

Sistem Monitoring Keseimbangan Beban 3 – Phasa Berbasis Arduino Uno

Fanny J. Doringin¹, Stephy B. Walukow², Marson J. Budiman³

¹Jurusan Teknik Elektro Politeknik Negeri Manado, Manado 95252

E-mail : fannydoringin67@gmail.com

²Jurusan Teknik Elektro Politeknik Negeri Manado, Manado 95252

E-mail : stephiewalukow@gmail.com

³Jurusan Teknik Elektro Politeknik Negeri Manado, Manado 95252

E-mail : marsonbudi@gmail.com

ABSTRAK

Untuk mendapatkan hasil pendistribusian beban listrik yang optimal, maka keseimbangan beban antar phasa pada sistem instalasi 3-fasa haruslah menjadi salah satu syarat yang mutlak, karena jika terjadi ketidakseimbangan beban, maka akan berdampak pada rugi-rugi daya listrik. Salah satu hal yang dapat menyebabkan terjadinya ketidakseimbangan beban, adalah dalam hal penginstalasian peralatan listrik pada masing-masing phasa yang tidak sesuai dengan rancangan instalasi yang ada.

Penelitian ini bertujuan untuk membuat suatu sistem yang dapat memonitoring dan menginformasikan keadaan beban yang terpasang pada sistem instalasi 3-fasa, sehingga dapat memudahkan dalam proses penambahan dan pembagian beban pada masing-masing phasa, sehingga dapat menghindari terjadinya ketidakseimbangan beban.

Metode yang digunakan dalam penelitian ini adalah metode prototyping, yang diawali dari tahapan perancangan, pembuatan dan pengujian sistem secara simulasi.

Dari hasil pengujian yang dilakukan, maka didapat hasil bahwa sistem yang dibuat dapat mendeteksi dan menginformasikan saat terjadi ketidakseimbangan beban, dimana saat beban terdeteksi pada phasa R melalui sensor arus sebesar 1 ampere, pada phasa S terdeteksi sebesar 2 ampere dan pada phasa T terdeteksi sebesar 3 ampere, maka sistem akan menginformasikan melalui bunyi *buzzer* sebagai indikator alarm, serta menampilkan informasi melalui tampilan *lcd* bahwa terjadi ketidakseimbangan beban, beserta nilai arus yang terdeteksi di setiap phasa.

Kata Kunci

Prototyping, sistem monitoring, Keseimbangan Beban, 3-Phasa

1. PENDAHULUAN

Salah satu penyebab terjadinya ketidakseimbangan beban pada sistem instalasi 3 phasa, dapat menyebabkan rugi-rugi daya listrik. Ketidakseimbangan beban, diakibatkan karena pengoperasian beban yang terhubung ke masing-masing phasa tidak dapat diatur untuk beroperasi secara serempak, karena pengoperasian beban disesuaikan dengan kebutuhan dari peralatan listrik yang terpasang pada masing-masing phasa. Hal lainnya yang dapat menyebabkan terjadinya ketidakseimbangan beban, adalah dalam hal penginstalasian peralatan listrik pada masing-masing phasa yang tidak sesuai dengan rancangan instalasi yang ada.

Untuk meminimalisir terjadinya rugi-rugi daya listrik pada sistem instalasi 3 phasa akibat dari ketidakseimbangan beban, maka dibuat suatu sistem yang dapat mendeteksi terjadinya ketidakseimbangan beban antar phasa pada sistem instalasi 3 phasa, dan menginformasikannya melalui tampilan *lcd* dan bunyi *buzzer* saat terindikasi terjadi ketidakseimbangan beban antar phasa.

