

Perbandingan Kinerja Arduino Uno dan ESP32 Terhadap Pengukuran Arus dan Tegangan

¹ I Putu Ardi Wahyu Widyatmika, ² Ni Putu Ayu Widyanata Indrawati,
³ I Wayan Wahyu Adi Prastya, ⁴ I Ketut Darminta, ⁵ I Gde Nyoman Sangka,
⁶ Anak Agung Ngurah Gde Sapteka *)

1,2,3,4,5,6 Politeknik Negeri Bali

*)sapteka@pnb.ac.id

Abstrak

Mikrokontroler merupakan bagian inti dari proyek kontrol otomatis. Pemilihan mikrokontroler sangat berpengaruh terhadap hasil pengolahan data pada sistem. Atas dasar pemikiran tersebut maka dilakukan sebuah penelitian yang bertujuan untuk menemukan mikrokontroler dengan kemampuan pengolahan data yang baik. Dalam penelitian ini menggunakan mikrokontroler Arduino Uno dan mikrokontroler ESP32. Di mana kedua mikrokontroler tersebut memiliki spesifikasi yang berbeda. Data yang digunakan berupa nilai tegangan dan arus yang diperoleh dari sensor tegangan ZMPT101B dan sensor arus ACS712. Pengujian yang dilakukan menghasilkan nilai rata-rata *error* tegangan sebesar 0,387649 persen dan arus 3,095044 persen untuk mikrokontroler Arduino Uno. Sedangkan untuk mikrokontroler ESP32 menghasilkan rata-rata nilai *error* tegangan sebesar 0,312182825 persen dan arus sebesar 0,194657573 persen.

Kata Kunci: Mikrokontroler, ESP32, Arduino Uno, Tegangan, Arus

1 Pendahuluan

Seiring perkembangan ilmu pengetahuan dan teknologi, saat sekarang banyak muncul gagasan-gagasan bidang elektronika digital. Sistem digital berkembang diaplikasikan pada teknologi mikrokontroler [1]. Dalam proses pengembangan dan penelitian, mikrokontroler yang sering digunakan oleh peneliti adalah Arduino Uno. Arduino Uno adalah sebuah rangkaian yang dikembangkan dari mikrokontroler berbasis ATmega328 [2]. Arduino UNO dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau *battery* [3]. Di mana Arduino Uno memiliki pin ADC 10-bit, yang artinya nilai hasil konversi berkisar dari 0 hingga 1023. Arduino Uno pun memiliki kekurangan yaitu tidak dapat terkoneksi dengan *WiFi*, untuk dapat terkoneksi dengan *WiFi* maka harus menambahkan komponen *WiFi module*. Sebagai perbandingan dan untuk mengatasi kekurangan dari Arduino Uno tersebut, maka dipilihlah mikrokontroler yang berbeda yaitu mikrokontroler ESP32. ESP32 adalah mikrokontroler yang diperkenalkan oleh *Espressif System* merupakan penerus dari ESP8266 [4]. Selain itu ESP32 juga memiliki keunggulan dibandingkan dengan mikrokontroler yang lain, mulai dari *pin out* yang lebih banyak, *pin analog* yang lebih banyak, memori yang lebih besar, serta terdapat *low energy Bluetooth 4.0* [5]. Pada mikrokontroler ini sudah tersedia modul *WiFi* dalam *chip* prosesor *dual core* yang berjalan di instruksi Xtensa LX16 sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things* [6]. Memori ESP32 terdiri atas 448 kB ROM, 520 kB SRAM, dua 8 kB *RTC memory*, dan *flash memory* 4MB. Chip ini mempunyai 18 pin ADC (12-bit), empat unit SPI, dan dua unit I2C. Kelebihan utama mikrokontroler ini ialah harganya yang relatif murah, mudah diprogram, memiliki jumlah pin I/O yang memadai,

