

Purwa-rupa Sistem Monitoring Suhu, Kelembapan, Arus, Tegangan dan Jumlah Produksi Melalui Socket Programming

Prototype of Monitoring System for Temperature, Humidity, Current, Voltage and Total Production via Socket Programming

Heri Andrianto¹, Hery Andrian²

^{1, 2}Teknik Elektro, Universitas Kristen Maranatha

Jl. Prof. drg. Surya Sumantri, M.P.H. No. 65, Bandung

heri.andrianto@eng.maranatha.edu^{1*}, heryandrian.engineer@gmail.com²

Abstrak – Pada umumnya, pabrik memiliki banyak peralatan yang perlu diawasi. Peralatan-peralatan tersebut tersebar di beberapa lokasi yang berjauhan sehingga diperlukan suatu sistem monitoring untuk mengawasi peralatan-peralatan tersebut. Pada makalah ini, telah dikembangkan purwa-rupa sistem monitoring berbasis Arduino Mega2560 dan Ethernet Shield untuk monitoring kelembapan udara, suhu udara, arus listrik, tegangan listrik dan jumlah produksi barang. Monitoring dilakukan melalui aplikasi Delphi sebagai antarmuka berbasis desktop. Komunikasi data antara komputer dengan perangkat menggunakan socket programming. Dari hasil pengujian pengukuran kelembapan udara, suhu udara, arus listrik, tegangan listrik dan jumlah produksi barang terdapat error pengukuran kelembapan udara 10,2 %, error pengukuran suhu 6,87%, error pengukuran arus 5,88 %, error pengukuran tegangan 2,21 %, dan error perhitungan jumlah produksi barang 0%. Komunikasi data pada sistem monitoring berbasis Arduino Mega2560, Ethernet Shield dan aplikasi delphi memiliki tingkat keberhasilan sebesar 100%.

Kata Kunci: arduino mega2560, ethernet shield, socket programming, sistem monitoring.

Abstract – In generally, factories have a lot of equipment that needs to be supervised. These equipments are scattered in several locations that are far apart, therefore a monitoring system is needed to supervise these equipments. In this paper we have developed a prototype of monitoring system based on Arduino Mega2560 and Ethernet Shield to monitor air humidity, air temperature, electric current, electric voltage, and the amount of goods produced. Monitoring is done through the Delphi application as a desktop-based interface. Data communication between computers and devices uses the socket programming. From the test results for measuring air humidity, air temperature, electric current, electric voltage, and the amount of production of goods, there is an error measuring air humidity 10.2%, error measuring temperature 6.87%, error measuring current 5.88%, error measuring voltage 2.21%, and errors in calculating the amount of production of goods 0%. Data communication on monitoring systems based on Arduino Mega2560, Ethernet Shield and the Delphi application has a success rate of 100%.

Keywords: arduino mega2560, ethernet shield, socket programming, monitoring system.

1. Pendahuluan

Sebuah pabrik biasanya memiliki banyak peralatan produksi yang tersebar di beberapa lokasi. Kondisi lingkungan pabrik seperti suhu dan kelembapan perlu diawasi supaya peralatan beroperasi normal dan mencegah hal-hal yang tidak diinginkan. Energi listrik yang digunakan oleh peralatan-peralatan listrik juga perlu diawasi untuk keperluan melakukan efisiensi penggunaan energi listrik [1][2][3][4][5], serta untuk keperluan mendeteksi gangguan pada sistem kelistrikan [6]. Selain itu jumlah produksi barang juga perlu diawasi untuk kebutuhan optimalisasi peralatan produksi barang supaya menghasilkan jumlah produksi barang yang maksimal. Pada Pabrik yang luas, seringkali jarak antara satu peralatan dengan peralatan lainnya saling berjauhan, sehingga akan membutuhkan biaya yang besar jika membuat jalur komunikasi baru. Oleh karena itu dibutuhkan sistem *monitoring* yang tepat dan efisien untuk mengawasi peralatan-peralatan tersebut. Sistem *monitoring* yang berjalan pada jaringan komputer yang sudah ada di pabrik menjadi solusi yang paling tepat untuk jalur komunikasi data antara komputer *server* dengan *client* (*controller*). *Socket programming* dapat menjadi solusi untuk komunikasi data antara komputer *server* dan perangkat. Parameter-parameter kondisi lingkungan dan perangkat seperti kelembapan udara, suhu udara, arus listrik, tegangan listrik dan jumlah produksi barang dapat dikirimkan dari perangkat ke komputer *server* melalui jaringan komputer yang sudah ada menggunakan *socket programming*. Data yang diterima oleh komputer server kemudian ditampilkan pada aplikasi Delphi yang berfungsi sebagai antarmuka dengan pengguna.

Socket programming dikenal sebagai pemrograman komunikasi data melalui jaringan komputer antara dua aplikasi *endpoints* (*client-server*). *Socket programming* memungkinkan komunikasi antar aplikasi melalui jaringan berbasis protokol TCP/IP. Keunggulan dari *socket programming* yaitu walaupun antar aplikasi memiliki *platform* berbeda namun dapat saling berkomunikasi asalkan aplikasi-aplikasi tersebut menggunakan protokol transfer yang sama, keunggulan lainnya yaitu dapat menangani banyak *client* sekaligus. Secara umum komunikasi menggunakan *socket* terdiri dari dua macam, yaitu komunikasi *stream* dan komunikasi *datagram*. Komunikasi *stream* dikenal sebagai *connection oriented* sedangkan komunikasi *datagram* dikenal sebagai *connectionless*. Protokol standar untuk komunikasi *stream* yaitu *Transmission Control Protocol* (TCP), sedangkan komunikasi *datagram* yaitu *User Datagram Protocol* (UDP) [7][8][9]. Model aplikasi yang sering dibuat pada komunikasi *socket* adalah aplikasi *client-server* [10] [11].