Dalam menghasilkan sistem pendeteksi dan penginformasi ketidakseimbangan beban antar phasa pada sistem instalasi 3 phasa, maka metode yang digunakan adalah metode prototyping, dimana tahapan-tahapannya diawali dari studi pustaka yang bertujuan untuk mendapatkan referensi yang

sesuai dengan penelitian ini. Tahapan selanjutnya adalah tahapan perancangan sistem, dimana pada tahapan ini dilakukan proses perancangan dan pembuatan perangkat keras dan perangkat lunak untuk kebutuhan kerja sistem. Untuk mendapatkan data sehubungan dengan kerja dari sistem yang dibuat, maka dilakukan proses pengujian sistem melalui program simulasi proteus.

Sistem yang terpadu dalam pembuatan simulasi sistem pendeteksi dan penginformasi ketidakseimbangan beban pada sistem instalasi 3 phasa, menggunakan kontroler *Arduino Uno* yang berfungsi sebagai pengatur kerja keseluruhan sistem [1][2], sensor arus *acs712* yang berfungsi sebagai pendeteksi perubahan arus beban, dimana keluaran dari sensor arus ini berupa tegangan, dan untuk mendapatkan nilai tegangan keluaran, maka persamaan yang digunakan adalah:

$$V_{Output} = 2.5 \pm (0.185 \times I) \quad (1)$$

Dimana *I* adalah besarnya arus yang terdeteksi oleh sensor arus [3][4][5][6], Algoritma program (*flowchart*) yang berfungsi untuk langkah-langkah yang akan dilakukan dalam pembuatan program dan pemodelan sistem [7][8].

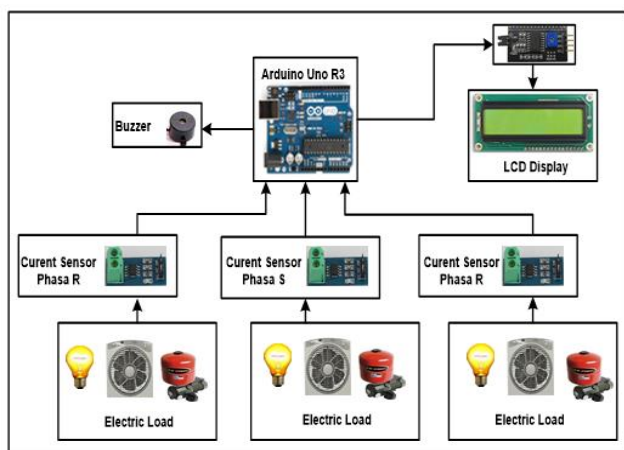
Untuk menjalankan sistem, dibutuhkan program yang ditanamkan kedalam mikrokontroler *Arduino Uno*, dimana perangkat lunak yang digunakan untuk pembuatan program adalah *Arduino IDE*, dengan mengacu pada diagram alir (*flowchart*) yang dibuat [9].

2. PEMBAHASAN

Metode yang digunakan dalam proses menghasilkan sistem monitoring keseimbangan beban 3-fasa adalah metode prototyping. Sesuai dengan tahapan untuk menghasilkan sistem monitoring keseimbangan beban 3-fasa, maka dilakukan proses prancangan sistem yang meliputi perancangan model sistem sebagai gambaran dari sistem yang akan dibuat yang dibuat dalam bentuk blok diagram sistem yang bertujuan untuk mendeskripsikan hubungan antar komponen kontrol yang saling terintegrasi sehingga membentuk suatu kesatuan sistem, perancangan perangkat lunak dalam bentuk diagram alir yang bertujuan untuk mendeskripsikan alur kerja program yang dibutuhkan sistem dalam proses memonitoring dan penginformasi keseimbangan beban, dan pembuatan sistem melalui program simulasi *proteus* berdasarkan hasil perancangan perangkat keras, serta pembuatan perangkat lunak berdasarkan algoritma program yang dihasilkan pada bagian perancangan perangkat lunak.

