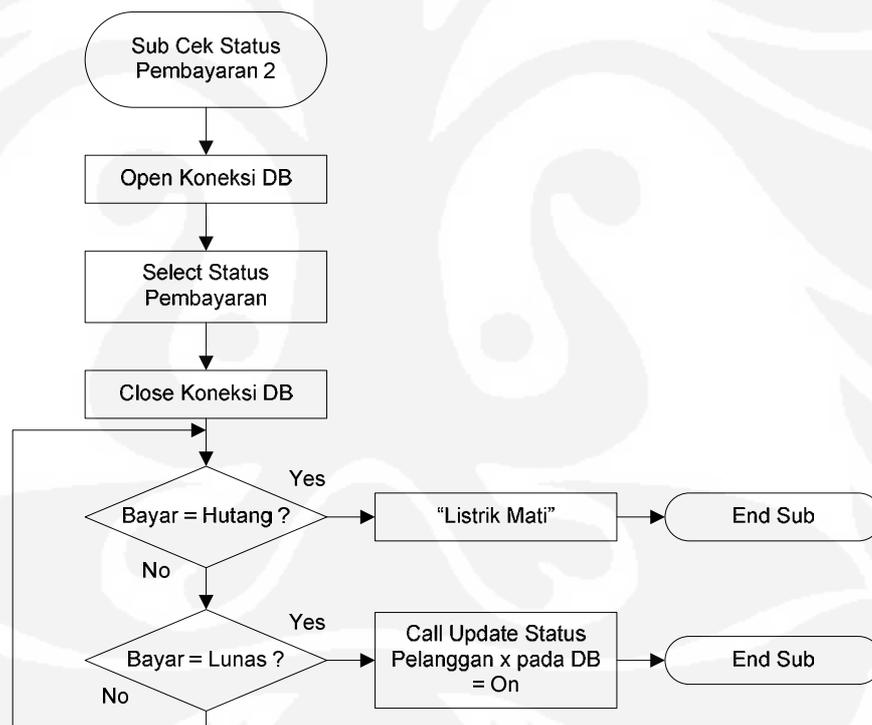
sub. Jika status pembayarannya tidak sama dengan hutang, maka akan dilanjutkan dengan perintah untuk mengecek apakah status pembayaran pelanggan x sama dengan lunas ?. Bila status pembayarannya sama dengan lunas, maka akan muncul “sudah lunas” pada text box sistem AMR bridge, lalu diakhiri dengan perintah untuk keluar dari sub. Bila status pembayaran tidak sama dengan lunas, maka alur algoritma pemrograman akan kembali ke perintah untuk mengecek apakah status pembayaran pelanggan x sama dengan hutang ?. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.11.



Gambar 3.11 Flow Chart Algoritma Sub Cek Status Pembayaran 1

Sub program cek status pembayaran 2 yang juga dipanggil pada sub program cek validasi, juga diawali dengan perintah untuk membuka koneksi ke database, memilih (select) status pembayaran dari pelanggan x, lalu menutup kembali koneksi dengan database. Kemudian dilanjutkan dengan perintah untuk mengecek apakah status pembayaran tersebut sama dengan hutang ?. Jika status pembayarannya sama dengan hutang, maka akan muncul “mati listrik” pada text

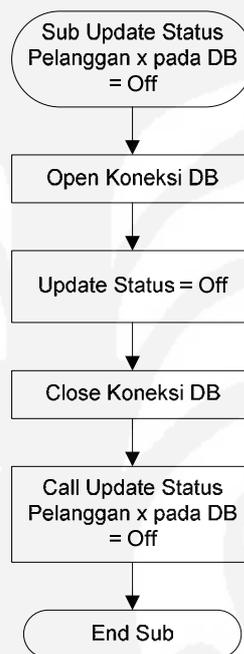
box sistem AMR bridge. Jika status pembayarannya tidak sama dengan hutang, maka akan dilanjutkan dengan perintah untuk mengecek apakah status pembayaran pelanggan x sama dengan lunas ?. Bila status pembayarannya sama dengan lunas, maka akan dilanjutkan dengan perintah untuk memanggil sub program update status pelanggan x pada database = on (flow chart dapat dilihat pada Gambar 3.14), lalu diakhiri dengan perintah untuk keluar dari sub. Bila status pembayaran tidak sama dengan lunas, maka alur algoritma pemrograman akan kembali ke perintah untuk mengecek apakah status pembayaran pelanggan x sama dengan hutang ?. Untuk lebih jelasnya, flow chart sub program ini dapat dilihat pada Gambar 3.12. Sub program cek status pembayaran 1 dan 2 berfungsi untuk memastikan keselarasan antara status on/ off dan status pembayaran pelanggan x dalam database.



