

Muhamad Hilmansyah Susanta<sup>1</sup>, Nana Sujana<sup>2</sup>

## Implementasi Koneksi *Internet of Things (IoT)* Pada *Arduino* Menggunakan *Module ESP8266*

Muhamad Hilmansyah Susanta<sup>1</sup>, Nana sujana<sup>2</sup>

Teknologi Komputer dan Informatika, Politeknik Pajajaran ICB Bandung Indonesia 40192

[muhamad.hilmansyah@poljan.ac.id](mailto:muhamad.hilmansyah@poljan.ac.id), [nana.sujana@poljan.ac.id](mailto:nana.sujana@poljan.ac.id)

### **Abstract**

*This study aims to implement an Internet of Things (IoT) system using an Arduino Uno integrated with an ESP8266 module as a network communication interface. The system is designed to enable remote control of electronic devices, particularly an LED, through an internet-based application. In this system, Arduino functions as the main data processing unit, while the ESP8266 module serves as a bridge connecting the system to a WiFi network. The research method includes hardware design, software development, and system testing, which covers WiFi connectivity, serial communication, output control, and system response. The testing results show that the ESP8266 module can connect to the WiFi network consistently with a 100% success rate and an average connection time of approximately 4.9 seconds. In addition, data communication between Arduino and ESP8266 operates properly, allowing the system to respond to user commands accurately and in real time. Based on these results, the developed system demonstrates stable and responsive performance, making it suitable for educational purposes as well as basic IoT implementation using Arduino.*

**Keywords:** *Internet of Things (IoT); Arduino Uno; Module ESP8266, WiFi*

### **Abstrak**

Penelitian ini bertujuan untuk mengimplementasikan sistem *Internet of Things (IoT)* dengan memanfaatkan *Arduino Uno* yang diintegrasikan dengan *Module ESP8266* sebagai media komunikasi berbasis jaringan *internet*. Sistem dirancang untuk memungkinkan pengendalian perangkat elektronik, khususnya lampu *LED*, secara jarak jauh melalui *aplikasi* berbasis *internet*. *Arduino* berfungsi sebagai pengolah data utama, sedangkan *ESP8266* digunakan sebagai penghubung antara sistem dengan jaringan *WiFi*. Metode yang digunakan meliputi perancangan perangkat keras, pengembangan perangkat lunak, serta pengujian sistem secara bertahap yang mencakup koneksi *WiFi*, komunikasi *serial*, kontrol *output*, dan respons sistem. Hasil pengujian menunjukkan bahwa *Module ESP8266* mampu terhubung dengan jaringan *WiFi* secara konsisten dengan tingkat keberhasilan mencapai 100% dan waktu koneksi rata-rata sekitar 4,9 detik. Selain itu, komunikasi data antara *Arduino* dan *ESP8266* berjalan dengan baik, sehingga sistem dapat merespons perintah pengguna secara cepat dan akurat. Berdasarkan hasil tersebut, sistem yang dikembangkan dapat bekerja secara stabil dan responsif, serta layak digunakan sebagai media pembelajaran maupun implementasi dasar dalam pengembangan teknologi *IoT* berbasis *Arduino*.

**Kata kunci :** *Internet of Things (IoT), Arduino Uno, Module ESP 8266; WiFi*

**Corresponding author :** [muhamad.hilmansyah@poljan.ac.id](mailto:muhamad.hilmansyah@poljan.ac.id)

## PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi pada era *digital* telah mendorong munculnya berbagai inovasi, salah satunya adalah *Internet of Things (IoT)*. *IoT* merupakan konsep yang memungkinkan perangkat fisik saling terhubung melalui jaringan *internet* untuk bertukar data secara otomatis tanpa intervensi manusia secara langsung (Suryadi & Prasetyo, 2018; Kurniawan & Setiawan, 2020). Teknologi ini telah banyak diterapkan dalam berbagai bidang seperti industri, pendidikan, dan sistem *smart home* karena kemampuannya dalam meningkatkan efisiensi dan efektivitas sistem (Wibisono & Nugroho, 2021).