2.1. Perancangan sistem

Perancangan perangkat keras dibuat dalam bentuk blok diagram yang menggambarkan hubungan antara *input/output* dengan kontroler *arduino uno* guna proses penyeimbangan beban antar fasa. Perncangan perangkat keras berupa blok giagram diperlihatkan pada Gambar 1,



Gambar 1. Blok Diagram Sistem

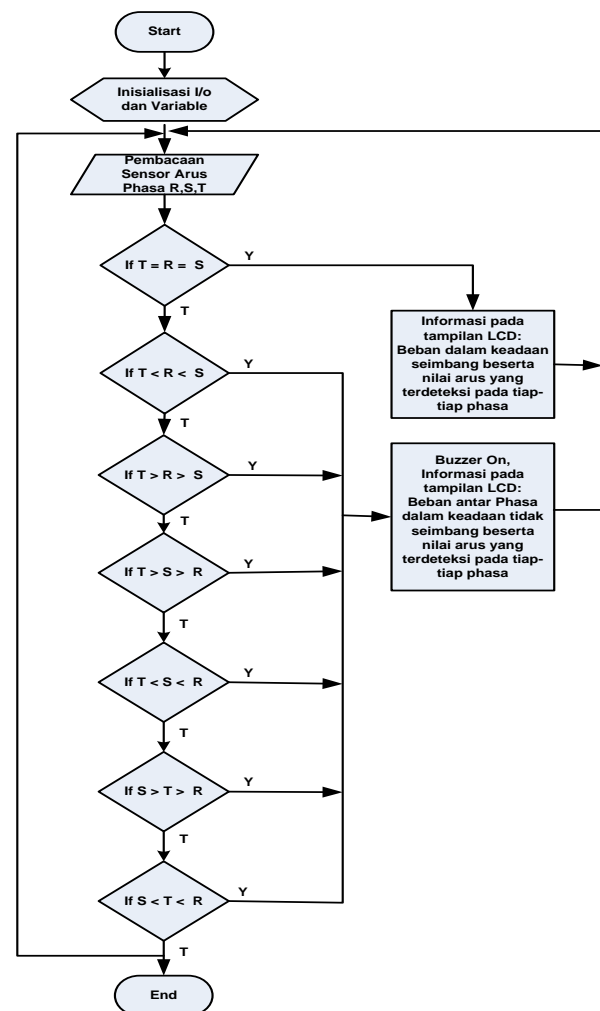
Adapun deskripsi blok diagram sistem pada Gambar 1 sebagai berikut:

1. Kontroler, berfungsi sebagai pengontrol kerja dari keseluruhan sistem, untuk proses pendeteksi dan penginformasi kondisi beban pada sistem instalasi 3 fasa.

2. Sensor arus *acs712*, berfungsi untuk mendeteksi perubahan arus dari masing-masing fasa dari ketiga fasa R, S dan fasa T, dimana data dari ketiga sensor arus tersebut diinputkan ke kontroler untuk diolah datanya.
3. *Buzzer* berfungsi sebagai media penginformasi berupa suara, saat terindikasi terjadi ketidakseimbangan beban antar fasa.
4. *Lcd display* berfungsi sebagai media penginformasi kondisi beban pada sistem instalasi 3 fasa.

2.2. Flow Chart Sistem

Perancangan perangkat lunak berupa algoritma program bertujuan untuk mendeskripsikan langkah-langkah atau alur kerja program yang akan dibuat untuk kebutuhan kerja sistem. Algoritma program diperlihatkan pada Gambar 2.



Gambar 2. Algoritma Sistem

Adapun deskripsi Algoritma sistem pada Gambar 2 sebagai berikut:

1. Inisialisasi *I/O* dan variabel merupakan bagian program untuk mendeklarasikan variabel-variabel *input/output* dan variabel-variabel yang nantinya akan dipergunakan

