

Prototype Automated Manipulator Robot Menggunakan Mikrokontroler NodeMCU ESP8266 Berbasis Internet of Things (IoT)

Eka Revadiaz¹, Mohammad Fatkhurrohman^{2*}, Didik Aribowo^{3*}

¹²³Universitas Sultan Ageng Tirtayasa

*Corresponding author, e-mail: 2283170017@untirta.ac.id

Abstrak

Peran robot manipulator dalam dunia industri era Revolusi Industri 4.0 menjadi sangat krusial karena mampu memberikan dampak yang signifikan terhadap dunia industri terutama dalam efisiensi dan efektivitas produksi. Melihat kondisi pandemi saat ini, proses produksi di industri sedikit terhambat karena pekerja diharuskan bekerja dari jarak jauh dan kontrol robot dilakukan di tempat. Dengan berkembangnya Revolusi Industri 4.0, maka dibuatlah *Prototype Automated Manipulator Robot* yang dapat dikontrol secara nirkabel baik secara manual maupun otomatis dengan menggunakan mikrokontroler NodeMCU ESP8266 dan teknologi *Internet Of Things (IoT)*. Metode penelitian yang dilakukan dalam penelitian ini menggunakan metode prototyping dengan pengujian sistem minimum dan kinerja robot. Hasil pengujian pengukuran sistem minimum dan kinerja kontrol robot yang telah dilakukan sebanyak 15 kali menunjukkan bahwa prototipe robot Manipulator otomatis dapat dioperasikan dengan baik dengan persentase kesalahan keseluruhan kurang dari 1% dan dapat dikontrol secara nirkabel dan otomatis melalui aplikasi Blynk.

Keyword: robot manipulator, revolusi industri 4.0, mikrokontroler, nodemcu esp8266, internet of things.

Abstract

The role of manipulator robots in the industrial world of the Industrial Revolution era 4.0 becomes very crucial because it is able to have a significant impact on the industrial world, especially in production efficiency and effectiveness. Seeing the current pandemic conditions, the production process in the industry is slightly hampered because workers are required to work remotely and robot control is carried out on-site. With the development of the Industrial Revolution 4.0, therefore made Prototype Automated Manipulator that can be controlled wirelessly either manually or automatically by using microcontroller NodeMCU ESP8266 and Internet of Things (IoT) technology. The research method conducted in this study using prototyping method with minimum system testing and robot performance. The results of testing the minimum system measurement and robot control performance that has been done 15 times show that the Prototype Automated Manipulator Robot can be operated properly with an overall error percentage of less than 1% and can be controlled wirelessly and automatically through the Blynk application.

Keywords: robots, industrial revolution 4.0, microcontroller, nodemcu esp8266, internet of things.

PENDAHULUAN

Peran serta eksistensi robot dalam kehidupan manusia pada abad ke-21 ini menjadi sangat penting bahkan telah menjadi bagian dari kehidupan manusia itu sendiri. Mulai dari robot industri, robot kantor, bahkan telah hadir dalam kehidupan kita pada skala rumah. Namun, peran robot pada dunia industri menjadi sangat krusial karena robot tersebut dapat memberikan dampak yang signifikan dalam proses produksi di industri khususnya dalam efisiensi produksi.

Robot industri biasanya terdiri dari lengan bersendi (*manipulator multi-linked*) dan efektor akhir yang melekat pada permukaan tetap. Selain itu, Organisasi Internasional untuk Standardisasi (ISO) dalam ISO 8373, mendefinisikan robot industri sebagai “manipulator yang dikendalikan secara otomatis, dapat diprogram ulang, multiguna, dapat diprogram dalam tiga *axis* atau lebih, yang dapat diperbaiki di tempat secara langsung untuk digunakan dalam aplikasi otomasi industri [1]. Mikrokontroler merupakan salah satu komponen teknologi yang berperan penting di era Revolusi Industri 4.0. Hal ini dikarenakan sistem mikrokontroler mampu melayani setiap ruang dan pergerakan dalam produktivitas di dunia industri [2].

Beberapa penelitian sebelumnya juga telah membahas mengenai rancang bangun/prototype robot manipulator. Pada penelitian yang dilakukan oleh Salmon dan Bartolomius Harpad “*Prototype Lengan Robot Berbasis Arduino dengan Menggunakan Algoritma Kinematics*” [3]. Pada penelitian ini, peneliti membuat prototype lengan robot berbasis Arduino dengan menerapkan algoritma kinematik yang telah di coding ke Arduino dengan kontrol RemoteXY yang disematkan pada *smartphone* android dengan *bluetooth* sebagai jembatan antara mikrokontroler dan *smartphone*. Pada penelitian lain tentang robot manipulator juga telah dilakukan oleh Fadhili Rahman, Faridah, Andi Ikram Nur dan Andi Nadar Makkaraka “*Rancang Bangun Prototype Manipulator Lengan Robot Menggunakan Motor Servo Berbasis Mikrokontroler*” [4]. Pada penelitian ini mereka merancang bangun sebuah manipulator lengan robot dengan menggunakan empat buah motor servo dan dikendalikan atau dikontrol oleh potensiometer berbasis mikrokontroler.

Berdasarkan penelitian yang telah dilakukan oleh beberapa peneliti serta melihat kondisi pandemi saat ini, dimana proses produksi industri sedikit terhambat karena pada staff dan karyawan harus bekerja secara *remotely* maka mau tidak mau pekerja harus dirumahkan. Maka dengan perkembangan revolusi industri yang sedang terjadi saat ini, maka robot manipulator pun dapat dikendalikan secara otomatis dan secara *wireless* atau nirkabel dengan bantuan *Internet of Things* (IoT) dan dapat mengontrol robot manipulator tersebut secara otomatis dan *wireless* berkat penggunaan mikrokontroler salah satunya dengan NodeMCU ESP8266, serta kita juga dapat mengatur nilai sudut pergerakan, kecepatan putaran servo dan pergerakan robot manipulator dalam melakukan tugasnya.