Gambar 3.12 Flow Chart Algoritma Sub Cek Status Pembayaran 2

Sub program update status pelanggan x pada database = off yang dipanggil pada sub program cek status pembayaran 1, diawali dengan perintah untuk

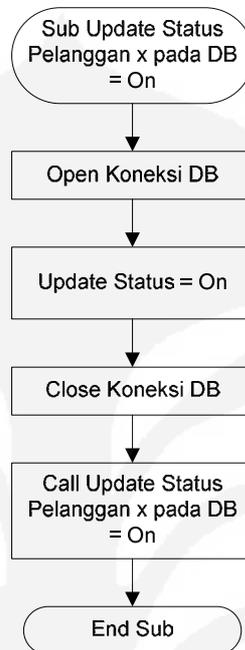
membuka koneksi ke database. Kemudian dilanjutkan dengan perintah untuk mengubah (*update*) status pelanggan x menjadi off. Kemudian dilanjutkan dengan perintah untuk menutup kembali koneksi dengan database. Kemudian dilanjutkan dengan perintah untuk memanggil sub program kirim status = off ke mikro controller (Gambar 3.16). Setelah itu, diakhiri dengan perintah untuk keluar dari sub. Untuk lebih jelasnya, dapat dilihat pada Gambar 3.13.



Gambar 3.13 Flow Chart Algoritma Sub Update Status Pelanggan x pada DB  
Sama dengan Off

Berikutnya, sub program update status pelanggan x pada database = on yang dipanggil pada sub program cek status pembayaran 2, mempunyai algoritma yang sama dengan sub program update status pelanggan x pada database = off. Sub program update status pelanggan x pada database = on yang dipanggil pada sub program cek status pembayaran 1, diawali dengan perintah untuk membuka koneksi ke database. Kemudian dilanjutkan dengan perintah untuk mengubah (*update*) status pelanggan x menjadi on. Kemudian dilanjutkan dengan perintah untuk menutup kembali koneksi dengan database. Kemudian dilanjutkan dengan perintah untuk memanggil sub program kirim status = on ke mikro controller

(Gambar 3.15). Setelah itu, diakhiri dengan perintah untuk keluar dari sub. Untuk lebih jelasnya, dapat dilihat pada Gambar 3.14.

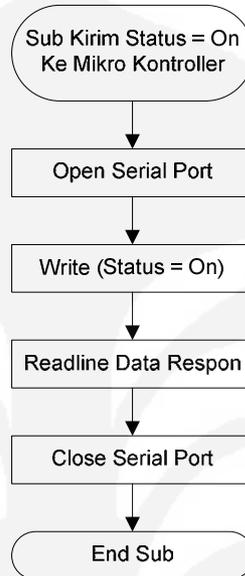


Gambar 3.14 Flow Chart Algoritma Sub Update Status Pelanggan x pada DB  
Sama dengan On

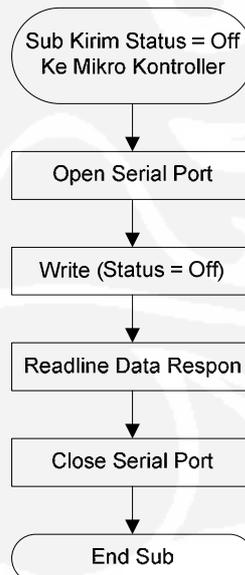
Sub program kirim status = on ke mikro controller yang dipanggil pada sub program update status pelanggan x pada database = on, diawali dengan perintah untuk membuka gerbang komunikasi serial. Lalu dilanjutkan dengan perintah untuk mengirim data “On” ke mikro controller. Kemudian dilanjutkan dengan perintah untuk menerima data respon dari mikro controller. Lalu dilanjutkan dengan perintah untuk menutup kembali gerbang komunikasi serial. Setelah itu, diakhiri dengan perintah untuk keluar dari sub (*End Sub*). Untuk lebih jelasnya, dapat dilihat pada Gambar 3.15.

