

**RANCANG BANGUN PERANGKAT MONITOR RADIASI GAMMA
BERBASIS MIKROKONTROLER ESP32 MENGGUNAKAN DETEKTOR
GEIGER-MÜLLER UNTUK PEMANTAUAN RADIASI DAN
LINGKUNGAN TERPADU**

SKRIPSI

Oleh:
RATNA RAHAYU
210604110069



**PROGRAM STUDI FISIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

**RANCANG BANGUN PERANGKAT MONITOR RADIASI GAMMA
BERBASIS MIKROKONTROLER ESP32 MENGGUNAKAN DETEKTOR
GEIGER-MÜLLER UNTUK PEMANTAUAN RADIASI DAN
LINGKUNGAN TERPADU**

SKRIPSI

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi salah satu persyaratan dalam
Memperoleh gelar Sarjana Sains (S.Si)**

**Oleh:
RATNA RAHAYU
210604110069**

**PROGRAM STUDI FISIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

HALAMAN PERSETUJUAN

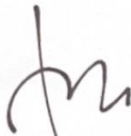
RANCANG BANGUN PERANGKAT MONITOR RADIASI GAMMA
BERBASIS MIKROKONTROLER ESP32 MENGGUNAKAN DETEKTOR
GEIGER-MÜLLER UNTUK PEMANTAUAN RADIASI DAN
LINGKUNGAN TERPADU

SKRIPSI

Oleh :
Ratna Rahayu
NIM. 210604110069

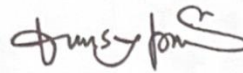
Telah Diperiksa dan Disetujui untuk Diuji
Pada Tanggal, 16 Juni 2025

Pembimbing I,



Farid Samsu Hananto M.T
NIP. 197405132003121001

Pembimbing II,



Mubasyroh, S.S., M.Pd.I
NIP. 197905022023212024

Mengetahui,
Ketua Program Studi



Dr. Ammar Tazi, M.Si
NIP. 19740730 200312 1 002

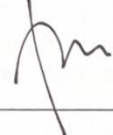
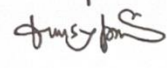
HALAMAN PENGESAHAN

RANCANG BANGUN PERANGKAT MONITOR RADIASI GAMMA
BERBASIS MIKROKONTROLER ESP32 MENGGUNAKAN DETEKTOR
GEIGER-MÜLLER UNTUK PEMANTAUAN RADIASI DAN LINGKUNGAN
TERPADU

SKRIPSI

Oleh:
Ratna Rahayu
NIM. 210604110069

Telah Dipertahankan di Depan Dewan Penguji
dan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Sains (S.Si)
Pada Tanggal, 24 Juni 2025

Penguji Utama	<u>Dr. Imam Tazi, M.Si</u> NIP. 19740730200312002	
Ketua Penguji	<u>Dr. Erna Hastuti, M.si</u> NIP. 198111192008012009	
Sekretaris Penguji	<u>Farid Samsu Hananto, M.T</u> NIP. 197405132003121001	
Anggota Penguji	<u>Mubasyiroh, S.S., M.Pd.I</u> NIP. 197905022023212024	



PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini :

Nama : RATNA RAHAYU

NIM : 210604110069

Jurusan : FISIKA

Fakultas : SAINS DAN TEKNOLOGI

Judul Penelitian : Rancang Bangun Perangkat Monitor Radiasi Gamma Berbasis Mikrokontroler ESP32 Menggunakan Detektor *Geiger-Müller* Untuk Pemantauan Radiasi dan Lingkungan Terpadu

Menyatakan dengan sebenarnya-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang tertulis dikutip dalam naskah ini serta disebutkan dalam sumber kutipan dan daftar pustaka. Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur jiplakan maka saya bersedia untuk mempertanggungjawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 25 Juni 2025
Yang Membuat Pernyataan



Katna Rahayu
NIM. 210604110069

MOTTO

"Tuhan tidak akan membebani seseorang melainkan sesuai dengan kesanggupannya." (QS. Al-Baqarah: 286)

HALAMAN PERSEMBAHAN

Segala puji dan syukur hanya milik Allah *Subhanahu wa Ta'ala*, yang telah memberikan Rahmat, hidayah, serta kekuatan, sehingga skripsi ini dapat terselesaikan dengan baik. Sholawat serta salam senantiasa tercurah kepada Nabi Muhammad *Shallallahu 'Alaihi Wasallam*, yang telah menjadi teladan hidup bagi umat manusia.

Skripsi ini penulis persembahkan dengan penuh rasa hormat dan terima kasih kepada:

1. Allah SWT yang Maha Pengasih lagi Maha Penyayang, segala proses ini hanya mungkin terjadi karena kasih dan kehendak-Mu. Dalam tiap sesak napas dan doa yang lirih, dalam gelap malam yang sepi dan Langkah yang nyaris menyerah, Engkau tidak pernah meninggalkanku. Terima kasih atas kekuatan, ketabahan, dan cahaya yang selalu Kau limpahkan, meski penulis sering lupa cara bersyukur.
2. Kedua orang tua ku tersayang, Bapak Misno dan Ibu Budiliatin. Terima kasih penulis ucapkan atas segala pengorbanan dan ketulusan yang diberikan hingga saat ini. Meskipun Bapak dan Ibu tidak sempat merasakan pendidikan hingga di bangku perkuliahan, namun selalu senantiasa memberikan yang terbaik untuk anak-anak nya, tak kenal lelah mendoakan, mengusahakan, memberikan dukungan baik secara moral maupun finansial, serta memprioritaskan pendidikan dan kebahagiaan anak-anak nya. Semoga dengan adanya skripsi ini dapat membuat Bapak dan Ibu lebih bangga karena telah berhasil menjadikan anak perempuan nya ini menyandang gelar sarjana seperti yang diharapkan. Besar harapan penulis semoga Bapak dan Ibu selalu

sehat, panjang umur, dan bisa menyaksikan keberhasilan lainnya yang penulis akan raih di masa yang akan datang.

3. Kakakku, Mas Chandra, terimakasih atas perhatian dan dukungan yang selalu hadir, meski tidak selalu dalam bentuk ucapan. Kehadiran dan semangat yang Mas berikan secara tidak langsung menjadi dorongan besar bagi penulis dalam menyelesaikan studi ini.
4. Seluruh keluarga besar, baik dari pihak ayah maupun ibu, dari kakek, nenek, paman, bibi, hingga para sepupu yang selalu memberikan semangat, perhatian, dan doa yang begitu berarti. Terima Kasih atas kebersamaan, kekeluargaan, dan doa yang menguatkan penulis.
5. Untuk penulis sendiri, terima kasih telah bertahan, terus berproses, dan tidak menyerah meski perjalanan tidak selalu mudah, dengan menguras banyak nya air mata dan kegelisahan, yang tetap berdiri meski sempat jatuh berkali-kali. Ini adalah bukti bahwa usaha tidak akan mengkhianati hasil, dan bahwa kegigihan akan selalu menemukan jalannya.

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Allah SWT, berkat rahmat dan karunia-Nya akhirnya penulis dapat menyelesaikan proposal skripsi berjudul “Rancang Bangun Perangkat Monitor Radiasi Gamma Berbasis Mikrokontroler ESP32 Menggunakan Detektor *Geiger-Müller* Untuk Pemantauan Radiasi dan Lingkungan Terpadu” sholat serta salam tetap tercurahkan kepada baginda tercinta Rasulullah Muhammad SAW yang telah memberikan kita jalan yang terang benderang yakni Addinul islam waliman.

Dalam kesempatan ini, penulis mengucapkan terimakasih kepada Bapak Farid Samsu Hananto dan Ibu Mubasyiroh, selaku dosen pembimbing yang telah memberikan motivasi, bimbingan serta arahan kepada penulis selama proses penyusunan proposal skripsi ini. Selain itu dukungan dari orang tua tercinta saya bapak Misno dan ibu Budiliatin yang selalu mendo’akan setiap langkah saya serta sahabat-sahabat saya telah membantu baik secara langsung maupun tidak langsung dalam menyelesaikan penulisan skripsi ini.

Penulis mengakui bahwa skripsi ini kurang dari cukup. Oleh karena itu, penulis dengan rendah hati meminta maaf jika ada kesalahan atau kekurangan membangun akan digunakan untuk perbaikan dimasa mendatang. Semoga Allah SWT memberikan rahmat dan hidayat kepada kita semua. Sehingga dengan penuh keremdhahan hati dan rasa hormat penulis mengucapkan terimakasih sebesar-besarnya kepada :

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.

2. Prof. Dr. Hj. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. Imam Tazi, M.Si., selaku Ketua Program Studi Fisika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
4. Bapak Farid Samsu Hananto M.T, sebagai dosen pembimbing yang senantiasa memberikan ilmu pengetahuan, inovasi dan meluangkan waktunya untuk membimbing penulis selama proses penyusunan skripsi.
5. Ibu Mubasyiroh, S.S.,M.Pd.I, sebagai dosen pembimbing kedua yang senantiasa menenangkan penulis, memberikan arahan, memberikan ilmu pengetahuan, dan membimbing penulis hingga skripsi ini berjalan dengan lancar dan baik.
6. Bapak Umar Sahiful Hidayat, M.Eng., selaku Pembimbing Luar yang senantiasa memberikan arahan, ilmu dan masukan selama di BRIN Yogyakarta.
7. Bapak Ikhsan Shobari, M.Eng., selaku Pembimbing Perancangan di BRIN Yogyakarta yang senantiasa memberikan ilmu, masukan, arahan, serta bimbingan dalam menyelesaikan laporan ini.
8. Bapak Dr. Imam Tazi, M.Si., selaku ketua penguji skripsi.
9. Ibu Dr. Erna Hastuti, M.Si selaku anggota penguji skripsi.
10. Seluruh Dosen Fisika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang, yang telah memberikan ilmu, membimbing dan memberikan pengarahan selama perkuliahan.

11. Kedua orang tua ku tersayang, Bapak Misno dan Ibu Budiliatin. Terima kasih penulis ucapkan atas segala pengorbanan dan ketulusan yang diberikan hingga saat ini.
12. Teman-teman seperjuangan Fisika 2021, yang senantiasa memberikan semangat, bantuan dan kebersamaan selama proses ini.
13. Teman-Teman *BFF*, yaitu Siti Maylia, Salma Qurrotu, Nasywa Zafirah, Chaerunnisa, yang selalu mendukung penulis dalam keadaan apapun, selalu membersamai penulis, dan selalu membantu penulis.
14. Teman *cucumber*, Zara Azaria Suyuti dan Amelya Tanjung, yang selalu menanyakan keadaan penulis, yang selalu menemani penulis, dan tidak pernah meninggalkan penulis dalam keadaan apapun.
15. Teman-Teman Kita Elins Brin, Rahma Wury Aprilliya, Asola Rahmabillah, Amelya Tanjung, yang membantu penulis selama pengambilan Tugas Akhir di BRIN Yogyakarta, dan *mensupport* satu sama lain untuk mencapai titik ini.

DAFTAR ISI

COVER	i
HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN	Error! Bookmark not defined.
PERNYATAAN KEASLIAN TULISAN	iii
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	ix
DAFTAR ISI	xii
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
DAFTAR LAMPIRAN	xvi
ABSTRAK	xvii
ABSTRACT	xviii
مستخلص البحث	xix

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	5
1.5 Batasan Masalah	5

BAB II KAJIAN PUSTAKA

2.1 Pengertian Radiasi	6
2.2 Jenis-Jenis Radiasi	6
2.2.1 Radiasi Pengion	7
2.2.2 Radiasi Non-Pengion	7
2.3 Radiasi Gamma	8
2.4 Detektor <i>Geiger-Muller</i>	9
2.5 Catu daya Tegangan Tinggi (<i>High Voltage</i>)	13
2.6 Mikrokontroler ESP32 WROOM DEVKIT V1	15
2.7 Sumber Radiasi Cesium-137	17
2.8 Sensor BME-280	18
2.9 Sensor GPS (NEO-6M)	21
2.10 TFT ILI9488	22
2.11 Software Arduino IDE 1.8.9	24
2.12 Penelitian Terdahulu	27

BAB III METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian	33
3.2 Jenis Penelitian	33
3.3 Alat dan Bahan Penelitian	33
3.3.1 Alat-Alat Penelitian	33
3.3.2 Bahan-Bahan Penelitian	34