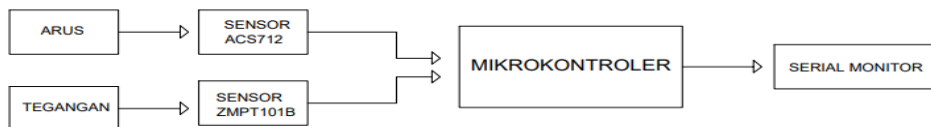
serta memiliki adapter WiFi [7]. ESP32 memiliki pin ADC 12-bit, yang artinya bernilai 0 hingga 4095. Untuk mengukur arus yang mengalir lewat blok terminal digunakan modul sensor ACS712. Menurut Fachri dkk, sensor ini dapat mengukur arus positif dan negatif dengan kisaran -5A sampai 5A. Sensor ini memerlukan suplai daya sebesar 5V [8]. Ketelitian dalam pembacaan sensor dioptimalkan dengan cara pemasangan komponen yang ada di antara penghantar yang menghasilkan medan magnet dengan *hall transducer* secara berdekatan [9]. Untuk mengukur tegangan menggunakan sensor ZMPT101B. Sensor tegangan ZMPT101B adalah suatu modul sensor tegangan AC yang menggunakan trafo isolasi dengan rasio tegangan 1:1. Sensor ini memiliki kelebihan antara lain transforme ukuran kecil, akurasi tinggi, konsistensi yang baik, untuk tegangan dan pengukuran daya. Modul ini dilengkapi dengan *trim* multi-putar potensiometer untuk menyesuaikan *output* ADC [10]. Sensor ini memiliki akurasi tinggi, konsistensi yang baik untuk tegangan dan tenaga pengukuran serta dapat mengukur sampai 250V AC [11].

Berdasarkan uraian tersebut, maka pada artikel dilakukan kajian tentang perbandingan kinerja ESP32 dan Arduino Uno terhadap pengukuran nilai tegangan dan arus. Data tersebut pastinya akan ada perbedaan karena ESP32 dan Arduino UNO memiliki bit yang berbeda. Semakin besar bit yang dimiliki mikrokontroler maka data yang diproses lebih banyak dalam satu waktu. Data itu nantinya akan digunakan untuk menemukan selisih kesalahan pembacaan nilai sensor (nilai *error*), sehingga dapat diketahui mikrokontroler yang lebih baik dalam pengolahan data pembacaan tegangan dan arus.

2 Metode

2.1 Diagram Balok

Purwarupa pengukuran sensor ACS712 dan sensor ZMPT101B ini memiliki diagram balok seperti Gambar 1.

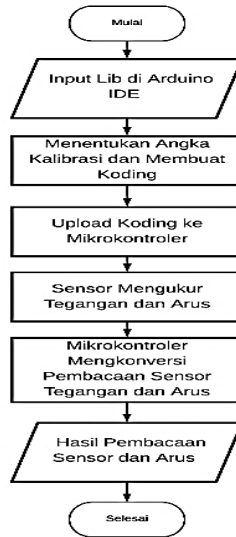


Gambar 1. Diagram balok purwarupa pengukuran sensor ACS712 dan sensor ZMPT101B.

Purwarupa ini menggunakan dua sensor yaitu sensor arus ACS712 dan sensor tegangan ZMPT101B. Pada sensor ACS712 akan mengukur besar arus listrik yang masuk ke terminal sensor, lalu data tersebut menuju ke mikrokontroler. Sensor ZMPT101B akan mengukur besar tegangan yang masuk ke sensor tersebut dan mengirim hasil data pengukuran ke mikrokontroler yang digunakan. Mikrokontroler berfungsi untuk mengolah data yang diterima dari dua sensor tersebut. Data-data tersebut akan diolah dan diterjemahkan sesuai perintah koding yang telah disematkan pada mikrokontroler. Hasil data yang telah diolah oleh mikrokontroler akan dimunculkan pada *Serial Monitor* di Arduino IDE.

2.2 Diagram Alir Kalibrasi Sensor

Kalibrasi sensor arus ACS712 dan sensor tegangan ZMPT101B ini memiliki diagram alir seperti Gambar 2.

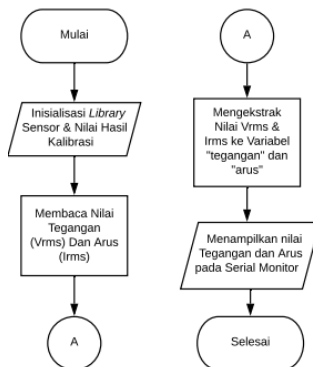


Gambar 2. Diagram alir kalibrasi sensor

Pada *software* Arduino IDE, dapat memasukkan *library* yang akan digunakan. Selanjutnya, membuat koding dan menentukan angka kalibrasi sehingga nantinya hasil yang didapat akan mendekati nilai sebenarnya. Setelah selesai membuat koding, *upload* koding ke mikrokontroler ESP32. Jika sudah berhasil diupload, maka data hasil pengukuran dari sensor arus ACS712 dan sensor tegangan ZMPT101B akan masuk ke mikrokontroler. Pada mikrokontroler, pembacaan kedua sensor tersebut akan dikonversi terlebih dahulu. Setelah dikonversi, hasil pembacaan sensor dan arus akan ditampilkan melalui *Serial Monitor* pada Arduino IDE.

