

## Rancangan *Monitoring* Ketersediaan Data Radar Menggunakan ESP8266 Berbasis Aplikasi Telegram

Raihan Muhammad Farid<sup>1</sup>, Iqbal Aurelio Tawakal<sup>2</sup>, Feti Fatonah<sup>3</sup>

<sup>1</sup>Politeknik Penerbangan Palembang, Palembang, Indonesia

<sup>2,3</sup>Teknik Navigasi Udara, Politeknik Penerbangan Indonesia Curug, Tangerang, Indonesia

E-mail: [iqbalaurelio@gmail.com](mailto:iqbalaurelio@gmail.com)

### Abstrak

Radar memiliki peran yang sangat penting dalam dunia penerbangan yaitu sebagai alat pengamatan (*surveillance*), data yang keluar dari radar masuk ke peralatan automasi kemudian diproses agar dapat menampilkan target pada *SDD* (*surveillance data display*) yang digunakan pemandu lalulintas penerbangan untuk mengontrol pesawat. Sehingga data radar harus terus aktif agar dapat menampilkan informasi pada layar *SDD*. Jika data radar tersebut mati, maka target tidak dapat tampil pada *SDD*, hal ini dapat menyulitkan pemandu lalulintas penerbangan dalam mengontrol pesawat. Oleh karena itu, penulis memiliki sebuah ide untuk merancang sebuah alat yang digunakan untuk memonitoring ketersediaan data radar menggunakan ESP8266 berbasis aplikasi Telegram. Input yang masuk ke rancangan ini berasal dari data radar yang diambil dari *splitter*. Rancangan ini juga menggunakan sensor INA219 yang dapat memonitoring tegangan dan arus data radar. Rancangan *monitoring* ketersediaan data radar ini diharapkan dapat memudahkan teknisi dalam memantau status data radar dimanapun dan kapanpun dengan menggunakan *handphone* melalui aplikasi Telegram.

**Kata Kunci:** radar, data radar, *SDD*, *monitoring*, Telegram

### Pendahuluan

Peran radar dalam dunia penerbangan adalah sebagai alat pengamatan (*surveillance*) yang digunakan oleh pemandu lalulintas penerbangan untuk mengarahkan pesawat agar sampai ke tempat tujuannya dengan selamat dan menghindari terjadinya tabrakan antar pesawat di udara. Pemanduan lalu lintas penerbangan di bandara besar cukup sibuk, sehingga peralatan radar di bandara besar harus beroperasi selama 24 jam *non-stop*, oleh karena itu teknisi harus menjaga peralatan radar agar beroperasi dengan normal. Hal ini dikarenakan berkaitan dengan ketersediaan (*availability*) sebagai standart performa pelayanan lalulintas penerbangan (Mervin C. Budge & Shawn R. German, 2015).

Data yang keluar dari radar akan digabungkan dengan data dari ADS-B dan diolah pada peralatan automasi. Hasil dari olahan data tersebut ditampilkan di *surveillance data display* (*SDD*). Tampilan dari *SDD* ini yang akan digunakan oleh pemandu lalulintas udara untuk mengarahkan pesawat ke tempat tujuannya. Oleh karena itu data radar yang keluar dari radar dan masuk ke automasi ini harus tetap aktif agar tidak terjadi *miss*.

Kondisi yang terjadi pada data radar yang masuk ke automasi yaitu ketika target dari radar hilang, maka seorang pemandu lalulintas penerbangan (*user*) akan melapor ke teknisi, kemudian teknisi akan melapor kepada penyedia layanan satelit untuk pengiriman data radar dari daerah lain yang dimana pada bandara menggunakan jasa penyedia layanan satelit untuk melihat apakah ada permasalahan dari media pengiriman data radar. Jika tidak ada permasalahan, maka pihak penyedia layanan satelit akan melaporkannya kembali kepada teknisi untuk mencari permasalahan lebih lanjut, hal ini membuat teknisi harus menunggu informasi dari pihak penyedia layanan satelit dan dapat memakan waktu.