Beberapa penelitian mengenai penggunaan *socket programming* sudah dilakukan oleh peneliti sebelumnya [12][13][14][15]. Ayurzana dkk mengembangkan sistem pengukuran air jarak jauh otomatis. Sistem ini mengirimkan data penggunaan air terukur jarak jauh secara otomatis dari pemancar dengan modem CDMA melalui jaringan SK-Telecom. Data penggunaan air diterima melalui *local area network* (LAN) TCP/IP dan ditampilkan di browser internet. Pada sistem ini modul *radio frequency* (RF) telah terpasang pada masing-masing meteran air. Program *Client/Server* dikembangkan menggunakan *socket programming* [12]. Jitbanyud dkk mengembangkan sistem kelas laboratorium komputer yang *powerful* dengan mensimulasikan prosedur pembelajaran yang ideal, guru dan siswa dapat tetap berhubungan satu sama lain dalam komunikasi 2 arah dan langsung di internet. *Socket programming* digunakan untuk membuat interaksi antara setiap komputer [13]. Maata dkk mengembangkan aplikasi *client-server* menggunakan *socket programming* di lingkungan komputasi terdistribusi yang bertujuan untuk mendemonstrasikan prinsip dan konsep di balik *socket programming* serta *library* yang tersedia di Java [14]. Ramadhan dkk mengembangkan sistem *monitoring* daya listrik menggunakan sensor YHDC SCT-013-000 dan NodeMCU yang dapat dipantau melalui web. Data dikirim dari perangkat *monitoring* ke *database server* menggunakan *protocol* *Websocket* [15].

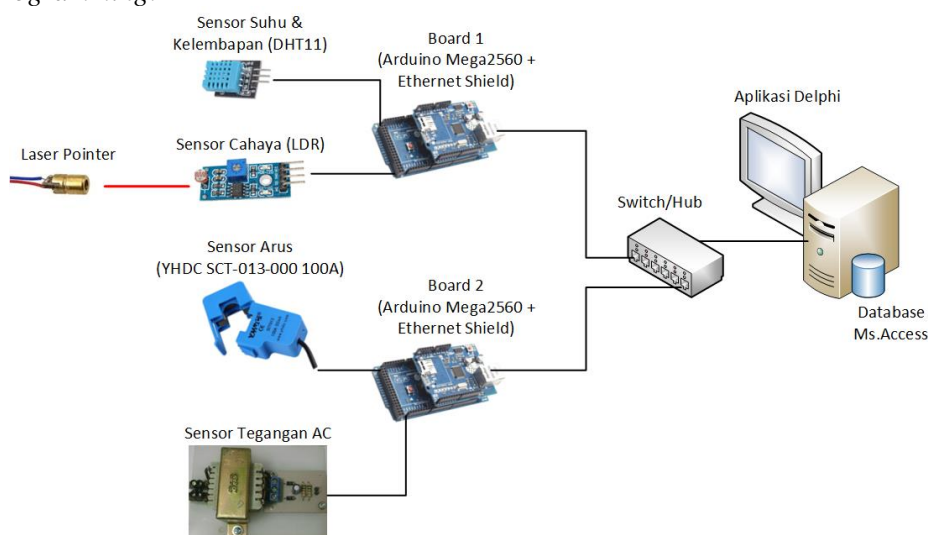
Penelitian-penelitian sebelumnya tentang *socket programming* belum ada yang diaplikasikan untuk keperluan industri (pabrik) seperti untuk pemantauan kelembapan udara, suhu udara, arus listrik, tegangan listrik, dan jumlah produksi barang. Pada makalah ini telah direalisasikan purwa-rupa sistem *monitoring berbasis Arduino Mega2560* melalui jaringan komputer menggunakan *socket programming* untuk pemantauan kelembapan udara, suhu udara, arus listrik, tegangan listrik, dan jumlah produksi barang. Data yang dikumpulkan dari perangkat-

perangkat berupa data kelembapan udara, suhu udara, arus listrik, tegangan listrik, dan jumlah produksi barang. Data dari perangkat-perangkat dikirimkan ke komputer *server* melalui jaringan komputer.

2. Metode Penelitian

Pada penelitian ini telah diimplementasikan purwa-rupa (*prototyping*) sistem *monitoring* suhu, kelembapan, arus, tegangan dan jumlah produksi melalui *socket programming*. Asumsi penempatan perangkat *monitoring* dalam kondisi riil yaitu perangkat *monitoring* dipasang pada jaringan komputer yang sudah tersedia di pabrik. Sistem *monitoring* ini masih dapat diterapkan di dunia nyata khususnya untuk jarak jauh (jarak antara perangkat *monitoring* dengan komputer server) karena menggunakan jaringan komputer (*Wire*). Jumlah perangkat *monitoring* yang dipasang pada jaringan komputer bisa banyak, tergantung kebutuhan. Masing-masing perangkat *monitoring* terhubung dengan beberapa sensor. Diasumsikan penempatan suatu perangkat *monitoring* dan sensor pada kondisi riil diusahakan sedekat mungkin sehingga tidak terjadi *drop* tegangan. Perangkat *monitoring* dan sensor ditempatkan dekat dengan mesin produksi atau panel listrik atau dengan kata lain perangkat *monitoring* dan sensor berada pada lokasi yang sama dengan mesin produksi atau panel listrik.

Metode penelitian terdiri dari empat langkah yaitu langkah pertama merancang arsitektur sistem *monitoring*, langkah kedua merancang dan merealisasikan purwa-rupa perangkat keras dan perangkat lunak sistem *monitoring*, langkah ketiga menguji purwa-rupa sistem *monitoring*, langkah keempat mengevaluasi purwa-rupa sistem *monitoring*. Langkah pertama yaitu merancang arsitektur sistem *monitoring* berbasis Arduino Mega2560 menggunakan *socket programming* yang terdiri dari komputer, perangkat monitoring, switch TP-Link SF1008D. Perangkat *monitoring* dibangun menggunakan sensor arus (YHDC SCT-013-000 100A), sensor tegangan *alternating current* (AC), *modul laser pointer*, sensor cahaya menggunakan *light dependent resistor* (LDR), sensor suhu dan kelembapan udara (DHT 11), *Ethernet Shield* W5100, board Arduino Mega2560. Gambar 1 memperlihatkan diagram blok sistem *monitoring* kelembapan udara, suhu udara, arus listrik, tegangan listrik, dan jumlah produksi barang berbasis Arduino Mega2560 menggunakan *socket programming*.



