

SISTEM PENGENDALI PERALATAN ELEKTRONIK SERTA PEMANTAUAN SUHU RUANGAN BERBASIS MIKROKONTROLER DENGAN MEDIA KOMUNIKASI JALA-JALA

**Vinsensius Rahmat Setyo Purnomo¹, Budihardja Murtianta²,
Darmawan Utomo³**

Program Studi Sistem Komputer Fakultas Teknik Elektronika dan Komputer
Universitas Kristen Satya Wacana, Salatiga.

vinsensius_rsp@yahoo.com¹, budihardja@yahoo.com², darmawan@staff.uksw.edu³

INTISARI

Penggunaan peralatan elektronik seperti lampu penerangan ataupun perangkat listrik lainnya semakin meningkat. Pada umumnya setiap peralatan elektronik memiliki tombol saklar yang letaknya berada di dekat peralatan sehingga pengguna harus mendekat untuk mengaktifkannya.

Pada makalah ini akan dijelaskan perancangan sebuah sistem pengendali jarak jauh peralatan elektronik secara nyala/mati menggunakan Komunikasi Jala-Jala / *Power Line Carrier* (PLC) sebagai metode komunikasi antar modul. Selain untuk pengendalian jarak jauh, sistem ini juga dapat digunakan untuk pemantauan suhu ruangan.

Cara kerja sistem yang dibuat ialah dengan cara mengirimkan data termodulasi *Frequency Shift Keying* (FSK) yang berisi perintah atau kondisi suhu ruangan melalui komunikasi jala-jala, data tersebut dilewatkan melalui kabel bertegangan 220 VAC. Pengujian sistem pada saluran listrik bertegangan 220 VAC dengan titik terjauh sepanjang 85 meter berhasil dilakukan karena sistem memiliki mekanisme *collision detection* yaitu sebuah algoritma untuk memastikan perintah yang dikirimkan ke modul *slave* berhasil diterima, dengan cara mengirim ulang perintah sebanyak sepuluh kali pengiriman jika modul *master* belum mendapat *ack* dari

modul *slave*. Sistem ini terdiri dari dua buah modul *master* dan empat buah modul *slave*, yang mana setiap modul *master* dapat mengendalikan dua buah modul *slave*. Setiap modul *master* dan modul *slave* dirancang dengan mikrokontroler ATmega8535 sebagai komponen pengendali utama. Sistem yang dibuat dapat dipakai di saluran bertegangan 220 VAC satu fasa, dan dapat berkomunikasi secara *half-duplex* dengan jarak terjauh 85 meter.

Kata Kunci : Sistem pengendali beban AC, Pemantau suhu, Komunikasi jala-jala, *Power line carrier*.

1. PENDAHULUAN

Pada bagian ini akan dijelaskan mengenai tujuan dan latar belakang dari sistem yang dibuat

1.1. Tujuan

Merancang dan merealisasikan sebuah sistem yang dapat digunakan untuk mengendalikan peralatan elektronik secara nyala atau mati dan dapat melakukan pemantauan suhu ruangan berbasis mikrokontroler melalui media komunikasi jala-jala.

1.2. Latar Belakang

Seiring dengan perkembangan teknik elektronika, semakin meningkat pula penggunaan peralatan elektronik seperti lampu penerangan ataupun kipas angin. Pada umumnya setiap peralatan elektronik memiliki tombol saklar yang letaknya berada di dekat peralatan untuk mengaktifkannya.

Dalam suatu lingkungan kerja jarak antar ruang pada umumnya saling berjauhan, maka tidak efektif apabila harus berjalan jauh sekedar untuk mengaktifkan peralatan elektronik tersebut, yang tentunya akan boros waktu dan tenaga. Atau pun secara konvensional dapat dibuat suatu sistem saklar jarak jauh, akan tetapi sistem tersebut akan membutuhkan banyak kabel yang digunakan untuk perpanjangan tombol saklar yang akan memboroskan biaya, selain itu dalam proses pembuatan instalasi kabel saklar yang merepotkan.