Adapun rancangan *monitoring* dengan menggunakan ESP8266 yang pernah dibuat sebelumnya antara lain (Gusti et al., n.d.) tentang "Monitoring Penggunaan Daya Listrik Sebagai Implementasi Internet Of Things Berbasis Esp8266", kemudian (Roihan et al., n.d.) tentang "Monitoring Kebocoran Gas menggunakan Mikrokontroler Arduino Uno dan ESP8266 berbasis Internet of Thing, dan juga (Mitha Djaksana et al., 2020) tentang "Perancangan Sistem Monitoring dan Kontroling Penggunaan Daya Arus Listrik Berbasis Android".

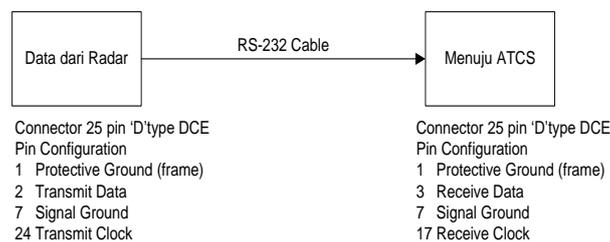
Oleh karena itu, Penulis memiliki ide untuk membuat sebuah alat yang dapat memonitoring ketersediaan data radar yang bisa di akses dengan melalui *handphone* dengan menggunakan aplikasi Telegram, sehingga bila terjadi masalah yang serupa maka teknisi bisa langsung mengetahui apakah yang bermasalah berasal dari data radar yang tidak masuk atau ada permasalahan didalam automasinya tanpa menunggu jawaban dari pihak penyedia layanan satelit. Hal ini dapat mempersingkat waktu dalam menyelesaikan masalah yang terjadi dan meningkatkan ketersediaan (*availability*) dari peralatan tersebut.

## Landasan Teori

### 1. MSSR Mode S

*Monopulse Secondary Surveillance Radar (MSSR) Mode S* merupakan sebuah fasilitas pengamatan penerbangan yang digunakan untuk mendeteksi pesawat terbang yang dipasang pada posisi tertentu di sekitar lingkungan Bandar Udara di dalam/di luar sesuai fungsinya. Prinsip kerja *secondary* Radar adalah dengan mengirimkan pulsa interogasi kepada transponder yang sudah terpasang di pesawat. Pulsa interogasi ini berisikan informasi identifikasi dan ketinggian pesawat, kemudian transponder mengirimkan pulsa balasan (reply) yang sinkron dengan pulsa interogasi (DJPU, n.d.).

Data dari radar ditransmisikan secara simplex dari *ouput* Radar menuju peralatan ATCS (*Air Traffic Controller System*). Komunikasi data yang digunakan yaitu komunikasi serial aliran sinkronus dengan *clock* yang berasal dari sumber (*radar site*). Setiap komunikasi berisikan dua sinyal, yaitu data dan *clock* (ICAO, 2020). Untuk mengirimkan data dari radar menuju ATCS, maka diperlukan sebuah koneksi *point to point* seperti pada gambar berikut ini :



**Gambar 1. Koneksi Data Radar**

### 2. Komunikasi Serial

Komunikasi serial merupakan komunikasi data dengan mengirimkan data secara satu per satu pada waktu tertentu, sehingga bentuk komunikasi ini lebih lambat dibandingkan komunikasi parallel seperti pada *port* printer yang dapat mengirimkan delapan bit sekaligus dalam sekali detik. Komunikasi data serial sinkronus, maksudnya adalah *clock* dikirimkan bersama dengan data serial, tetapi *clock* tersebut dibangkitkan sendiri baik dari sisi pengirim maupun penerima.

Protokol komunikasi serial yang paling umum adalah RS232, RS485, RS422, USB, dan *Ethernet*. RS-232 merupakan standar komunikasi serial yang didefinisikan sebagai antarmuka antara perangkat terminal data (Data Terminal Equipment/DTE) dan perangkat komunikasi data (Data Communications Equipment/DCE) menggunakan pertukaran data biner secara serial. Didalam definisi tersebut, DTE adalah perangkat computer dan DCE sebagai modem walaupun pada kenyataannya tidak semua produk antarmuka adalah DCE yang sesungguhnya.