2.3 Diagram Alir Koding

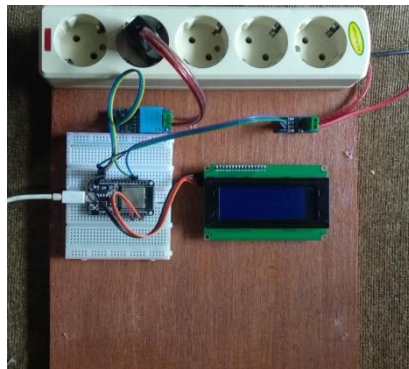
Diagram alir koding menggunakan *platform* Arduino IDE dengan mikrokontroler Arduino Uno dan ESP32 seperti Gambar 3.



Gambar 3 Diagram alir koding

2.4 Gambar Instalasi

Gambar instalasi alat yang telah dirangkai seperti Gambar 4.



Gambar 4 Instalasi alat

3 Hasil

3.1 Hasil pengukuran menggunakan Arduino Uno

Pada pengujian pengukuran arus dan tegangan pertama, menggunakan mikrokontroler Arduino Uno yang dibandingkan dengan alat ukur. Keduanya diberikan beban yang sama, dan didapatkan hasil seperti pada Gambar 6 dan Tabel 1.



Gambar 5 Hasil pengukuran tegangan dan arus menggunakan alat ukur

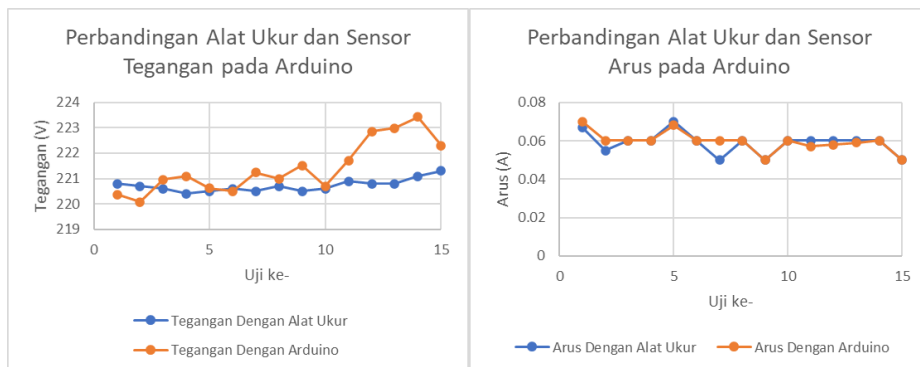


Gambar 6 Hasil pengukuran menggunakan Arduino Uno

Tabel 1 Hasil perbandingan alat ukur dan sensor pada Arduino

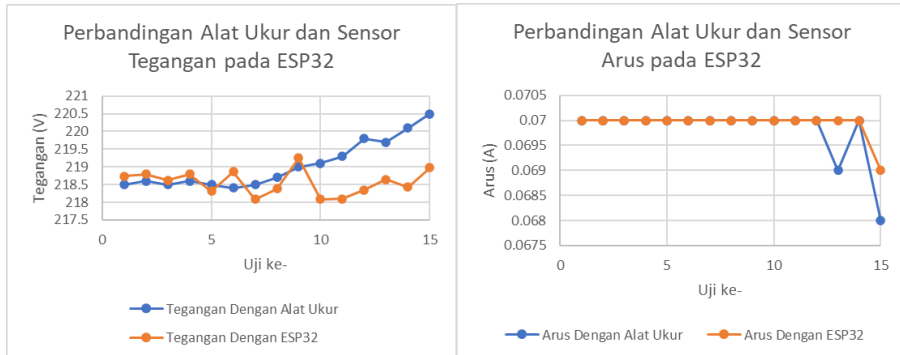
ARDUINO UNO					
ALAT UKUR		SENSOR		ERROR (%)	
V	I	V	I	V	I
220,8	0,067	220,37	0,07	0,194746	4,477612
220,7	0,055	220,08	0,06	0,280924	9,090909
220,6	0,06	220,97	0,06	0,167724	0
220,4	0,06	221,1	0,06	0,317604	0
220,5	0,07	220,62	0,068	0,054422	2,857143
220,6	0,06	220,51	0,06	0,040798	0
220,5	0,05	221,24	0,06	0,335601	20
220,7	0,06	220,99	0,06	0,1314	0
220,5	0,05	221,51	0,05	0,45805	0
220,6	0,06	220,7	0,06	0,045331	0
220,9	0,06	221,71	0,057	0,366682	5
220,8	0,06	222,85	0,058	0,928442	3,333333
220,8	0,06	222,98	0,059	0,987319	1,666667
221,1	0,06	223,43	0,06	1,053822	0
221,3	0,05	222,3	0,05	0,451875	0
Rata-rata				0,387649	3,095044

Perbandingan hasil data untuk pengukuran tegangan menggunakan Arduino Uno dan alat ukur dapat dilihat pada Gambar 7.