Dalam pengembangan sistem *IoT*, *mikrokontroler* memiliki peran penting sebagai pusat kendali perangkat. *Arduino* merupakan salah satu *platform mikrokontroler* yang banyak digunakan karena kemudahan penggunaan dan fleksibilitasnya dalam berbagai *aplikasi* (Kadir, 2017). Namun, *Arduino* memiliki keterbatasan dalam hal konektivitas *internet* sehingga memerlukan *module* tambahan agar dapat terhubung ke jaringan *IoT*.

Salah satu *module* yang *mikrokontroler* digunakan adalah *ESP8266*, yaitu *module WiFi* yang mampu menghubungkan *mikrokontroler* ke jaringan *internet*. Penelitian oleh Prabowo et al. (2023) menunjukkan bahwa penggunaan *ESP8266* pada sistem *Arduino* mampu meningkatkan efisiensi komunikasi data secara *real-time*. Selain itu, penelitian oleh Putra et al. (2022) menunjukkan bahwa integrasi *sensor* dengan *ESP8266* dapat digunakan untuk sistem *monitoring* jarak jauh secara efektif.

Penelitian lain oleh Hidayat dan Rahman (2021) menyatakan bahwa implementasi *IoT* berbasis *Arduino* dan *ESP8266* dapat meningkatkan kinerja sistem kontrol secara signifikan. Saputra dan

Wibowo (2020) juga menunjukkan bahwa sistem berbasis *IoT* menggunakan *ESP8266* memiliki tingkat keandalan yang baik dalam pengiriman data.

Berdasarkan uraian tersebut, penelitian ini bertujuan untuk mengimplementasikan koneksi *Internet of Things (IoT)* pada *Arduino* menggunakan *ESP8266* sehingga sistem mampu melakukan komunikasi data secara *real-time* melalui jaringan *internet*.

## LANDASAN TEORI

### *Arduino Uno*

*Arduino* merupakan *platform open-source* berbasis *mikrokontroler* yang dirancang untuk memudahkan pengguna dalam mengembangkan sistem elektronik interaktif. *Platform* ini terdiri dari perangkat keras (*hardware*) berupa papan *mikrokontroler* dan perangkat lunak (*software*) berupa *Arduino IDE* yang digunakan untuk menulis, mengompilasi, dan mengunggah program ke dalam *board Arduino* (Kadir, 2017).

*Arduino* banyak digunakan dalam pengembangan sistem *embedded* karena memiliki kemudahan dalam pemrograman serta dukungan komunitas yang luas. Bahasa pemrograman yang digunakan pada *Arduino* merupakan turunan dari bahasa *C/C++* yang telah disederhanakan sehingga mudah dipahami oleh pemula maupun peneliti (Kurniawan & Setiawan, 2020).

Salah satu jenis *Arduino* yang paling populer adalah *Arduino Uno* yang menggunakan *mikrokontroler ATmega328P*. *Board* ini memiliki 14 *pin digital input/output*, 6 *pin analog input*, serta mendukung komunikasi *serial*, *SPI*, dan *I2C* yang memungkinkan integrasi dengan berbagai *sensor* dan *module* tambahan (Wibisono & Nugroho, 2021).

Dalam implementasi *Internet of Things (IoT)*, *Arduino* sering digunakan sebagai pengendali utama yang bertugas membaca data dari *sensor* dan mengontrol aktuator. Namun, *Arduino* memiliki keterbatasan dalam konektivitas jaringan sehingga memerlukan *module* tambahan seperti *ESP8266* untuk dapat terhubung ke *internet* (Suryadi & Prasetyo, 2018)



Gambar 1. *Arduino Uno*

### Module ESP 8266

*ESP8266* merupakan *module WiFi* yang berfungsi untuk memberikan kemampuan konektivitas *internet* pada perangkat *mikrokontroler* seperti *Arduino*. *Module* ini dikembangkan oleh Espressif Systems dan menjadi salah satu komponen utama dalam pengembangan sistem *IoT* karena harganya yang terjangkau dan kemampuannya yang cukup tinggi (Hidayat & Rahman, 2021).