Sub program kirim status = off ke mikro controller yang dipanggil pada sub program update status pelanggan x pada database = off, diawali dengan perintah untuk membuka gerbang komunikasi serial. Lalu dilanjutkan dengan perintah untuk mengirim data “Off” ke mikro controller. Kemudian dilanjutkan dengan perintah untuk menerima data respon dari mikro controller. Lalu dilanjutkan

dengan perintah untuk menutup kembali gerbang komunikasi serial. Setelah itu, diakhiri dengan perintah untuk keluar dari sub (*End Sub*). Untuk lebih jelasnya, dapat dilihat pada Gambar 3.16.



Gambar 3.15 Flow Chart Algoritma Sub Kirim Status = On ke Mikro Kontroller

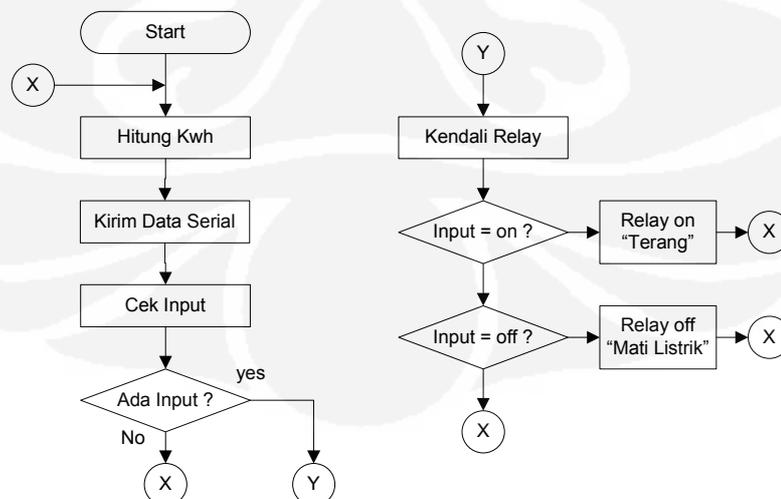


Gambar 3.16 Flow Chart Algoritma Sub Kirim Status = Off ke Mikro Kontroller

### 3.3 Perancangan Algoritma Sistem dengan BASCOM-AVR

Algoritma pemrograman mikro controller dengan BASCOM (*Basic Compiler*) AVR ini berfungsi sebagai *counter* KWH (*Kilo Watt Hour*) untuk simulasi data AMR dan menerima data serial dari windows aplikasi vb.net (AMR Bridge) untuk mengendalikan relay. Aliran algoritma pemrograman mikro controller tersebut dapat dilihat pada Gambar 3.17. Ketika mikro controller di start, maka akan menjalankan perintah hitung kwh. Perintah ini merupakan perintah untuk counter data kwh sebagai simulasi data AMR. Kemudian dilanjutkan dengan perintah kirim data serial. Perintah ini merupakan perintah untuk mengirim data serial yang terdiri dari nama, status, dan nilai kwh pelanggan x. Data serial tersebut digabung terlebih dahulu sebelum dikirim dengan pemisah berupa karakter “ ; ” (Nama;StatusListrik;Kwh).

Kemudian dilanjutkan dengan perintah cek input. Jika ada input data serial, maka aliran algoritma dilanjutkan dengan perintah kendali relay. Perintah ini merupakan perintah untuk mengecek data serial yang diterima dari sistem AMR bridge. Jika data serial yang diterima = “off”, maka relay off (memutus aliran listrik) kemudian kembali ke perintah hitung kwh. Jika data serial yang diterima = “on”, maka relay on (tidak memutus aliran listrik) kemudian kembali ke perintah hitung kwh. Bila tidak ada data serial yang diterima, maka alur algoritma akan kembali ke perintah hitung kwh.