3.4 Prosedur Penelitian.....	34
3.4.1 Studi Literatur	36
3.5 Perancangan Sistem	37
3.5.1 Perancangan Perangkat Keras (<i>Hardware</i>)	37
3.5.2 Perancangan Program Arduino 1.8.9	46
3.5.3 Perancangan Pengiriman Data ke <i>Google Sheet</i>	47
3.6 Tahap Pengambilan Data.....	58
3.6.1 Pengambilan Data Kalibrasi Pemantauan Radiasi	58
3.6.2 Pengambilan Data Kalibrasi Lingkungan	58
3.6.3 Pengambilan Data Kalibrasi Tegangan Baterai.....	59
3.6.4 Pengambilan Data Kalibrasi Posisi	59
3.7 Tahap Pengolahan Data.....	60
3.7.1 Pengolahan Data Kalibrasi Pemantauan Radiasi	60
3.7.2 Pengolahan Data Kalibrasi Lingkungan	61
3.7.3 Pengolahan Data Tegangan Baterai.....	63
3.7.4 Pengolahan Data Kalibrasi Posisi	64
 BAB IV HASIL DAN PEMBAHASAN	
4.1 Hasil Penelitian	65
4.1.1 Hasil Integrasi Modul dengan ESP32 WROOM DEVKIT V1	65
4.1.2 Hasil Perancangan Program Arduino IDE 1.8.9	67
4.1.3 Hasil Tampilan Display TFT ILI9488 dan <i>Spreadsheet</i>	77
4.1.4 Hasil Data Kalibrasi	78
4.2 Pembahasan.....	87
4.3 Keterkaitan Hasil Penelitian Dalam Perspektif Islam.....	94
 BAB V PENUTUP	
5.1 Kesimpulan	100
5.2 Saran.....	100
 DAFTAR PUSTAKA.....	102
LAMPIRAN.....	105

DAFTAR GAMBAR

Gambar 2. 1 Skema Detektor <i>Geiger-Müller</i>	10
Gambar 2. 2 Kurva Plateau Antara Cacah Ion Terhadap Tegangan Dalam Detektor Isian Gas	12
Gambar 2. 3 ESP32 WROOM DEVKIT V1	15
Gambar 2. 4 Sumber Radioaktif Cesium-137	18
Gambar 2. 5 Sensor BME-280	19
Gambar 2. 6 Sensor GPS NEO-6M	21
Gambar 2. 7 TFT ILI9488 Module	23
Gambar 2. 8 Tampilan Arduino IDE 1.8.9	26
Gambar 3. 1 Prosedur Penelitian Monitoring Radiasi dan Lingkungan	35
Gambar 3. 2 Blok Diagram Perancangan Perangkat Keras (<i>Hardware</i>)	37
Gambar 3. 3 Schematic Rangkaian Monitor Radiasi dan Lingkungan	39
Gambar 3. 4 Rancangan PCB Hardware Monitor Lingkungan ESP32	41
Gambar 3. 5 PCB Kit Detektor <i>Geiger-Müller</i> Signal Conditioner	45
Gambar 3. 6 Display TFT ILI9488 Monitor Radiasi dan Lingkungan	46
Gambar 3. 7 Tampilan <i>Spreadsheet</i> data Monitor Radiasi dan Lingkungan	47
Gambar 3. 8 Blok Diagram Perancangan Program	46
Gambar 4. 1 Hasil Perancangan Sistem Perangkat Keras (<i>Hardware</i>)	65
Gambar 4. 2 Inisialisasi Library Arduino IDE 1.8.9	68
Gambar 4. 3 Inisialisasi PIN dan Variabel yang digunakan	69
Gambar 4. 4 Pengiriman Data ke <i>Google Sheets</i>	70
Gambar 4. 5 Pengaturan Waktu dengan Zona yang sesuai	71
Gambar 4. 6 Setting Setup untuk Tampilan Display	72
Gambar 4. 7 Inisialisasi masing-masing sensor ke Setup	73
Gambar 4. 8 Fungsi loop dan rumus dari masing-masing sensor	73
Gambar 4. 9 Listing Persamaan <i>Battery Voltage</i> dan Konversi Paparan	75
Gambar 4. 10 Set Tampilan TFT	75
Gambar 4. 11 Listing Data Logging ke <i>Spreadsheet</i>	76
Gambar 4. 12 Hasil Display Tampilan TFT	77
Gambar 4. 13 Hasil Display Tampilan <i>Spreadsheet</i>	78
Gambar 4. 14 Hasil paparan dan cacahan dari surveymeter standar	79
Gambar 4. 15 Hasil paparan dan cacahan dari display TFT	80
Gambar 4. 16 Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di luar ruangan	81
Gambar 4. 17 Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di ruangan bersuhu dingin (berAC)	82
Gambar 4. 18 Pembuktian Kalibrasi Tegangan Baterai dengan <i>Voltmeter</i>	83
Gambar 4. 19 Dokumentasi Kalibrasi Posisi	85
Gambar 4. 20 Pembuktian Rentang Titik 1 Pada Antar Perangkat	85
Gambar 4. 21 Pembuktian Rentang Titik 2 Pada Antar Perangkat	86
Gambar 4. 22 Pembuktian Rentang Titik 3 Pada Antar Perangkat	86
Gambar 4. 23 Pembuktian Rentang Titik 4 Pada Antar Perangkat	86
Gambar 4. 24 Pembuktian Rentang Titik 5 Pada Antar Perangkat	87

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	30
Tabel 3. 1 Tahapan Pengiriman Data ke <i>Google Sheet</i>	47
Tabel 3. 2 Pengambilan Data Kalibrasi Pemantauan Radiasi.....	61
Tabel 3. 3 Pengambilan Data Kalibrasi Lingkungan.....	61
Tabel 3. 4 Pengambilan Data Tekanan	62
Tabel 3. 5 Pengambilan Data Kalibrasi Tegangan Baterai	63
Tabel 3. 6 Pengambilan Data Kalibrasi Posisi.....	64
Tabel 4. 1 Keterangan komponen pada gambar 4.1	66
Tabel 4. 2 Data Kalibrasi Kit Detektor <i>Geiger Muller</i> dengan Surveymeter Standar	79
Tabel 4. 3 Hasil data kalibrasi Sensor BME-280 dengan Termometer Jam Digital	81
Tabel 4. 4 Hasil Kalibrasi Tekanan dengan Konversi dari <i>Altitude</i>	84
Tabel 4. 5 Hasil Kalibrasi Tegangan Baterai dengan <i>Voltmeter</i>	83
Tabel 4. 6 Hasil Data Kalibrasi <i>Track Logger</i> dengan Sensor NEO-6M	84

DAFTAR LAMPIRAN

Lampiran 1. 1	Gambar Perancangan Perangkat Keras	105
Lampiran 1. 2	Gambar Hasil Perancangan Perangkat Keras	107
Lampiran 1. 3	Listing Program Arduino IDE 1.8.9.....	111

ABSTRAK

Rahayu, Ratna. 2025. **Rancang Bangun Perangkat Monitor Radiasi Gamma Berbasis Mikrokontroler ESP32 Menggunakan Detektor *Geiger-Müller* Untuk Pemantauan Radiasi dan Lingkungan Terpadu.** Skripsi. Program Studi Fisika. Fakultas Sains dan Teknologi. Universitas Islam Negeri Maulana Malik Ibrahim Malang. Dosen Pembimbing (I) Farid Samsu Hananto, M.T. (II) Mubasyiroh, S.S., M.Pd.I.

Kata kunci: Radiasi gamma, *Geiger-Müller*, ESP32, pemantauan lingkungan, IoT, sensor BME-280, GPS NEO-6M, TFT ILI9488.

Radiasi gamma merupakan bentuk radiasi pengion yang tidak kasat mata dan berpotensi membahayakan kesehatan manusia serta lingkungan, terutama di sekitar fasilitas nuklir. Penelitian ini bertujuan merancang dan membangun sistem pemantauan radiasi gamma berbasis mikrokontroler ESP32 yang terintegrasi dengan detektor Geiger-Müller serta sensor lingkungan dan posisi (GPS). Sistem ini dirancang agar portabel, ekonomis, dan mampu melakukan pemantauan terpadu terhadap parameter radiasi, lingkungan (suhu, kelembapan, tekanan), dan posisi geografis secara real-time. Data yang diperoleh ditampilkan melalui layar TFT ILI9488 serta dikirimkan ke penyimpanan cloud menggunakan koneksi Wi-Fi. Proses perancangan mencakup pengembangan perangkat keras dan perangkat lunak menggunakan Arduino IDE, serta integrasi sensor melalui protokol komunikasi I2C, UART, dan SPI. Hasil pengujian menunjukkan bahwa perangkat mampu mendeteksi dan mengirimkan data radiasi serta parameter lingkungan dengan tingkat akurasi hingga 95%. Sistem ini berpotensi digunakan sebagai solusi pemantauan radiasi yang efisien dan adaptif, khususnya untuk lingkungan sekitar reaktor atau area berisiko tinggi terhadap radiasi.

ABSTRACT

Rahayu, Ratna. 2025. **Design and Development of a Gamma Radiation Monitoring Device Based on ESP32 Microcontroller Using a Geiger-Müller** Detector for Integrated Radiation and Environmental Monitoring. Undergraduate Thesis. Department of Physics, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang. Supervisor (I) Farid Samsu Hananto, M.T., and (II) Mubasyiroh, S.S., M.Pd.I.

Keywords: Gamma radiation, Geiger-Müller, ESP32, environmental monitoring, IoT, BME-280 sensor, NEO-6M GPS, TFT ILI9488.

Gamma radiation is a type of ionizing radiation that is invisible to the human eye and can pose serious risks to both human health and the environment, particularly around nuclear facilities. This research aims to design and develop a portable, cost-effective monitoring system using an ESP32 microcontroller integrated with a Geiger-Müller detector, environmental sensors, and a GPS module. The system enables real-time monitoring of radiation exposure levels, environmental parameters (temperature, humidity, and air pressure), and geographical location. Data collected by the sensors is displayed on a TFT ILI9488 screen and transmitted to cloud storage via Wi-Fi. The design process includes both hardware and software development using the Arduino IDE, with sensor integration via I2C, UART, and SPI communication protocols. Testing results indicate that the device is capable of detecting and transmitting radiation and environmental data with up to 95% accuracy. This system offers a practical and adaptable solution for radiation monitoring, especially in high-risk environments such as those surrounding nuclear reactors.

مستخلص البحث

راهايو، راتنا. 2025. تصميم وبناء جهاز لمراقبة إشعاع غاما يعتمد على المتحكم الدقيق ESP32 باستخدام كاشف غيغر-ميلر لرصد الإشعاع والبيئة المتكاملة. بحث جامعي. قسم الفيزياء. كلية العلوم والتكنولوجيا. جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف : (I) فريد سامسو هاننطو، الماجستير (II) مباشرة، الماجستير

الكلمة الرئيسية: إشعاع غاما، جهاز غيغر-ميلر، ESP32، مراقبة البيئة، إنترنت الأشياء، مستشعر BME-280، نظام تحديد المواقع M6-NEO، شاشة TFT ILI 9488.

تشكل الإشعاعات غاما نوعًا من الإشعاع المؤين غير المرئي والذي يمكن أن يشكل خطرًا على صحة البشر والبيئة، خاصةً حول المنشآت النووية. تهدف هذه الدراسة إلى تصميم وبناء نظام لمراقبة إشعاع غاما قائم على المتحكم الدقيق ESP32، المتكامل مع كاشف Geiger-Müller وكذلك مستشعرات البيئة والموقع (GPS). تم تصميم هذا النظام ليكون محمولًا واقتصاديًا، وقادرًا على إجراء مراقبة متكاملة لمؤشرات الإشعاع والبيئة (درجة الحرارة، الرطوبة، الضغط)، والموقع الجغرافي في الوقت الحقيقي. تُعرض البيانات التي تم الحصول عليها من خلال شاشة TFT ILI 9488، وتُرسل إلى التخزين السحابي باستخدام اتصال Wi-Fi. تشمل عملية التصميم تطوير الأجهزة والبرمجيات باستخدام Arduino IDE، بالإضافة إلى تكامل المستشعرات عبر بروتوكولات الاتصال C 12 و UART و SPI. أظهرت نتائج الاختبار أن الجهاز قادر على اكتشاف وإرسال بيانات الإشعاع ومؤشرات البيئة بدقة عالية. تظهر نتائج الاختبار أن الجهاز قادر على اكتشاف وإرسال بيانات الإشعاع والمعلومات البيئية بدقة تصل إلى 95%. يتمتع هذا النظام بإمكانية استخدامه كحل فعال ومتكيف لمراقبة الإشعاع، وخاصة في البيئات المحيطة بالمفاعل أو في المناطق ذات المخاطر العالية من الإشعاع.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Radiasi gamma adalah jenis radiasi elektromagnetik yang memiliki energi tinggi dan dapat menembus berbagai bahan, termasuk tubuh manusia. Dalam konteks reaktor nuklir, seperti Reaktor Kartini yang berada di Direktorat Pengelolaan Fasilitas Ketenaganukliran (DPFK) – Badan Riset dan Inovasi Nasional (BRIN), pemantauan radiasi gamma sangat penting untuk menjaga keselamatan operasional dan perlindungan lingkungan. Pengukuran radiasi gamma yang akurat membantu dalam mendeteksi adanya kebocoran radiasi yang dapat membahayakan pekerja dan masyarakat di sekitar reaktor. Oleh karena itu, diperlukan sistem pemantauan yang efektif dan terjangkau untuk memastikan reaktor beroperasi dalam batas aman yang ditetapkan (Anonim, 2020).