Berdasar permasalahan ini, telah dibuat sebuah sistem pengendali jarak jauh melalui media kabel listrik PLN bertegangan AC 220 V dengan frekuensi 50-60 Hz.

Metode pengiriman data melalui kabel instalasi listrik PLN tersebut dikenal dengan nama Komunikasi Jala-Jala, dalam bahasa Inggris dikenal dengan istilah *Power Line Communication / Power Line Carrier (PLC)*. Pada metode komunikasi *Power Line Carrier* data dikirimkan melalui kabel listrik PLN, sehingga tidak diperlukan lagi instalasi kabel komunikasi lain. Selain untuk mengendalikan peralatan elektronik sistem juga dilengkapi dengan sensor suhu untuk memantau kondisi suhu ruangan. Keuntungan yang didapatkan apabila menggunakan sistem ini adalah pengguna tidak perlu berpindah tempat untuk mengendalikan peralatan elektronik, instalasi kabel listrik PLN lebih efektif karena dapat digunakan untuk media komunikasi, dapat menghemat biaya, tenaga dan waktu.

2. TINJAUAN PUSTAKA

Pada bab ini akan mengenai komunikasi jala-jala / *power line carrier (PLC)* dan modulasi *binary frequency shift keying (BFSK)*.

2.1. Komunikasi Jala-Jala / *Power Line Carrier (PLC)*

Sejarah teknologi komunikasi jala-jala pada awalnya dikembangkan oleh Dr. Paul Brown pada tahun 1991, beliau yang melakukan riset tentang kelayakan menggunakan jala – jala listrik sebagai media komunikasi [1].

Pada prinsipnya, komunikasi jala-jala adalah menumpangkan sinyal data pada listrik PLN sebagai media transmisinya. Sinyal yang ditumpangkan adalah sinyal data yang telah termodulasi. Salah satu keuntungan yang didapat apabila menggunakan komunikasi jala-jala adalah tidak perlu membuat instalasi jalur komunikasi lagi karena komunikasi jala-jala menggunakan instalasi listrik yang sudah ada sebagai media transmisi. Jadi, pada setiap area yang telah memiliki jaringan listrik PLN bisa digunakan metode komunikasi jala-jala [4]. Kekurangan komunikasi jala-jala adalah komunikasi ini tidak dapat terhubung apabila di antara kedua titik terdapat transformator.

Prinsip penumpangkan sinyal pembawa pada listrik PLN adalah superposisi dua buah gelombang yang memiliki perbedaan karakteristik. Gelombang pembawa memiliki frekuensi yang cukup tinggi (125 kHz) namun memiliki amplitudo yang kecil, sedangkan listrik PLN hanya memiliki frekuensi 50 Hz namun amplitudonya

cukup tinggi. Dengan kata lain proses penumpangan sinyal pembawa pada listrik PLN adalah dengan cara menjumlahkan kedua gelombang tersebut.

2.2. Modulasi *Binary Frequency Shift Keying* (BFSK)

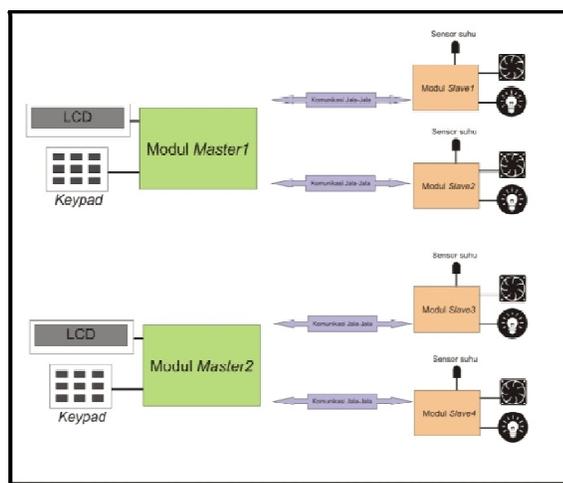
Binary Frequency Shift Keying (BFSK) adalah sebuah metode modulasi data digital, sesuai dengan namanya metode modulasi ini mengubah nilai frekuensi *carrier* untuk merepresentasikan data digital berdasarkan pada kondisi data.