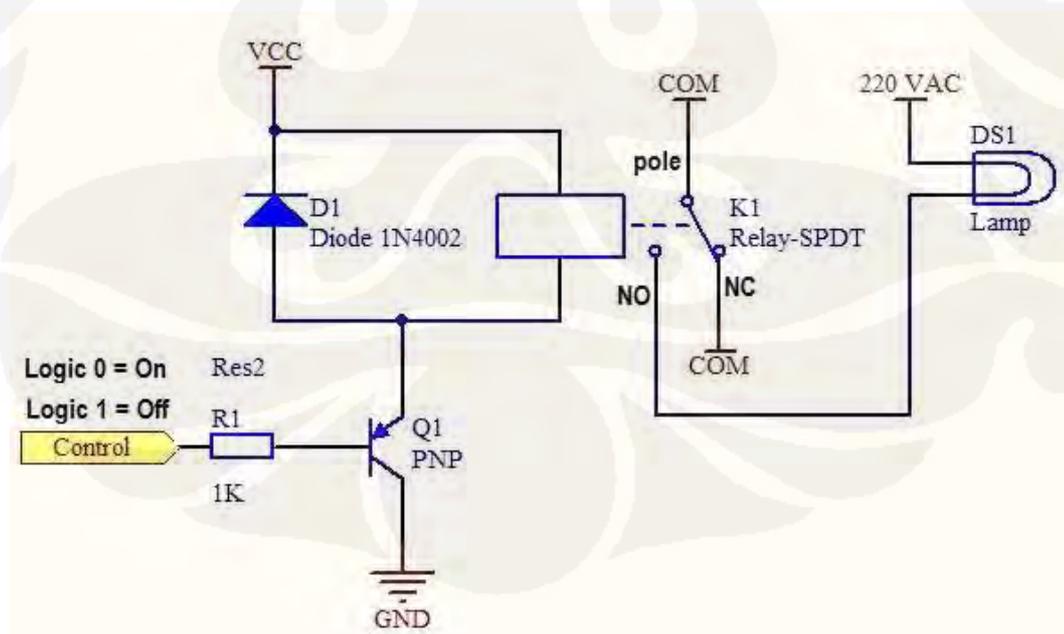


Gambar 3.17 Flow Chart Sistem dengan BASCOM-AVR

### 3.4 Perancangan Rangkaian Kendali Relay

Perancangan rangkaian pengendali relay menggunakan transistor pnp yang dirangkai sebagai switch elektronik. Switch elektronik tersebut kemudian dikendalikan oleh mikro controller dengan memberi input logika 1 atau 0 pada bagian basis transistor. Ketika basis transistor diberi logika 0, maka bagian *collector* transistor yang diberi vcc akan terhubung dengan bagian *emitor* transistor yang diberi ground. Dengan begitu solenoid/ coil mendapat beda tegangan, sehingga bagian *pole* dari relay terhubung ke bagian NO (*Normally Open*) relay yang telah dihubungkan dengan tegangan 220 VAC. Hal tersebut menyebabkan lampu mendapat beda tegangan sehingga bisa menyala.

Sebaliknya bila basis transistor diberi logika 1, maka solenoid tidak akan mendapat beda tegangan, sehingga bagian pole dari relay tetap terhubung pada bagian NC (*Normally Close*) relay yang dihubungkan dengan COM (*common*). Hal tersebut menyebabkan lampu tidak mendapat beda tegangan, sehingga tidak bisa menyala. Menyala atau tidaknya lampu merupakan simulasi pemutusan aliran listrik melalui relay yang diumpamakan terhubung pada kabel listrik sebelum *circuit breaker* dalam AMR. Untuk lebih jelasnya dapat dilihat pada Gambar 3.18.



Gambar 3.18 Rangkaian Kendali Relay

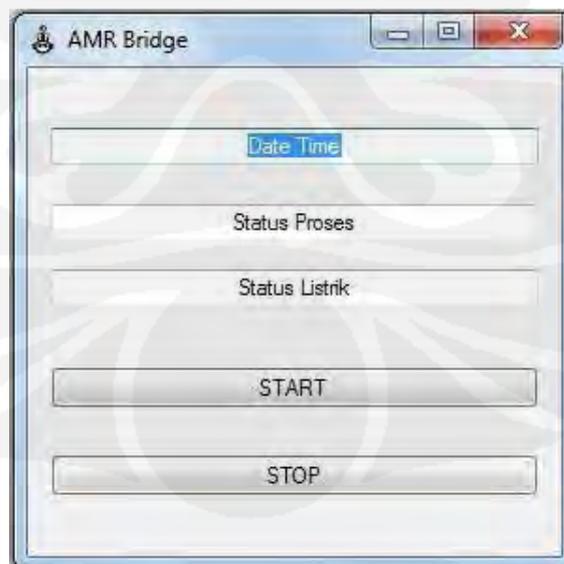
## BAB IV

### ANALISA HASIL

Bab keempat ini akan menampilkan hasil dari sistem simulasi pengendali AMR berbasis TCP/IP dan analisa dari sistem tersebut.