*ESP8266* memiliki *prosesor* internal dan memori yang memungkinkan *module* ini tidak hanya digunakan sebagai perangkat tambahan, tetapi juga dapat berfungsi sebagai *mikrokontroler* mandiri. *Module* ini mendukung komunikasi melalui protokol *TCP/IP* sehingga dapat digunakan untuk mengirim dan menerima data melalui jaringan *internet* (Saputra & Wibowo, 2020).

Salah satu tipe *ESP8266* yang umum digunakan adalah *ESP-01* dan *NodeMCU*. *ESP-01* biasanya digunakan sebagai *module* tambahan untuk *Arduino*, sedangkan *NodeMCU* sudah dilengkapi dengan *USB*

*interface* sehingga dapat diProgram secara langsung tanpa menggunakan *mikrokontroler* tambahan (Prabowo et al., 2023).

Dalam implementasinya, *ESP8266* digunakan untuk menghubungkan sistem ke *platform IoT* seperti Blynk atau *web server* sehingga memungkinkan pengguna untuk melakukan *monitoring* dan kontrol perangkat secara jarak jauh melalui *smartphone* atau komputer (Putra et al., 2022).



Gambar 2. *Module ESP8266*

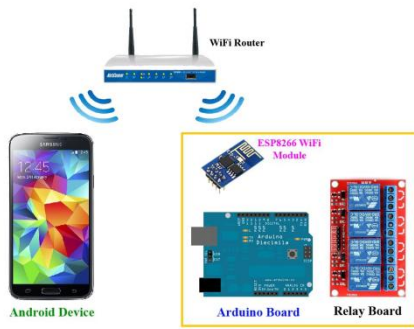
## METHODOLOGI

### Arsitektur sistem

Sistem yang dirancang merupakan sistem *Internet of Things (IoT)* yang memungkinkan pengguna mengontrol lampu *LED* dari jarak jauh menggunakan *smartphone*. Sistem ini terdiri dari beberapa komponen utama, yaitu *Arduino* sebagai pengendali utama, *Module ESP8266* sebagai penghubung ke jaringan *internet*, serta *aplikasi mobile* sebagai antarmuka pengguna.

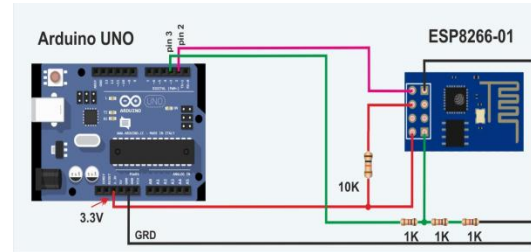
Alur kerja sistem dimulai dari pengguna yang memberikan perintah melalui *aplikasi* pada *smartphone*. Perintah tersebut dikirimkan melalui jaringan *internet* ke *Module ESP8266*. Selanjutnya, *ESP8266* meneruskan data ke *Arduino* untuk diProses. *Arduino* kemudian mengontrol kondisi *LED* (menyala atau mati) sesuai dengan perintah yang diterima.

Sebaliknya, sistem juga dapat mengirimkan informasi status *LED* kembali ke pengguna melalui jaringan *internet* secara *real-time*. Dengan demikian, sistem bersifat dua arah (*bidirectional*).



Gambar 3. Arsitektur Sistem

Arduino dan ESP8266, sehingga mendukung implementasi sistem berbasis *Internet of Things (IoT)* seperti kontrol perangkat jarak jauh dan *monitoring* data secara *real-time*.