Dalam transmisi data digital ada dua macam kondisi yaitu ‘1’ dan ‘0’ atau *high* dan *low*. Berdasar kondisi tersebut data akan dimodulasi secara BFSK, sinyal *carrier* atau sinyal pembawa akan dibuat bervariasi untuk merepresentasikan data. Sinyal pembawa yang dibentuk berupa gelombang sinus. Frekuensi sinyal pembawa dibagi menjadi dua yaitu f_1 dan f_2 . Frekuensi sinyal pembawa f_1 digunakan untuk merepresentasi data yang bernilai ‘0’ dan frekuensi f_2 digunakan untuk merepresentasikan data yang bernilai ‘1’ [3, h.146-148].

3. PERANCANGAN SISTEM

Pada bagian ini akan dijelaskan perancangan sistem pengendali peralatan elektronik serta pemantauan suhu ruangan yang menggunakan komunikasi jala-jala sebagai media komunikasi sistem. Perancangan terdiri dari perancangan perangkat keras dan perancangan perangkat lunak.

3.1. Gambaran Sistem



Gambar 1. Blok diagram sistem yang dirancang

Gambar 1 menunjukkan blok diagram sistem yang dirancang, secara umum sistem yang dirancang terdiri dari dua bagian utama yaitu modul *master* dan modul *slave*. Setiap modul *master* dapat mengontrol dua buah modul *slave* dan dalam pengoperasiannya modul *master* dan modul *slave* saling berkomunikasi melalui media komunikasi jala-jala.

3.2. Perancangan Perangkat Keras

Pada bagian ini akan dijelaskan mengenai perancangan perangkat keras sistem. Perancangan perangkat keras meliputi perancangan pada modul *master* dan pada modul *slave*.

3.2.1. Perangkat Keras Modul Master

Modul *master* adalah bagian di mana pengguna dapat berinteraksi langsung dengan sistem, melalui bagian ini pengguna dapat memberikan perintah untuk mengontrol peralatan elektronik yang terhubung pada modul *slave* ataupun mengamati kondisi peralatan elektronik dan keadaan suhu pada modul *slave*. Ada empat bagian utama pada modul *master* yaitu pengendali utama, modem PLC, antarmuka dan catu daya.

Pengendali utama merupakan pusat dari modul dan bertugas untuk mengontrol lalu-lintas data pada sistem, semua bagian pada modul *master* berpusat pada bagian ini. Beberapa fungsi bagian pengendali utama antara lain menerima input dari bagian antarmuka yakni *keypad*, menampilkan data yang diterima pada LCD, mengatur mode kerja modem PLC dan menyediakan komunikasi serial dengan komputer agar sistem dapat terhubung dengan komputer pemantau [5]. Bagian ini dirancang berbasis mikrokontroler AVR ATmega8535 buatan Atmel Corporation [2].

Modem PLC dirancang dengan IC LM1893 buatan National Semiconductor, LM1893 adalah IC khusus untuk untuk membuat modem *power line carrier*. Tipe modulasi yang digunakan dalam IC ini adalah *frequency shift keying* (FSK) dengan rentang frekuensi sinyal pembawa antara 50 kHz sampai 300 kHz. Hasil modulasi yang dilakukan oleh modem berupa gelombang sinus dengan frekuensi $\pm 127,75$ kHz untuk merepresentasikan data '0' dan $\pm 122,25$ kHz untuk merepresentasikan data '1' [7, h.14].

Bagian antarmuka berfungsi untuk sarana interaksi pengguna dengan sistem yang dirancang. Bagian ini secara keseluruhan terhubung dan dikendalikan oleh bagian pengendali utama. Ada dua komponen utama pada bagian antarmuka yaitu *keypad* 4x4 dan LCD karakter 16x2.

