



RANCANG BANGUN SISTEM SCADA BERBASIS RASPBERRY PI

Putra Arisandy
Department of Electrical Engineering
Faculty of Industrial Technology Universitas Islam Indonesia
Yogyakarta Indonesia
Email : putra.elco@gmail.com



UNIVERSITAS ISLAM INDONESIA

Abstrak

Sistem SCADA berbasis Raspberry Pi untuk menggantikan fungsi komputer cukup efektif untuk meminimalkan biaya pengadaan dan biaya operasional. Perancangan sistem SCADA berbasis *open source* juga dapat menggantikan fungsi *software* SCADA yang memiliki lisensi berbayar. Pada penelitian ini sistem SCADA pada Raspberry Pi digunakan untuk mengakuisi data dari tiga buah *slave* mikrokontroler dengan komunikasi RS485. Satu buah *slave* dihubungkan dengan PLC LG Master K200S dengan komunikasi RS232C untuk mengakuisi data proses produksi pada *plant* konveyor. Untuk melakukan komunikasi dengan seluruh *slave* menggunakan komunikasi RS485 dan RS232C, format protokol dibangun dengan metode *request* dan *response* agar dapat melakukan proses *reading/writing* menuju alamat *slave* yang tepat. Komunikasi RS485 antara Raspberry Pi dengan seluruh *slave* membutuhkan waktu sampling sebesar 333 ms dengan tingkat *error* data sebesar 22,66 % yang disebabkan oleh *noise*.

1. Pendahuluan

Penggunaan Raspberry Pi sebagai pengganti PC dapat memangkas biaya pengadaan dan operasional yang harus dikeluarkan. Selain itu sistem SCADA berbasis *open source* juga dapat menggantikan fungsi *software* SCADA yang memiliki lisensi berbayar.

Pada penelitian ini, Raspberry Pi digunakan sebagai *master* proses kontrol instalasi penerangan, monitoring suhu, dan monitoring proses produksi pada plan konveyor. Raspberry Pi akan dihubungkan dengan 3 *slave* mikrokontroler menggunakan komunikasi RS485. Salah satu *slave* akan terhubung dengan *Programmable Logic Controller* (PLC) melalui komunikasi RS232C.

Proses komunikasi antar kontroler membutuhkan sebuah protokol komunikasi agar proses pertukaran data dapat diterima sesuai dengan alamat yang dituju. Protokol komunikasi RS485 dirancang sendiri oleh peneliti. Sedangkan komunikasi RS232C dengan PLC menggunakan protokol yang sudah ditetapkan oleh *vendor* pembuat sesuai dengan *datasheet* PLC tersebut.

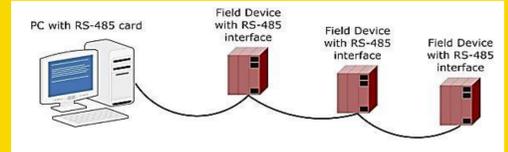
2. Teori

Komunikasi RS485 merupakan komunikasi yang menggunakan dua buah kabel untuk menghubungkan antar perangkat. RS485 banyak digunakan untuk menangani komunikasi jarak jauh mencapai 1220 meter tanpa menggunakan *repeater*. Kelebihan lain dari komunikasi RS485 adalah dapat menghubungkan lebih dari 32 perangkat komunikasi [1].

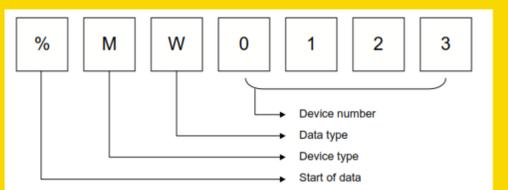
Komunikasi RS485 tidak mengatur standar protokol yang harus digunakan. Karena itu *user* bebas dalam menentukan protokol yang akan digunakan untuk komunikasi antar perangkat. Kebebasan dalam penentuan protokol ini juga membuat RS485 menjadi dasar utama terbentuknya komunikasi yang sering digunakan pada dunia industri seperti Profibus, Interbus, dll.