Gambar 7 Perbandingan alat ukur dan sensor pada Arduino

Perbandingan hasil data untuk pengukuran tegangan menggunakan ESP32 dan alat ukur dapat dilihat pada Gambar 8.



Gambar 8 Grafik perbandingan alat ukur dan sensor pada esp32

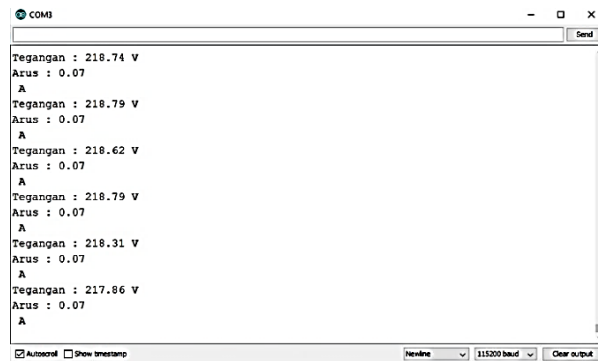
Saat dilakukan pengujian dengan mikrokontroler Arduino Uno, dipasang beban berupa *charger handphone* dan *charger laptop*. Pengujian tersebut dilakukan dengan membandingkan kedua sensor dan alat ukur. Hal ini bertujuan untuk melihat tingkat akurasi data yang dihasilkan oleh sensor. Nilai penyimpangan dalam (%) dihitung menggunakan Persamaan (1).

$$Error (\%) = \frac{|Nilai alat ukur - Nilai sensor|}{Nilai alat ukur} \times 100\% \tag{1}$$

Dapat dilihat data tegangan sumber yang terukur oleh sensor ZMPT101B menghasilkan hasil data yang sedikit kurang stabil dengan rata-rata *error* yang didapat sebesar 0,387649%. Untuk pengukuran arus oleh sensor ACS712 didapatkan hasil yang kurang stabil dengan rata-rata *error* yang didapat sebesar 3,095044%.

3.2 Hasil pengukuran menggunakan ESP32

Pada pengujian pengukuran arus dan tegangan kedua, menggunakan mikrokontroler ESP32 yang dibandingkan dengan alat ukur. Keduanya diberikan beban yang sama didapatkan hasil seperti pada Gambar 9 dan Tabel 2.



Gambar 9 Hasil pengukuran menggunakan ESP32.

Tabel 2 Hasil perbandingan alat ukur dan sensor pada ESP32

ESP32					
ALAT UKUR		SENSOR		ERROR (%)	
V	I	V	I	V	I
218,5	0,07	218,74	0,07	0,109839817	0
218,6	0,07	218,79	0,07	0,086916743	0
218,5	0,07	218,62	0,07	0,054919908	0
218,6	0,07	218,79	0,07	0,086916743	0
218,5	0,07	218,31	0,07	0,086956522	0
218,4	0,07	218,86	0,07	0,210622711	0
218,5	0,07	218,08	0,07	0,19221968	0
218,7	0,07	218,38	0,07	0,146319159	0
219	0,07	219,25	0,07	0,114155251	0
219,1	0,07	218,09	0,07	0,460976723	0
219,3	0,07	218,1	0,07	0,547195622	0
219,8	0,07	218,35	0,07	0,659690628	0
219,7	0,069	218,65	0,07	0,477924442	1,449275362
220,1	0,07	218,43	0,07	0,758746025	0
220,5	0,068	218,98	0,069	0,689342404	1,470588235
Rata-rata				0,312182825	0,194657573

Saat dilakukan pengujian dengan mikrokontroler ESP32, dipasang beban berupa *charger handphone* dan *charger laptop*. Pengujian tersebut dilakukan dengan membandingkan kedua sensor dan alat ukur. Hal ini bertujuan untuk melihat tingkat akurasi data yang dihasilkan oleh sensor. Dapat dilihat data tegangan sumber yang terukur oleh sensor ZMPT101B menghasilkan hasil data yang sedikit kurang stabil tetapi lebih baik jika dibandingkan dengan mikrokontroler Arduino Uno, rata-rata error yang didapat sebesar 0,312182825%. Untuk pengukuran arus oleh sensor ACS712

didapatkan hasil yang hampir sangat stabil dengan rata-rata *error* yang didapat sebesar 0,194657573%. Percobaan dilakukan berulang hingga 15 kali guna mendapatkan hasil yang akurat.