Catu daya merupakan bagian yang bertanggung jawab dalam menyediakan daya yang dibutuhkan oleh seluruh bagian modul *master*. Rangkaian catu daya dibuat dengan beberapa komponen yaitu 1 buah trafo CT 300 mA, empat buah dioda 1N4002 yang disusun *bridge*, dua buah kapasitor elektrolit 100 μ F / 25 V dan satu buah IC LM7818.

3.2.2. Perangkat Keras Modul *Slave*

Modul ini dinamakan modul *slave* karena tugas utama modul ini hanya menjalankan perintah yang diberikan oleh modul *master*. Pada modul inilah pengguna dapat menghubungkan peralatan elektronik yang nantinya akan dikontrol melalui modul *master*, selain itu pada modul ini juga terdapat sensor suhu yang digunakan untuk mengukur suhu ruangan tempat modul *slave* berada. Ada lima bagian utama dari modul *slave* yaitu pengendali utama, modem PLC, sensor suhu, saklar, dan catu daya. Selanjutnya akan dijelaskan mengenai perancangan dan fungsi masing-masing bagian dari modul *slave*.

Bagian pengendali utama dirancang dengan mikrokontroler ATmega8535 sebagai pusat pengolah data dan mengendalikan bagian yang lainnya. Berbeda dengan pengendali utama modul *master* perancangan pengendali utama modul *slave* jauh lebih sederhana dikarenakan fungsi bagian ini juga tidak sekompleks bagian *master*. Fitur yang terdapat pada bagian ini adalah *buzzer* sebagai *alarm* dan regulator tegangan 5V yang nantinya dibutuhkan juga oleh sensor suhu dan bagian saklar.

Sensor suhu yang digunakan pada sistem adalah sensor suhu DS18B20 buatan Dallas Semiconductor. Penggunaan sensor suhu ini cukup mudah karena pada penggunaannya sensor suhu DS18B20 hanya membutuhkan satu buah pin mikrokontroler sebagai untuk komunikasi dengan mikrokontroler itu sendiri [6].

Rangkaian saklar dirancang dengan komponen utama optotriac MOC3020 dan triac BT136. Optotriac MOC3020 dipilih karena dengan menggunakan optik maka rangkaian pengendali utama akan terisolasi dari rangkaian bertegangan tinggi

sehingga akan lebih aman bagi rangkaian pengendali apabila terjadi kesalahan pada rangkaian saklar.

3.3. Perancangan Perangkat Lunak

Pada bagian ini akan dijelaskan mengenai perancangan protokol komunikasi data dan aplikasi *desktop* pemantau kondisi sistem.

3.3.1. Protokol Komunikasi Data

Metode komunikasi yang digunakan pada sistem ini adalah komunikasi serial yang memanfaatkan fasilitas komunikasi serial yang sudah ada pada mikrokontroler dengan kecepatan pengiriman 600 bps. Mode komunikasi yang digunakan dalam perancangan ini adalah *halfduplex* yang berarti modul *master* maupun modul *slave* dapat saling berkomunikasi dua arah secara bergantian.

Pada saat modul berkomunikasi ketika modul *master* memberikan perintah atau ketika modul *slave* mengirimkan kondisinya, data yang dikirimkan berupa *string* karakter. *String* yang dikirimkan diawali dengan karakter '*' sebagai tanda awal data dan diakhiri dengan karakter '#' sebagai tanda akhir dari *string* yang dikirim. Tabel 1 menunjukkan data yang dikirim oleh modul *master1*.

Tabel 1. Data yang dikirim oleh modul *Master1*

<i>String</i>	Fungsi
*1ddd#	Perintah untuk meminta kondisi modul <i>slave1</i>
*2ddd#	Perintah untuk meminta kondisi modul <i>slave2</i>
*1s'x'x'#	Perintah untuk mengubah kondisi peralatan elektronik pada <i>slave1</i>
*2s'x'x'#	Perintah untuk mengubah kondisi peralatan elektronik pada <i>slave2</i>

Penjelasan perintah-perintah yang dikirimkan oleh modul *master1* adalah sebagai berikut. *String* *1ddd# dan *2ddd# merupakan perintah untuk meminta kondisi modul *slave*. Karakter '*' adalah penanda awal *string*, karakter berikutnya adalah alamat modul *slave* '1' untuk *slave1* dan '2' untuk *slave2* kemudian karakter 'd' adalah sebagai kode untuk perintah meminta kondisi sedangkan karakter '#' adalah tanda akhir dari *string*.