Setiap perangkat kabel akan mengirim dan menerima sinyalnya sendiri karena RS-232 dikategorikan sebagai Data Terminal Equipment (DTE) atau Data Communication Equipment (DCE). Biasanya ada konektor 25 pin pada kabel (ada juga konektor 9 pin). Standar menetapkan bahwa modem memiliki konektor perempuan dengan fungsi pin DCE, sedangkan terminal dan komputer memiliki konektor laki-laki dengan tujuan pin DTE (Nasser, 2010).

### 3. ESP8266 D1 mini

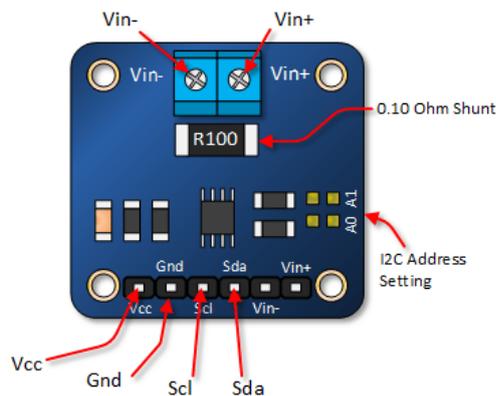
ESP 8266 D1 mini merupakan sebuah perangkat modul wifi *microcontroller* berbasis *chip* ESP 8266. Kelebihan dari *Microcontroller* jenis ini yaitu memiliki bentuk yang kecil dan harganya yang murah. ESP 8266 D1 mini memiliki fungsi yang tidak jauh berbeda dengan Arduino yang fungsinya sebagai mikrokontroler open source dan sering digunakan dalam pembuatan aplikasi Internet of Things (IoT). Chipset ESP 8266 merupakan sebuah chip *microcontroller* yang memiliki fitur wifi, sehingga memungkinkan *microcontroller* ini dapat terhubung ke jaringan wifi (Gusti et al., n.d.).



**Gambar 1. ESP8266 d1 mini**

**4. INA 219 Sensor**

INA 219 sensor merupakan sebuah alat yang digunakan untuk mengukur dua parameter sekaligus, yaitu tegangan dan arus listrik. INA 219 Voltage sensor dapat di program melalui aplikasi Arduino IDE (lady ada, 2022). SDA dan SCL adalah pin untuk protokol I2C. Pin ini mengacu pada Serial Data dan Serial Clock. Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didisain khusus untuk mengirim maupun menerima data.



**Gambar 2. INA 219**

**5. Buzzer**

*Buzzer* adalah sebuah komponen elektronika yang dapat menghasilkan getaran suara dalam bentuk gelombang bunyi (Ahmad Rizaldy et al., n.d.). *Buzzer* sering digunakan karena memiliki bentuknya yang kecil dan penggunaan daya yang minim. Cara kerja dari *buzzer* sendiri yaitu ketika suatu aliran listrik mengalir pada rangkaian *buzzer*, maka terjadi pergerakan mekanis pada *buzzer* tersebut yang akan menghasilkan suara.



**Gambar 3. Buzzer**

**6. Aplikasi Telegram**

Sebelum munculnya smartphone, Telegram banyak digunakan. Di masa lalu, telegram adalah layanan yang disediakan oleh kantor pos untuk mengirim pesan tertulis yang mendesak melalui jarak yang jauh. Namun, karena teknologi maju dengan cepat, fasilitas ini menjadi usang dan tidak lagi diperlukan. Kini, sebuah startup

yang membuat aplikasi dengan nama Telegram telah mengklaimnya. Layanan pesan instan berbasis cloud bernama Telegram mengutamakan kecepatan dan keamanan (Fitriansyah, n.d.). Telegram dirancang untuk memudahkan pengguna saling berkirim pesan teks, audio, video, gambar dan sticker dengan aman. Secara default, semua data yang diangkut akan menggunakan enkripsi standar industri. Oleh karena itu, pesan yang terkirim sepenuhnya aman dari pihak ketiga. Selain teks, foto, dan video, Telegram dapat digunakan untuk berbagi kontak dari perangkat orang lain, musik, file zip, lokasi realtime, dokumen, dan lainnya.