Tujuan dari penelitian ini adalah untuk merancang dan menghasilkan sebuah *prototype automated manipulator robot* serta mengetahui kinerja dari *prototype automated manipulator robot* yang dapat dikontrol secara nirkabel menggunakan mikrokontroler NodeMCU ESP8266 berbasis teknologi *Internet of Things* (IoT)

Robot Manipulator

Perkembangan teknologi khususnya di bidang mekanika dan juga elektronika mendorong peningkatan teknologi yang pesat pada bidang robotika. Dengan kecanggihan teknologi yang berkembang saat ini, semakin banyak pekerjaan-pekerjaan manusia yang mulai tergantikan oleh robot, salah satunya robot manipulator [5]. Robot manipulator atau robot lengan adalah salah satu contoh dari sekian banyaknya perangkat mekatronika dan salah satu jenis robot yang sudah mulai banyak digunakan dalam kehidupan manusia khususnya dalam dunia industri [6]. Manipulator merupakan sebuah sistem mekanika yang digunakan untuk memanipulasi sebuah gerakan seperti mengangkat, mengambil, memindahkan, serta memanipulasi benda kerja untuk mempermudah dan meringankan pekerjaan manusia. Sebuah robot manipulator dirancang sedemikian rupa – menyerupai lengan manusia yang mempunyai sifat fleksibel supaya mudah digunakan untuk melakukan sebuah pekerjaan dengan cepat, presisi serta tingkat efisiensi yang lebih baik. Robot manipulator memiliki sistem kerja yang dilengkapi oleh aktuator dan memiliki *Degree of Freedom* (DOF) atau derajat kebebasan dengan jumlah tertentu agar mampu bekerja persis seperti lengan manusia [7]. Fungsi utama dari sebuah robot terutama robot manipulator dalam dunia industri saat ini adalah menggantikan tugas manusia karena robot tersebut mampu melakukan pekerjaan yang beras secara presisi, cepat, efektif dan aman [7].

Internet of Things (IoT)

Internet of Things adalah perkembangan ilmiah yang menjanjikan untuk mengoptimalkan kehidupan berdasarkan sensor cerdas dan perangkat pintar yang bekerja bersama melalui Internet. Dalam penerapannya, *Internet of Things* juga dapat secara otomatis mengidentifikasi, mencari, melacak, memantau, dan memicu peristiwa terkait secara *real-time* [8]. *Internet of Things* (IoT) juga merupakan metode yang bertujuan untuk memaksimalkan manfaat koneksi Internet dan mengirimkan serta memproses data atau informasi melalui Internet secara nirkabel, virtual, dan mandiri. Teknologi IoT terkoneksi ke berbagai terminal pengumpulan data melalui Internet dan jaringan komunikasi lainnya. Informasi tentang lingkungan objek ditangkap secara *real time*, diubah ke format data yang sesuai dikirim melalui jaringan lalu dikirim ke pusat data. Data tersebut kemudian diproses oleh prosesor cerdas menggunakan *cloud computing* dan teknologi komputasi pintar lainnya yang dapat memproses data dalam jumlah besar untuk mencapai tujuan dari IoT tersebut [9].

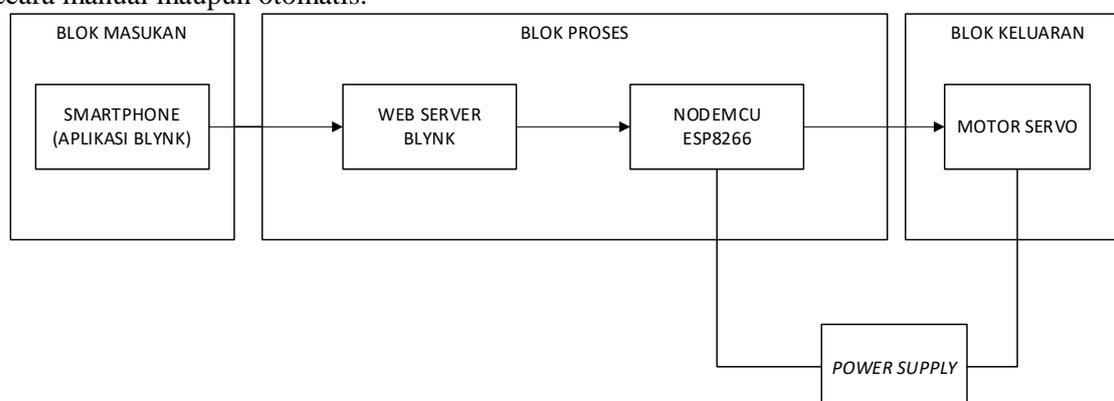
NodeMCU ESP8266

Mikrokontroler dapat dianalogikan sebagai sebuah personal komputer yang semua atau sebagian besar elemennya dikemas pada satu *chip* IC, dikenal dengan nama lain *single chip computer* [10]. NodeMCU adalah papan elektronik berbasis *chip* ESP8266, yang memiliki fungsi mikrokontroler dan berfungsi untuk melakukan koneksi Internet (WiFi) [11]. NodeMCU memiliki banyak pin I/O sehingga mikrokontroler ini dapat digunakan untuk mengembangkan sebuah sistem *monitoring* dan *controlling* untuk proyek *Internet of Things* (IoT) maupun proyek lokal.

METODE

Penelitian ini menggunakan metode penelitian *prototyping*. *Prototyping* adalah proses yang digunakan untuk membantu mengembangkan perangkat keras dan perangkat lunak berupa model perangkat keras dan perangkat lunak yang akan dirancang sebagai versi awal [12]. Metode ini dilakukan secara bertahap yaitu pertama mengembangkan prototipe sederhana kemudian mengembangkannya dari waktu ke waktu hingga prototipe dikembangkan dan jika sistem belum sempurna maka perlu dievaluasi kembali [13]. *Prototype* adalah bentuk dasar/prototipe dari suatu sistem atau subsistem dan mewakili model produk/alat yang akan dirancang [14]. Secara umum ada beberapa tahapan dalam pendekatan *prototyping* ini, yang pertama adalah tahap desain, pembuatan perangkat keras/lunak, dan terakhir adalah tahap evaluasi. Proses pertama dimulai pada fase desain. Pada tahap ini kegiatan yang dilakukan adalah pembuatan model atau prototype berdasarkan permasalahan yang ada. Fokusnya adalah pada desain antarmuka fungsional prototipe yang dimaksud. Langkah selanjutnya adalah membuat prototipe. Pada tahap ini, prototipe keseluruhan dari rencana pemecahan masalah dibuat. Proses selanjutnya adalah evaluasi. Langkah ini merupakan kegiatan evaluasi prototipe atau model yang dibuat. Jika ada bagian yang tidak sesuai dengan harapan, maka harus diubah. Prototipe dievaluasi oleh pengguna dan digunakan untuk memilih kebutuhan pengembangan perangkat lunak. Iterasi yang terjadi selama pembuatan prototipe memungkinkan pengembang untuk mendefinisikan dan menanggapi keinginan dan kebutuhan pengguna. Bagian terakhir berkaitan dengan hasil. Pada tahap ini merupakan hasil akhir dari prototype atau model yang telah dihasilkan sesuai dengan yang diinginkan. Langkah selanjutnya dalam pendekatan ini adalah merancang sistem aplikasi, yang melibatkan pembangunan sistem berdasarkan hasil dari langkah sebelumnya [15].