Gambar 5. Rangkaian Sistem

### Rangkaian Sistem

Rangkaian sistem ini mengintegrasikan *Arduino Uno* dengan *Module ESP8266* (ESP-01) untuk memungkinkan komunikasi data melalui jaringan *WiFi*. *Arduino Uno* berfungsi sebagai pengendali utama yang memproses data, sedangkan *ESP8266* bertindak sebagai *module* komunikasi yang menghubungkan sistem ke *internet*.

Komunikasi antara *Arduino* dan *ESP8266* menggunakan antarmuka *serial (UART)*, yaitu melalui *pin TX (transmit)* dan *RX (receive)*. Data yang dikirimkan dari *Arduino* akan diteruskan ke *ESP8266* untuk dikirim ke jaringan *internet*, begitu pula sebaliknya.

Dalam perancangan rangkaian, perlu diperhatikan bahwa *ESP8266* bekerja pada tegangan 3.3V, sehingga tidak dapat langsung dihubungkan dengan *pin 5V* dari *Arduino*. Oleh karena itu, digunakan rangkaian pembagi tegangan pada jalur *TX Arduino* ke *RX ESP8266* untuk menurunkan tegangan agar sesuai dengan spesifikasi *module*.

Selain itu, *pin CH\_PD (ChIP Enable)* harus dihubungkan ke tegangan 3.3V agar *Module ESP8266* dapat aktif. *Pin GPIO0* juga dihubungkan ke 3.3V untuk memastikan *module* berada pada mode operasi normal.

Dengan konfigurasi ini, sistem dapat melakukan pertukaran data secara stabil antara

Tabel 1. Konfigurasi *Pin Arduino Uno* dan *Module ESP 8266*

<i>Pin Arduino Uno</i>	<i>Pin module ESP 8266</i>
3.3V	Vcc
GND	GND
D2	Output
3.3V	CH_PD
3.3V	GPIO0
RX(Pin0)	TX
TX(pin1)	RX
TX → 1kΩ → 2kΩ → GND	RX ESP8266

### Cara Kerja Sistem

Proses kerja sistem dimulai dari tahap Mulai (Start), yaitu ketika seluruh perangkat seperti *Arduino Uno* dan *Module ESP8266* diaktifkan. Pada tahap ini, sistem akan melakukan inisialisasi, di mana *Arduino* menyiapkan program yang akan dijalankan, sedangkan *ESP8266* mempersiapkan koneksi komunikasi dengan *Arduino* serta konfigurasi jaringan *WiFi*.

Setelah proses inisialisasi selesai, *Module ESP8266* akan mencoba melakukan koneksi ke jaringan *WiFi* menggunakan *SSID* dan *password* yang telah ditentukan

sebelumnya. Pada tahap ini terdapat proses pengambilan keputusan, yaitu apakah *ESP8266* berhasil terhubung ke jaringan atau tidak.

Jika tidak terhubung, maka sistem akan mengulangi proses koneksi hingga berhasil. Namun, jika berhasil terhubung, maka *ESP8266* akan memperoleh alamat *IP* sebagai identitas pada jaringan *internet*.

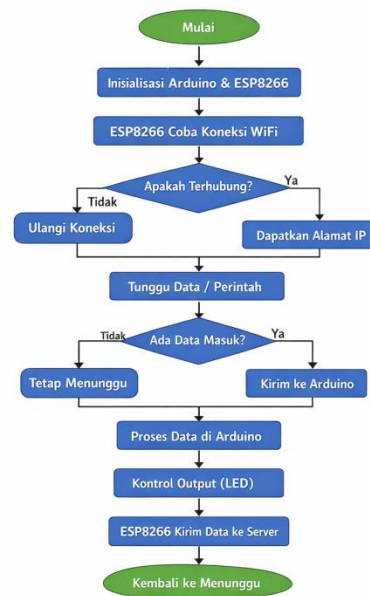
Setelah koneksi berhasil, sistem masuk ke tahap menunggu data atau perintah dari pengguna atau *server*. Pada kondisi ini, sistem akan terus memonitor apakah terdapat data yang masuk melalui jaringan *internet*.