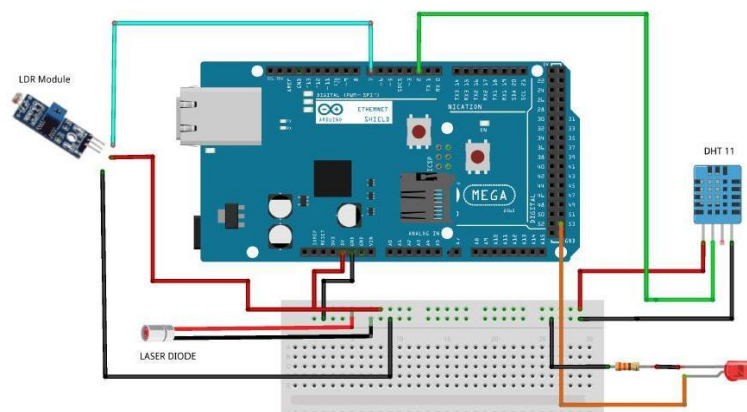
Gambar 1. Arsitektur sistem *monitoring* berbasis Arduino Mega2560 menggunakan *socket programming*.

Arduino Mega2560 bertugas mengatur penerimaan data dari sensor dan tempat terhubungnya *ethernet shield* sebagai media komunikasi *socket* antara Arduino Mega2560 dengan aplikasi Delphi yang terdapat pada komputer. Dari Gambar 1 dapat dilihat bahwa Arduino Mega2560 pertama hanya digunakan untuk membaca data dari sensor DHT 11 dan sensor cahaya (LDR). Arduino Mega2560 kedua digunakan untuk membaca data dari sensor arus SCT dan sensor tegangan AC. Data hasil pembacaan sensor dikirimkan ke komputer server melalui

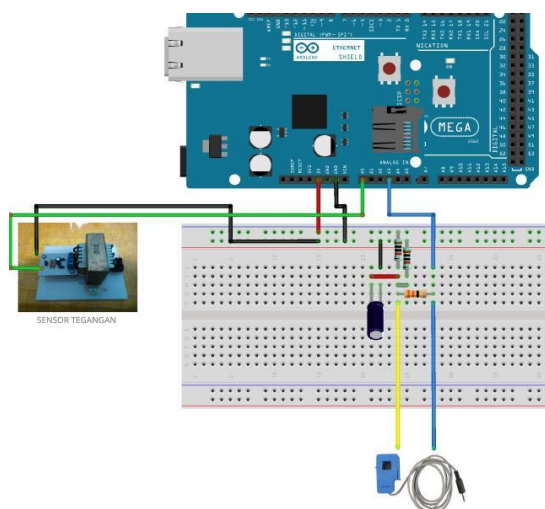
ethernet shield lalu diterima dan ditampilkan oleh komputer (aplikasi Delphi) serta disimpan pada *database*. Data yang disimpan dalam *database* meliputi *log connection*, *log monitoring*, dan *log jumlah produksi*.

Modul *laser pointer* dan LDR digunakan untuk mendeteksi barang. Cara mengukur jumlah produksi barang yaitu dengan mendeteksi barang yang ada diantara modul *laser pointer* dan LDR. Modul *laser pointer* memancarkan cahaya yang tepat mengenai LDR. Jika tidak ada barang diantara modul *laser pointer* dan LDR, maka LDR akan menerima cahaya dari modul *laser pointer* dan *counter* jumlah barang tidak akan bertambah. Sebaliknya jika ada barang diantara modul *laser pointer* dan LDR, maka LDR tidak akan menerima cahaya dari modul *laser pointer*, kemudian *counter* jumlah barang akan bertambah satu. Perubahan *output digital* modul sensor cahaya (LDR) dari kondisi tidak ada barang ke kondisi ada barang dihitung sebagai penambahan 1 barang, jika tidak ada perubahan *output digital* dari modul sensor cahaya (LDR) maka tidak dihitung penambahan 1 barang.

Langkah kedua yaitu merancang dan merealisasikan purwa-rupa perangkat keras dan perangkat lunak sistem *monitoring*. Gambar 2 dan Gambar 3 memperlihatkan perancangan perangkat keras sistem *monitoring* yang terdiri dari Arduino Mega2560 pertama dan Arduino Mega2560 kedua. Komponen yang terhubung ke Arduino Mega2560 pertama yaitu sensor suhu dan kelembapan udara (DHT11) serta sensor cahaya (LDR). Output sensor DHT11 terhubung ke pin 2 Arduino Mega2560 pertama. Selain itu sensor cahaya (LDR) juga terhubung ke Arduino Mega2560 pertama melalui pin 7. Sedangkan Arduino Mega2560 kedua terhubung dengan sensor tegangan AC melalui pin A0 dan terhubung dengan sensor arus SCT melalui pin A3.