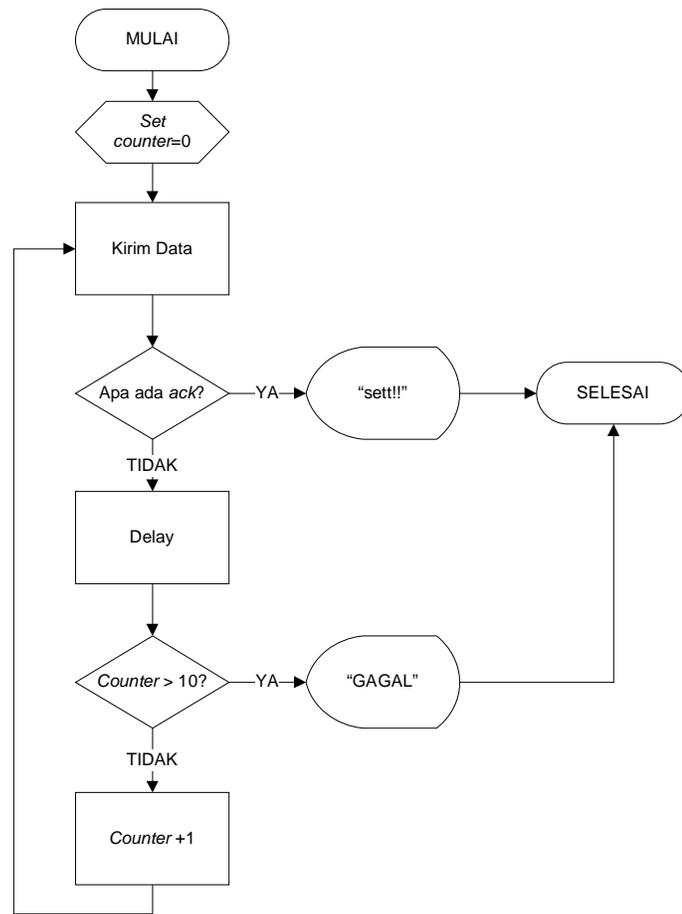
Tabel 2 menunjukkan data yang dikirim oleh modul *Slave1* dan *Slave2*.

Tabel 2. Data yang dikirim oleh modul *Slave1* dan *Slave2*

String	Fungsi
*1aaaaa#	Ack dari <i>slave1</i>
*2aaaaa#	Ack dari <i>slave2</i>
*1d'ss'x'x'#	Kondisi pada <i>slave1</i>
*2d'ss'x'x'#	Kondisi pada <i>slave2</i>

Penjelasan dari Tabel 2 adalah sebagai berikut. *String* *1aaaaa# dan *2aaaaa# memiliki fungsi untuk memberi tahu *master1* jika perintah telah diterima oleh *slave*, karakter '*' sebagai penanda awal, karakter berikutnya menunjukkan nama *slave* pengirim '1' untuk *slave1* dan '2' untuk *slave2*. Kemudian karakter 'a' sebagai kode *ack*, dan terakhir karakter '#' sebagai penanda akhir.

Karena pada sistem yang dirancang terdapat dua buah modul *master* maka tidak menutup kemungkinan adanya *data collision* atau tumbukan data. Tumbukan data dapat terjadi ketika terjadi pengiriman data secara bersamaan sehingga menyebabkan data rusak. Oleh karena itu dibutuhkan sebuah mekanisme *collision detection*. Mekanisme berfungsi untuk mendeteksi dan kemudian menghindari terjadinya tumbukan data, sehingga data dapat terkirim. Gambar 2 menunjukkan diagram alir mekanisme *collision detection*.