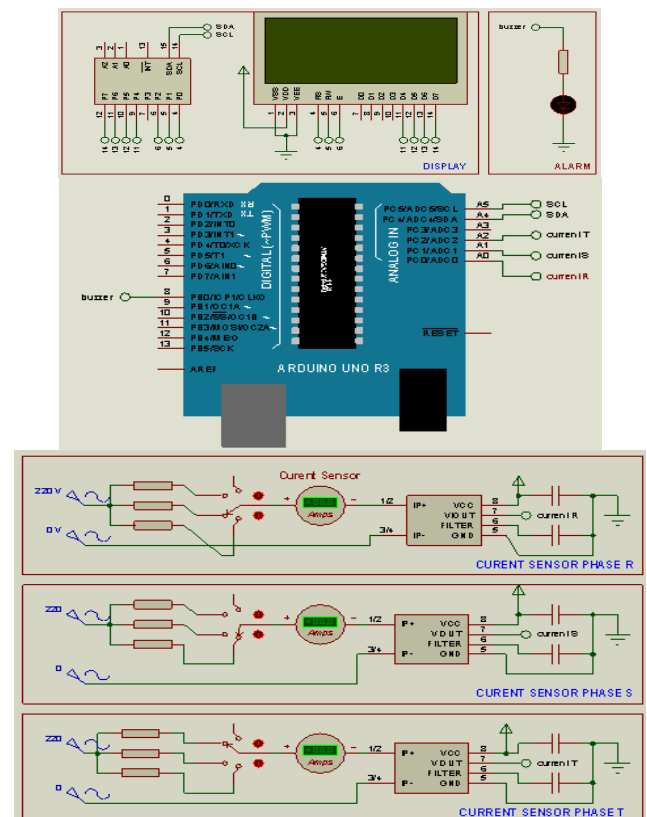
dalam program, baik sebagai penyimpan data ataupun untuk keperluan pengolahan data.

2. Membaca data sensor arus untuk fase R, S dan Fase T
3. Langkah selanjutnya adalah membandingkan besarnya arus untuk fase R, fase S dan Fase T. jika besarnya arus pada fase R sama dengan besarnya arus pada fase S dan fase T, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan seimbang dan besarnya beban yang terdeteksi pada masing-masing fase.
4. Jika besarnya arus pada fase R lebih besar dari besarnya arus pada fase T, dan besarnya arus pada fase R lebih kecil dari besarnya arus pada fase S, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing fase, serta sistem akan mengaktifkan *buzzer* sebagai alarm.
5. Jika besarnya arus pada fase R lebih kecil dari besarnya arus pada fase T, dan besarnya arus pada fase R lebih besar dari besarnya arus pada fase S, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing fase, serta sistem akan mengaktifkan *buzzer* sebagai alarm.
6. Jika besarnya arus pada fase S lebih kecil dari besarnya arus pada fase T, dan besarnya arus pada fase S lebih besar dari besarnya arus pada fase R, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing fase, serta sistem akan mengaktifkan *buzzer* sebagai alarm.
7. Jika besarnya arus pada fase S lebih besar dari besarnya arus pada fase T, dan besarnya arus pada fase S lebih kecil dari besarnya arus pada fase R, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing fase, serta sistem akan mengaktifkan *buzzer* sebagai alarm.
8. Jika besarnya arus pada fase T lebih kecil dari besarnya arus pada fase S, dan besarnya arus pada fase T lebih besar dari besarnya arus pada fase R, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing fase, serta sistem akan mengaktifkan *buzzer* sebagai alarm.
9. Jika besarnya arus pada fase T lebih besar dari besarnya arus pada fase S, dan besarnya arus pada fase T lebih kecil dari besarnya arus pada fase R, maka sistem akan menginformasikan melalui tampilan lcd bahwa beban dalam keadaan tidak seimbang dan besarnya beban yang terdeteksi pada masing-masing

fasa, serta sistem akan mengaktifkan *buzzer* sebagai alarm.

2.3. Pembuatan Simulasi Sistem

Simulasi sistem dibuat dengan menggunakan program simulasi *Proteus*, dengan mengatur komponen-komponen penunjang sistem kontrol sesuai dengan fungsi dan kebutuhannya dalam sistem dan menghubungkannya dengan kontroler *arduino uno*, sehingga membentuk suatu kesatuan sistem kontrol yang terintegrasi. Pembuatan simulasi sistem kontrol diperlihatkan pada Gambar 3.