Perancangan penelitian ini berupa instrumen perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*) untuk mendukung perancangan *prototype* pada penelitian. Instrumen *hardware* yang digunakan dalam penelitian terdiri dari NodeMCU V3 ESP8266, motor servo MG996R, *Power Supply* 12V 20A, DC – DC *Buck Converter* dan LCD 16X2 I2C. Instrumen *software* terdiri dari *software* Arduino IDE untuk memprogram mikrokontroler dan aplikasi Blynk yang tersedia pada *Playstore* (Android) dan *App Store* (iOS) yang akan digunakan sebagai program untuk mengontrol secara IoT menggunakan *smartphone* dan mikrokontroler NodeMCU akan menerima perintah yang diberikan melalui aplikasi dan menjalankan/mengaktifkan robot manipulator. NodeMCU akan menjalankan program kendali robot manipulator yang dapat dikendalikan oleh *smartphone* pada jarak jauh dengan memanfaatkan modul ESP8266 baik secara manual maupun otomatis.



Gambar 1. Blok Diagram Sistem *Prototype Automated Manipulator Robot*

Gambar 1 merupakan blok diagram keseluruhan sistem *prototype* yang akan diimplementasikan pada perancangan *prototype* ini yang meliputi blok masukan, blok proses dan blok keluaran.

- Blok Masukan
Pada tahapan ini, Aplikasi Blynk sebagai masukan yang akan digunakan untuk mengendalikan dan mengatur pergerakan robot berdasarkan perintah yang diberikan *user*. Perintah dari *user* yang telah diberikan melalui *smartphone* pada aplikasi Blynk selanjutnya akan diproses dan dikirimkan ke *web server* Blynk lalu di transmisikan dan diterima oleh modul ESP8266. Selanjutnya, data yang diterima akan dikirimkan ke NodeMCU untuk diproses.
- Blok Proses
Data masukan yang diterima dari *web server* Blynk, selanjutnya akan diproses oleh NodeMCU ESP8266 untuk mengambil keputusan pengendalian sesuai dengan perintah yang diberikan oleh *user*. NodeMCU ESP8266 akan mengeksekusi perintah yang diberikan dengan mengirimkan data ke port I/O pada mikrokontroler.
- Blok Keluaran
Pada bagian keluaran ini adalah motor servo/robot akan menerima nilai/data yang diterima oleh NodeMCU ESP8266 melalui pin I/O sudah ditentukan dan robot akan bergerak sesuai sudut dan kecepatan yang telah ditentukan melalui masukan dari *user* melalui aplikasi yang nantinya akan bergerak secara manual ataupun otomatis.

A. Rancangan Perangkat Keras (*Hardware*)

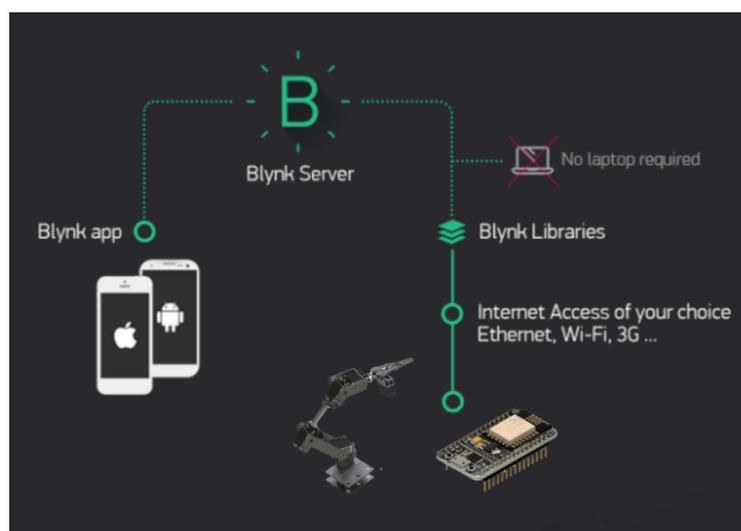
Pada perancangan perangkat keras (*hardware*) *prototype* ini, agar semua komponen dapat tersusun dengan rapi maka seluruh komponen disusun di atas *acrylic* dengan ukuran A2 42 cm x 60 cm dan A4 29,7 cm x 21 cm. Komponen-komponen tersebut terdiri dari; *servo base*, *servo waist*, *servo shoulder*, *servo elbow*, *servo wrist*, *servo end effector*, *port output motor servo*, *switch on off*, *power source*, *power supply switching DC*, *LCD 16x2 I2C*, *stiker desain info prototype dan barcode*, *dc – dc buck converter bracket*, *nodemcu esp8266 development board bracket*. Pembuatan dasar kerangka menggunakan *acrylic* dan setiap komponen dikencangkan dengan baut dan diberi *spacer* serta kaki meja untuk menopang *prototype*. Rancangan desain perangkat keras dapat dilihat pada Gambar 2.