Pemantauan radiasi gamma di reaktor nuklir sangat penting karena radiasi ini tidak dapat terlihat oleh mata manusia dan dapat menyebabkan kerusakan serius jika tidak terdeteksi. Perangkat pemantau yang ada saat ini seringkali besar, mahal, dan sulit dipasang di berbagai lokasi. Oleh karena itu, ada kebutuhan untuk mengembangkan sistem monitor radiasi gamma yang lebih kecil dan lebih ekonomis namun tetap efektif dalam mendeteksi radiasi gamma (Hart, 2018). Monitor mini ini tidak hanya akan meningkatkan kapasitas pemantauan di area strategis, tetapi juga memungkinkan deteksi dini terhadap perubahan Tingkat radiasi yang dapat mengindikasikan masalah pada lingkungan.

Detektor *Geiger-Müller* (GM) adalah salah satu jenis detektor radiasi yang sering digunakan untuk pengukuran radiasi gamma karena kemampuannya dalam

mendeteksi radiasi gamma dengan efisiensi yang baik. Detektor ini bekerja dengan menghasilkan pulsa listrik setiap kali radiasi gamma menembus gas dalam tabung detektor, yang kemudian dihitung untuk menentukan tingkat radiasi (Jones, 2019). Sistem berbasis *Geiger-Müller* (GM) sebagai pendeteksi radiasi memungkinkan deteksi radiasi gamma dengan akurasi tinggi dan biaya yang lebih rendah, sementara ESP32 memberikan kemampuan koneksi nirkabel Wi-Fi dan Bluetooth, memiliki kemampuan integrasi yang luas dengan berbagai jenis sensor, termasuk sensor lingkungan (seperti sensor suhu, kelembaban, gas, dan debu) dan sensor posisi berbasis GPS.

Adanya detektor *Geiger-Müller* untuk mengukur radiasi gamma, ditambah dengan integrasi mikrokontroler modern seperti ESP32, bermanfaat untuk merancang sistem pemantauan yang tidak hanya efektif tetapi juga ekonomis dalam penggunaannya di lingkungan.

Sebagaimana yang dijelaskan pada surah Al-Baqarah ayat 164 :

إِنَّ فِي خَلْقِ السَّمُوتِ وَالْأَرْضِ وَاخْتِلَافِ اللَّيْلِ وَالنَّهَارِ وَالْفُلْكِ الَّتِي تَجْرِي فِي الْبَحْرِ بِمَا يَنْفَعُ النَّاسَ وَمَا أَنْزَلَ اللَّهُ مِنَ السَّمَاءِ مِنْ مَّاءٍ فَأَحْيَا بِهِ الْأَرْضَ بَعْدَ مَوْتِهَا وَبَثَّ فِيهَا مِنْ كُلِّ دَابَّةٍ وَتَصْرِيفِ الرِّيْحِ وَالسَّحَابِ الْمُسَخَّرِ بَيْنَ السَّمَاءِ وَالْأَرْضِ لَآيَاتٍ لِّقَوْمٍ يَعْقِلُونَ

Artinya : “*Sesungguhnya dalam penciptaan langit dan bumi, dalam pergantian malam dan siang, dalam kapal yang berlayar di laut dengan membawa apa yang bermanfaat bagi manusia, dan apa yang diturunkan Allah dari langit berupa air yang dihidupkan bumi setelah mati, dan dalam penyebaran di bumi segala jenis hewan, dan dalam peredaran angin dan awan yang dikendalikan antara langit dan bumi, terdapat tanda-tanda bagi kaum yang menggunakan akal.*”

Ayat ini menegaskan bahwa segala sesuatu yang ada di alam ini merupakan tanda bagi mereka yang berpikir. Dalam hal ini, perkembangan teknologi pemantauan radiasi gamma adalah salah satu bentuk pemanfaatan ilmu

pengetahuan dan teknologi untuk melindungi umat manusia. Ini merupakan contoh bagaimana manusia dapat menggunakan akalnya, yang diberikan oleh Allah, untuk menciptakan solusi yang tidak hanya bermanfaat, tetapi juga menjaga keberlanjutan kehidupan dan kelestarian bumi. Dengan menggunakan akal dan teknologi, kita dapat mencegah kerusakan akibat radiasi berbahaya dan memastikan keselamatan serta Kesehatan umat manusia dan bumi yang merupakan amanah dari Allah SWT (Shihab, 2002).

Penelitian (Habib, 2016) mengembangkan sistem monitoring jarak jauh radiasi gamma secara realtime berbasis web server dimana pada penelitian tersebut sama-sama menggunakan detektor *Geiger-Müller* sebagai pendeteksi radiasi gamma, pada penelitian tersebut menggunakan mikrontroler Atmega28 untuk memproses data ke modul GSM yang nanti nya data yang ditangkap akan dibaca melalui web server atau jarak jauh.

Melalui penelitian yang ada sebelumnya, penulis mengembangkan penelitian ini dengan pendekatan yang lebih efisien, ekonomis, serta memperhatikan aspek keselamatan dan keamanan pekerja yang berada di area dengan tingkat paparan radiasi yang tinggi. Penelitian ini menggunakan detektor *Geiger-Müller* sebagai sensor untuk mengukur paparan radiasi dan mengandalkan kombinasi mikrokontroler ESP32, yang memiliki kelebihan dalam konektivitas Wi-Fi dan Bluetooth. Alat ini dirancang portabel sehingga dapat digunakan pada jarak jauh, menghindari kontak langsung antara tubuh manusia dan radiasi berbahaya. Selain itu, alat ini mengintegrasikan sensor lingkungan dan sensor posisi untuk mendeteksi pengaruh lingkungan terhadap paparan radiasi tersebut.

Penelitian ini diharapkan mampu mendeteksi dan memantau radiasi gamma secara langsung serta mengirimkan data ke display monitor, dengan memanfaatkan kombinasi detektor *Geiger-Müller* dan mikrokontroler ESP32 yang dapat mendeteksi paparan radiasi secara jarak jauh, sehingga memungkinkan respons lebih cepat terhadap potensi paparan radiasi yang berbahaya, serta mendukung upaya mitigasi risiko kesehatan bagi pekerja dan lingkungan.

1.2 Rumusan Masalah

1. Bagaimana merancang sistem perangkat monitor radiasi gamma berbasis mikrokontroler ESP32 yang mampu melakukan pemantauan radiasi dan lingkungan terpadu?
2. Bagaimana mengintegrasikan perangkat monitor radiasi gamma berbasis mikrokontroler ESP32 dengan sensor lingkungan, sensor posisi dan display penampil?
3. Bagaimana menampilkan data radiasi, data lingkungan dan data posisi?
4. Bagaimana mengirimkan data radiasi, data lingkungan, dan data posisi pada penyimpanan cloud?

1.3 Tujuan Penelitian

1. Merancang sistem perangkat keras monitor radiasi gamma berbasis mikrokontroler ESP32 yang terintegrasi dengan detektor *Geiger-Müller* dan sensor lingkungan terpadu untuk pemantauan radiasi dan lingkungan.
2. Mengimplementasikan detektor *Geiger-Müller* (GM) untuk mendeteksi radiasi gamma.
3. Mengintegrasikan perangkat monitor radiasi gamma berbasis mikrokontroler ESP32 dengan sensor lingkungan terpadu dan sensor posisi.

4. Menampilkan dan mengirimkan data radiasi, data lingkungan terpadu dan data posisi pada display penampil dan penyimpanan *cloud*.

1.4 Manfaat Penelitian

1. Memberikan solusi pemantauan radiasi gamma yang lebih ekonomis dengan menggunakan komponen lokal, yang dapat meningkatkan keselamatan paparan radiasi gamma dan lingkungan.
2. Menyediakan data dan desain yang dapat digunakan sebagai referensi untuk penelitian lanjutan dalam pengembangan teknologi pemantauan radiasi.
3. Meningkatkan keselamatan, keamanan dan kesehatan para pekerja yang terpapar radiasi, terutama pada industri yang berkaitan dengan bahan radioaktif atau radiasi gamma.
4. Perangkat ini diharapkan dapat digunakan untuk mendeteksi radiasi di area lingkungan Reaktor Kartini.

1.5 Batasan Masalah

1. Penelitian ini menggunakan detektor *Geiger-Müller* 4011 untuk mendeteksi paparan radiasi gamma.
2. Penelitian ini menggunakan sensor BME280, dan sensor GPS NEO-6M untuk mendeteksi data lingkungan dan data posisi.
3. Mikrokontroler yang digunakan adalah mikrokontroler ESP32 WROOM DEVKIT V1.
4. Sistem menggunakan layar LCD TFT ILI9488, untuk menampilkan parameter pengukuran.
5. Data hasil pengukuran dikirim melalui Wi-Fi.
6. Perangkat menggunakan catudaya mandiri yang dapat diisi ulang.

BAB II

KAJIAN PUSTAKA

2.1 Pengertian Radiasi

Radiasi merupakan pancaran energi yang melewati suatu materi atau ruangan. Radiasi dapat berbentuk partikel, panas, maupun gelombang elektromagnetik dari suatu sumber energi (Sutapa dkk, 2016). Dalam fisika, radiasi di deskripsikan sebagai setiap proses dimana energi bergerak melalui media atau melalui ruang, dan akhirnya diserap oleh benda lain (Sprawls, 2016).

Radiasi juga diartikan sebagai proses terjadinya perpindahan panas (kalor) tanpa menggunakan zat perantara (Rozak, 2016). Kalor dapat dipancarkan melalui bentuk gelombang cahaya, gelombang radio dan gelombang elektromagnetik. Radiasi dalam bentuk gelombang elektromagnetik atau disebut juga dengan foton adalah jenis radiasi yang tidak mempunyai massa dan muatan listrik, misalnya adalah sinar gamma dan sinar-x.

Secara umum radiasi dapat dibagi menjadi dua kategori utama, yaitu radiasi pengion dan non-pengion. Radiasi pengion memiliki energi yang cukup tinggi untuk melepaskan *electron* dari atom atau molekul, sehingga menghasilkan ion. Contoh radiasi pengion meliputi sinar gamma, sinar-X, dan partikel alfa atau beta. Sebaliknya, radiasi non-pengion, seperti gelombang radio dan Cahaya tampak, tidak cukup kuat untuk mengionisasi atom dan molekul (Knoll, 2020).

2.2 Jenis-Jenis Radiasi

Secara garis besar, radiasi digolongkan ke dalam dua kelompok, yakni radiasi pengion dan radiasi non pengion yang memiliki perbedaan masing-masing sebagai berikut :

2.2.1 Radiasi Pengion

Radiasi pengion adalah jenis radiasi yang dapat menyebabkan proses ionisasi (terbentuknya ion positif dan ion negatif) apabila berinteraksi dengan materi. Radiasi yang termasuk dalam jenis radiasi pengion adalah partikel alfa, partikel beta, sinar gamma, sinar-X dan neutron. Setiap jenis radiasi memiliki karakteristik khusus, jadi, radiasi pengion yang selanjutnya disebut radiasi adalah gelombang elektromagnetik dan partikel bermuatan yang karena energi yang dimilikinya mampu mengionisasi media yang dilaluinya.

Untuk memantau dan mengukur tingkat radiasi pengion, digunakan perangkat seperti detektor *geiger-müller*, dosimeter, dan kamera pencitraan radiasi. Detektor *geiger-müller* adalah salah satu teknologi yang paling umum digunakan untuk mendeteksi partikel dan sinar pengion, terutama karena kesederhanaannya dalam mendeteksi pulsa Listrik akibat ionisasi (Wahyudi, 2017).

2.2.2 Radiasi Non-Pengion

Radiasi non-pengion adalah jenis radiasi yang tidak akan menyebabkan efek ionisasi apabila berinteraksi dengan materi. Berbeda dengan radiasi pengion, radiasi ini tidak mampu melepaskan electron dari atom atau molekul tetapi dapat menyebabkan eksitasi electron ke Tingkat energi yang lebih tinggi. Contohnya seperti, gelombang radio, mikro, sinar inframerah, dan lain-lain (Suryani, 2018).

Walaupun dianggap lebih aman dibandingkan radiasi pengion, paparan radiasi non-pengion dalam intensitas tinggi atau jangka Panjang dapat memberikan dampak negatif. Misalnya, paparan sinar ultraviolet berlebihan

dapat menyebabkan kerusakan kulit seperti sunburn dan meningkatkan risiko kanker kulit (WHO, 2020).

2.3 Radiasi Gamma

Radiasi tinggi mampu melepas energi dalam jumlah yang sangat besar, dan apabila melalui materi akan dapat menimbulkan ion-ion dalam materi yang dilaluinya sehingga disebut juga sebagai radiasi pengion (berebein dkk, 2020). Salah satu radiasi pengion yaitu sinar gamma. Sinar gamma merupakan salah satu gelombang elektromagnetik yang memiliki daya tembus tinggi sehingga dapat melewati tubuh manusia.

Radiasi gamma adalah bentuk radiasi elektromagnetik dengan energi tinggi dan panjang gelombang pendek. Radiasi ini memiliki kemampuan penetrasi yang sangat tinggi dibandingkan dengan radiasi alpha dan beta, sehingga memerlukan perhatian khusus dalam pengawasan dan pemantauan (Knoll, 2018). Radiasi gamma dihasilkan dari proses peluruhan radioaktif dalam inti atom dan sering digunakan dalam berbagai aplikasi, termasuk dalam pengawasan reaktor nuklir (Singh & Kumar, 2020).