Gambar 3. Simulasi sistem

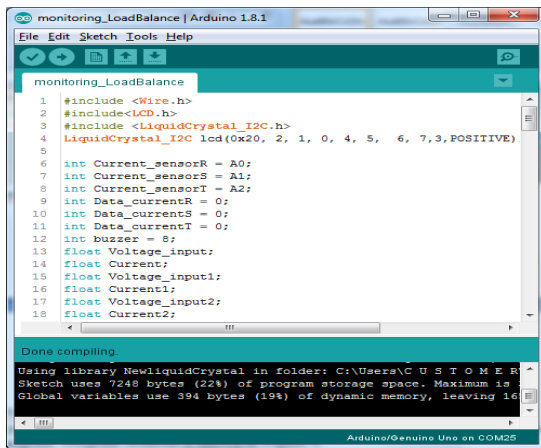
Keterangan Gambar 3:

Mikrokontroler Arduino uno digunakan sebagai media kontroler untuk mengolah data yang diinputkan melalui 3 buah sensor arus, dimana ketiga buah sensor arus tersebut merepresentasikan arus yang mengalir pada fase R, S dan T. Masing-masing sensor arus terhubung dengan beban yang direpresentasikan dengan tahanan (Resistor) yang masing-masing nilai resistansinya sebesar 220 Ohm untuk mewakili arus beban 1 Ampere; 110 Ohm untuk mewakili arus beban 2 Ampere dan 73,33 yang mewakili arus beban 3 Ampere. Setelah data yang diterima dari sensor arus diolah oleh kontroler, maka data tersebut akan dioutputkan melalui tampilan *LCD* sebagai informasi nilai arus yang terdeteksi pada masing-masing fase, baik arus yang mengalir pada fase R, fase S dan fase T. Jika terdeteksi terjadi ketidakseimbangan beban, maka *Buzzer* akan

berbunyi yang ditandai dengan menyalaunya indikator LED.

2.4. Pembuatan Perangkat Lunak

Pembuatan perangkat lunak dibuat melalui perangkat lunak yang telah disediakan oleh Arduino yaitu *Arduino IDE*, dimana pembuatan perangkat lunak dilakukan dengan mengacu pada hasil rancangan perangkat lunak berupa diagram alir yang telah dibuat. Proses pembuatan perangkat lunak iperlihatkan pada Gambar 4.



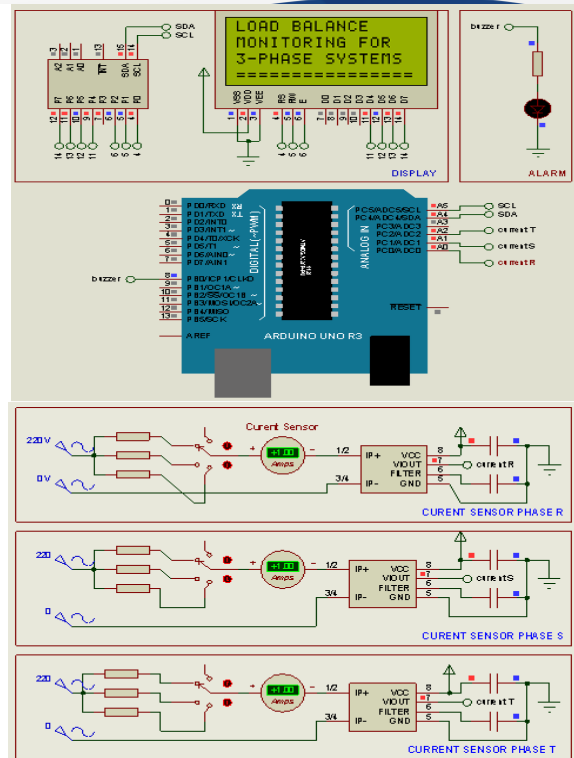
Gambar 4. Proses pembuatan perangkat lunak