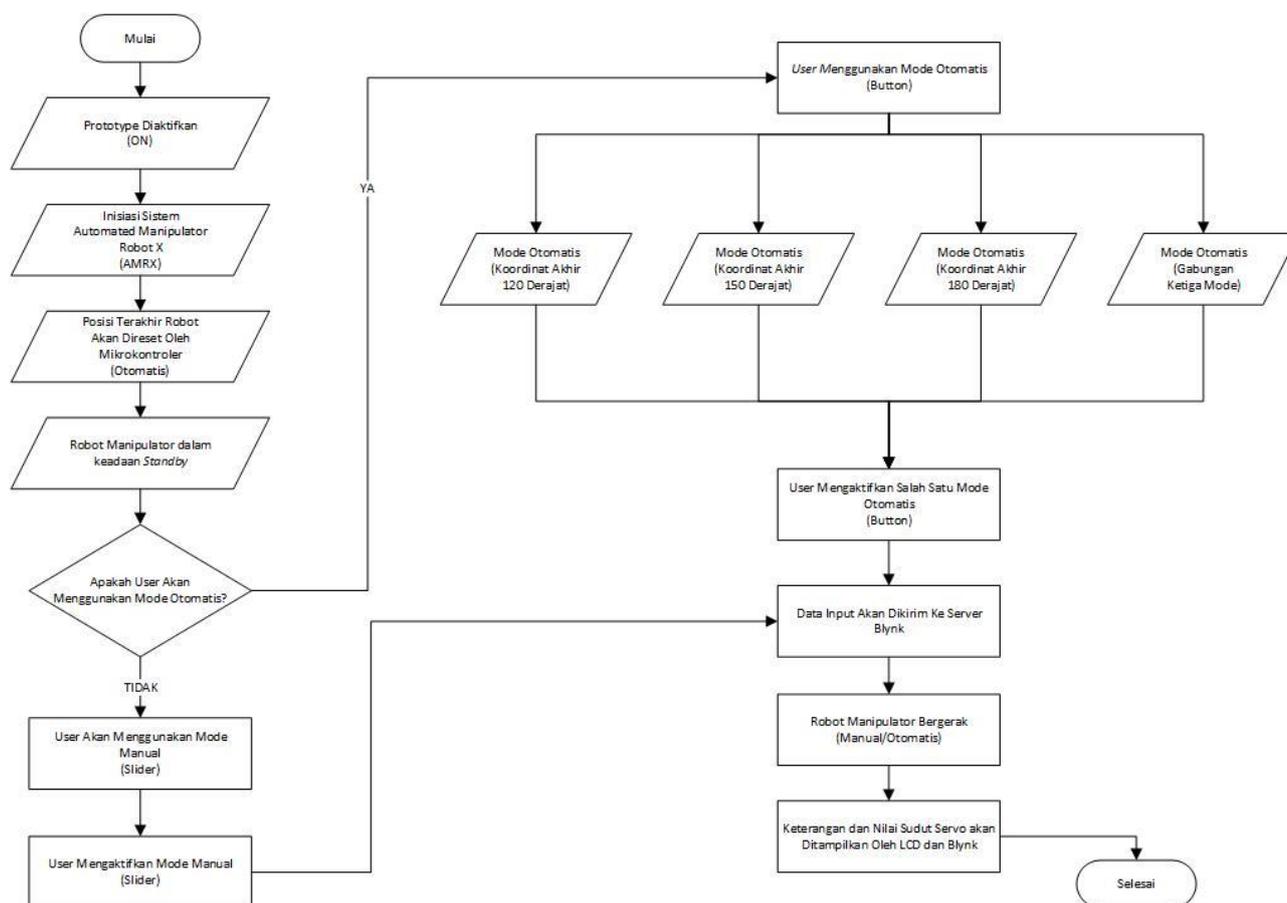
Gambar 2. Rancangan Desain *Prototype* & Desain Info *Prototype*

B. Rancangan Perangkat Lunak (*Software*)

Pada perancangan perangkat lunak (*software*) ini peneliti menggunakan dua buah *software* dan aplikasi yaitu Arduino IDE dan Blynk yang digunakan untuk memprogram mikrokontroler NodeMCU ESP8266 agar mampu mengendalikan motor servo MG996R menggunakan aplikasi Blynk dengan menerapkan konsep Internet of Things (IoT) dan dikendalikan secara *wireless* atau *remote*. Skema sistem kerja *prototype* dan *Flowchart* sistem kerja *prototype* ditunjukkan pada Gambar 3 dan 4.



Gambar 3. Skema Sistem Kerja *Prototype*



Gambar 4. *Flowchart* Sistem Kerja *Prototype*

Berdasarkan *flowchart* sistem kerja *prototype* pada Gambar 4 dapat dideskripsikan bahwa kerja *prototype* dimulai pada saat *prototype* tersebut diaktifkan dengan menghubungkan stop kontak ke terminal yang memiliki tegangan satu fase. Pada saat *prototype* mendapatkan tegangan dan *switch* ditekan ON, maka tegangan akan mengalir ke seluruh komponen sistem dari *prototype* robot manipulator. Lalu, NodeMCU ESP8266 sebagai mikrokontroler akan memproses dan mengeksekusi program yang telah di *upload* ke NodeMCU. Pada saat

program dijalankan, teks yang telah diprogram akan dimunculkan oleh LCD dan posisi terakhir dari robot manipulator akan secara otomatis di *reset* dengan nilai yang telah diatur pada program untuk posisi *standby*.

Pada saat mode *standby*, maka *user* dapat menggunakan robot manipulator yang mampu dikendalikan secara IoT tersebut dengan dua mode, yaitu manual dan otomatis. Apabila *user* memilih mode manual maka *user* harus menggunakan *widget slider* yang terdapat pada aplikasi Blynk. Setiap *widget slider* yang ada memiliki fungsi masing-masing untuk menggerakkan robot manipulator dengan konfigurasi motor servo yang alamatnya berbeda-beda, jika *user* menggeser *slider* servo 1 maka nilai *slider* akan dikirim ke server Blynk dan servo 1 akan bergerak sesuai dengan sudut yang diberikan, lalu nilai sudut servo akan ditampilkan pada LCD dan seterusnya. Namun, apabila *user* hendak memerintahkan robot manipulator secara otomatis maka *user* harus menggunakan *widget button*.

Pada aplikasi Blynk, peneliti membuat empat buah *button* untuk mode otomatis. Yaitu mode untuk titik koordinat akhir 120°, 150°, 180° dan gabungan ketiganya (sudut z). Pada saat *user* menekan salah satu tombol misalkan tombol untuk koordinat akhir 120°, maka nilai *widget button* tersebut akan dikirim ke server Blynk yang kemudian akan diterima dan diproses oleh NodeMCU. Lalu, NodeMCU akan mengeksekusi program untuk *button* 120° sesuai dengan nilai yang telah diatur. Notifikasi mode otomatis akan ditampilkan pada layar LCD.