Gambar 2. Diagram alir mekanisme *collision detection*.

3.3.2. Aplikasi *Desktop* Pemantau Sistem

Aplikasi pemantau sistem diinstall pada komputer pemantau (PC), Fungsi utama utama aplikasi ini adalah untuk memantau kondisi sistem yang sedang berjalan. Aplikasi akan menampilkan kondisi modul *slave*. Aplikasi ini dirancang menggunakan Visual Studio 2010 menggunakan bahasa pemrograman C#.

Modul *master* akan mengirimkan data setiap satu menit sekali untuk memperbaharui kondisi sebuah modul *slave*. Data yang dikirimkan modul master berupa string karakter dengan bentuk “*slave&suhu&d1&d2#”. Penjelasan dari data tersebut adalah sebagai berikut, karakter ‘*’ merupakan awalan atau *header* dari sebuah data, karakter ‘&’ merupakan pemisah data yang dikirimkan, karakter ‘#’ merupakan tanda dari akhir sebuah data. Sebagai contoh modul *master* mengirimkan data sebagai berikut *s1 &29&1 &0#, dari data tersebut didapatkan informasi :

- alamat *slave* : *slave 1*
- suhu : 29°C
- *device1* : ON
- *device2* : OFF

4. PENGUJIAN DAN ANALISIS

Pengujian ini dilakukan untuk mengetahui kinerja sistem yang telah dibuat ketika dirangkai pada sebuah jalur listrik AC 220V. Pengujian yang dilakukan adalah pengujian pengiriman perintah oleh modul *master* serta pengujian kinerja mekanisme *collision detection*.

Kriteria pengujian yang telah dilakukan adalah sebagai berikut :

- Seluruh modul dirangkai pada seutas kabel yang memiliki aliran listrik satu fasa 220V.
- Jarak antara *master1* dan *master2* adalah ± 1 meter.
- Jarak antara *master2* dan *slave1* adalah 50 meter.
- Jarak antara *slave1* dan *slave2* adalah 10 meter.
- Jarak antara *slave2* dan *slaveA* adalah 10 meter.
- Jarak antara *slaveA* dan *slaveB* adalah ± 15 meter.

Pada perancangan, *master1* hanya dapat mengendalikan *slave1* dan *slave2* dan *master2* hanya dapat mengendalikan *slaveA* dan *slaveB*.

Pengujian pertama yang dilakukan adalah pengujian terhadap kinerja modul *master* dalam memberikan perintah untuk mengubah kondisi modul *slave*. Setiap modul *master* mengirimkan sepuluh kali pengiriman perintah seting untuk setiap modul *slave* selanjutnya dicatat hasilnya. Ada dua kriteria dalam pengujian ini yaitu keberhasilan modul *slave* dalam mengubah kondisi peralatan elektronik dan pengembalian *ack* dari modul *slave* yang menandakan modul *slave* telah melakukan seting peralatan elektronik.

Tabel 3 menunjukkan hasil pengujian pada *master1* dan Tabel 4 menunjukkan hasil pengujian pada *master2*.

Tabel 3. Hasil pengujian *Master1*.

Pengujian	Slave1		Slave2	
	seting	ack	seting	ack
1	√	√	√	√
2	√	√	√	√
3	√	x	√	√
4	√	√	√	√
5	√	√	√	√
6	√	√	√	√
7	√	x	√	√
8	√	√	√	√
9	√	√	√	x
10	√	√	√	√
Prosentase Keberhasilan (%)	10/10= 100%	8/10= 80%	10/10= 100%	9/10= 90%

Tabel 4. Hasil pengujian *Master2*.