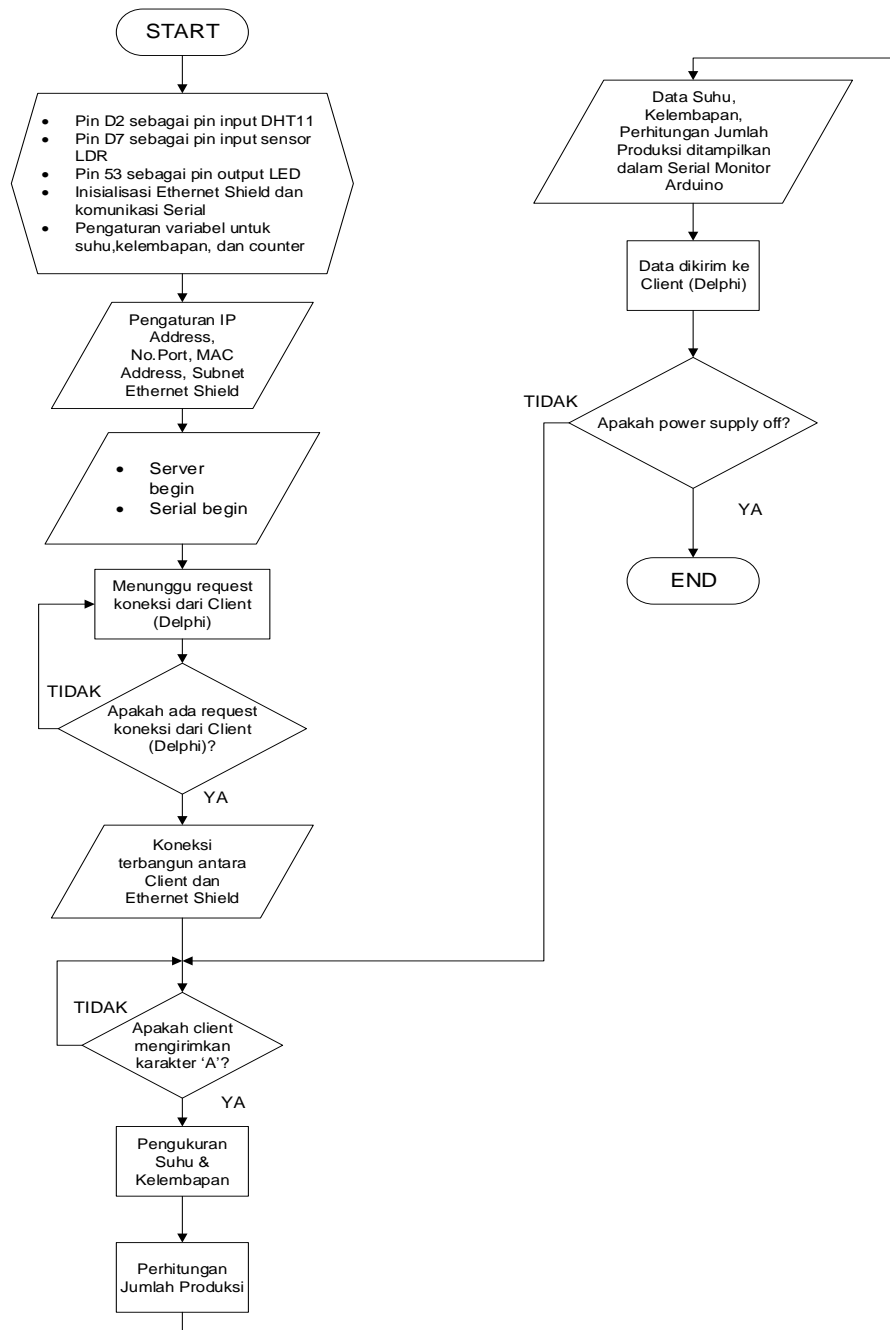


Gambar 2. *Wiring diagram* Arduino Mega2560 pertama.



Gambar 3. *Wiring diagram* Arduino Mega2560 kedua.

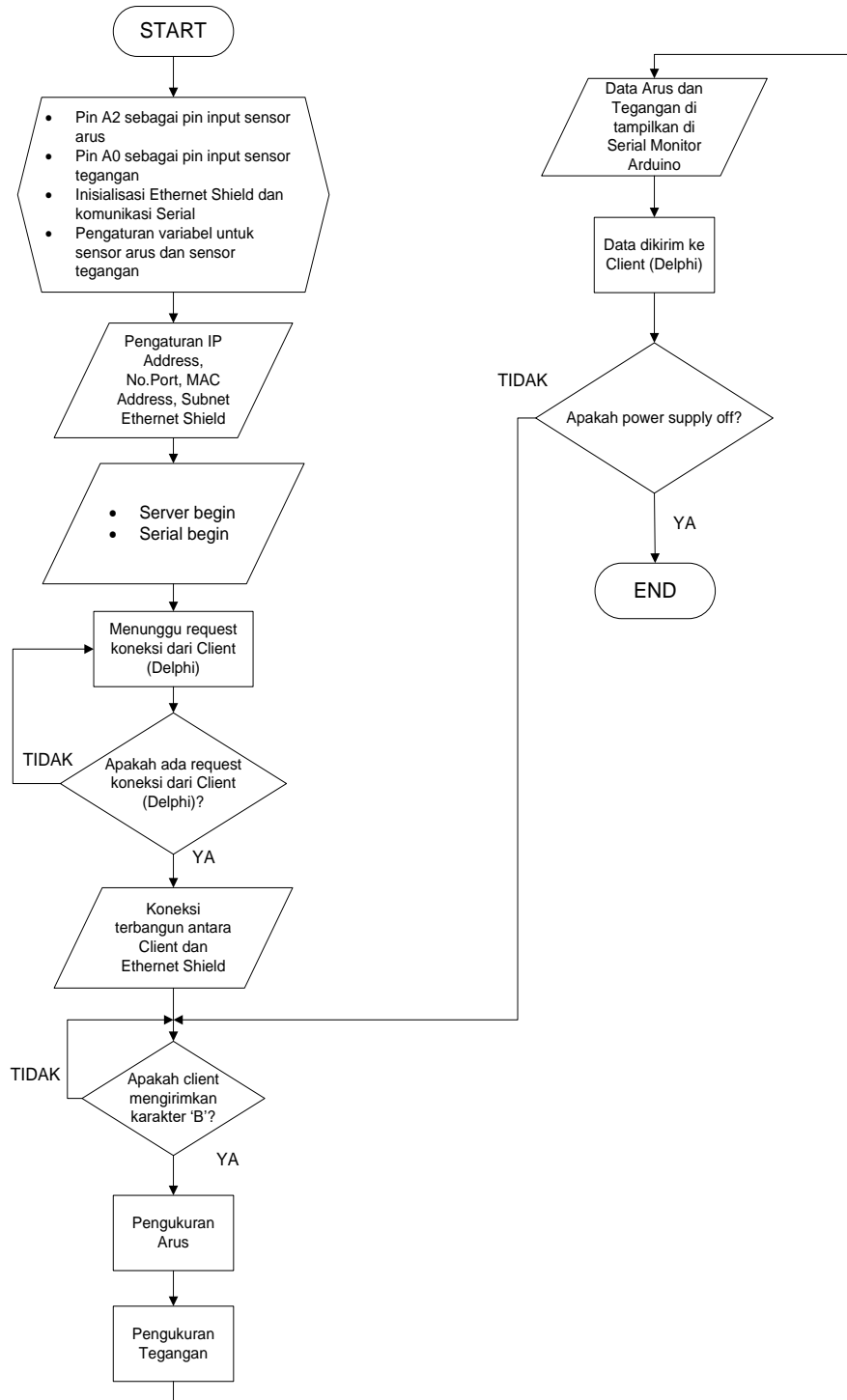
Perancangan perangkat lunak (*software*) dibagi menjadi dua bagian utama, yaitu perangkat lunak pada Arduino Mega2560 dan perangkat lunak pada komputer. Gambar 4 dan Gambar 5 memperlihatkan perancangan perangkat lunak berupa *flowchart* program pada Arduino Mega2560 pertama dan Arduino Mega2560 kedua. Gambar 6 memperlihatkan perancangan perangkat lunak berupa *flowchart* program aplikasi pada komputer.



Gambar 4. *Flowchart* program pada Arduino Mega2560 pertama.