Menurut buku teks oleh Knoll (2018), radiasi gamma memiliki energi yang cukup untuk menembus material dan memerlukan detektor yang sensitif untuk mengukurnya secara efektif (Bailey & Moore, 2021). Radiasi ini memerlukan sistem pengawasan yang canggih untuk melindungi lingkungan dan kesehatan manusia dari potensi bahaya radiasi yang berlebih (Miller & Collins, 2022).

Proses peluruhan radioaktif yang menghasilkan radiasi gamma seringkali tidak terlihat, sehingga memerlukan perangkat deteksi yang sangat sensitif, seperti

detektor scintillation atau detektor semikonduktor, untuk memonitor tingkat radiasi secara real-time (Jain & Kaur, 2022).

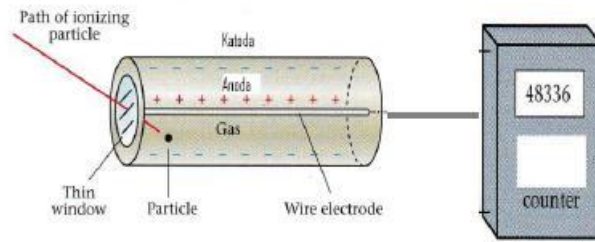
Karakteristik radiasi gamma meliputi energi yang sangat tinggi, panjang gelombang yang pendek, kemampuan penetrasi yang luar biasa, serta interaksinya yang unik dengan materi. Energi foton radiasi gamma biasanya berkisar antara 100 keV hingga lebih dari 10 MeV. Energi tinggi ini memungkinkan radiasi gamma untuk menembus berbagai material yang sangat padat. Panjang gelombangnya yang kurang dari 10 pikometer menjadikannya salah satu bentuk radiasi paling kuat dalam spektrum elektromagnetik. Hal ini memungkinkan radiasi gamma digunakan untuk menembus material seperti logam, beton, bahkan jaringan biologis (Widodo, 2021).

Radiasi gamma menghasilkan pulsa listrik melalui ionisasi gas yang ada di dalam detektor *Geiger-Müller*. Radiasi gamma menyebabkan elektron terlepas dari atom dalam gas di dalam tabung detektor, dan partikel bebas yang dihasilkan bergerak menuju elektroda. Dengan adanya tegangan tinggi, terjadi penggandaan ionisasi yang menghasilkan arus listrik besar dalam waktu singkat, yang kemudian tercatat sebagai pulsa listrik. Detektor *Geiger-Müller* menghitung jumlah pulsa ini untuk memberikan informasi mengenai intensitas radiasi gamma yang ada di lingkungan tersebut.

2.4 Detektor *Geiger-Muller*

Detektor *Geiger-Müller* (GM) adalah salah satu alat yang paling umum digunakan untuk mendeteksi radiasi ionisasi, termasuk radiasi gamma. Detektor ini berfungsi dengan prinsip ionisasi, di mana radiasi yang melewati tabung *Geiger-Müller* menyebabkan ionisasi gas di dalam tabung dan menghasilkan

pulsa listrik yang dapat diukur (Zhao & Li, 2022).



Gambar 2. 1 Skema Detektor *Geiger-Müller* Shoufika, 2017)

Menurut artikel oleh Bailey et al. (2021), detektor *Geiger-Müller* dikenal karena kesederhanaannya dan kemampuannya untuk mendeteksi berbagai jenis radiasi dengan efisiensi tinggi. Penelitian oleh Miller dan Collins (2022) juga menunjukkan bahwa detektor GM adalah alat yang efektif dalam pengawasan radiasi di lingkungan nuklir karena kemampuannya untuk memberikan respons cepat terhadap radiasi gamma.

Detektor *Geiger-Müller* adalah alat untuk mendeteksi radiasi berdasarkan pasangan ion yang dibentuk di dalam tabung yang berisi gas. Detektor isian gas biasanya terdiri dari sebuah tabung berdinding logam yang diisi dengan gas dan mempunyai kawat di tengah nya. Dinding tabung merangkap sebagai katoda sedang kawat sebagai anoda (Shoufika, 2017).

Detektor *Geiger-Müller*, atau sering disebut sebagai tabung GM, beroperasi berdasarkan prinsip ionisasi gas. Ketika radiasi gamma memasuki tabung yang berisi gas pengisi (seperti helium, neon, atau argon), ia mengionisasi gas tersebut. Proses ini menghasilkan pasangan ion positif dan elektron bebas. Karena adanya medan listrik yang diterapkan di dalam tabung, ion dan elektron tersebut akan bergerak menuju elektroda yang berbeda, menghasilkan pulsa listrik yang dapat diukur (Richter & Schaefer, 2019).

Penelitian oleh Richter et al. (2019) menjelaskan secara mendetail mengenai struktur dan komponen penting dari detektor GM, termasuk elektrodanya, gas pengisi, dan mekanisme penguatan pulsa (Papadopoulos, 2020).

Selain itu, studi oleh Papadopoulos (2020) menyoroti kelebihan detektor GM dalam hal sensitivitas tinggi dan kemudahan penggunaan di lingkungan radiasi rendah hingga sedang.

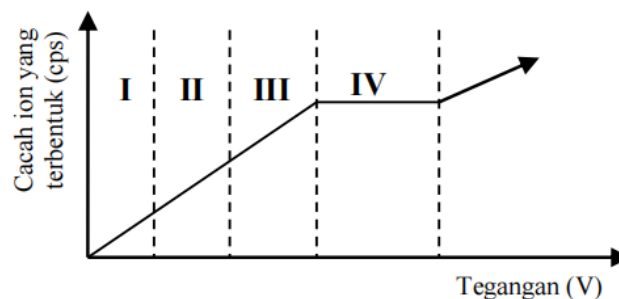
Pada penelitian Spatz, operasi pencacahan pada *Geiger-Müller* menyebabkan perubahan karakteristik kurva *plateau*, setelah digunakan kurva *plateau* dengan segera sedikit meningkat, dan setelah terus digunakan kurva *plateau* menjadi tajam. Perubahan pengamatan karakteristik kurva *plateau* karena dekomposisi dari gas detector oleh muatan. Dan dalam penelitian Babak, sumber tegangan tinggi dirancang untuk menciptakan tegangan nominal di anoda *Geiger-Muller counter* di tengah perhitungan tegangan karakteristik (sekitar 400V). Stabilisasi tegangan *output* dilakukan dengan membatasi amplitudo yang keluar pada sirkuit primer (Babak, 2014).

Detektor isian gas prinsip kerjanya memanfaatkan terjadinya ionisasi gas isian pada medium aktif dalam detector akibat adanya interaksi dengan zarah radiasi maka akan timbul pasangan ion-elektron. Dengan adanya beda potensial pada anoda dan katoda maka akan timbul medan listrik, sehingga pasangan ion-elektron akan terpisahkan. Ion akan bergerak ke arah katoda dan elektron bergerak ke anoda (Sayono & Sujitno, 2010).

Pencacahan yang terbentuk bergantung pada besar tegangan yang dikenakan pada detektor. Ketika tegangan yang terus dinaikkan tidak lagi ada kesebandingan dan tinggi pulsa tidak bergantung pada besar tenaga radiasi yang dideteksi daerah

ini dinamakan daerah *Geiger-Müller*. Daerah *Geiger-Müller* yang terdapat pada (gambar 2.2) yang dianggap kurva *plateau* merupakan rentang tegangan dalam tabung pencacah gas.

Pada daerah I (daerah rekomendasi), ketika tegangan yang dikenakan masih rendah, *electron* dan ion positif yang terbentuk akan segera bergabung kembali. Kalau tegangan terus dinaikkan, maka kemungkinan terjadinya penggabungan kembali elektron dan ion positif dapat diabaikan. Ion-ion yang sampai elektroda akan menghasilkan suatu sinyal pulsa. Daerah ini dinamakan daerah II (daerah ionisasi). Karena kenaikan tegangan, pada daerah III (daerah proporsional electron yang dibebaskan akan mempunyai tenaga gerak yang cukup besar untuk mengakibatkan ionisasi sekunder, mengionkan atom-atom gas lainnya karena tumbukan. Hal ini menaikkan jumlah muatan yang dikumpulkan pada elektroda dan menaikkan tinggi pulsa yang dihasilkan. Dalam daerah IV (daerah *Geiger-Müller*), tegangan terus dinaikkan maka tidak ada lagi kesebandingan dan tinggi pulsa tidak lagi bergantung besar tenaga radiasi yang dideteksi. Daerah ini dinamakan daerah *Geiger-Müller*. Di atas daerah *Geiger Muller*, apabila tegangan terus dinaikkan maka akan terjadi lucutan listrik secara terus menerus dalam tabung gas dan akibatnya detektor menjadi rusak.



Gambar 2. 2 Kurva *Plateau* Antara Cacah Ion Terhadap Tegangan Dalam Detektor Isian Gas (*Shoufika, 2017*)

2.5 Catu daya Tegangan Tinggi (*High Voltage*)

Catu daya tegangan tinggi (*high voltage power supply*) adalah komponen elektronik yang dirancang untuk menghasilkan tegangan tinggi yang stabil dan terkontrol. Dalam banyak aplikasi, seperti deteksi radiasi, pengujian isolasi, dan akselerator partikel, diperlukan catu daya yang mampu menghasilkan tegangan lebih dari 100 V, bahkan mencapai ratusan kilovolt, tanpa menyebabkan gangguan pada sistem (Yang dan Chen, 2018). Tujuan utama dari catu daya ini adalah untuk menyediakan sumber energi dengan kualitas tinggi, mengatur fluktuasi tegangan, serta menyediakan proteksi terhadap kerusakan komponen akibat tegangan yang tidak stabil. Tegangan tinggi ini umumnya dibutuhkan dalam aplikasi yang memerlukan penghantaran energi melalui media isolasi atau untuk memfasilitasi proses ionisasi dalam perangkat seperti tabung *Geiger-Müller* (GM) yang digunakan untuk mendeteksi radiasi.

Catu daya tegangan tinggi umumnya bekerja dengan mengubah tegangan rendah (biasanya 12 V atau 24 V) menjadi tegangan tinggi yang diperlukan melalui beberapa tahap. Tahap pertama adalah penggunaan konverter DC-DC atau trafo step-up untuk meningkatkan tegangan input menjadi lebih tinggi. Pada trafo step-up, hubungan antara tegangan primer dan sekunder dapat dijelaskan dengan rumus (Suyanto, 2019):

$$V_s = V_p \times \left(\frac{N_s}{N_p} \right) \quad (2.1)$$

di mana V_s adalah tegangan sekunder, V_p adalah tegangan primer, N_s adalah jumlah lilitan pada kumparan sekunder, dan N_p adalah jumlah lilitan pada kumparan primer. Jika $N_s > N_p$, maka tegangan sekunder V_s lebih besar dari tegangan primer V_p , sehingga transformator ini disebut sebagai step-up

transformator. Jika $N_s < N_p$, maka tegangan sekunder V_s lebih kecil dari tegangan primer V_p , sehingga transformator ini disebut sebagai step-down transformator. Selanjutnya, untuk menjaga kestabilan, catu daya ini menggunakan rangkaian pengatur tegangan yang memastikan bahwa tegangan output tetap dalam batas yang aman dan stabil.

Catu daya tegangan tinggi memiliki beberapa komponen utama yang bekerja secara sinergis untuk memastikan kinerja yang optimal. Komponen tersebut antara lain trafo step-up untuk meningkatkan tegangan, dioda penyearah untuk mengubah tegangan AC menjadi DC, Dioda penyearah biasanya mengikuti hukum dasar untuk arus searah, yaitu hukum ohm:

$$V = I \times R \quad (2.2)$$

di mana V adalah tegangan (volt), I adalah arus (ampere), dan R adalah resistansi (ohm). Rectifier merupakan sebuah komponen elektronik yang digunakan untuk mengubah arus bolak-balik (AC) menjadi arus searah (DC). Proses ini dikenal sebagai penyearahan (rectification). Penyearah sangat penting dalam berbagai aplikasi, terutama dalam perangkat elektronik yang memerlukan sumber energi DC, komponen yang biasanya terdapat pada rectifier yaitu dioda dan kapasitor. Kemudian terdapat transistor atau pengatur tegangan untuk menjaga kestabilan output. Selain itu, penggunaan sirkuit proteksi sangat penting, seperti pemutus arus dan pengatur suhu untuk mencegah kerusakan akibat lonjakan tegangan atau *overheat*. Semua komponen ini dirancang dengan ketelitian tinggi untuk memastikan keandalan dan keselamatan perangkat (Prasetyo, 2020).

Dalam aplikasi seperti deteksi radiasi, catu daya tegangan tinggi digunakan untuk memberi tegangan yang cukup agar tabung detektor dapat berfungsi dengan

baik. Sebagai contoh, dalam tabung *Geiger-Müller*, tegangan tinggi diperlukan untuk menghasilkan pulsa listrik ketika radiasi partikel atau foton mengenai gas di dalam tabung. Pulsa ini kemudian digunakan untuk mendeteksi adanya radiasi. Dalam aplikasi ini, catu daya harus memberikan tegangan yang stabil dan dapat dikendalikan dengan baik agar hasil deteksi radiasi tetap akurat dan dapat diandalkan (Setiawan, 2021).