HASIL DAN PEMBAHASAN

A. Perancangan Perangkat Keras (*Hardware*)

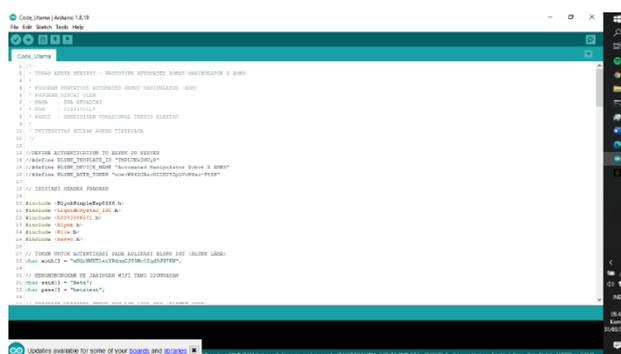
Perakitan prototype dimulai dengan membuat sebuah desain rangkaian elektronik dari hardware yang akan digunakan. Prototype ini menggunakan *power supply* DC dan DC-DC *Buck Converter Stepdown* untuk menggerakkan motor servo dan menhidupkan seluruh komponen yang terdapat dalam rangkaian hardware seperti NodeMCU atau LCD 16x2. Untuk pengontrolan motor servo, pin yang digunakan dalam penelitian ini yaitu D0, D1, D2, D3, D4 dan D5. Sedangkan, untuk input LCD menggunakan pin D6 dan D7. Untuk menempatkan semua komponen tersebut, peneliti menggunakan akrilik dengan ukuran diameter panjang 60 cm dan lebar 42 cm. Realisasi dari perancangan *prototype* dapat dilihat pada Gambar 5.



Gambar 5. Realisasi dari Rancangan *Prototype*

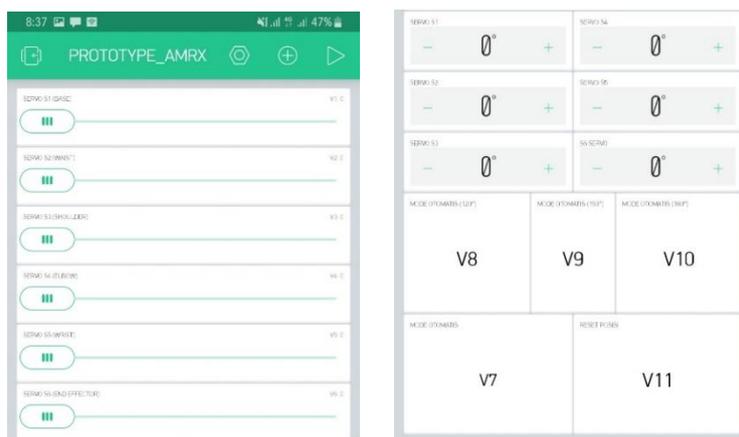
B. Perancangan Perangkat Lunak (*Software*)

Penulisan program untuk *prototype* ini dilakukan menggunakan *compiler* Arduino IDE. Pemrograman yang dilakukan yakni menuliskan beberapa *header* dan variabel-variabel input yang akan digunakan (menyesuaikan komponen yang dipakai) seperti *library* servo, *wire*, LCD, Blynk, dan ESP8266. Dalam hal ini, peneliti mengidentifikasi pin-pin apa saja yang digunakan, inisiasi hardware pada program, pengaturan output yang diinginkan seperti posisi servo/tampilan LCD. Proses pemrograman ditunjukkan pada Gambar 6.



Gambar 6. Realisasi Program *Prototype*

Setelah melakukan pemrograman dan menyesuaikannya dengan aplikasi yang digunakan, kemudian peneliti melakukan pendesainan *layout* aplikasi Blynk dengan memasukkan beberapa *widget* yang diperlukan untuk pengontrolan *prototype* berbasis IoT. Peneliti menggunakan *widget slider* horizontal, *numeric input*, dan *push button*. Kemudian alamat-alamat *input widget* tersebut disesuaikan dengan program yang telah disusun agar masukan yang diberikan melalui aplikasi Blynk sesuai dengan *input* yang di inisiasi pada program sehingga *output* yang diinginkan dapat terkirim sesuai dengan alamat yang diberikan dan nilai yang dikirim. Aplikasi Blynk dan program yang peneliti buat dapat diakses langsung melalui QR code.



Gambar 7. Realisasi Aplikasi Untuk Mengontrol *Prototype Automated Manipulator Robot*

Gambar 7 merupakan realisasi dari perancangan *software* berupa aplikasi Blynk. Adapun cara kerja aplikasi Blynk dalam mengontrol *prototype* ini yaitu pada saat pengguna ingin menggerakkan robot secara manual, pengguna tersebut dapat menggeser *widget slider* yang terdapat pada aplikasi atau dengan memasukkan nilai sudut pada *widget numeric input*. Maka pada saat nilai sudut diberikan servo/robot manipulator akan bergerak sesuai nilai yang diberikan. Dan apabila ingin menggunakan mode otomatis, maka pengguna tinggal menekan *widget button* yang tersedia dibawah *widget numeric input*.

Pengujian *Power Supply*

Pengujian tegangan pada *power supply* dilakukan untuk mengetahui seberapa besar tegangan yang dihasilkan dan seberapa akurat dengan agar tegangan yang dihasilkan sesuai dan tidak melebihi dari spesifikasi yang dibutuhkan. Tegangan output yang dibutuhkan oleh peneliti untuk *prototype* ini sebesar 12V DC. Hasil pengujian pengukuran tegangan *power supply* dapat dilihat pada Tabel 1.