3. PENGUJIAN SISTEM

Pengujian sistem dilakukan melalui program simulasi Proteus, dimana pengujian sistem dilakukan untuk mengetahui apakah sistem yang dibuat telah bekerja sesuai dengan hasil rancangan. Adapun proses pengujian sistem meliputi:

3.1. Pengujian Sistem Saat Pertama Kali Diaktifkan

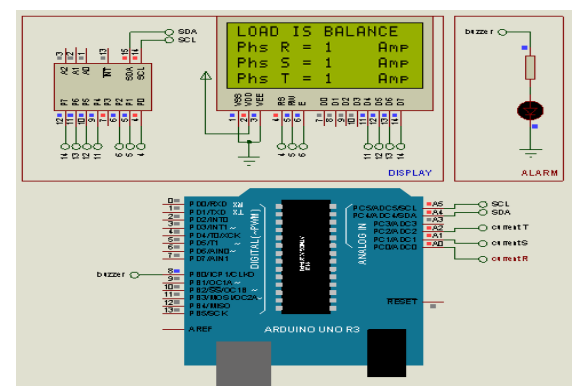
Saat sistem pertama kali diaktifkan, maka sistem akan menampilkan informasi sehubungan dengan fungsi dari sistem tersebut, dimana informasinya ditampilkan melalui tampilan LCD berupa Sistem Monitoring Keseimbangan Beban 3-Phasa. Tampilan awal saat sistem pertama kali diaktifkan diperlihatkan pada Gambar 5.

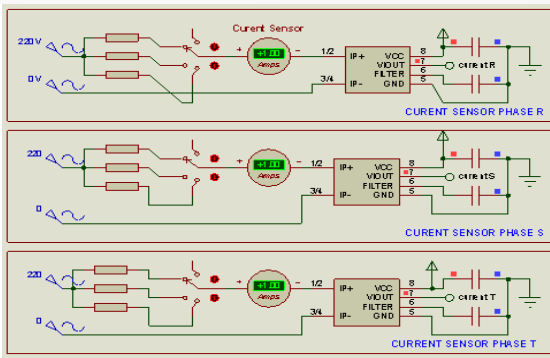


Gambar 5. Sistem pertama kali diaktifkan

3.2. Pengujian Sistem Dalam Keadaan Seimbang

Setelah proses setup dijalankan selama 2 detik untuk menampilkan informasi sehubungan dengan fungsi sistem, maka sistem akan masuk ke bagian pembacaan nilai tegangan yang diinputkan oleh sensor arus. Pembacaan nilai tegangan melalui sensor arus, akan di olah oleh kontroler menjadi nilai arus sesuai dengan besaran arus yang terdeteksi oleh sensor arus. Proses pengujian dilakukan dengan cara mengatur posisi dari ke tiga salkar yang mewakili fasa R, S dan T pada bagian output dari resistor 220 Ohm, dimana nilai arusnya sebesar 1 Ampere. Adapun pengujian sistem dalam keadaan seimbang diperlihatkan pada Gambar 6.





Gambar 6. Pengujian sistem dalam keadaan seimbang

Dari Gambar 6 terlihat bahwa saat sistem mendeteksi besarnya arus yang pada masing-masing fase (fase R, S dan T) sama besar, maka sistem akan menginformasikan bahwa beban dalam keadaan seimbang, dimana besarnya arus yang terdeteksi adalah sebesar 1 ampere di tiap-tiap fase.