2.6 Mikrokontroler ESP32 WROOM DEVKIT V1

ESP32 WROOM DEVKIT V1 merupakan modul mikrokontroler yang sangat populer dalam pengembangan sistem *Internet of Things* (IoT) karena memiliki integrasi Wi-Fi dan Bluetooth, kemampuan komputasi yang kuat, serta biaya yang rendah. Modul ini banyak digunakan dalam aplikasi sensor lingkungan karena kompatibilitasnya dengan berbagai sensor dan kemampuannya untuk mengolah data secara real-time. Salah satu penggunaan yang signifikan dari ESP32 adalah dalam sistem pemantauan lingkungan, termasuk pendeteksian paparan radiasi melalui sensor radiasi, sensor lingkungan dan integrasi GPS. Teknologi ini semakin penting untuk memantau kondisi lingkungan di area yang berpotensi terkontaminasi radiasi.



Gambar 2. 3 ESP32 WROOM DEVKIT V1

ESP32 WROOM DEVKIT V1 dapat digunakan untuk mengambil data dari beberapa sensor dan mengirimkannya ke server melalui koneksi Wi-Fi. Dengan

demikian, pengguna dapat dengan mudah memantau kondisi lingkungan yang dipantau oleh sensor dari jarak jauh dan dalam waktu nyata tanpa harus berinteraksi langsung dengan perangkat. Dengan teknologi ini, pengguna dapat dengan mudah memantau lingkungan yang diukur oleh sensor secara akurat dan tepat waktu tanpa harus berada di dekat sensor tersebut (Nurkamid & Widodo, 2021).

ESP32 WROOM DEVKIT V1 adalah modul yang paling umum, ideal untuk kebanyakan proyek IoT dengan kemampuan Wi-Fi dan Bluetooth. ESP32 adalah chip yang cukup lengkap, terdapat prosesor, penyimpanan, dan akses pada GPIO (General Purpose Input Output). Board ini memiliki dua versi, yaitu 30 GPIO dan 36 GPIO. Keduanya memiliki fungsi yang sama tetapi versi yang 30 dipilih karena memiliki dua pin GND. Semua pin diberi label bagian atas board sehingga mudah untuk dikenali. Board ini memiliki interface USB to UART yang mudah diprogram dengan program pengembangan aplikasi seperti Arduino IDE. Sumber daya board bisa diberikan melalui konektor micro USB (Wagyana, 2019). 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. ESP32 memiliki pin ADC 12-bit, yang artinya bernilai 0 hingga 4095.

Kumar et al. (2020) dalam penelitiannya menyebutkan bahwa ESP32 memungkinkan integrasi dengan berbagai sensor lingkungan seperti sensor suhu, kelembapan, dan sensor gas. Keunggulan ESP32 yang mendukung protokol komunikasi seperti I2C, SPI, dan UART memungkinkan pengguna untuk menghubungkan beragam sensor untuk memantau berbagai parameter lingkungan secara bersamaan. Selain itu, ESP32 juga memiliki mode deep sleep yang berguna

untuk menghemat daya, sehingga cocok digunakan dalam aplikasi lapangan yang memerlukan pengoperasian jangka panjang dengan sumber daya terbatas.

2.7 Sumber Radiasi Cesium-137

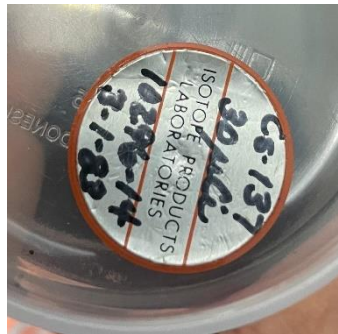
Cesium-137 (Cs-137) merupakan isotop radioaktif yang terbentuk sebagai produk sampingan dalam reaksi fisi nuklir. Isotop ini memiliki waktu paruh sekitar 30,1 tahun dan memancarkan radiasi gamma serta partikel beta selama peluruhannya. Radiasi gamma yang dipancarkan memiliki energi sekitar 662 keV, yang dapat menembus materi dengan baik. Karena waktu paruh yang panjang, cesium-137 bisa bertahan lama di lingkungan, membuatnya menjadi salah satu isotop yang diawasi ketat untuk pemantauan radiasi (Nasution, 2014).

Radiasi gamma yang dipancarkan oleh cesium-137 merupakan jenis radiasi elektromagnetik yang memiliki daya tembus tinggi. Radiasi ini dapat menembus tubuh manusia dan menyebabkan kerusakan pada sel serta jaringan. Paparan jangka panjang terhadap radiasi gamma dapat meningkatkan risiko terkena kanker dan penyakit lainnya. Oleh karena itu, penting untuk menggunakan alat deteksi seperti detektor Geiger-Müller untuk memantau keberadaan cesium-137 di lingkungan dan memastikan tingkat radiasi tetap dalam batas yang aman (Iskandar, 2015).

Detektor *Geiger-Müller* (GM) adalah alat yang sering digunakan untuk mengukur radiasi gamma, termasuk radiasi yang dipancarkan oleh cesium-137. Alat ini bekerja dengan mendeteksi ionisasi yang terjadi ketika radiasi gamma mengenai tabung Geiger-Müller. Sinyal yang dihasilkan oleh ionisasi ini kemudian dapat diukur untuk mengetahui tingkat radiasi di suatu area. Penggunaan detektor GM dalam pemantauan radiasi cesium-137 sangat penting untuk menjaga

keselamatan masyarakat, terutama di daerah yang rawan terpapar radiasi dari reaktor nuklir atau sumber radiasi lainnya (Setiawan, 2016).

Sumber utama cesium-137 adalah reaktor nuklir, di mana cesium ini terbentuk sebagai hasil sampingan dari proses pemecahan inti dalam reaktor. Selain itu, cesium-137 juga digunakan dalam beberapa aplikasi industri, seperti pengukuran densitas material dan terapi radiasi dalam pengobatan kanker. Namun, kecelakaan nuklir seperti yang terjadi di Chernobyl pada tahun 1986 dan Fukushima pada tahun 2011 mengakibatkan pelepasan cesium-137 dalam jumlah besar ke lingkungan sekitar. Pelepasan ini menyebabkan kontaminasi tanah dan air, serta meningkatkan perhatian terhadap pemantauan radiasi (Sari, 2018). Pencemaran cesium-137 dapat menyebabkan dampak jangka panjang terhadap lingkungan karena isotop ini memiliki waktu paruh yang panjang. Di daerah yang terkontaminasi, cesium-137 dapat bertahan di tanah dan air selama beberapa dekade.



Gambar 2. 4 Sumber Radioaktif Cesium-137

2.8 Sensor BME-280

Sensor BME-280 merupakan produk dari Bosch Sensortec yang menggabungkan sensor digital dari temperatur, kelembaban relative, tekanan, dan ketinggian berdasarkan prinsip pengeinderaan yang telah teruji. Sensor BME-280 memiliki dimensi yang kecil dan konsumsi daya yang rendah sehingga mudah

diterapkan dan digerakkan oleh baterai seperti telepon genggam, jam tangan, modul GPS, dan lain-lain.

Salah satu sensor yang banyak digunakan dalam aplikasi pemantauan lingkungan adalah sensor BME280, yang mampu mengukur suhu, kelembapan, dan tekanan udara. Dengan adanya GPS dan sensor tambahan seperti sensor radiasi, sistem pemantauan lingkungan dapat dioptimalkan untuk berbagai keperluan, termasuk mitigasi dampak dari paparan radiasi.



Gambar 2. 5 Sensor BME-280

Kumar et al. (2020), IoT dapat mengubah cara pemantauan lingkungan dilakukan, karena menyediakan akses real-time terhadap data yang dapat digunakan untuk membuat keputusan cepat dalam mengatasi permasalahan lingkungan. Salah satu sensor yang banyak digunakan dalam aplikasi pemantauan lingkungan adalah sensor BME-280, yang mampu mengukur suhu, kelembapan, dan tekanan udara. Dengan adanya GPS dan sensor tambahan seperti sensor radiasi, sistem pemantauan lingkungan dapat dioptimalkan untuk berbagai keperluan, termasuk mitigasi dampak dari paparan radiasi.

BME280 adalah sensor lingkungan multifungsi yang dikembangkan oleh Bosch. Sensor ini dapat mengukur tiga parameter utama lingkungan: suhu, kelembapan relatif, dan tekanan atmosfer. Fungsionalitas yang luas ini menjadikan

BME280 sebagai pilihan yang ideal untuk berbagai aplikasi IoT yang membutuhkan pemantauan kondisi lingkungan. Tidak hanya itu sensor BME-280 dirancang dengan konsumsi daya yang sangat rendah, membuatnya ideal untuk perangkat portable dan aplikasi berbasis baterai, dengan ukuran nya yang kecil, sensor ini mudah diintegrasikan ke dalam berbagai perangkat.

Sensor BME-280 menggunakan protocol komunikasi 12C yang memanfaatkan dua jalur untuk mentransfer data: SCL (Serial Clock Line) dan SDA (Serial Data Line). Protokol 12C komunikasi antar perangkat di dalam satu bus yang hanya menggunakan dua jalur, yaitu SCL untuk sinyal jam dan SDA untuk data. SCL adalah jalur yang mengirimkan sinyal jam untuk menyinkronkan transfer data antar perangkat. Setiap perangkat yang terhubung ke bus 12C menggunakan sinyal jam ini untuk menerima data dengan sedangkan SDA adalah jalur yang mengangkut data yang diikirim dari perangkat master (misalnya mikrokontroler) ke perangkat slave (misalnya sensor BME-280) (Bosch, 2014).

Penelitian yang dilakukan oleh Ahmed et al. (2021) menunjukkan bahwa sensor BME-280 adalah perangkat yang sangat efisien dalam mengukur berbagai parameter lingkungan dengan tingkat akurasi yang tinggi. Sensor ini bekerja dengan konsumsi daya rendah, yang membuatnya ideal untuk digunakan dalam aplikasi berbasis IoT dengan daya baterai terbatas. BME280 juga mendukung berbagai protokol komunikasi seperti I2C dan SPI, yang memungkinkannya untuk dengan mudah diintegrasikan dengan platform mikrokontroler seperti ESP32 untuk pemantauan jarak jauh dan pengumpulan data secara real-time.

Sensor BME-280 memiliki peran yang cukup berguna dalam pemantauan radiasi dan lingkungan. Meskipun sensor BME-280 tidak secara langsung

mengukur radiasi, tetapi ia memberikan data lingkungan yang dapat digunakan untuk memahami faktor-faktor yang mempengaruhi radiasi.

2.9 Sensor GPS (NEO-6M)

Modul GPS NEO-6M merupakan modul yang memiliki performa baik. Time to First Fix (TIFF) adalah salah satu parameter penting dimana didefinisikan sebagai kecepatan modul GPS untuk dapat mengakses data almanac dan ephemeris dari satelit. TIFF, dan NEO-6M ini memiliki TIFF paling lama 27 detik. Spesifikasi lain adalah, tegangan kerja maksimum 3,6 V, konsumsi arus maksimum 67 mA, tipe receiver 50 channel, update rate 5 Hz, akurasi kecepatan 0,1 m/s, dengan komunikasi interface UART (U-Blox, 2017). Pin yang dimiliki modul GPS Neo-6M berjumlah 4 pin yang diantaranya adalah. Pin GND yang berfungsi sebagai ground dari tegangan sumber dari modul GPS Neo-6M, Pin VCC yang berfungsi sebagai port untuk pemberian tegangan sumber pada modul GPS Neo-6M, pin TX(transmitter) dan pin RX(Receiver) yang digunakan dalam komunikasi serial.



Gambar 2. 6 Sensor GPS NEO-6M

Receiver GPS ini termasuk yang fleksibel dengan harga yang lumayan murah, serta memiliki ukuran yang lumayan kecil. Modul penerima GPS Ublox Neo-6m ini mampu mengolah hingga 50 *channel* sinyal yang akurat dan tepat dengan waktu *Cold* TTFF kurang dari 27 detik. Sebagai perbandingan, rata-rata GPS *Navigator*

yang diproduksi secara umum memiliki waktu *Cold* TTFF lebih dari 50 detik (U-Blox, 2017).

Modul ini dirancang untuk menerima sinyal GPS dari satelit dan memberikan informasi mengenai posisi (lintang, bujur, dan ketinggian) di permukaan bumi. Prinsip dasar kerja GPS adalah menggunakan sinyal dari minimal empat satelit untuk melakukan trilaterasi, yang memungkinkan perangkat untuk menentukan posisinya dengan akurat. GPS Neo 6M mengimplementasikan teknologi ini dengan dukungan kemampuan penerimaan sinyal yang baik, meskipun hanya menggunakan perangkat yang relatif kecil dan murah. Modul ini juga dilengkapi dengan antena built-in yang cukup sensitif untuk menangkap sinyal dari satelit, meskipun dapat juga menggunakan antena eksternal untuk meningkatkan kinerja penerimaan sinyal dalam kondisi tertentu. GPS Neo 6M memiliki daya rendah, menjadikannya pilihan ideal untuk aplikasi berbasis baterai atau perangkat yang membutuhkan konsumsi energi yang efisien (Fadhilah & Putra, 2023).