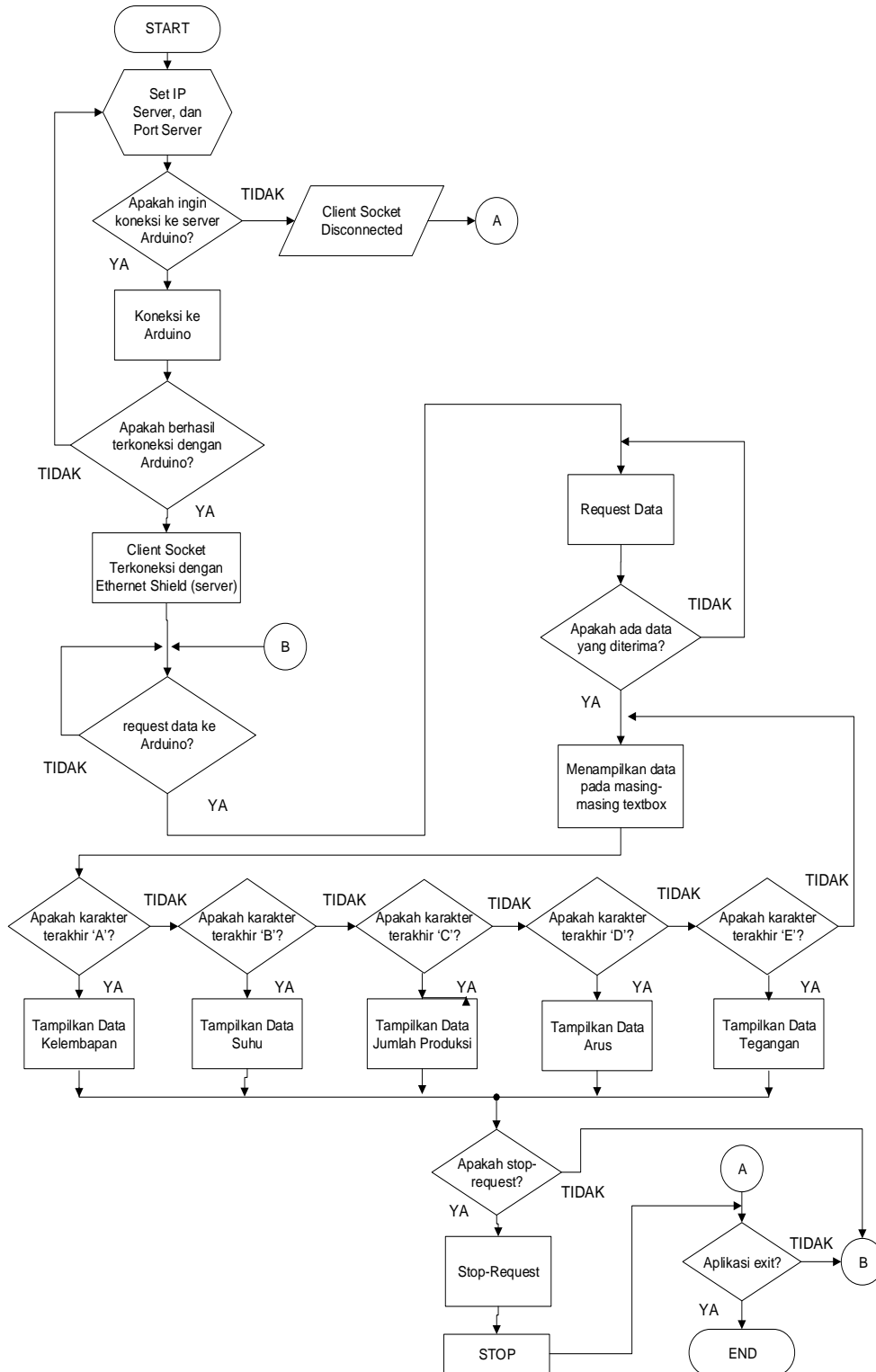
Alur program pada Arduino Mega2560 pertama yaitu, melakukan inisialisasi pin input, pin output, variabel, dan *ethernet shield*. Kedua, pengaturan alamat IP, nomor port, MAC Address, subnet pada *ethernet shield*. Ketiga, menjalankan *ethernet shield* sebagai server. *Ethernet shield* menunggu permintaan koneksi dari komputer. Jika ada permintaan koneksi, maka proses koneksi terbangun antara

ethernet shield dan komputer, jika tidak ada koneksi, maka *ethernet shield* akan tetap menunggu permintaan koneksi dari *client*. Setelah itu, komputer meminta data suhu, kelembapan, dan jumlah produksi barang dengan mengirimkan karakter ‘A’ ke Arduino Mega2560 pertama. Jika karakter ‘A’ diterima, maka Arduino Mega2560 pertama melakukan proses pengukuran suhu, kelembapan, dan perhitungan jumlah produksi. Kemudian hasil pengukuran suhu, kelembapan dan perhitungan jumlah produksi dikirimkan ke komputer. Selama Arduino Mega2560 mendapatkan *power supply*, maka proses akan kembali ke proses *client* mengirimkan karakter ‘A’ untuk meminta data.



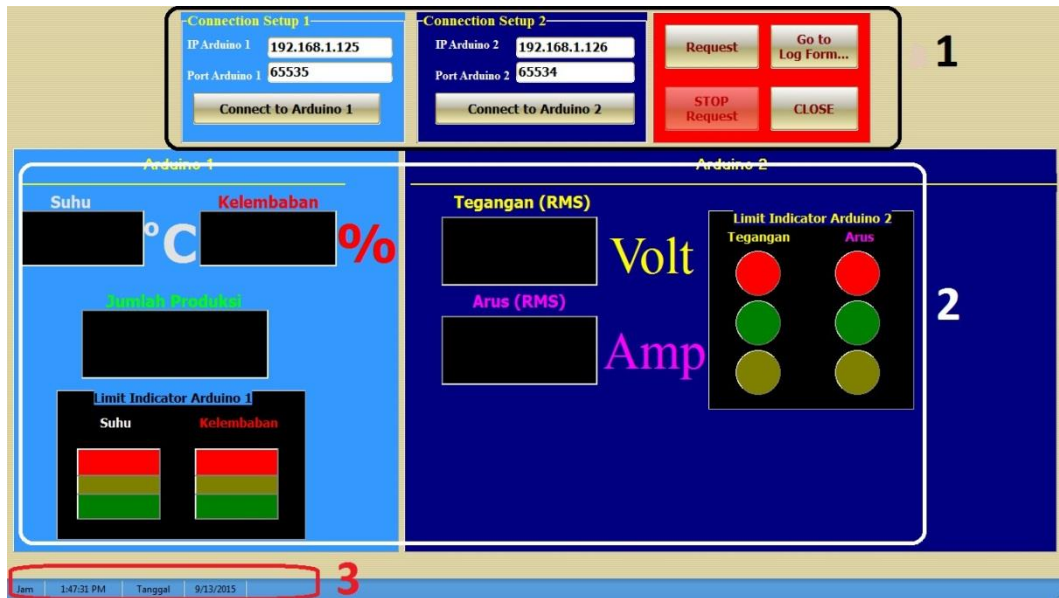
Gambar 5. Flowchart program pada Arduino Mega2560 kedua.