Master K200S tipe A dan C (K3P-07AS dan K3P-07CS) memiliki fitur pendukung untuk melakukan komunikasi dengan *external device* tanpa harus menggunakan modul Cnet I/F [2]. Meskipun tidak memiliki seluruh fungsi yang dapat dilakukan oleh modul Cnet I/F, fitur tersebut sangat berguna bagi *user* yang ingin melakukan pengembangan sistem yang murah berbasis jaringan RS232C.

Untuk melakukan komunikasi dengan *external device* menggunakan RS232C, terdapat standar protokol yang harus dipatuhi sesuai dengan *datasheet* PLC LG Master K200S. Kesalahan dalam mengaplikasikan protokol tersebut akan membuat proses penulisan dan pembacaan tidak akan dieksekusi oleh PLC.



Header (ENQ)	Station No	Instruction	Data	Tail (EOT)	Frame check (BCC)
--------------	------------	-------------	------	------------	-------------------

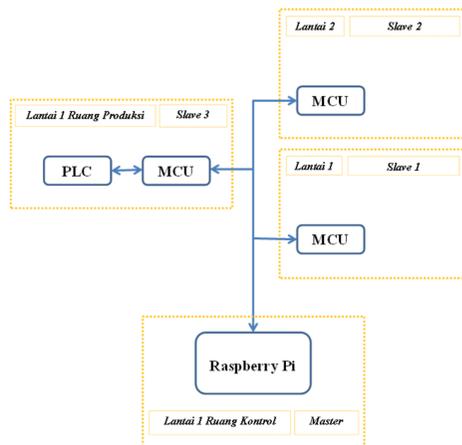


3. Desain Sistem

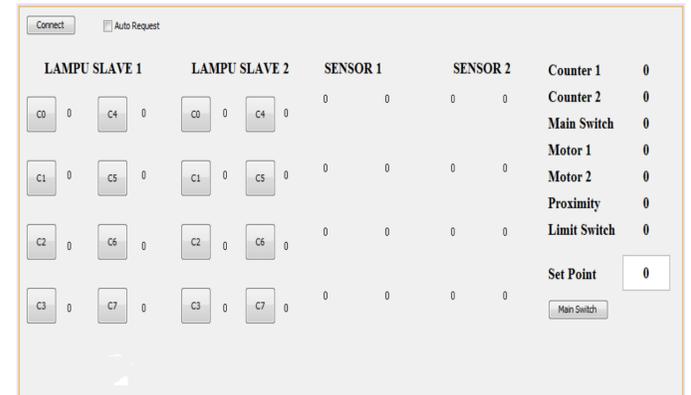
Pada sistem ini Raspberry Pi berperan sebagai *master* untuk memantau dan mengatur seluruh proses yang ada pada *plant* antara lain : mengontrol instalasi penerangan, *monitoring* suhu ruangan, *monitoring* proses produksi pada *plant* konveyor, dan mengontrol proses produksi pada *plant* konveyor. Raspberry Pi terhubung dengan dua mikrokontroler ATmega16 dan satu Arduino Mega menggunakan komunikasi RS485.

Mikrokontroler ATmega16 digunakan untuk memantau suhu pada 8 ruangan menggunakan sensor LM35 dan menghidup matikan lampu pada 8 ruangan. Arduino Mega berperan untuk menerima perintah dari *master* melalui komunikasi RS485 dan meneruskannya menuju PLC LG Master K200S menggunakan komunikasi RS232C. PLC LG Master K200S berperan untuk *monitoring* dan mengendalikan proses produksi pada *plant* konveyor.

Proses pembuatan GUI dibangun dengan JAVA SWING berbasis pemrograman JAVA JDK/JRE 8 menggunakan *software* Netbeans IDE 8.1. Tujuan utama digunakannya bahasa pemrograman JAVA adalah karena fleksibilititas, dimana pemrograman dapat dilakukan melalui komputer dan di-*run* langsung menggunakan Raspberry Pi tanpa harus melakukan *copy-paste source code*.