2.10 TFT ILI9488

TFT (*Thin-Film Transistor*) ILI 9488 adalah sebuah modul layar sentuh (LCD) yang menggunakan teknologi TFT untuk menampilkan gambar dan informasi dalam bentuk grafis. Modul ini dilengkapi dengan driver ILI 9488, yang merupakan controller layar TFT dengan resolusi tinggi yang digunakan dalam banyak aplikasi, mulai dari perangkat portabel seperti smartphone hingga sistem embedded berbasis mikrokontroler. Driver ini mendukung antarmuka komunikasi SPI (Serial Peripheral Interface) untuk menghubungkan dengan mikrokontroler atau sistem lainnya. Teknologi TFT memungkinkan layar untuk menampilkan gambar dengan

kualitas tinggi dan warna yang lebih tajam dibandingkan dengan jenis layar lain, seperti layar LCD berbasis LED (Putra & Wijaya, 2021).

Prinsip dasar kerja dari modul TFT ILI 9488 adalah dengan menggunakan teknologi transistor untuk mengontrol cahaya di setiap piksel layar. Setiap piksel pada layar TFT terdiri dari tiga sub-piksel (merah, hijau, biru), yang masing-masing dikendalikan oleh transistor untuk menghasilkan warna tertentu. Driver ILI 9488 bertugas untuk mengendalikan setiap transistor dalam array piksel, memastikan



Gambar 2. 7 TFT ILI9488 *Module*

bahwa setiap piksel menampilkan warna yang tepat sesuai dengan data yang diterima dari mikrokontroler. ILI 9488 memanfaatkan antarmuka SPI untuk komunikasi dengan perangkat lain, dan dapat beroperasi dengan frekuensi tinggi untuk memberikan tampilan yang lebih responsif dan halus (Nugroho & Rahmat, 2020).

Cara kerja TFT ILI 9488 dimulai dengan menerima data gambar atau informasi melalui antarmuka SPI dari mikrokontroler atau sistem sumber data lainnya. Setelah data diterima, driver ILI 9488 mengonversi data digital tersebut menjadi sinyal yang mengatur transistor pada setiap piksel di layar TFT. Setiap transistor diatur untuk mengontrol intensitas cahaya dan warna yang ditampilkan pada layar. Dengan demikian, TFT ILI 9488 menghasilkan gambar atau teks yang ditampilkan dengan jelas dan presisi. Selain itu, modul ini mendukung mode orientasi layar yang

dapat diprogram, memungkinkan pengguna untuk menampilkan gambar baik dalam posisi landscape (horizontal) maupun portrait (vertikal) sesuai kebutuhan. Dalam sistem touchscreen, modul ILI 9488 juga menerima input sentuhan dari pengguna yang diterjemahkan menjadi koordinat pada layar, memungkinkan kontrol interaktif pada antarmuka grafis.

TFT ILI 9488 memiliki berbagai spesifikasi teknis yang menjadikannya pilihan yang populer untuk berbagai aplikasi. Layar ini umumnya memiliki resolusi 480x320 piksel, yang memberikan kualitas tampilan yang cukup tinggi untuk aplikasi grafis. TFT ILI 9488 bekerja dengan tegangan operasi sekitar 3.3V hingga 5V, membuatnya kompatibel dengan berbagai mikrokontroler dan platform pengembangan. Layar ini juga dilengkapi dengan touchscreen kapasitif yang memungkinkan interaksi langsung melalui sentuhan. Modul ini mendukung berbagai mode tampilan, termasuk mode portrait dan landscape, serta mampu memberikan tampilan dengan kedalaman warna 16-bit (RGB565). Ini memungkinkan pengguna untuk menampilkan gambar dengan lebih banyak nuansa warna, yang sangat penting dalam aplikasi visual. Selain itu, ILI 9488 memiliki konsumsi daya yang efisien, membuatnya cocok untuk aplikasi berbasis baterai yang membutuhkan penghematan daya.

2.11 Software Arduino IDE 1.8.9

Arduino IDE adalah software yang digunakan untuk membuat sketch pemrograman atau dengan kata lain Arduino IDE sebagai media untuk pemrograman pada board yang ingin diprogram. Arduino IDE ini berguna untuk mengedit, membuat, meng-*upload* ke board yang ditentukan, dan meng-*coding* program tertentu. Arduino IDE dibuat dari Bahasa pemrograman JAVA, yang

dilengkapi dengan library C/C++ (wiring), yang membuat operasi input/output lebih mudah. Software Arduino IDE ini, kita dapat memodelkan sifat dari parameter rangkaian analog dan digital. Kemampuan yang disediakan Arduino IDE adalah dapat memodelkan berbagai rancangan rangkaian, menguji suatu rangkaian dengan berbagai kemungkinan komponen, memeriksa sifat dari keseluruhan rangkaian dengan melakukan analisis AC/DC transient (Kamal, 2023)

Kode program yang ditulis untuk Arduino dikenal sebagai sketsa. Perangkat lunak yang digunakan untuk mengembangkan sketsa tersebut untuk Arduino umumnya dikenal sebagai Arduino IDE. IDE ini berisi bagian- bagian berikut di dalamnya (Leo, 2016):

Editor Teks : Disinilah kode yang disederhanakan dapat ditulis menggunakan versi C++ yang disederhanakan Bahasa pemrograman

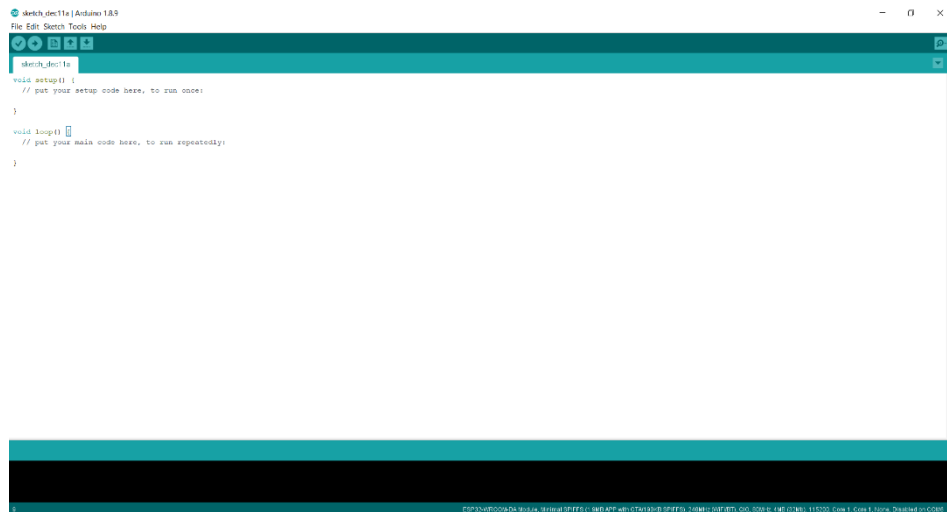
Area Pesan : Menampilkan kesalahan dan juga memberikan umpan balik tentang penyimpanan dan pengeksporan kode.

Teks : Konsol menampilkan keluaran teks oleh lingkungan Arduino termasuk kesalahan lengkap pesan dan informasi lainnya

Console toolbar : Toolbar ini berisi berbagai tombol Verify, Upload, New Open, Save, dan Serial Monitor, Di sudut kanan bawah jendela, ditampilkan Development Board dan Serial Port yang digunakan.

Antarmuka pengguna Arduino IDE dirancang agar mudah digunakan, bahkan untuk pemula. Arduino IDE 1.8.9 menawarkan berbagai fitur, termasuk editor kode dengan penyempurnaan untuk sintaksis dan debugging. Kesederhanaan antarmuka memudahkan pengguna baru untuk cepat memahami cara kerja perangkat ini (Patel

dan Kumar, 2019). Hal ini diperkuat oleh penelitian di Indonesia yang dilakukan oleh Arifin et al. (2019), yang menunjukkan bahwa antarmuka pengguna Arduino



Gambar 2. 8 Tampilan Arduino IDE 1.8.9

IDE telah berhasil menarik perhatian mahasiswa di sekolah menengah untuk mempelajari dasar-dasar pemrograman dan elektronik. Fitur seperti auto-complete dan notifikasi error juga mempermudah debugging, yang sangat penting dalam pengembangan perangkat keras.

Arduino IDE 1.8.9 mendukung berbagai sistem operasi, termasuk Windows, macOS, dan Linux, yang menjadikannya sangat fleksibel dan dapat diakses oleh banyak pengembang di seluruh dunia. Tero (2019) menjelaskan bahwa kompatibilitas lintas platform ini memungkinkan pengembang untuk memilih sistem operasi yang sesuai dengan kebutuhan mereka tanpa terbatas oleh perangkat keras atau perangkat lunak tertentu.

Salah satu aspek penting dari Arduino IDE 1.8.9 adalah pembaruan yang rutin dilakukan untuk meningkatkan stabilitas dan kinerja perangkat lunak. Pembaruan ini memperbaiki bug dan meningkatkan kecepatan dalam proses pengunggahan kode, yang memudahkan pengembang dalam menjalankan proyek mereka.

Drosdowski & Müller (2019) menekankan bahwa pembaruan rutin meningkatkan performa dan kompatibilitas papan Arduino, yang meminimalkan gangguan dalam pengembangan. Penelitian di Indonesia oleh Rahman et al. (2019) juga menunjukkan bahwa pembaruan ini mengurangi jumlah error yang ditemukan pengguna, meningkatkan efisiensi pengembangan proyek embedded di kalangan mahasiswa.

2.12 Penelitian Terdahulu

Berikut ini merupakan penelitian-penelitian yang telah dilakukan sebelumnya :

1. *“Desain dan Perekayasaan Area Monitor Berbasis Mikrokontroler Menggunakan Detektor Geiger-Müller”* oleh (Nugroho & Setyadi, 2005).

Penelitian ini bertujuan mengembangkan alat monitor berbasis mikrokontroler menggunakan detektor *Geiger-Müller*. Perangkat ini mampu mendeteksi radiasi gamma dengan akurasi tinggi di area terbatas. Data diproses oleh mikrokontroler sederhana dan ditampilkan pada layar local tanpa adanya konektivitas jaringan. Persamaan dengan penelitian penulis adalah penggunaan detektor *Geiger-Müller* sebagai sensor utama untuk deteksi radiasi gamma. Namun, penelitian ini tidak mencakup teknologi IoT, koneksi jaringan, atau integrasi pemantauan parameter lingkungan terpadu yang menjadi focus penelitian penulis.

2. *“Pemanfaatan Modul Geiger-Müller untuk Mendeteksi Radiasi pada Pengolahan Limbah B3 Rumah Sakit”* oleh (Mahda, 2022)

Penelitian ini bertujuan memanfaatkan modul *Geiger-Müller* untuk mendeteksi radiasi pada pengolahan limbah B3 rumah sakit. Penelitian ini

menekankan pentingnya alat pendeteksi radiasi untuk menjaga Tingkat radiasi pada limbah rumah sakit sesuai standar keselamatan. Persamaan dengan penelitian penulis adalah penggunaan detektor *Geiger-Müller* untuk mendeteksi radiasi gamma. Namun, penelitian ini terbatas pada aplikasi medis tertentu, sedangkan penelitian penulis mencakup pemantauan lingkungan secara umum dengan integrasi teknologi IoT.

3. *“Rancang Bangun Sistem Monitoring Kualitas Udara Menggunakan ESP32 dan Protokol MQTT”* oleh (Himawan, 2021).

Penelitian ini bertujuan mengembangkan sistem monitoring kualitas udara berbasis ESP32 dengan protocol MQTT. Penelitian ini memanfaatkan sensor gas (misalnya MQ 7) untuk memantau kualitas udara secara real-time. Protokol komunikasi MQTT memungkinkan pengiriman data secara efisien ke platform IoT. Persamaan dengan penelitian penulis adalah penggunaan ESP32 dan teknologi IoT untuk pemantauan real-time. Namun, penelitian ini hanya berfokus pada kualitas udara tanpa mencakup deteksi radiasi gamma dan pemantauan lingkungan terpadu seperti yang dilakukan dalam penelitian penulis.

4. *“Rancang Bangun Sistem Pemantauan Kualitas Udara Berbasis IoT dengan Sensor DHT11 dan Sensor MQ135”* oleh (Muttaqin et al, 2024).