Alur program pada Arduino Mega2560 kedua mempunyai alur proses yang sama dengan Arduino Mega2560 pertama. Perbedaannya hanya dalam penggunaan pin-pin input, pengukuran yang dilakukan dan karakter kode yang diterima oleh *ethernet shield* pada Arduino Mega2560 kedua. Arduino Mega2560 kedua digunakan untuk pengukuran tegangan dan arus. Komputer mengirimkan karakter 'B' untuk meminta data arus dan tegangan ke Arduino Mega2560 kedua.



Gambar 6. Flowchart program aplikasi sistem monitoring pada komputer.

Gambar 7 merupakan tampilan halaman utama program aplikasi *monitoring* pada komputer yang dikembangkan menggunakan Delphi. *Graphical user interface* (GUI) program akan menampilkan data dari Arduino Mega2560 pertama yaitu suhu udara, kelembapan udara, dan jumlah produksi serta data dari Arduino Mega2560 kedua yaitu besaran tegangan AC dan arus AC. Gambar 8 menunjukkan realisasi purwa-rupa sistem *monitoring* kelembapan udara, suhu udara, arus listrik, tegangan listrik, dan jumlah produksi barang secara keseluruhan.



Gambar 7. Realisasi perangkat lunak aplikasi sistem *monitoring* pada komputer.



Gambar 8. Realisasi purwa-rupa perangkat keras sistem *monitoring*

3. Hasil dan Pembahasan

Pada makalah ini, pengujian pembacaan suhu udara, kelembapan udara, arus listrik dan tegangan listrik dilakukan dengan cara membandingkan hasil pembacaan dari perangkat yang dikembangkan dengan alat ukur, kemudian menentukan *error* pengukuran. Sedangkan pengujian

jumlah produksi barang dilakukan dengan cara membandingkan hasil membacaan dari perangkat dengan hasil perhitungan manual. Hasil pengukuran suhu dengan menggunakan sensor DHT 11 dan alat ukur suhu digital Constant HT100 dapat dilihat pada Tabel 1. *Error* pengukuran suhu maksimal yaitu sebesar 6,87% dengan selisih 1,8 °C. Hasil pengukuran suhu menggunakan sensor DHT 11 dalam penelitian ini masih dalam spesifikasi akurasi pengukuran suhu dari sensor DHT 11 yaitu ± 2 °C.

Tabel 1. Hasil pengukuran suhu udara menggunakan DHT11 dan Constant HT100

No	DHT11 (°C)	Constant HT100 (°C)	Error (%)
1	28	26,2	6,87
2	28	26,2	6,87
3	27	26,2	3,05
4	28	26,2	6,87
5	28	26,2	6,87
6	28	26,2	6,87
7	28	26,2	6,87
8	27	26,2	3,05
9	27	26,2	3,05
10	27	26,2	3,05

Hasil pengukuran kelembapan udara dengan menggunakan sensor DHT 11 dan alat ukur kelembapan digital Constant HT100 ditunjukkan pada Tabel 2. *Error* pengukuran kelembapan maksimal didapat 10,2 % dengan selisih 5 % RH. Gambar 9 memperlihatkan hasil pengukuran suhu dan kelembapan menggunakan DHT11 dan Constant HT100. Hasil pengukuran kelembapan menggunakan sensor DHT 11 dalam penelitian ini masih dalam spesifikasi akurasi pengukuran kelembapan dari sensor DHT 11 yaitu ± 5 % RH.

Tabel 2. Hasil pengukuran kelembapan udara menggunakan DHT11 dan Constant HT100

No	DHT11 (%)	Constant HT100 (%)	Error (%)
1	44	49	10,2
2	44	49	10,2
3	44	49	10,2
4	44	49	10,2
5	44	49	10,2
6	44	49	10,2
7	44	49	10,2
8	44	49	10,2
9	44	49	10,2
10	44	49	10,2



Gambar 9. Pengukuran Suhu dan Kelembapan DHT11 dan Constant HT100

Hasil pengukuran arus dan tegangan listrik menggunakan sensor dibandingkan hasil pengukuran arus dan tegangan listrik menggunakan alat ukur pada beberapa jenis beban listrik. Nilai arus listrik yang didapatkan dari pembacaan sensor arus SCT dibandingkan dengan nilai arus listrik yang diukur

menggunakan clamp meter digital. Nilai tegangan listrik yang didapatkan dari pembacaan sensor tegangan dibandingkan dengan nilai tegangan listrik yang diukur menggunakan multimeter digital. Beban 1 terdiri dari charger laptop, charger handphone, dan adaptor switch hub. Tabel 3 dan Tabel 4 memperlihatkan hasil pengukuran arus dan tegangan listrik pada beban 1.

Tabel 3. Hasil pengukuran arus listrik pada beban 1

No	Clamp meter (A)	Sensor Arus SCT (A)	Error (%)
1	0,18	0,18	0
2	0,17	0,17	0
3	0,17	0,18	5,88
4	0,17	0,18	5,88
5	0,17	0,18	5,88
6	0,17	0,18	5,88
7	0,17	0,18	5,88
8	0,17	0,18	5,88
9	0,17	0,18	5,88
10	0,17	0,18	5,88

Tabel 4. Hasil pengukuran tegangan listrik pada beban 1

No	Multimeter Digital (V)	Sensor Tegangan (V)	Error (%)
1	217	218	0,46
2	217	217	0
3	217	217	0
4	216	214	0,92
5	216	216	0
6	216	212	1,85
7	216	214	0,92
8	216	218	0,92
9	216	217	0,46
10	216	217	0,46

Tabel 3 memperlihatkan hasil pengukuran arus listrik yang menunjukkan tingkat *error* maksimal pengukuran arus listrik pada beban 1 adalah 5,88 %. Tabel 4 memperlihatkan hasil pengukuran tegangan listrik yang menunjukkan bahwa tingkat *error* maksimal pengukuran tegangan pada beban 1 adalah 1,85 %. Gambar 10 memperlihatkan pengukuran arus dan tegangan listrik pada beban 1.