#### 7. Arduino IDE

Arduino merupakan sebuah platform dari *physical computing* yang bersifat *open source*. Perlu dipahami bahwa kata “platform” di sini merupakan sebuah pilihan kata yang tepat. Arduino bukan hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari hardware, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah software yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan mengupload ke dalam *memory microcontroller* (Hakiki et al., 2020). Konfigurasi pada Arduino IDE adalah hal utama untuk menjalankan program, karena Arduino IDE dapat menampilkan hasil atau error pada kode program sebelum dimasukkan ke mikrokontroler dan dapat memberikan hasil yang baik dan jelas.

### Metode Penelitian

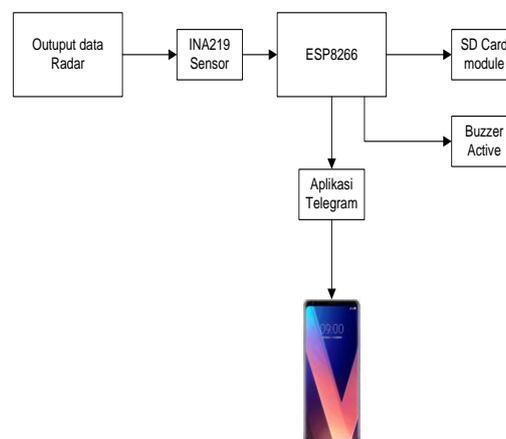
Metode R&D (Research and Development) digunakan dalam penelitian ini sebagai metode penelitian. Metode penelitian dan pengembangan, kadang-kadang dikenal sebagai R&D dalam bahasa Inggris, adalah teknik penelitian yang digunakan untuk membuat produk tertentu dan mengevaluasi keefektifannya. (Amiles & Rennes, 2015) mendefinisikan penelitian dan pengembangan (R&D) sebagai metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut. Berdasarkan definisi yang diberikan di atas, dapat ditarik kesimpulan bahwa metode R&D adalah suatu teknik penelitian yang digunakan untuk menciptakan produk tertentu dan menyempurnakan produk yang sesuai dengan standar dan kriteria produk yang dihasilkan melalui berbagai tahapan dari validasi atau pengujian. Untuk mengumpulkan data yang dibutuhkan, peneliti melakukan penelitian terlebih dahulu. Pembuatan sistem diuji dan dievaluasi setelah pengembangan selesai.

### Metode Pengumpulan Data

Dalam memperoleh data yang dibutuhkan, penulis menggunakan metode dalam pengumpulan datanya yaitu Studi pustaka dan Observasi, Metode studi pustaka dilakukan dengan memanfaatkan buku referensi dan database online untuk mengumpulkan data dan bahan yang akan digunakan dalam penyusunan rancangan. Observasi digunakan untuk mengumpulkan rincian teknis mendasar tentang komponen elektronik yang akan dipilih untuk rangkaian perangkat keras. Sebagai bagian dari kebutuhan sistem, penulis juga menggunakan pendekatan observasi untuk menemukan informasi perangkat lunak.

### Pembahasan

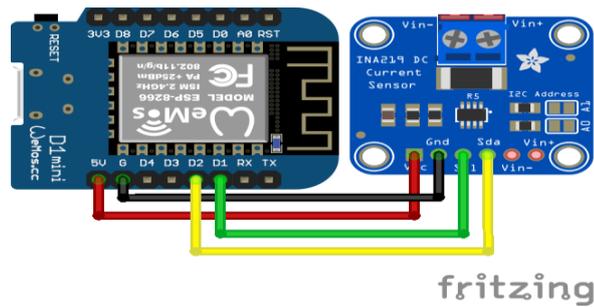
- A. Perancangan Alat
  - 1. Blok Diagram



**Gambar 4. Gambaran umum perancangan**

Dari gambar 4, ditunjukkan bahwa data radar yang keluar dari output data radar yaitu dari *splitter* data radar masuk kedalam INA219 *sensor* agar diketahui besar arus dan tegangan yang lewat, kemudian masuk ke ESP8266 yang telah dirangkai dengan *buzzer active* sebagai indicator suara bila terjadi alarm dan *SD Card module* untuk menyimpan rekaman hasil *monitoring* ketersediaan data radar. ESP8266 yang tersambung dengan *wifi* diprogram agar dapat memonitoring ketersediaan data radar dan memberikan notifikasi bila ada data radar yang mati melalui Aplikasi Telegram yang dapat diakses melalui *smartphone* oleh para teknisi.