Untuk mengetahui besarnya nilai tegangan yang masuk rangkaian ADC pada kontroler, maka nilai tegangan tersebut harus dikonversi ke dalam data 16 bit, seperti contoh, pada proses pembacaan nilai input tegangan dari sensor arus *acs712*, dimana untuk arus beban yang terdeteksi sebesar 2 ampere pada fase R, maka nilai tegangan yang diinputkan ke kontroler *Arduino Uno* melalui jalur *analog to digital converter (ADC)* yang berfungsi untuk merubah besaran analog ke dalam bentuk data digital, akan diolah oleh kontroler menjadi nilai tegangan, dimana prosesnya sebagai berikut:

$$\begin{aligned} \text{Tegangan_input} &= ((\text{Data input_arusR} * \text{Tegangan} \\ &\quad \text{referensi}) / \text{data 16 bit}) \\ &= ((594 * 4.98) / 1024) \\ &= 2.89 \text{ Volt} \end{aligned}$$

Setelah nilai tegangan diperoleh, maka sistem akan mengubah nilai tegangan yang diperoleh ke nilai arus, melalui proses sebagai berikut:

$$\begin{aligned} \text{Nilai Arus} &= ((\text{Tegangan_input} - 2.5) / 0.186) \\ &= ((2.89 - 2.5) / 0.186) \\ &= 2.09 \text{ Ampere} \end{aligned}$$

Untuk lebih jelasnya proses mengkonversi nilai tegangan dan arus dari hasil pembacaan sensor arus, diperlihatkan pada Tabel 1.

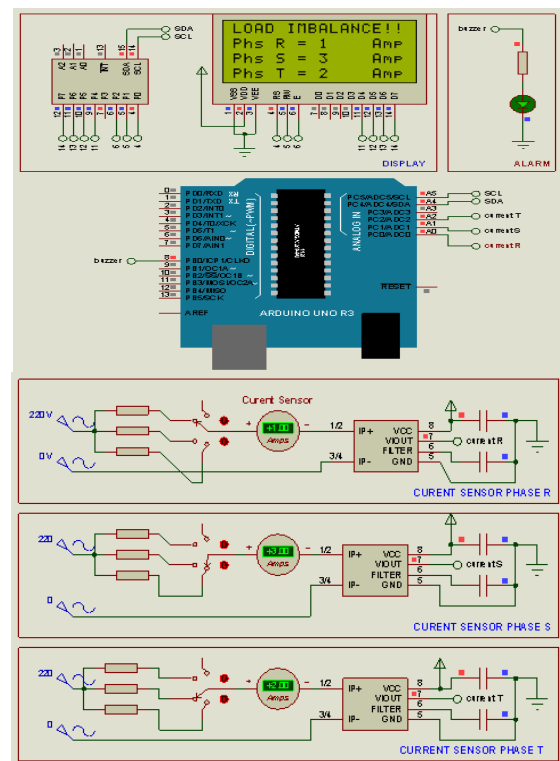
Tabel 1. Nilai konversi pembacaan arus ke tegangan

Arus beban (Ampere)	Data pembacaan sensor arus	Nilai konversi tegangan (Volt)	Nilai konversi arus (Ampere)
1	553	2.69	1.02
2	594	2.89	2.09
3	635	3.08	3.11

3.3. Pengujian Sistem Saat Terdeteksi Tidak Seimbang

Pengujian dilakukan dengan cara mengatur beban pada fase R sebesar 1 ampere, fase S sebesar 3 ampere dan

fase T sebesar 2 ampere. Pengujian sistem saat terdeteksi terjadinya ketidakseimbangan beban diperlihatkan pada Gambar 7.



Gambar 7. Pengujian sistem saat terdeteksi terjadinya ketidakseimbangan beban

Hasil pengujian pada Gambar 7 memperlihatkan bahwa saat terindikasi bahwa beban yang terpasang pada sistem instalasi 3-fase dalam keadaan tidak seimbang, dimana terdeteksi besarnya arus pada fase R sebesar 1 ampere, pada fase S sebesar 3 ampere dan pada fase T sebesar 2 ampere, maka sistem secara otomatis akan mengaktifkan *Buzzer* yang berfungsi sebagai penginformasi bahwa sistem dalam keadaan tidak seimbang. Disamping itu juga sistem akan menginformasikan besarnya arus yang terdeteksi pada masing-masing fase (fase R, S dan T), dimana data pengukuran dan monitoring besarnya arus beban masing-masing fase (fase R, S dan T) diperlihatkan pada Tabel 2.