4 Kesimpulan

Berdasarkan hasil percobaan yang telah dilakukan dengan menggunakan dua mikrokontroler, dapat disimpulkan bahwa pemilihan mikrokontroler juga dapat mempengaruhi hasil data pengukuran. Hal tersebut dikarenakan mikrokontroler merupakan otak atau pengendali dari suatu komponen agar dapat sesuai dengan tujuan yang diinginkan. Kelebihan mikrokontroler ESP32 yaitu selain dilengkapi dengan modul Wifi juga dilengkapi dengan pin input ADC yang lebih banyak. Terbukti dari penggunaan mikrokontroler ESP32 mendapatkan rata-rata nilai *error* untuk pengukuran tegangan sebesar 0,312182825 persen dan pengukuran arus sebesar 0,194657573 dibandingkan saat menggunakan mikrokontroler Arduino Uno di mana menghasilkan rata-rata nilai *error* untuk pengukuran tegangan sebesar 0,387649 persen dan pengukuran arus sebesar 3,095044 persen.

5 Ucapan Terima kasih

Terima kasih kami sampaikan kepada editor dan reviewer Jurnal Otomasi Kontrol dan Instrumentasi atas review yang diberikan sehingga artikel ini dapat dipublikasikan.

6 Referensi

- [1] Wibowo, A., *Pengaturan Kipas Berbasis Mikrokontroler Dengan Menggunakan Suhu*, Skripsi, Universitas Negeri Semarang, 2011.
- [2] Silvia, A.F., Haritman, E., dan Muladi, Y., *Rancang Bangun Akses Kontrol Pintu Gerbang Berbasis Arduino Dan Android*, ELECTRANS, 13(1), 2014.
- [3] Sokop, S.J., Mamahit, D.J., dan Sompie, S.R.U.A., *Trainer Periferal Antarmuka Berbasis Mikrokontroler Arduino Uno*, Jurnal Teknik Elektro Dan Komputer, 5(3), 2016.
- [4] Zaini, M., Safrudin, dan Bachrudin, M., *Perancangan Sistem Monitoring Tegangan, Arus Dan Frekuensi Pada Pembangkit Listrik Tenaga Mikrohidro Berbasis IoT*, TESLA: Jurnal Teknik Elektro, 22(2), 2020.
- [5] Pradisti, R., dan Mustaziri, *Rancang Bangun Alat Penghitung Biaya Penggunaan Listrik Kamar Kos Secara Otomatis Berbasis Arduino Menggunakan Sensor Arus*, TEKNIKA (Jurnal Ilmiah Bidang Ilmu Rekayasa), 12(2), 2018.
- [6] Wahyudi, J., *Audit Energi Di Bidang Tata Cahaya Untuk Gedung Kampus Bonaventura UAJY*, Skripsi, Universitas Atma Jaya Yogyakarta, 2014.
- [7] Wagya, A., dan Rahmat, *Prototipe Modul Praktik untuk Pengembangan Aplikasi Internet of Things (IoT)*, SETRUM: Sistem Kendali Tenaga Elektronika Telekomunikasi Komputer, 8(2), 2019
- [8] Fachri, M.R., Sara, I.D., dan Away, Y., *Pemantauan Parameter Panel Surya Berbasis Arduino secara Real Time*, Jurnal Rekayasa Elektrika (JRE), 11(4), 2015.
- [9] Ilham, A.A., Syafaruddin, dan Ramschie, A.A.S., *Sistem Monitoring Dan Kendali Kerja Air Conditioning Berbasis Mikrokontroller ATmega 8535*, Jurnal Ristek, 2(1), 2013.

- [10] Arifin, N., Lubis, R.S., dan Gapy, M, *Rancang Bangun Prototype Power Meter 1 Fasa Berbasis Mikrokontroller Atmega328P*, KITEKTRO, 4(1), 2019.
- [11] Arsyad, B.M., Sofwan, A., dan Nugroho, A., *Perancangan Sistem Kontrol Over/Under Voltage Relay Berbasis Mikrokontroler Pada Saluran Tegangan 220VAC*, TRANSMISI, 21(1), 2019.