GUI (Graphical User Interface)



4. Hasil dan Analisa

Dari hasil pengujian yang dilakukan terhadap komunikasi RS485 antara *master* dengan seluruh *slave*, format data protokol RS485 yang dirancang oleh penulis dapat bekerja seperti yang diinginkan. Semua proses *writing/reading* data untuk *slave* tertentu juga berhasil dieksekusi sesuai dengan alamat *slave* yang dituju.

```
Request Slave 1 -> Sent : $01R#
Response Slave 1 -> Received : $01 293 295 363 353 342 335 329 347 0 1 0 0 0 1 0 0 #
Request Slave 2 -> Sent : $02R#
Response Slave 2 -> Received : $02 175 146 128 129 109 100 79 204 1 1 1 0 0 0 1 0 #
Request Slave 3 -> Sent : $03R#
Response Slave 3 -> Received : $03 1 0 0 1 1 0 0 0 #
Sent : $01R#
Received : $01 293 294 362 352 340 334 328 347 0 1 0 0 0 1 0 0 #
Sent : $02R#
Received : $02 175 146 129 130 110 100 79 204 1 1 1 0 0 0 1 0 #
Sent : $03R#
Received : $03 1 0 0 1 1 0 0 0 #
```

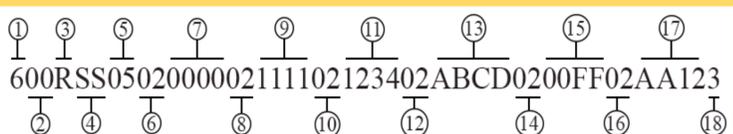
```
Data awal Slave 1 -> Received : $01 290 292 361 340 315 311 307 332 0 1 0 0 1 0 1 #
Sent : $02R#
Received : $02 182 154 140 163 116 94 88 202 1 1 0 0 1 0 1 1 #
Sent : $03R#
Mengubah nilai B0 pada Slave 1 -> Sent : $01WB00#
Sent : $01R#
Received : $01 288 289 359 337 312 308 305 331 0 1 0 0 1 0 1 #
Sent : $02R#
Received : $02 184 155 141 163 116 93 87 203 1 1 0 0 1 0 1 1 #
```

Sampling time komunikasi RS485 yang digunakan pada sistem ini adalah sebesar 333 ms dengan tingkat kerusakan data sebesar 22,66% yang disebabkan oleh *noise*. Nilai tersebut dipilih karena persentase rata - rata antara 333 ms, 350 ms, dan 400 ms tidak berbeda jauh. Selain itu dengan menggunakan 333 ms, total waktu yang digunakan untuk mengakses seluruh *slave* kurang dari 1 detik atau hanya sebesar 999 ms.

```
Sent : $01R#
Received : $01 293 294 362 353 342 335 329 347 0 0 0 0 0 0 0 0 #
Sent : $02R#
Received : $02 175 146 128 129 109 100 79 204 1 1 1 0 0 0 1 0 #
Sent : $03R#
Received : $03 0 0 0 0 0 0 0 0 #
```

Kerusakan data akibat Noise

Jumlah Request Data	Sampling Time (ms)	Jumlah Error Pada Percobaan			Rata - Rata	% Rata - Rata
		1	2	3		
300	100	140	168	152	152	50,66 %
	200	98	98	118	98	32,66 %
	300	95	83	76	83	27,66 %
	333	82	61	68	68	22,66 %
	350	59	73	66	66	22 %
	400	55	70	67	67	22,33 %



5. KESIMPULAN

1. Format protokol komunikasi RS485 untuk menangani komunikasi lebih dari 2 perangkat dirancang dengan format *header*, alamat perangkat, data, dan *tail*.
2. Format protokol komunikasi RS232C LG Master K200S sesuai dengan format yang telah dituliskan di *datasheet*.
3. *Sampling time* terbaik yang dibutuhkan Raspberry Pi untuk mengakses seluruh *slave* adalah sebesar 333 ms per *slave*.
4. Kerusakan data komunikasi RS485 disebabkan oleh *noise* dan *sampling time* yang terlalu cepat.

6. Referensi

[1] BB SmartWorx, "A Practical Guide to Using RS-422 and RS-485 Serial Interfaces," RS-422 and RS-485 Applications Ebook, Oct. 2010. [Online] Available : <http://www.bb-elec.com/Learning-Center/All-White-Papers/Serial/RS-422-and-RS-485-Applications-eBook.aspx>.

[2] LG Industrial Systems, "User's Manual LG Programmable Logic Controller Master-K," Datasheet, Accessed on Feb. 08, 2018. [Online] Available: http://www.ehaegypt.com/uploads/K200_300_1000S_yi2jwqxz.pdf