Jika tidak ada data masuk, maka sistem akan tetap berada pada kondisi menunggu. Namun, jika terdapat data masuk, maka data tersebut akan diterima oleh *ESP8266* dan kemudian dikirimkan ke *Arduino* melalui komunikasi *serial*.

Selanjutnya, *Arduino* akan melakukan pemrosesan data untuk menentukan tindakan yang harus dilakukan. Berdasarkan hasil pemrosesan tersebut, *Arduino* akan mengontrol *output*, misalnya menyalakan atau mematikan *LED* sesuai dengan perintah yang diterima.

Setelah *output* dijalankan, *Arduino* akan mengirimkan status atau data hasil eksekusi kembali ke *ESP8266*. *Module ESP8266* kemudian akan mengirimkan data tersebut ke *server* atau *aplikasi* pengguna melalui jaringan *internet*.

Tahap terakhir adalah sistem kembali ke kondisi menunggu (*loop*), di mana seluruh proses akan berulang secara terus-menerus selama sistem masih aktif. Dengan demikian, sistem mampu bekerja secara *real-time* dan responsif terhadap setiap perintah yang diberikan oleh pengguna.



Gambar 6. Flowchart

## PENGUJIAN DAN ANALISA

### Pengujian Sistem

Pengujian sistem dilakukan dalam beberapa tahap untuk memastikan seluruh komponen bekerja dengan baik. Pengujian koneksi *WiFi* bertujuan untuk memastikan *Module ESP8266* dapat terhubung ke jaringan yang tersedia dengan mengamati keberhasilan koneksi serta waktu yang dibutuhkan hingga terhubung. Selanjutnya, pengujian komunikasi *serial* dilakukan untuk memastikan pertukaran data antara *Arduino* dan *ESP8266* melalui antarmuka *UART* berjalan dengan baik, di mana data yang dikirim berupa perintah ON dan OFF. Pengujian kontrol *LED* bertujuan untuk memastikan bahwa *LED* dapat merespons perintah dari *aplikasi* dengan benar, baik dalam kondisi menyala maupun mati. Selain itu, dilakukan juga pengujian respons sistem untuk mengukur waktu yang dibutuhkan sejak perintah dikirim oleh pengguna hingga *LED* memberikan respon,

sehingga dapat diketahui tingkat kecepatan dan kinerja sistem secara keseluruhan.

### Hasil Pengujian

Berikut Tabel Hasil Pengujian pada Penelitian ini.

Tabel .3 Hasil Pengujian Koneksi antara *Arduino Uno* dengan jaringan *Internet*

No	Percobaan	Status	Waktu (detik)
1	Uji 1	Berhasil	5
2	Uji 2	Berhasil	4
3	Uji 3	Berhasil	6
4	Uji 4	Berhasil	5
5	Uji 5	Berhasil	4
6	Uji 6	Berhasil	5
7	Uji 7	Berhasil	6
8	Uji 8	Berhasil	5
9	Uji 9	Berhasil	4
10	Uji 10	Berhasil	5
Rata-Rata			4.9

Berdasarkan hasil pengujian koneksi *WiFi* yang telah dilakukan sebanyak 10 kali percobaan, diperoleh bahwa seluruh percobaan menunjukkan status koneksi berhasil, sehingga tingkat keberhasilan sistem mencapai 100%. Waktu yang dibutuhkan untuk terhubung ke jaringan *WiFi* bervariasi antara 4 hingga 6 detik, dengan nilai rata-rata sebesar 4,9 detik. Variasi waktu koneksi ini dipengaruhi oleh kondisi kestabilan jaringan dan proses inisialisasi *Module ESP8266* saat melakukan koneksi ke *access point*. Secara keseluruhan, hasil ini menunjukkan bahwa sistem yang dirancang memiliki performa koneksi yang stabil, konsisten, dan responsif, sehingga layak digunakan dalam implementasi *Internet of Things (IoT)* berbasis *Arduino* dan *Module ESP8266*.

### KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian alat maka dapat diambil kesimpulan sebagai berikut :

1. Implementasi *Internet of Things (IoT)* menggunakan *Arduino Uno* dan *Module ESP8266* berhasil dilakukan dengan baik.
2. Sistem mampu terhubung ke jaringan *WiFi* dengan tingkat keberhasilan mencapai 100% dan rata-rata waktu koneksi sekitar 4,9 detik.
3. Komunikasi data antara *Arduino* dan *ESP8266* melalui antarmuka *serial (UART)* berjalan dengan lancar tanpa kendala.
4. Sistem kontrol *LED* dapat bekerja sesuai perintah ON dan OFF dari *aplikasi* dengan respon yang cepat.
5. Secara keseluruhan, sistem yang dirancang memiliki kinerja yang stabil, responsif, dan layak digunakan sebagai media pembelajaran *IoT* dasar.

Beberapa saran dan masukan untuk pengembangan dari alat ini diantaranya :

1. Perlu ditambahkan sistem keamanan seperti autentikasi atau enkripsi data untuk meningkatkan keamanan jaringan.
2. Penggunaan *platform* berbasis *cloud* dapat dikembangkan agar sistem lebih fleksibel dan *scalable*.
3. Sistem dapat dikembangkan lebih lanjut dengan menambahkan *sensor* untuk fungsi *monitoring* selain kontrol.
4. Disarankan menggunakan *Module ESP8266* versi terbaru atau yang lebih stabil untuk meningkatkan performa koneksi.
5. Perlu dilakukan pengujian lebih lanjut pada berbagai kondisi jaringan untuk mendapatkan hasil yang lebih optimal dan akurat.

## DAFTAR PUSTAKA

- Hidayat, T., & Rahman, A. (2021). Implementasi *Internet of Things* berbasis *Arduino* menggunakan *Module ESP8266*. *Jurnal Informatika*, 9(1), 15–22.
- Saputra, E., & Wibowo, S. (2020). Sistem *monitoring* suhu berbasis *IoT* menggunakan *Arduino* dan *ESP8266*. *Jurnal Teknologi dan Sistem Komputer*, 8(3), 120–126.
- Prabowo, N., Sari, D., & Nugraha, A. (2023). Implementasi sistem kontrol berbasis *mikrokontroler Arduino Uno*. *Jurnal Energi Elektrik*, 12(1), 45–50.
- Putra, R., Sari, D., & Nugraha, A. (2022). Implementasi *sensor BH1750* pada sistem *monitoring* berbasis *mikrokontroler*. *Jurnal Teknologi Elektro*, 10(2), 30–36.
- Firmansyah, R., & Hidayat, D. (2019). Perancangan sistem kendali lampu berbasis *IoT* menggunakan *ESP8266*. *Jurnal Teknik Elektro Indonesia*, 11(2), 85–92.
- Suryadi, A., & Prasetyo, B. (2018). Implementasi sistem *monitoring* berbasis *Internet of Things* pada perangkat *mikrokontroler*. *Jurnal Ilmiah Teknologi Informasi*, 7(1), 25–30.
- Kurniawan, D., & Setiawan, A. (2020). Pengembangan sistem *IoT* untuk *monitoring* lingkungan berbasis *Arduino*. *Jurnal Teknologi Informasi dan Komunikasi*, 9(2), 55–62.
- Kadir, A. (2017). *Pemrograman Arduino*. Yogyakarta: Andi.
- Sugiyono. (2018). *Metode penelitian kombinasi (mixed methods)*. Bandung: Alfabeta.
- Wibisono, G., & Nugroho, Y. (2021). Implementasi *Module ESP8266* pada

sistem *smart home* berbasis *Arduino*. *Jurnal Teknik Komputer*, 10(1), 1–8.