Penelitian ini bertujuan merancang sistem pemantauan kualitas udara berbasis IoT menggunakan sensor DHT11 untuk mengukur suhu dan kelembapan, serta sensor MQ135 untuk mendeteksi gas polutan. Data yang dikumpulkan dikirim secara real-time ke aplikasi berbasis web untuk monitoring jarak jauh. Sistem ini dirancang untuk memantau kondisi

kualitas udara di ruang publik atau lingkungan kerja. Persamaan dengan penelitian penulis adalah adanya integrasi teknologi IoT untuk pemantauan lingkungan. Mikrokontroler yang digunakan juga menggunakan mikrokontroler EPS32 yang mempunyai penyimpanan ke cloud. Selain itu, kedua penelitian sama-sama berfokus pada pemantauan parameter lingkungan untuk tujuan pengawasan kondisi lingkungan. Perbedaan dengan penelitian penulis adalah penelitian ini hanya mencakup kualitas udara, tanpa melibatkan deteksi radiasi gamma. Sedangkan penelitian penulis melibatkan deteksi radiasi gamma. Sensor untuk parameter lingkungan juga berbeda, penulis menggunakan sensor BME-280 dan sensor GPS NEO-6M untuk memantau parameter lingkungan.

5. "*Pemantauan Paparan Radiasi Lingkungan Terpadu dengan Komunikasi GSM/GPRS*" oleh (Ikhsan Shobari et al., 2014).

Penelitian ini menghasilkan sistem pemantauan paparan radiasi lingkungan terpadu berbasis komunikasi GSM/GPRS. Sistem ini mampu mengukur parameter seperti radiasi, suhu udara, kecepatan angin, dan curah hujan secara real-time dan mengirimkan data melalui layanan SMS. Modul Arduino berbasis ATmega 328 digunakan untuk akuisisi data dari sensor, kemudian data dikirimkan ke computer proses menggunakan teknologi GSM.GPRS. Persamaan dengan penelitian penulis adalah sama-sama memantau radiasi gamma sebagai salah satu parameter utama. Detektor yang digunakan juga sama-sama menggunakan detektor *Geiger-Müller*. Perbedaan dengan penelitian penulis adalah penelitian ini menggunakan GSM/GPRS, sedangkan penelitian penulis menggunakan teknologi IoT

berbasis Wi-Fi melalui modul ESP32. Penelitian ini mengirimkan data melalui SMS dalam format CSV sederhana, sedangkan penelitian penulis memiliki akses data yang lebih fleksibel melalui platform berbasis cloud/spreadsheet.

Tabel 2. 1 Penelitian Terdahulu

No	Judul Penelitian	Peneliti dan Tahun	Hasil Penelitian	Penjelasan	Perbedaan	Persamaan
1.	Desain dan Perencanaan Area Monitor Berbasis Mikrokontroler Menggunakan Detektor Geiger-Müller	Nugroho & Setyadi, 2005	Menghasilkan alat monitor radiasi dengan detektor <i>Geiger-Müller</i> berbasis mikrokontroler. Perangkat mampu mendeteksi radiasi gamma di area terbatas dengan akurasi tinggi.	Menggunakan tabung Geiger-Müller untuk mendeteksi radiasi gamma. Data diolah menggunakan mikrokontroler sederhana dan ditampilkan di layar lokal tanpa jaringan.	Sama-sama menggunakan detektor <i>Geiger-Müller</i> sebagai sensor utama dan fokus pada deteksi radiasi gamma.	Tidak mencakup teknologi IoT, koneksi jaringan, atau parameter lingkungan terpadu. Tidak menggunakan ESP32.
2.	Pemanfaatan Modul Geiger-Müller untuk Mendeteksi Radiasi pada Pengolahan Limbah B3 Rumah Sakit	Mahda, 2022	Perangkat pendeteksi radiasi digunakan untuk memastikan tingkat radiasi pada limbah rumah sakit tetap aman sesuai standar.	Difokuskan untuk aplikasi spesifik di pengolahan limbah medis berpotensi radioaktif menggunakan detektor <i>Geiger-Müller</i> .	Sama-sama menggunakan detektor <i>Geiger-Müller</i> untuk mendeteksi radiasi gamma.	Hanya untuk aplikasi medis. Tidak menggunakan IoT, ESP32, atau monitoring lingkungan terpadu.

3.	Rancang Bangun Sistem Monitoring Kualitas Udara Menggunakan ESP32 dan Protokol MQTT	Himawan, 2021	Sistem berhasil memonitor kualitas udara secara real-time dengan integrasi sensor gas dan protokol komunikasi MQTT berbasis ESP32.	Menggunakan ESP32 untuk konektivitas IoT dan protokol MQTT untuk data real-time. Difokuskan untuk pemantauan kualitas udara.	Sama-sama menggunakan ESP32 dan teknologi IoT untuk monitoring real-time.	Fokus pada kualitas udara, bukan deteksi radiasi gamma atau lingkungan terpadu.
4.	Rancang Bangun Sistem Pemantauan Kualitas Udara Berbasis IoT dengan Sensor DHT11 dan Sensor MQ135	Muttaqin et al., 2024	Menghasilkan perancangan sistem pemantauan kualitas udara berbasis IoT menggunakan sensor DHT11 untuk mengukur suhu dan kelembapan, serta sensor MQ135 untuk mendeteksi gas polutan. Sistem ini dirancang untuk memantau kondisi kualitas udara di ruang publik atau lingkungan kerja.	Data yang dikumpulkan dikirim secara real-time ke aplikasi berbasis web untuk monitoring jarak jauh.	Adanya integrasi teknologi IoT untuk pemantauan lingkungan. Mikrokontroler yang digunakan juga menggunakan mikrokontroler ESP32 yang menyai penyimpanan ke cloud.	Penelitian ini hanya mencakup kualitas udara, tanpa melibatkan deteksi radiasi gamma. Sensor yang digunakan penulis sensor BME-280 dan sensor GPS NEO-6M.

					Selain itu, kedua [enelitian sama-sama berfokus pada pemanfaatan parameter lingkungan untuk tujuan pengawasan kondisi lingkungan.	
--	--	--	--	--	---	--

BAB III

METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian tentang “Rancang Bangun Perangkat Monitor Radiasi Gamma Berbasis Mikrokontroler ESP32 Menggunakan Detektor *Geiger-Müller* untuk Pemantauan Radiasi dan Lingkungan” ini dilaksanakan mulai pada bulan Agustus sampai bulan November 2024 di Badan Riset dan Inovasi Nasional (BRIN) Yogyakarta.

3.2 Jenis Penelitian

Jenis Penelitian yang dibuat adalah penelitian rancang bangun monitoring radiasi gamma berbasis mikrokontroler ESP 32 menggunakan detektor *Geiger-Müller* untuk pemantauan radiasi dan lingkungan. Jenis Output yang diharapkan pada penelitian kali ini adalah untuk mendeteksi paparan radiasi dan lingkungan yang terjadi di area lingkungan reaktor kartini.

3.3 Alat dan Bahan Penelitian

3.3.1 Alat-Alat Penelitian

Adapun alat-alat yang dibutuhkan dalam penelitian ini yaitu :

- a. Power Supply (GW Instek GPE-4323)
- b. Multimeter (Fluke 117, Sanwa 510A)
- c. Oscilloscope (Texttronix TDS 2024C)
- d. Soldering Kit
- e. Signal Generator (GW Funtion GeneratorGFG-8015G)
- f. Laptop/Komputer
- g. Arduino IDE 1.8.9

- h. Diptrace Launcher

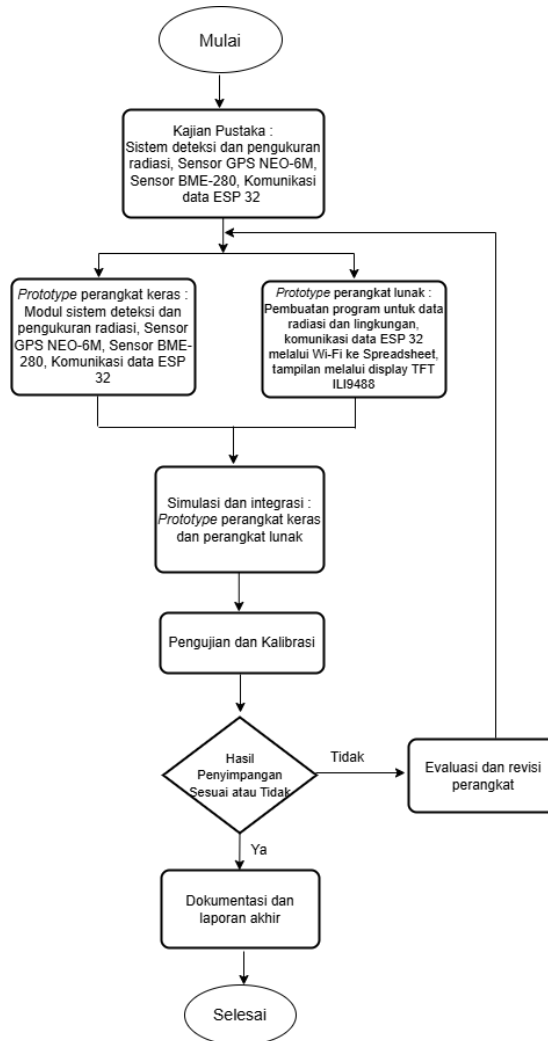
3.3.2 Bahan-Bahan Penelitian

Adapun bahan-bahan yang dibutuhkan yaitu :

- a. Kit Detektor *Geiger-Müller* (GM) 4011
- b. Mikrokontroler ESP32 WROOM DEVKIT V1
- c. Unit Display (Layar LCD/LED) ILI9488
- d. Papan Sirkuit Cetak (PCB)
- e. Resistor, kapasitor, dioda, transistor, transformator dan IC555
(Integrated Circuit)
- f. Papan Breadboard
- g. Kabel Jumper
- h. Sumber Radiasi (Cesium-137)
- i. Buzzer
- j. LED
- k. Powerbank
- l. Sensor Posisi (GPS NEO-6M)
- m. Sensor Lingkungan (BME280)
- n. Jam Digital
- o. *Track Logger*

3.4 Prosedur Penelitian

Prosedur penelitian dari rancang bangun perangkat monitor radiasi gamma berbasis mikrokontroler ESP32 menggunakan detektor *Geiger-Müller* ditunjukkan pada gambar berikut :



Gambar 3. 1 Prosedur Penelitian Monitoring Radiasi dan Lingkungan

Berikut merupakan penjelasan mengenai tahapan-tahapan dalam prosedur penelitian dari monitoring radiasi dan lingkungan :

- a. Kajian pustaka mengenai sistem deteksi dan pengukuran radiasi, Sensor GPS NEO-6M, Sensor BME-280, dan komunikasi data EPS 32 dilakukan.
- b. Peralatan dan bahan yang akan dirangkai disiapkan.
- c. Modul Surveymeter Kit dirangkai dengan pre-amplifier dan catu daya tegangan tinggi sehingga membentuk satu kesatuan sensor paparan radiasi yang berfungsi untuk mendeteksi dan mengukur tingkat radiasi lingkungan.
- d. Sensor BME-280, Sensor GPS NEO-6M, dan Sensor Radiasi dirangkai.

- e. Detektor *Geiger-Müller*, Sensor BME-280, dan Sensor GPS NEO-6M diintegrasikan dengan mikrokontroler ESP32.
- f. Pengujian Sensor Radiasi dengan sumber radioaktif standar dilakukan, Pengujian Sensor BME-280, Pengujian Sensor GPS-NEO-6M.
- g. Program untuk komunikasi data berbasis Wi-Fi dibuat, dengan pengiriman data ke platform IoT (Spreadsheet).
- h. Antarmuka berbasis spreadsheet yang akan menampilkan data pemantauan radiasi dan lingkungan melalui display monitor TFT ILI9488 dikembangkan.
- i. Data yang dihasilkan kemudian dikalibrasikan menggunakan perangkat referensi atau alat standar yang sudah terverifikasi.
- j. Evaluasi dilakukan dan revisi dilakukan jika eror masih di atas 5%, kemudian pengujian dilakukan kembali.
- k. Dokumentasi untuk setiap tahapan penelitian dilakukan, dan dokumentasi akhir setelah evaluasi tanpa revisi diselesaikan.

3.4.1 Studi Literatur

Studi literatur merupakan tahapan penting dalam menelaah, mengkaji, dan mengumpulkan berbagai sumber pustaka atau referensi ilmiah yang relevan dengan topik penelitian. Tujuan utama dari studi literatur ini adalah untuk memahami konsep-konsep dasar yang berkaitan dengan objek penelitian, seperti teori, metode, teknologi serta perkembangan terbaru yang telah dilakukan oleh peneliti sebelumnya.