Pengujian	SlaveA		SlaveB	
	seting	ack	seting	ack
1	√	√	√	√
2	√	√	√	√
3	√	√	√	√
4	√	√	√	√
5	√	√	√	x
6	√	√	√	√
7	√	√	√	√
8	√	√	√	√
9	√	√	√	√
10	√	√	√	√
Prosentase Keberhasilan (%)	10/10= 100%	10/10= 100%	10/10= 100%	9/10= 90%

Dari hasil pengujian yang ditunjukkan pada Tabel 3 dan Tabel 4, baik *master1* maupun *master2* dapat memberikan perintah dengan baik, akan tetapi terkadang modul *master* tidak dapat menerima *ack* yang telah dikirimkan oleh modul *slave*. Hal tersebut dikarenakan terjadinya tumbukan pengiriman data, ketika modul *slave* mengirimkan *ack* kepada modul *master* pada saat yang bersamaan modul *master*

yang lainya sedang meminta kondisi modul *slave* yang lain pula, sehingga mengganggu jalur komunikasi. Apabila *ack* yang dikirimkan oleh modul *slave* tidak diterima oleh modul *master*, maka modul *master* akan tetap mengirimkan perintah walaupun sebenarnya modul *slave* telah mengubah kondisi sesuai dengan perintah yang telah dikirimkan sebelumnya.

Pengujian selanjutnya adalah pengujian mekanisme *collision detection*, fungsi utama dari mekanisme ini adalah untuk memastikan perintah yang dikirimkan oleh modul *master* dapat diterima oleh modul *slave*. Cara kerja mekanisme ini adalah dengan cara mengirimkan perintah secara berulang apabila modul *master* belum mendapatkan *ack*, pengulangan pengiriman data dilakukan sebanyak sepuluh kali sebelum dinyatakan gagal karena modul *master* tidak menerima *ack*. Keberhasilan mekanisme tersebut dapat terlihat ketika kedua modul *master* mengirimkan perintah kepada masing-masing *slave* pada saat yang bersamaan. Mekanisme *collision detection* dianggap berhasil ketika kedua buah modul master dapat mengirimkan perintah pada masing-masing *slavenya*. Pengujian dilakukan sebanyak dua puluh kali pengujian, Tabel 5 menunjukkan hasil pengujian mekanisme *collision detection*.

Tabel 5. Hasil pengujian mekanisme *collision detection*.

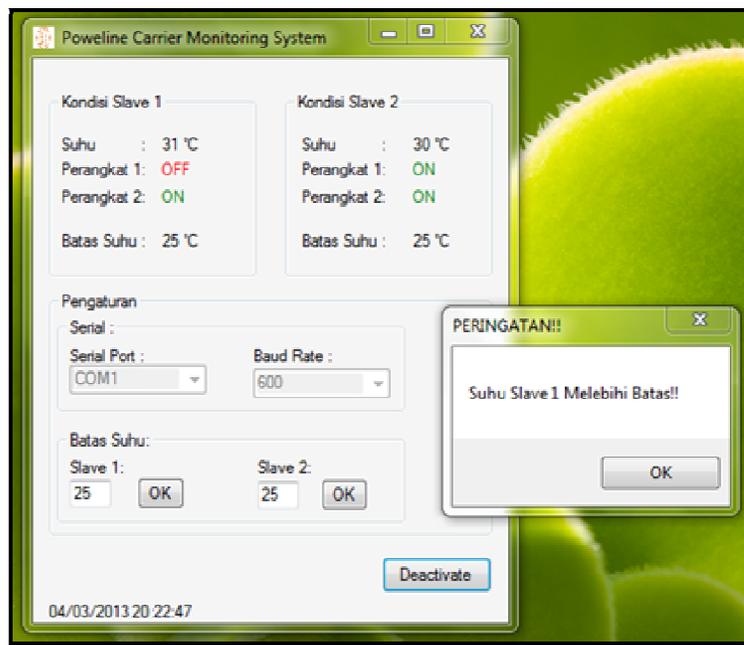
Pengujian	Keberhasilan pengiriman perintah	
	<i>Master1</i>	<i>Master2</i>
1	√	√
2	√	√
3	x	√
4	√	√
5	√	√
6	√	√
7	√	√
8	√	x
9	√	√
10	√	√
11	√	√
12	√	√
13	√	√
14	x	√
15	√	√
16	x	√
17	√	√
18	√	√
19	√	x
20	√	√
Prosentase Keberhasilan (%)	17/20= 85%	18/20= 90%