(a)



(b)

Gambar 10 (a) Pengukuran arus listrik pada beban 1, (b) Pengukuran tegangan listrik pada Beban 1

Beban 2 terdiri dari lampu dengan nilai daya yang berbeda-beda (15 W, 25 W, 40 W, 60 W, dan 75 W). Hasil pengukuran arus dan tegangan listrik pada beban 2 dapat dilihat pada Tabel 5 dan Tabel 6. Tabel 5 memperlihatkan hasil pengukuran arus dengan tingkat *error* lebih dari 100% ($>100\%$), *error* tersebut disebabkan oleh arus yang dikonsumsi oleh beban sangat kecil jika dibandingkan dengan *range* pengukuran arus sensor (YHDCC SCT-013-000 100A) yaitu antara 0 sampai dengan 100 A, dan jika dilihat dari besarnya nilai daya yang dikonsumsi, seiring dengan daya yang bertambah besar maka tingkat *error* semakin kecil. K.A. Santoso dan D.A Prasetya [2] melakukan pengukuran arus menggunakan sensor arus ACS712 pada beban berupa lampu (35 W, 40 W, 80 W, 100 W dan 200 W) dengan *error* maksimal pengukuran arus sebesar 3,496%. Jika dibandingkan dengan hasil penelitian K.A. Santoso dan D.A Prasetya, maka sensor arus (YHDCC SCT-013-000 100A) yang digunakan pada makalah ini tidak akurat untuk mengukur arus pada beban dengan konsumsi arus yang sangat kecil, pernyataan ini sesuai dengan penelitian R.P Pratama [16] yang juga menyatakan bahwa sensor arus SCT-013-000 tidak dapat membaca arus lebih kecil dari 0.03 A. Sensor arus ACS712 lebih akurat untuk mengukur arus pada beban dengan konsumsi arus yang sangat kecil.

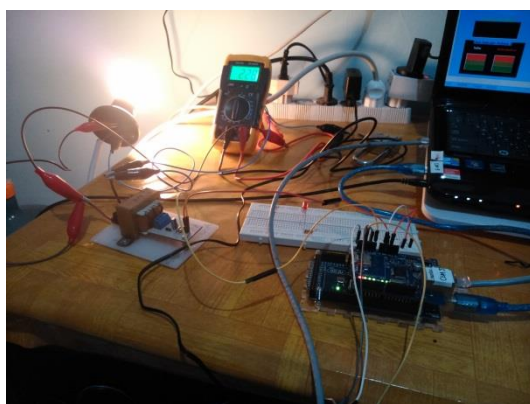
Tabel 6 memperlihatkan hasil pengukuran tegangan listrik dengan tingkat *error* maksimal pengukuran tegangan listrik pada beban 2 adalah 2,21 %. K.A. Santoso dan D.A Prasetya [2] melakukan pengukuran tegangan menggunakan sensor tegangan ZMPT101B pada tegangan 110 sampai dengan 230 volt menggunakan *voltage regulator* dengan *error* maksimal pengukuran tegangan sebesar 2%. Jika dibandingkan dengan hasil penelitian K.A. Santoso dan D.A Prasetya, maka sensor tegangan AC yang digunakan pada makalah ini memiliki tingkat akurasi yang hampir sama dengan sensor tegangan ZMPT101B. Gambar 11 memperlihatkan pengukuran arus listrik dan tegangan listrik pada beban 2.

Tabel 5. Hasil pengukuran arus listrik pada beban 2

No	Daya Beban (W)	Clamp Meter (A)	Sensor Arus (A)	Error (%)
1	15	0,04	0,30	650
2	25	0,08	0,34	325
3	40	0,13	0,39	200
4	60	0,19	0,46	142
5	75	0,25	0,53	112

Tabel 6. Hasil pengukuran tegangan listrik pada beban 2

No	Daya Beban (W)	Multi Meter Digital (V)	Sensor Tegangan (V)	Error (%)
1	15	227	222	2,20
2	25	226	221	2,21
3	40	226	224	0,88
4	60	227	222	2,20
5	75	228	225	1,31



(a)



(b)

Gambar 11. (a) Pengukuran tegangan listrik pada beban 2, (b) Pengukuran arus listrik pada beban 2

Beban 3 yang terdiri dari pompa air dengan tekanan 3 Psi (1520 Watt). Tabel 7 dan Tabel 8 memperlihatkan hasil pengukuran arus listrik dan tegangan listrik pada beban 3. Tabel 7 memperlihatkan hasil pengukuran arus listrik, *error* maksimal pada pengukuran arus listrik beban 3 yaitu sebesar 3,46 %. Tabel 8 memperlihatkan hasil pengukuran tegangan listrik, tingkat *error* maksimal pengukuran tegangan listrik pada beban 3 yaitu 1,83 %. A.R Suharso dkk [1] menggunakan sensor arus ACS712 dan sensor tegangan ZMPT101B untuk mengukur arus dan tegangan pada alat pemanas ruangan dengan *error* maksimal 3.61% untuk pengukuran arus dan 0.24% untuk pengukuran tegangan. Jika dibandingkan dengan hasil penelitian A.R Suharso dkk, maka sensor arus (YHDCC SCT-013-000 100A) yang digunakan pada makalah ini memiliki tingkat akurasi yang hampir sama dengan sensor arus ACS712 pada beban dengan konsumsi arus yang besar. Sensor tegangan AC yang digunakan pada makalah ini memiliki *error* pengukuran tegangan yang lebih besar dibandingkan dengan sensor tegangan ZMPT101B.