2. Perancangan Input sensor



**Gambar 5. Koneksi ESP8266 dan INA219**

Pada gambar 5, sensor INA219 berfungsi untuk mendeteksi tegangan dan arus yang masuk yang dapat diprogram melalui aplikasi Arduino IDE. Untuk mengetahui besar tegangan yang dimonitoring yaitu dengan cara memberi masukan pada Vinput, kemudian memberikan tegangan sebesar +5V (VCC) yang berfungsi untuk mengaktifkan sensor, lalu menghubungkan pin SCL dan SDA ke pin yang ada pada microcontroller.



**Gambar 6. Susunan seluruh komponen yang digunakan**

Pada gambar 6, penulis merangkai seluruh komponen yang digunakan dalam perancangan ini yaitu INA219, ESP8266, Power Supply, Buzzer, dan LED.

3. Perancangan Pengolahan Input dan Output.



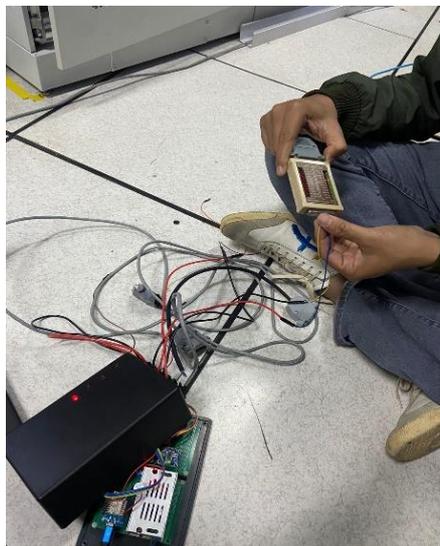
```
1 #include "TFT.h"
2 #include <Wire.h>
3 #include <Adafruit_GFX.h>
4 #include <SPI.h>
5 #include <RTC.h>
6
7 #define Buzzer
8
9 String ssid = "R001-R001-000000";
10 String pass = "1234567890123456";
11 String token = "1110405314d131919148u0T0W6L3C3A-1080u0yP?";
12
13 const int chipSelect = 14;
14
15 #define TFT_2M0120_000 (0x80);
16 #define TFT_2M0120_000000 (0x01);
17 #define TFT_2M0120_000000 (0x04);
18 #define TFT_2M0120_000000 (0x08);
19
20 String RadarState = "Aktif";
21 String RadarStateBaru = "Aktif";
22 String RadarStateLama = "Aktif";
23 String RadarStateBaru = "Aktif";
24
25 String dataRadar;
26 String dataRadarBaru;
27 String dataRadarLama;
28 String dataRadarBaru;
29
```

**Gambar 7. Pembuatan source code untuk menjalankan program**

Pada gambar 7 ini merupakan bagian pengolahan input dan output, untuk menjalankan rancangan ini sesuai dengan perintah yang diinginkan maka penulis membuat *source code* pada aplikasi Arduino IDE kemudian mengunggahnya kedalam board ESP8266. Apabila data radar mati, maka buzzer akan berbunyi dan juga ESP8266 akan memberikan notifikasi alarm melalui aplikasi Telegram.

**A. Uji coba Alat**

Uji coba Alat dilakukan untuk mengetahui apakah rancangan telah berjalan sesuai dengan hasil yang diinginkan. Penulis melakukan ujicoba secara langsung menggunakan data radar yang diambil dari *splitter* data radar, hasilnya adalah Rancangan monitoring ketersediaan data radar menggunakan ESP266 berbasis aplikasi Telegram dapat memberikan notifikasi ke aplikasi Telegram ketika data radar mati (tidak aktif) dan ketika data radar muncul kembali (aktif). Rancangan ini juga dapat melakukan pembacaan tegangan, arus, dan daya pada data radar yang dapat ditampilkan pada aplikasi Telegram dan menyimpan hasilnya secara otomatis pada kartu microSD. Penulis juga melengkapi rancangan ini dengan *buzzer* yang akan berbunyi ketika data radar tidak aktif dan akan mati ketika data radar aktif kembali.