#### 4.1 Hasil Tampilan Sistem AMR Bridge

Ketika tombol start ditekan, sistem AMR bridge ini akan langsung menjalankan fungsinya sesuai dengan yang telah dijelaskan dalam flow chart algoritma keseluruhan sistem AMR bridge (Gambar 3.2). Pada text box pertama, akan ditampilkan tanggal dan jam yang diambil dari komputer. Text box kedua akan menampilkan status proses dari sistem berbasis vb.net yang diberi nama AMR bridge ini. Isi dari status proses telah dijelaskan dalam Gambar 3.8, Gambar 3.10, dan Gambar 3.11. Pada text box ketiga, akan ditampilkan status listrik. Jika pelanggan belum membayar tagihan listrik pada tanggal terakhir yang telah ditetapkan, maka status listrik akan berisi “Mati Listrik” dan aliran listrik pelanggan tersebut akan diputus. Sebaliknya jika pelanggan sudah melunasi tagihan listrik, maka status listrik akan berisi “Terang”. Tampilan dari sistem AMR bridge dapat dilihat pada Gambar 4.1.



Gambar 4.1 Tampilan Sistem AMR Bridge

## 4.2 Hasil Percobaan Koneksi

Koneksi database yang dimaksud dalam Table 4.1 adalah berhasil terhubung atau tidaknya aplikasi AMR bridge dalam komputer client ke database di komputer server melalui jaringan lokal (LAN). Koneksi serial yang dimaksud dalam percobaan ini adalah berhasil atau tidaknya aplikasi AMR bridge menerima dan mengirim data serial dari dan ke mikro controller. Trigger relay yang dimaksud dalam percobaan ini adalah berhasil atau tidaknya, trigger yang dilakukan oleh mikro controller untuk mengendalikan relay berdasarkan data dari database status pelanggan listrik. Berikutnya, Table 4.2 akan melengkapi hasil percobaan koneksi ini.

Tabel 4.1. Hasil Percobaan 1.1

Percobaan	Koneksi Database	Koneksi serial	Trigger Relay
1	berhasil	berhasil	berhasil
2	berhasil	berhasil	berhasil
3	berhasil	berhasil	berhasil
4	berhasil	berhasil	berhasil
5	berhasil	berhasil	berhasil
6	berhasil	berhasil	berhasil
7	berhasil	gagal	gagal
8	berhasil	berhasil	berhasil
9	berhasil	berhasil	berhasil
10	berhasil	berhasil	berhasil

## 4.3 Hasil Percobaan Sistem AMR Bridge

Table 4.2 menunjukkan data dari sistem AMR bridge secara garis besar. Pada tabel tersebut dicantumkan data tanggal, status (On/ Off), status pembayaran, lampu, dan waktu. Data tanggal menunjukkan kapan sistem AMR bridge diaktifkan. Status (On/ Off) dan status pembayaran menunjukkan data dalam database pelanggan x yang terdapat dalam komputer server. Data lampu, menunjukkan keselarasan antara status (On/ Off) dengan status pembayaran. Jika status “On” dan pembayaran “lunas”, maka lampu akan menyala. Sebaliknya, jika status “Off” dan pembayaran “hutang”, maka lampu akan padam. Namun, bila

status (On/ Off) dan status pembayaran tidak selaras, maka program pengecekan validasi status yang terdapat dalam system AMR akan bekerja secara otomatis untuk menyelaraskannya. Terakhir, data waktu menunjukkan waktu yang dibutuhkan untuk menyalakan atau mematikan lampu, mulai dari perubahan status dalam database di komputer server.

Table 4.2 Hasil Percobaan 1.2

Percobaan	Tanggal	Status (On/ Off)	Status Pembayaran	Lampu	Waktu (detik)
1	11	On	Lunas	menyala	4
2	11	Off	Hutang	padam	4
3	11	On	Lunas	menyala	5
4	11	Off	Hutang	padam	4
5	11	On	Lunas	menyala	4
6	11	Off	Hutang	padam	4
7	11	On	Lunas	menyala	4
8	11	Off	Hutang	padam	4
9	11	On	Lunas	menyala	4
10	11	Off	Hutang	padam	4

#### 4.4 Hasil Percobaan Akurasi Data

Percobaan akurasi data dilakukan dengan cara mengamati data yang diterima oleh sistem AMR bridge dari sistem mikro kontroller dengan data yang dikirim oleh sistem AMR bridge ke database pelanggan x dalam komputer server setiap lima detik. Tabel 4.3 menunjukkan nilai data kwh yang diterima oleh sistem AMR bridge dari mikro kontroller, data kwh yang terkirim ke dalam database pelanggan x, persentase akurasi antara data-data tersebut, dan delay yang terjadi saat pengiriman data kwh ke dalam database pelanggan x.