Tabel 7. Hasil pengukuran arus listrik pada beban 3

No	Clampmeter (A)	Sensor Arus (A)	Error (%)
1	6,97	7,17	2,87
2	6,97	7,16	2,72
3	6,97	7,14	2,44
4	6,95	7,19	3,45
5	6,94	7,16	3,17
6	6,95	7,16	3,02
7	6,94	7,12	2,59
8	6,94	7,13	2,74
9	6,93	7,17	3,46
10	6,94	7,13	2,74

Tabel 8. Hasil pengukuran tegangan listrik pada beban 3

No	Multimeter Digital (V)	Sensor Tegangan (V)	Error (%)
1	217	218	0,46
2	217	217	0
3	218	214	1,83
4	217	220	1,38
5	217	218	0,46
6	217	218	0,46
7	217	217	0
8	217	217	0
9	217	217	0
10	217	217	0

Tabel 9 memperlihatkan hasil perhitungan jumlah produksi yang ditampilkan di komputer serta perhitungan dengan manual. *Sampel* perhitungan jumlah produksi diambil 10 kali perhitungan sampai 50 kali perhitungan, hasil nilai perhitungan manual dan perhitungan nilai di komputer menunjukkan nilai yang sama. Dari Tabel 9 dapat dilihat bahwa sistem *counter* jumlah produksi barang berfungsi dengan baik.

Tabel 9. Hasil perhitungan jumlah produksi menggunakan cara manual dan komputer

No	Manual	Komputer	Error (%)
1	10	10	0
2	20	20	0
3	30	30	0
4	40	40	0
5	50	50	0

4. Kesimpulan

Sistem *monitoring* keadaan suhu udara, kelembapan udara, arus listrik dan tegangan listrik serta jumlah produksi barang berbasis Arduino Mega2560 melalui *socket programming* telah berhasil dirancang dan direalisasikan serta dapat berfungsi dengan baik. Pengukuran suhu dan kelembapan menggunakan sensor DHT 11 memiliki *error* maksimal yang masih sesuai dengan spesifikasi akurasi sensor DHT 11. *Error* maksimal pengukuran arus dan tegangan pada beban pertama (charger laptop, charger handphone, dan adaptor switch hub) didapatkan sebesar 5,88 % untuk pengukuran arus dan 1,85

% untuk pengukuran tegangan. Pengukuran arus dan tegangan pada beban kedua berupa lampu dengan daya yang bervariasi (15 W, 25 W, 40 W, 60 W, dan 75 W) didapatkan *error* pengukuran > 100 % untuk pengukuran arus dan 2,21 % untuk pengukuran tegangan. Sedangkan pengukuran arus dan tegangan pada beban ketiga berupa pompa air dengan daya 1520 W didapatkan *error* maksimal sebesar 3,46 % untuk pengukuran arus dan 1,83 % untuk pengukuran tegangan. Dari hasil pengujian arus listrik menggunakan sensor arus (YHDCC SCT-013-000 100A) dapat disimpulkan bahwa sensor arus (YHDCC SCT-013-000 100A) tidak akurat jika digunakan untuk mengukur arus yang sangat kecil (<1 A). Sensor arus (YHDCC SCT-013-000 100A) tepat digunakan untuk mengukur arus 1 A sampai dengan 100 A.

Referensi

- [1] A. Rakhman Suharso, A. Nugraha, and D. Oktarina Dwi Handayani, "Sistem Monitor dan Kontrol Konsumsi Listrik Rumah Tangga Berbasis IoT dan Android," *J. Rekayasa Teknol. Nusa Putra*, vol. 7, no. 2, pp. 1–11, 2021.
- [2] K. A. Santoso and D. A. Prasetya, "Rancang Bangun KWh Meter Digital Berbasis IoT," in *Simposium Nasional RAPI XIX Tahun 2020 FT UMS*, 2020, pp. 39–46.
- [3] M. Manfaluthy, A. Sunardi, S. Wilyanti, A. Jaenul, and E. Mulyana, "Design a Monitoring System of a Single-Phase Electricity Meter Based on the Internet of Things," *Spektra J. Fis. dan Apl.*, vol. 7, no. 3, pp. 113–126, 2022.
- [4] F. M. Dahunsi, S. O. Eniola, A. A. Ponnle, O. A. Agbolade, C. N. Udekwe, and A. O. Melodi, "A Review of Smart Energy Metering System Projects," *J. Elektron. dan Telekomun.*, vol. 21, no. 1, p. 70, 2021.
- [5] M. Sari *et al.*, "Pengembangan Wireless Sensor Network Pada Komunikasi Development of Wireless Sensor Network in Communication," *Jurnal Media Elektrik.*, vol. 19, no. 2, pp. 106–111, 2022.
- [6] W. R. Saputra, A. Muid, and T. Rismawan, "Rancang Bangun Sistem Monitoring Gangguan pada Gardu Listrik Menggunakan Arduino dan Website," *J. Coding*, vol. 04, no. 2, pp. 141–150, 2016.
- [7] E. Özceylan, "Implementation of Socket Programming Simulation Using Quantum Communication Technologies," in *2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 2022, pp. 1007–1012.
- [8] B. Shafiei, "Socket programming," *Adv. Environ. Biol.*, vol. 6, no. 5, pp. 1812–1822, 2012.
- [9] B. Shafiei and F. Iranmanesh, "Review socket programming," *Aust. J. Basic Appl. Sci.*, vol. 6, no. 7, pp. 354–364, 2012.
- [10] A. Baig, "Protocols and sockets [Internetprogramming]," *IEEE Potential*, vol. 18, no. 3, pp. 38–41, 1999.
- [11] M. Xue and C. Zhu, "The socket programming and software design for communication based on client/server," in *Proceedings of the 2009 Pacific-Asia Conference on Circuits, Communications and System, PACCS 2009*, 2009, pp. 775–777.
- [12] O. Ayurzana, Y. Park, J. Kwon, and H. Kim, "Development of Telemetry System of Water Supply using the RF module and TCP/IP Socket Program," in *SICE-ICASE International Joint Conference*, 2006, pp. 3744–3746.
- [13] A. Jitbanyud and N. Toaditsep, "The system of powerful computer laboratory class via socket programming," in *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010*, 2010, vol. 4, pp. 638–641.
- [14] R. L. Maata, R. Cordova, B. Sudramurthy, and A. Halibas, "Design and Implementation of Client-Server Based Application Using Socket Programming in a Distributed Computing Environment," in *2017 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2017*, 2018, pp. 1–4.
- [15] Z. Ramadhan, S. R. Akbar, and G. E. Setyawan, "Implementasi Sistem Monitoring Daya Listrik Berbasis Web dan Protokol Komunikasi Websocket," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 1, pp. 205–211, 2019.
- [16] R. P. Pratama, "Rancang Bangun Alat Pengukur Data Kelistrikan dengan Aplikasi

Android MQTT Dash,” *Elektriese J. Sains dan Teknol. Elektro*, vol. 12, no. 01, pp. 63–72, 2022.