Tabel 1. Hasil Pengukuran Tegangan Power Supply

Komponen	No.	Pengukuran	Vout	Vout Terukur	Selisih Pengukuran	Persentase Error
Power Supply 12V/20A	1	Tanpa Beban	AC 220V (Input)	222 V	2 V	0,91 %
			DC 12V (Output)	11,99 V	0,01 V	0,08 %
	2	Dengan Beban	AC 220V (Input)	222 V	2 V	0,91 %
			DC 12V (Output)	11,99 V	0,01 V	0,08 %

Pengujian DC – DC Buck Converter

Pengujian ini bertujuan untuk mengetahui tegangan output yang keluar dengan mengatur potensiometer agar output sesuai dengan yang diinginkan (setelah tegangan yang masuk dari *power supply* di *convert* oleh *DC – DC Buck Converter Step Down* agar tidak lebih dari 5 V). Hasil pengujian pengukuran tegangan *DC – DC Buck Converter* dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pengukuran Tegangan DC -DC Buck Converter Stepdown Power Supply

Komponen	No.	Pengukuran	Vout	Vout Terukur	Selisih Pengukuran	Persentase Error
DC – DC Buck Converter Stepdown	1	Tanpa Beban	DC 12V (Input)	11,99 V	0,01 V	0,08 %
			DC 5 V (Output)	5,01 V	0,01 V	0,20 %
	2	Dengan Beban	DC 12V (Input)	11,98 V	0,02 V	0,17 %
			DC 5 V (Output)	5,02 V	0,02 V	0,40 %

Pengujian Tegangan NodeMCU ESP8266

Pengujian tegangan pada mikrokontroler NodeMCU ESP8266 V3 ini bertujuan untuk mengetahui besarnya tegangan *output* yang diterima dari *power supply*. Dalam pengukuran tegangan ini dibagi menjadi dua fase, yakni saat kondisi tanpa beban dan dengan beban. Hasil pengujian pengukuran tegangan NodeMCU ESP8266 dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengukuran Tegangan NodeMCU ESP8266

Komponen	No.	Pengukuran	Vout	Vout Terukur	Selisih Pengukuran	Persentase Error
NodeMCU ESP8266	1	Tanpa Beban	DC 3,3 V	3,31 V	0,01 V	0,3 %
			DC 5 V	5,02 V	0,02 V	0,4 %
	2	Dengan Beban	DC 3,3 V	3,31 V	0,01 V	0,3 %
			DC 5 V	5,01 V	0,01 V	0,2 %

Pengujian Tegangan Motor Servo

Pengujian tegangan motor servo MG996R terbagi menjadi dua fase yakni saat motor servo 1 sampai 6 dalam kondisi 0/LOW (dalam kondisi nonaktif/tanpa tegangan) dan dalam kondisi 1/HIGH (dalam keadaan aktif/bertegangan). Pengujian ini dilakukan untuk mengetahui seberapa tegangan yang diterima oleh motor servo pada kondisi 0/LOW dan kondisi 1/HIGH agar tegangan yang diterima tidak melebihi spesifikasi yang diperlukan. Pengujian tegangan motor servo MG996R ini diukur dengan cara menghubungkan antara *output*

tegangan *power supply* dengan *ground* pada tegangan 5 V. Hasil pengujian pengukuran tegangan motor servo dapat dilihat pada Tabel 4.

Tabel 4. Hasil Pengukuran Tegangan Motor Servo

Servo	Kondisi Motor Servo	Tegangan Hasil Pengukuran (V)	Tegangan Pada Spesifikasi (V)	Eror %
1	0	0	0	0 %
	1	5,01	5,00	0,2%
2	0	0	0	0 %
	1	5,01	5,00	0,2%
3	0	0	0	0 %
	1	5,01	5,00	0,2%
4	0	0	0	0 %
	1	5,01	5,00	0,2%
5	0	0	0	0 %
	1	5,01	5,00	0,2%
6	0	0	0	0 %
	1	5,01	5,00	0,2%
Vrata-rata saat kondisi = 0			0 V	
Vrata-rata saat kondisi = 1			5,01 V	

Pengujian Jarak Kontrol *Prototype*

Pengujian jarak kontrol robot manipulator dilakukan dengan lima kategori jarak, yakni pada jarak 100 meter, 200 meter, 300 meter, 400 meter dan 500 meter. Pengujian ini dilakukan untuk mengetahui seberapa jauh robot manipulator dapat dikontrol melalui IoT dengan NodeMCU yang memiliki *chipset* WiFi ESP8266. Hasil pengujian jarak kontrol *prototype* dapat dilihat pada Tabel 5.

Tabel 5. Hasil Pengujian Jarak Kontrol *Prototype*

No.	Jarak (m)	Keterangan
1	100	Berjalan Baik
2	200	Berjalan Baik
3	300	Berjalan Baik
4	400	Berjalan Baik
5	500	Berjalan Baik

Pengujian Presisi Sudut Pergerakan *Prototype*

Pengujian presisi sudut pergerakan motor servo MG996R dilakukan dengan cara membandingkan sudut ideal yang diharapkan dengan sudut yang diukur dengan sebuah busur derajat. Semakin kecil selisih antara sudut hasil pengukuran (sudut robot) dengan sudut ideal maka semakin tinggi tingkat kepresisiannya. Hasil pengujian presisi sudut pergerakan *prototype* dapat dilihat pada Tabel 6.

Tabel 6. Hasil Pengujian Presisi Sudut Pergerakan *Prototype*

Link - n	Sudut Ideal			Sudut Robot			Selisih		
	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
Base	0°	90°	180°	0°	90,1°	180,1°	0°	0,1°	0,1°
Link - 1	0°	90°	180°	0°	90,7°	125,3°	0°	0,6°	5,3°
Link - 2	0°	90°	180°	0°	90,5°	180,5°	0°	0,5°	0,5°
Link - 3	0°	90°	180°	0°	90,2°	180,4°	0°	0,2°	0,4°
Link - 4	0°	90°	180°	0°	90,4°	180,1°	0°	0,4°	0,1°
End Effector	Tutup		Buka	Tutup		Buka	0		
	0 cm		5 cm	0 cm		5 cm			

Tabel 7 merupakan rentang pergeseran *slider* pengendali untuk sudut pergerakan motor servo agar perputaran sudut motor servo tidak melebihi sudut yang diperlukan.