Dari hasil pengujian yang ditunjukkan pada Tabel 5, dari dua puluh kali pengujian yang telah dilakukan lima diantaranya mekanisme *collision detection* gagal melakukan tugasnya. Jika mekanisme *collision detection* gagal maka yang terjadi adalah perintah tidak dapat diterima oleh modul *slave* atau perintah berhasil dikirimkan namun *ack* dari modul *slave* tidak diterima kembali oleh modul *master*. Apabila perintah gagal dikirimkan ke modul *slave* maka penampil LCD pada modul *master* akan menampilkan notifikasi “GAGAL” selama dua detik kemudian kembali

ke menu pengubah kondisi modul *slave*. Berdasarkan hasil pengujian tersebut mekanisme *collision detection* dianggap cukup berhasil dalam menghindari tumbukan data yang menyebabkan gagalnya komunikasi antara modul *master* dan *slave*.

Pengujian berikutnya adalah pengujian aplikasi *desktop*. Aplikasi *desktop* hanya dapat dijalankan pada komputer yang memakai sistem operasi windows. Komputer harus dihubungkan dengan modul *master* melalui komunikasi serial agar aplikasi *desktop* dapat terhubung dengan sistem. Sebelum komunikasi diaktifkan terlebih dahulu dilakukan pengaturan agar komunikasi dapat berjalan dengan baik, beberapa pengaturan yang dilakukan ialah pengaturan serial port, pengaturan kecepatan pengiriman data, dan pengaturan batas maksimal suhu pada aplikasi *desktop*.

Pada pengujian batas suhu diatur menjadi 25° C, ketika data dari modul *master* diterima oleh komputer pemantau dan data suhu *slave1* yang diterima melebihi batas suhu yang telah ditentukan maka aplikasi *desktop* memberikan peringatan yang bertuliskan “Suhu Slave 1 Melebihi Batas!!” selain itu aplikasi *desktop* juga membunyikan suara *alarm* untuk memberi peringatan pada pengguna. Gambar 3 menunjukkan hasil pengujian aplikasi *desktop*.



Gambar 3. Pengujian aplikasi *desktop*.

5. KESIMPULAN

Metode *power line carrier* atau komunikasi jala-jala berhasil diaplikasikan pada sistem pengendali daya secara nyala/mati dan sistem pemantauan suhu. Modul *master* maupun modul *slave* dapat saling berkomunikasi secara *half-duplex* pada jala-jala PLN 220 VAC, satu fasa dengan pengiriman data maksimal 1000 bps untuk jarak terjauh 85 meter. Dua modul *master*, Master1 dan Master2 yang terletak dalam jaringan sama dapat mengendalikan masing-masing *slavenya*.

DAFTAR PUSTAKA

- [1] Anonim, “Akses internet via kabel listrik PLN”, diakses dalam <http://by-1.blogspot.com/2010/01/akses-internet-via-kabel-listrik-pln.html> pada 8 April 2012.
- [2] ATMEL, “8-bit AVR Microcontroller with 8K Bytes In-System Programmable Flash”, Atmel Corporation, 2006.
- [3] Forouzan, Behrouz A., “Data Communications and Networking, Fourth Edition”, McGraw-Hill Companies Inc, New York, United States, 2007.
- [4] Iwan, “Komunikasi Via Jala-Jala Listrik”, diakses dalam <http://amatirindonesia.blogspot.com/2010/05/komunikasi-via-jala-jala-listrik.html> pada 4 April 2012.
- [5] Maxim, “+5V-Powered, Multichannel RS-232 Drivers/Receivers”, Maxim Integrated, 2010.
- [6] MAXIM, “DS18B20 Programmable Resolution 1-Wire Digital Thermometer”, Dallas Semiconductor-Maxim, 2008.
- [7] National Semiconductor, “LM1893 LM2893 Carrier-Current Transceiver”, National Semiconductor Corporation, 1995.