Delay dihitung dengan menggunakan *stop watch* yang dimulai saat sistem AMR bridge menerima data kwh dan diakhiri saat database menerima data kwh yang sama dengan data kwh yang diterima sistem AMR bridge. Sedangkan persentase akurasi didapat dengan cara membagi nilai kwh (dalam sistem AMR bridge) dengan nilai kwh (dalam database) setiap lima detik.

Table 4.3 Hasil Percobaan 1.3

Percobaan	Nilai Kwh (dalam sistem AMR Bridge)	Nilai Kwh (dalam database)	Delay (detik)	Persentase Akurasi
1	1055	1055	1	100%
2	1056	1056	1	100%
3	1057	1057	1	100%
4	1058	1058	1	100%
5	1059	1059	1	100%
6	1060	1060	1	100%
7	1061	1061	1	100%
8	1062	1062	1	100%
9	1063	1063	1	100%
10	1064	1064	1	100%

#### 4.5 Analisa

Dari hasil percobaan terlihat bahwa aplikasi AMR bridge berhasil melakukan koneksi dengan database dalam komputer server melalui kabel LAN. AMR bridge juga berhasil melakukan koneksi serial ke mikro controller. Data yang dikirim melalui koneksi serial tersebut juga telah berhasil memicu relay sesuai dengan ketentuan. Saat data serial yang diterima mikro controller sama dengan “On”, maka pemicu relay akan diberi logika 0. Dengan begitu bagian *pole* dari relay akan terhubung ke bagian NO (*Normally Open*) relay sehingga lampu menyala. Sebaliknya, saat data serial yang diterima mikro controller sama dengan “Off”, maka pemicu relay akan diberikan logika 1. Dengan begitu bagian pole dari relay akan tetap terhubung ke bagian NC (*Normally Close*) relay sehingga lampu tidak menyala. Waktu proses untuk menyalakan atau mematikan lampu cukup cepat untuk koneksi ke database di komputer server melalui jaringan lokal.

Dari hasil percobaan akurasi data, dapat terlihat bahwa sistem AMR bridge berhasil mengirim data kwh dengan sangat akurat. Hal tersebut masih sangat mungkin terjadi karena koneksi dari sistem AMR bridge ke database dalam komputer server masing menggunakan jaringan lokal. Bila telah menggunakan jaringan internet, kemungkinan besar akurasi akan berkurang karena pengaruh kepadatan data dan kerusakan data dalam jaringan internet.

## **BAB V**

### **KESIMPULAN**

- Aplikasi AMR Bridge telah berhasil menjembatani database dalam komputer server dan mikro controller pada AMR
- Aplikasi AMR Bridge telah berhasil menyalakan atau memadamkan lampu melalui mikro controller dan rangkaian trigger relay berdasarkan data dari database dalam komputer server dengan delay rata-rata empat detik.
- Aplikasi AMR Bridge telah berhasil mengirim data ke database dalam komputer server dengan akurasi 100% dan delay satu detik.

## DAFTAR ACUAN

1. Amrfinal.ppt  
<[www.tcil-india.com/new/new\\_site/white%20paper/amrfinal.ppt](http://www.tcil-india.com/new/new_site/white%20paper/amrfinal.ppt)>
2. Tamarkin, Tom D., Automatic Meter Reading, as published in Public Power Magazine volume 50 number 5, 1992
3. Pressman,R.S., Software Engineering: A Practitioner's Approach, The McGraw-Hill, 1997.
4. Cedeno,Nancy, The Internet Tool Kit, Sybex, 1995.
5. Net,Genesis & Devra,Hall, Build A Web Site, Canbridge, Massachusetts, 1995.
6. Craig Hunt, TCP/IP Network Administration, O'reilly & Associates, inc., 1992.
7. Albitz Paul & liu Cricket, DNS & BIND, O'reilly & Associates, inc., 1992.
8. Purbo, Onno W., TCP/IP, Elex Media Komputindo, 1998.
9. Ramadhan, Arief, VB.NET 2005, Elex Media Komputindo, 2006
10. Budiharto, Widodo, Kendali Cerdas Berbasis SMS/ Web/ TCP-IP, Elex Media Komputindo, 2009