**Gambar 8. Pengujian alat secara langsung**



**Gambar 9. Notifikasi otomatis pada Aplikasi Telegram**



**Gambar 10. Tampilan hasil monitoring data radar**

## Kesimpulan

Berdasarkan pada teori dan pembahasan yang telah diuraikan tentang rancangan monitoring ketersediaan data radar menggunakan ESP8266 berbasis Aplikasi Telegram, maka penulis menarik kesimpulan sebagai berikut :

1. Rancangan ini hanya dapat memonitoring ada tidaknya data radar serial dan clock yang keluar dari splitter data radar.
2. Rancangan ini dapat memudahkan teknisi untuk mengetahui bila terjadinya data radar yang hilang secara tiba-tiba melalui notifikasi otomatis telegram, sehingga teknisi dapat mengambil tindakan secepatnya.

## Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Airnav Medan yang telah memberi dukungan yang membantu pelaksanaan penelitian dan atau penulisan artikel.

## Daftar Pustaka

- Ahmad Rizaldy, Feti Fatonah, & Bambang Wijaya Putra. (n.d.). *RANCANGAN ALAT MONITORING PERUBAHAN SUHU PADA SHELTER VOR (VHF OMNIDIRECTIONAL RANGE), SHELTER GLIDE PATH DAN SHELTER LOCALIZER BERBASIS SMS GATEWAY DI BANDAR UDARA INTERNASIONAL HANG NADIM BATAM*. Amiles, & Rennes. (2015). *Research and Development (R&D)*.  
DJP. (n.d.). *PERATURAN DIREKTUR JENDERAL PERHUBUNGAN UDARA NOMOR : KP 077 TAHUN 2018*.

- Fitriansyah, F. (n.d.). *Penggunaan Telegram Sebagai Media Komunikasi Dalam Pembelajaran Online*.  
<https://doi.org/10.31294/jc.v20i2>
- Gusti, I., Mastawan, P., Putra, E., Ayu, I., & Giriantari, D. (n.d.). Monitoring Penggunaan Daya Listrik Sebagai Implementasi Internet of Things Berbasis Wireless Sensor Network. *Teknologi Elektro*, 16(03).
- Hakiki, M. I., Darusalam, U., & Nathasia, N. D. (2020). Konfigurasi Arduino IDE Untuk Monitoring Pendeteksi Suhu dan Kelembapan Pada Ruang Data Center Menggunakan Sensor DHT11. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 4(1), 150. <https://doi.org/10.30865/mib.v4i1.1876>
- ICAO. (2020). *PRELIMINARY SYSTEM INTERFACE CONTROL DOCUMENT FOR THE INTERCONNECTION OF ACC CENTERS OF THE CARSAM*.
- lady ada. (2022). Adafruit INA219 Current Sensor Breakout. *Adafruit Industries*, 1–28.  
<https://learn.adafruit.com/adafruit-ina219-current-sensor-breakout>
- Mervin C. Budge, & Shawn R. German. (2015). *Basic Radar Analysis*.
- Mitha Djaksana, Y., Eresha Jl Raya Puspiptek, S., Serpong, K., & Tangerang Selatan, K. (2020). PERANCANGAN SISTEM MONITORING DAN KONTROLING PENGGUNAAN DAYA LISTRIK BERBASIS ANDROID. *Jurnal Sistem Informasi Dan Teknologi Informasi*, 2(3), 13–24.
- Nasser, G. A. (2010). Pengendali Pintu Rumah Via Internet. *Jetc*, 5, 711–722.
- Roihan, A., Permana, A., & Mila, D. (n.d.). *MONITORING KEBOCORAN GAS MENGGUNAKAN MIKROKONTROLER ARDUINO UNO dan ESP8266 BERBASIS INTERNET OF THINGS*.