Tabel 7. Rentang Pergeseran Sudut Slider Pengendali Servo

Slider servo – n	Rentang Pergeseran
Slider servo 1	0 – 180
Slider servo 2	30 – 120
Slider servo 3	0 – 180
Slider servo 4	0 – 180
Slider servo 5	0 – 180
Slider servo 6	0 – 60

Pengujian Waktu Tempuh Pergerakan Robot Manipulator

Pengujian waktu tempuh pergerakan robot manipulator dilakukan untuk mengetahui seberapa lama waktu pergerakan yang dibutuhkan oleh setiap motor servo dalam satu kali *sweep* (0 – 180) dan kembali ke posisi semula (180 – 0). Hasil pengujian waktu tempuh pergerakan *prototype automated manipulator robot* dapat dilihat pada Tabel 8.

Tabel 8. Hasil Pengujian Waktu Tempuh Pergerakan Prototype

Servo - n	Waktu tempuh (0 – 180)	Waktu tempuh (180 - 0)
Servo 1	0,78 detik	0,78 detik
Servo 2	0,83 detik	0,85 detik
Servo 3	0,72 detik	0,86 detik
Servo 4	0,86 detik	0,85 detik
Servo 5	0,73 detik	0,72 detik
Servo 6	0,53 detik	0,54 detik

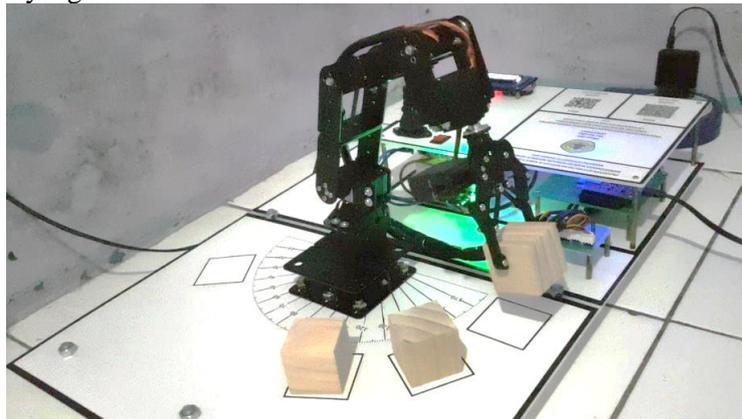
Tabel 9. Hasil Pengujian Akurasi Robot Manipulator Dalam Memindahkan Objek Benda

Percobaan Ke-	Posisi Awal Benda	Titik Koordinat Akhir Benda	Waktu Tempuh	Hasil Pengujian	Keterangan
1	30°	120°	14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
2			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
3			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
4			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
5			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
6			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
7			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
8	30°	150°	14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
9			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
10			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
11			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
12			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
13			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
14			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
15	30°	180°	14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)
			14 s	Berhasil	Benda Berhasil Dipindahkan (Sesuai Titik Koordinat Akhir)

Pengujian Keakurasian Robot

Pengujian keakurasian dilakukan untuk mengetahui seberapa akurat *prototype* robot manipulator dapat berfungsi dengan baik. Pengujian ini dilakukan dengan cara memindahkan objek benda dari posisi awal (30°) ke posisi akhir titik koordinat yang telah ditentukan (merujuk pada sudut istimewa 120° , 150° dan 180°). Hasil pengujian keakurasian *prototype* dapat dilihat pada Tabel 9.

Dari Tabel 9 menunjukkan bahwa dari percobaan yang telah dilakukan untuk mengetahui keakurasian *prototype* dalam memindahkan benda dari posisi awal ke posisi akhir (titik koordinat akhir benda) dan bagaimana performa serta kinerja robot saat difungsikan dengan mode otomatis, yakni robot manipulator bergerak dari titik awal 30° menuju titik posisi akhir yang telah ditentukan (120° , 150° dan 180°) robot manipulator mampu mengambil dan meletakkan objek benda dari posisi awal ke titik koordinat akhir objek sesuai dengan koordinat yang telah ditentukan.



Gambar 8. Proses Pengujian Akurasi Robot Manipulator Dalam Memindahkan Objek Benda

PENUTUP

Berdasarkan hasil yang diperoleh dari pengujian *Prototype Automated Manipulator Robot X Menggunakan Mikrokontroler Berbasis Internet of Things (IoT)* dapat ditarik kesimpulan bahwa *Prototype Automated Manipulator Robot X () Menggunakan Mikrokontroler NodeMCU ESP8266 Berbasis Internet of Things (IoT)* dapat berfungsi dengan baik. Persentase eror pengukuran dari pengujian tegangan di setiap komponen mendapatkan hasil yang sangat baik, tingkat eror di bawah 1% yang berarti semua komponen dapat bekerja dengan optimal dengan batas tegangan yang sudah diatur. Untuk pengujian kontrol robot manipulator, pengujian dilakukan mulai dari jarak 100 hingga 500 meter dan menunjukkan bahwa robot manipulator dapat dioperasikan dengan baik meskipun dalam jarak yang cukup jauh. Tingkat Presisi pergeseran sudut motor servo dengan sudut ideal kecil sehingga dapat dikatakan motor servo bergerak presisi. Dan untuk pengujian keakurasian *prototype* robot manipulator dengan mode otomatis yang telah dilakukan sebanyak 15 kali dengan posisi akhir objek benda yang berbeda-beda (120° , 150° dan 180° terhadap sumbu z) menunjukkan bahwa *prototype* dapat bergerak secara optimal dan efektif serta waktu tempuh yang dibutuhkan untuk setiap gerakan relatif konstan. Terdapat beberapa saran yang diharapkan dapat berguna untuk menyempurnakan *prototype* ini di masa yang akan datang, yaitu: desain robot manipulator yang lebih efisien dan optimal, penggunaan *konveyor* yang juga terhubung secara IoT, menggunakan metode yang berbeda seperti *inverse/forward kinematics*, menggunakan sensor yang dipadukan menggunakan AI bahkan *deep learning*. Agar *Prototype Automated Manipulator Robot* ini dapat menjadi solusi yang lebih nyata di dunia Industri dan sebagai sarana pendidikan.