Tabel 2. Pengukuran dan monitoring besarnya arus beban masing-masing fase (fase R, S dan T)

Arus beban Antar Pahasa (Ampere)	Nilai Konversi Data pembacaan arus Antar Pahasa			Hasil Monitoring
	R	S	T	
2 2 2	594	594	594	Beban Seimbang
1 2 3	553	594	635	Bebn Tidak Seimbang
2 3 1	594	635	553	Bebn Tidak Seimbang
3 1 2	635	553	594	Bebn Tidak Seimbang

4. KESIMPULAN

Dari hasil penelitian yang dilakukan maka dapat disimpulkan bahwa sistem monitoring keseimbangan beban 3-fasa dapat mendeteksi dan menginformasikan besarnya arus beban yang terpasang pada masing-masing fasa. Saat sistem terdeteksi dalam keadaan seimbang, dimana besarnya arus yang terdeteksi pada fasa R,S dan T sama besar yaitu sebesar 2 ampere, maka sistem akan menginformasikan melalui tampilan *LCD* bahwa sistem dalam keadaan seimbang. Jika sistem mendeteksi terjadinya ketidakseimbangan beban antar fasa, dimana besarnya arus yang terdeteksi pada fasa R = 1 ampere, fasa S = 3 ampere dan pada fasa T = 2 ampere, maka secara otomatis sistem akan mengaktifkan *Buzzer* sebagai indikasi terjadinya ketidakseimbangan beban, serta informasi besarnya arus yang terdeteksi pada masing-masing fasa melalui tampilan *LCD*.

UCAPAN TERIMA KASIH

Penulis mengucapkan banyak terima kasih kepada DRPM Ristek DIKTI dan Politeknik Negeri Manado yang telah memfasilitasi penelitian ini, serta Politeknik Negeri Bandung yang telah menyelenggarakan seminar IRWNS sebagai sarana berbagi dan bertukar pikiran demi penyempurnaan penelitian ini.

DAFTAR PUSTAKA

- [1] A. G. Smith, "Introduction To Arduino", 2011.
- [2] S. MT, "*Teori dan Aplikasi Mikrokontroler*", Cetakan Pertama, Graha Ilmu, Yogyakarta, 2005.
- [3] Allegro mikrosistem.inc. "*Datasheet ACS712*".
- [4] Husnawati, R. Passarella, Sutarno dan Rendyansyah, "*Perancangan dan Simulasi Energi Meter Digital Satu Fasa Menggunakan Sensor Arus ACS712*", *JNETI* Vol. 2. No. 4, November 2013.
- [5] A. Fitriandi, E. Komalasari, H. Gusmedi, "*Rancang Bangun Alat Monitoring Arus dan Tegangan Berbasis Mikrokontroler dengan SMS Gateway*", *Jurnal Rekayasa dan Teknologi Elektro*, Vol. 10, No. 2, 2016.
- [6] I. Ahmad, A. Ramschie. "*Sistem Monitoring Dan Kendali Kerja Air Conditioning Berbasis Mikrokontroler ATmega 8535.*", *Jurnal Ristek* Vol.2, No.1, Juni 2013.
- [7] A. Ramschie, J. Makal, V. Ponggawa, "Method of Freon Leak Detection and Dirty Air Filter in Air Conditioning for Electrical Savings", *International Journal of Computer Applications*, Vol. 172, No.1, 2017
- [8] A. Ramschie, J. Makal, V. Ponggawa, "Algorithms Air Conditioning Air Filter Detection System For Electric Energy Savings", *International Journal of Computer Application*, Vol. 156, No. 8, 2016.
- [9] S. Muhammad, "*Panduan Mudah Simulasi Dan Praktek Mikrokontroler Arduino*". Andi, 2013.