DAFTAR PUSTAKA

- [1] D. Mikelsten, V. Teigens, and P. Skalfist, *Kecerdasan Buatan: Revolusi Industri Keempat*. Cambridge Stanford Books. [Online]. Available: https://books.google.co.id/books?id=tR3NDwAAQBAJ&printsec=frontcover&source=gbs_atb#v=onepage&q&f=false

- [2] D. Nahrowi, D. Aribowo, and M. A. Hamid, "Pengembangan Trainer Kit Mikrokontroler ATmega16 untuk Sekolah Menengah Kejuruan," *J. Pendidik. Teknol. dan Kejuruan*, vol. 17, no. 2, pp. 145–155, 2020, [Online]. Available: <https://ejournal.undiksha.ac.id/index.php/JPTK/article/view/24366>
- [3] Salmon and B. Harpad, "Prototype Lengan Robot Berbasis Arduino Dengan Menggunakan Algoritma Kinematics," *J. Ilm. Matrik*, vol. 22, no. 2 SE-Articles, Jul. 2020, doi: 10.33557/jurnalatrik.v22i2.960.
- [4] F. Rahman, F. Faridah, A. I. Nur, and A. N. Makkarak, "Rancang Bangun Prototipe Manipulator Lengan Robot Menggunakan Motor Servo Berbasis Mikrokontroler," *ILTEK J. Teknol.*, vol. 15, no. 01, pp. 42–46, 2020, doi: 10.47398/iltek.v15i01.508.
- [5] E. Sitompul and Sodri, "Prototipe Manipulator Lengan Robot Berbasis Arduino dengan Metoda Kendali Lead-Through," *J. Elem.*, vol. 6, no. 1, pp. 1–14, 2020.
- [6] Marsono, I. A. Elbaith, and A. Qolik, "Prototype Arm Robotic 6 Axis Untuk Menyiapkan Kompetensi Pemrograman Matakuliah Mekatronika Mahasiswa Prodi D3 Teknik Mesin," *J. Tek. Mesin dan Pembelajaran*, vol. 1, no. 2, p. 1, 2018, doi: 10.17977/um054v1i2p1-7.
- [7] A. Darmawan, N. S. Salahuddin, and M. Karjadi, "PROTOTIPE ALAT PEMINDAH BARANG DI PELABUHAN BERBASIS ARDUINO," *J. Ilm. Teknol. dan Rekayasa*, vol. 23, no. No. 2, pp. 103–111, 2018, doi: <https://doi.org/10.35760/tr.2018.v23i2.2460>.
- [8] A. Junaidi, "Internet Of Things, Sejarah, Teknologi Dan Penerapannya : Review," *J. Ilm. Teknol. Inf.*, vol. 1, no. 3, pp. 62–66, 2015.
- [9] M. Jamil and M. Said, "The Utilization of Internet of Things (IoT) for Multi Sensor Data Acquisition using Thingspeak," *VOLT J. Ilm. Pendidik. Tek. Elektro*, vol. 3, no. 1, p. 13, 2018, doi: 10.30870/volt.v3i1.1962.
- [10] D. Desmira and D. Aribowo, "Desain Alat Pendeteksi Kebocoran Gas Elpiji Menggunakan Mikrokontroler Atmega16," *VOLT J. Ilm. Pendidik. Tek. Elektro UNTIRTA*, vol. 1, no. 1, pp. 73–80, 2016.
- [11] N. Hidayati, L. Dewi, M. F. Rohmah, and S. Zahara, "Prototype Smart Home Dengan Modul NodeMCU ESP8266 Berbasis Internet of Things (IoT)," *Tek. Inform. Univ. Islam Majapahit*, pp. 1–9, 2018.
- [12] D. Purnomo, "Model Prototyping Pada Pengembangan Sistem Informasi," *JIMP - J. Inform. Merdeka Pasuruan*, vol. 2, no. 2, pp. 54–61, 2017, doi: 10.37438/jimp.v2i2.67.
- [13] N. Renaningtias and D. Apriliani, "Penerapan Metode Prototype Pada Pengembangan Sistem Informasi Tugas Akhir Mahasiswa," *Rekursif J. Inform.*, vol. 9, no. 1, 2021, doi: 10.33369/rekursif.v9i1.15772.
- [14] R. C. Noor Santi, "Perancangan Interaksi Pengguna (User Interaction Design) Menggunakan Metode Prototyping," *J. Tek. Inform.*, vol. 9, no. 2, pp. 108–113, 2018, doi: 10.15408/jti.v9i2.5599.
- [15] N. Putri, N. Agung Prabowo, and R. A. Widyanto, "Implementasi Metode Prototyping pada Perancangan Aplikasi Electronic Ticket (E-Ticket) berbasis Android," *J. Komtika (Komputasi dan Inform.)*, vol. 3, no. 2, pp. 62–68, 2020, doi: 10.31603/komtika.v3i2.3474.

Biodata Penulis

Eka Revadiaz. Lahir di Tangerang, 17 Maret 1999. Mahasiswa S1 di Program Studi Pendidikan Vokasional Teknik Elektro FKIP Universitas Sultan Ageng Tirtayasa tahun 2017.

Mohammad Fatkhurrohman. Lahir di Kebumen, 05 April 1989. Menyelesaikan (S1) Pendidikan Teknik Elektro Universitas Negeri Yogyakarta pada tahun 2012, menyelesaikan S1 Teknik Informatika di STMIK Elrahma Yogyakarta pada tahun 2013, serta menyelesaikan Magister Pendidikan Pascasarjana (S2) Pendidikan Teknologi dan Kejuruan – Pendidikan Teknik Elektro Universitas Negeri Yogyakarta pada tahun 2015. Sejak tahun 2015 hingga sekarang menjadi staff pengajar tetap di Program Studi Pendidikan Vokasional Teknik Elektro FKIP Universitas Sultan Ageng Tirtayasa, Kota Serang – Banten.

Didik Aribowo, lahir di Jambi, 15 Februari 1982. Menyelesaikan S1 pada jurusan Teknik Elektro Universitas Muhammadiyah Yogyakarta dan menyelesaikan Pendidikan Pascasarjana S2 Magister Teknik Elektro Institut Teknologi Sepuluh November Surabaya. Sejak tahun 2008 – 2014 menjadi staff pengajar di Jurusan Teknik Elektro FT Universitas Sultan Ageng Tirtayasa. Dan sejak 2014 hingga sekarang menjadi staff pengajar tetap di Program Studi Pendidikan Vokasional Teknik Elektro FKIP Universitas Sultan Ageng Tirtayasa, Kota Serang – Banten.