Pada penelitian sebelumnya, sistem monitoring paparan radiasi banyak yang belum menggunakan teknologi IoT dan belum menggunakan parameter lingkungan yang juga untuk memonitoring lingkungan, oleh karena itu pada penelitian kali

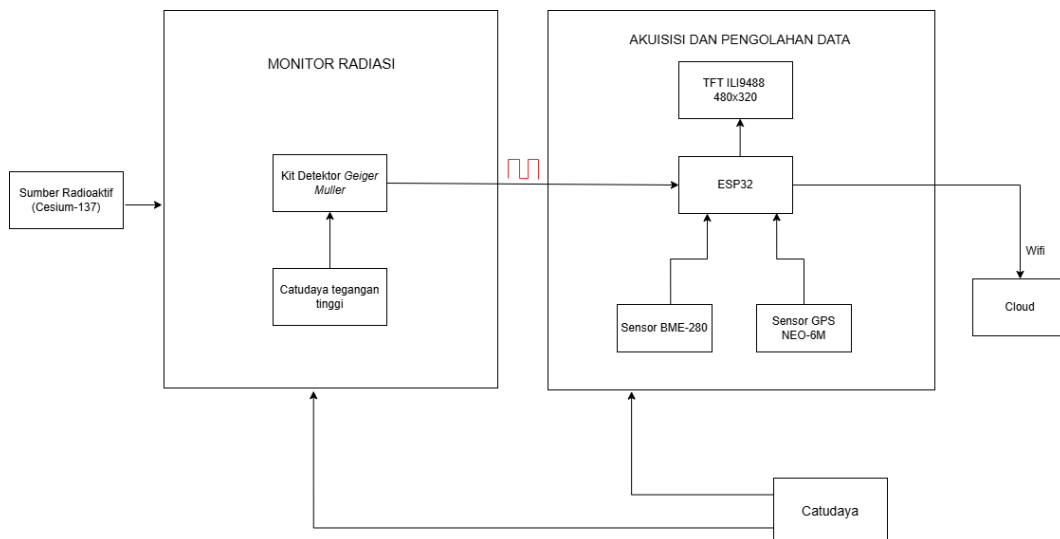
mengembangkan sistem monitoring paparan radiasi dan lingkungan yang diintegrasikan secara terpadu.

3.5 Perancangan Sistem

Perancangan Sistem pada penelitian kali ini yaitu mencakup pada perancangan perangkat keras (*Hardware*) dan perancangan program.

3.5.1 Perancangan Perangkat Keras (*Hardware*)

Perancangan modul perangkat keras monitoring radiasi gamma berbasis mikrokontroler ESP 32 menggunakan detektor *Geiger-Müller* adalah sebagai berikut :



Gambar 3. 2 Blok Diagram Perancangan Perangkat Keras (*Hardware*)

3.5.1.1 Integrasi Modul dengan Mikrokontroler ESP32

Dalam proses perancangan sistem monitoring radiasi dan lingkungan berbasis mikrokontroler, diperlukan integrasi beberapa komponen elektronik untuk membentuk satu kesatuan sistem yang mampu melakukan akuisisi data, pemrosesan, tampilan informasi, serta pengiriman data secara nirkabel. Mikrokontroler ESP32 digunakan sebagai pusat kendali utama karena memiliki

kemampuan pemrosesan data yang baik, dilengkapi dengan modul Wi-Fi, serta mendukung berbagai protokol komunikasi seperti UART, SP, dan I2C.

Rangkaian sistem terdiri atas beberapa komponen utama, yaitu sensor radiasi berbasis Modul Surveymeter Kit, sensor lingkungan BME-280 untuk mengukur suhu, kelembapan, dan tekanan udara, serta modul GPS NEO-6M untuk menentukan posisi geografis pengukuran, layar TFT ILI9488 digunakan sebagai antarmuka visual untuk menampilkan data hasil pengukuran yang sebagaimana ditunjukkan pada gambar 3.3.

Mikrokontroler ESP 32 sebagai pusat pengendali utama yang berfungsi untuk membaca data dari masing-masing sensor, mengolah informasi, serta menampilkan dan mengirimkan data ke platform penyimpanan berbasis cloud.

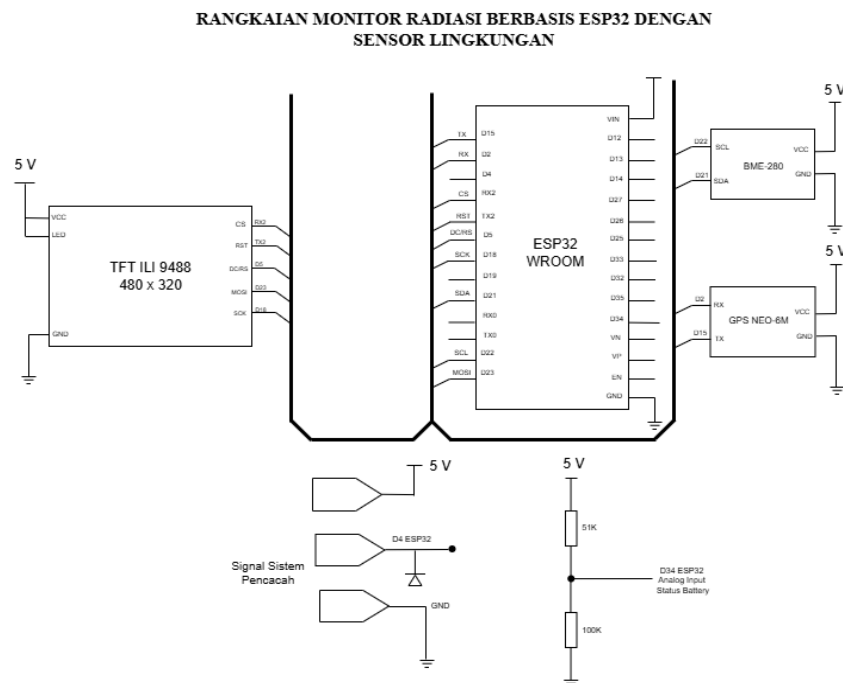
Layar TFT ILI9488 digunakan sebagai antarmuka visual untuk menampilkan data hasil pengukuran, layar ini menggunakan komunikasi Serial Peripheral Interface (SPI) yang dihubungkan ke ESP32 dengan konfigurasi pin sebagai berikut: pin CS pada TFT dihubungkan ke RX2 pada ESP32, pin RST dihubungkan ke pin TX2, pin DC/RS dihubungkan ke D5, pin MOSI dihubungkan ke D23, pin SCK dihubungkan ke pin D18, kemudian catu daya layar diberikan melalui pin VCC dan GND yang terhubung ke jalur catu daya dan GND ESP32.

Sensor BME-280 menggunakan protokol komunikasi I2C. Pin SDA pada BME-280 dihubungkan ke D21 ESP32, sedangkan pin SCL dihubungkan ke D22. Pin VCC dan GND pada sensor terhubung ke sumber tegangan 3.3 V dan GND pada ESP32.

Modul GPS NEO-6M ini berkomunikasi dengan ESP32 melalui antarmuka UART. Pin TX pada modul GPS NEO-6M dihubungkan ke pin D15 pada ESP32, pin RX dihubungkan ke pin D2. Modul tersebut mendapatkan catu daya dari pin VCC dan GND EPS32.

Sistem pendeteksi radiasi, kit detektor *geiger muller* dihubungkan ke ESP32 melalui pin D4 dan GND. Pin ini berfungsi sebagai input digital untuk menghitung jumlah pulsa radiasi yang diterima dari detektor.

Selain itu, untuk mengetahui kondisi daya, digunakan rangkaian pembagi tegangan, yang terdiri dari dua resistor, masing-masing 100 k Ω dan 51 k Ω , yang terhubung secara seri. Tegangan output dari titik tengah kedua resistor dihubungkan ke pin analog D34 pada ESP32 sebagai input analog (ADC). Nilai yang dibaca dari pin ini menunjukkan tegangan baterai dan digunakan untuk memantau status sistem baterai sistem.



Gambar 3. 3 *Schematic* Rangkaian Monitor Radiasi dan Lingkungan

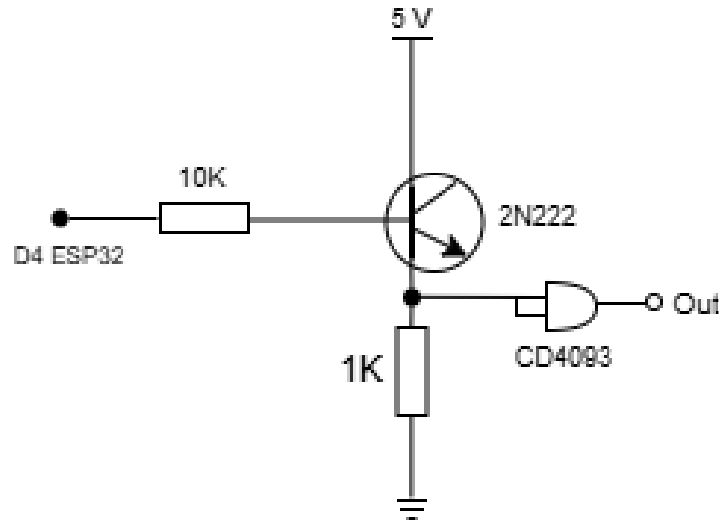
3.5.1.2 Integrasi Kit Detektor *Geiger-Müller* dengan ESP32

Penelitian ini dikaji sistem deteksi radiasi menggunakan skema *Geiger-Counter* yang nanti akan dihubungkan melalui tabung *Geiger-Müller* 4011. Sistem deteksi radiasi ini merupakan hasil adaptasi dari skematik detektor geiger berbasis NodeMCU, namun tanpa menggunakan mikrokontroler NodeMCU secara langsung. Yang diambil hanya bagian penguat dan pengolah sinyal dari skematik tersebut dan menerapkannya pada sistem dengan konfigurasi sendiri. Detektor radiasi yang digunakan adalah tabung *Geiger-Müller* 4011, yang memiliki karakteristik memerlukan tegangan tinggi sebesar 330 V.

Pulsa radiasi yang dihasilkan oleh tabung *Geiger-Müller* 4011 sangat kecil dan berdurasi singkat, sehingga tidak dapat langsung digunakan oleh mikrokontroler. Oleh karena itu, sinyal mentah dari tabung diproses terlebih dahulu oleh rangkaian buffer yang terdapat pada kit dev warna hijau seperti tampak pada gambar (3.5). PCB ini dirancang menggunakan komponen SMD (*Surface Mount Device*) untuk menghemat ruang dan meningkatkan efisiensi rangkaian. Di dalamnya terdapat *buffer amplifier* yang berfungsi untuk menstabilkan dan memperkuat sinyal agar dapat dikenali secara digital.

Setelah sinyal diperkuat dan dibentuk menjadi pulsa logika oleh buffer ini, sinyal tersebut kemudian dikirim ke modul ESP32, yang berfungsi sebagai pusat pemrosesan dan komunikasi data. ESP32 menerima pulsa logika ini sebagai input digital dan menghitung jumlah pulsa sebagai representasi dari tingkat radiasi. Dengan demikian, sistem ini berhasil menggabungkan sensor analog berbasis tabung Geiger dengan platform mikrokontroler digital modern (ESP32), menjadikannya perangkat deteksi radiasi yang modular, portabel, dan siap untuk

integrasi lebih lanjut seperti logging, pemetaan radiasi berbasis GPS, atau pengiriman data secara nirkabel.

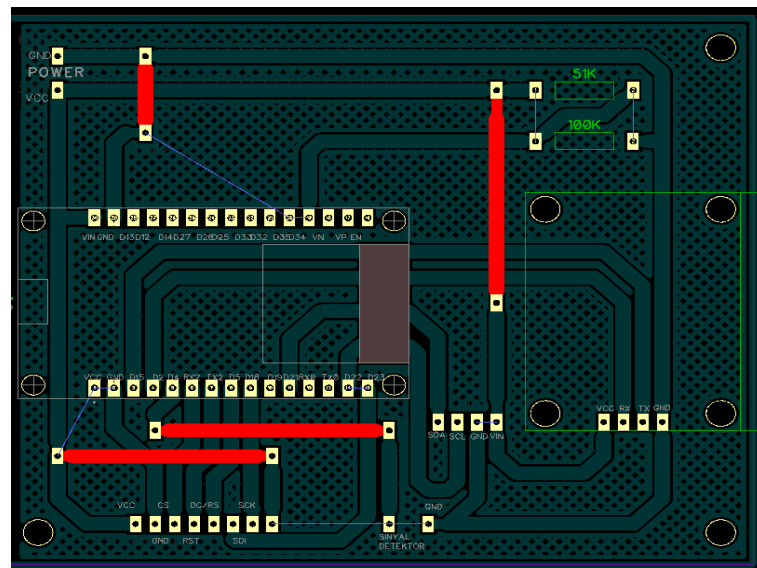


Gambar 3. 4 Rancangan *Schematic* Buffer Kit Detektor *Geiger-Müller*

3.5.1.3 Perancangan PCB Hardware

Rangkaian PCB (*Printed Circuit Board*) merupakan suatu media fisik yang digunakan untuk menghubungkan dan menyusun komponen-komponen elektronik secara sistematis dan permanen dalam satu kesatuan rangkaian elektronik. PCB berfungsi sebagai wadah pendukung mekanik serta jalur konduktif bagi sinyal listrik antar komponen. PCB berfungsi sebagai wadah pendukung mekanik serta jalur konduktif bagi sinyal listrik antar komponen. Desain PCB dibuat berdasarkan skematik rangkaian yang telah dirancang sebelumnya, Dimana jalur-jalur tembaga (*trace*) menggantikan penggunaan kabel penghubung konvensional sehingga menghasilkan rangkaian lebih rapi, dan tahan terhadap gangguan eksternal.

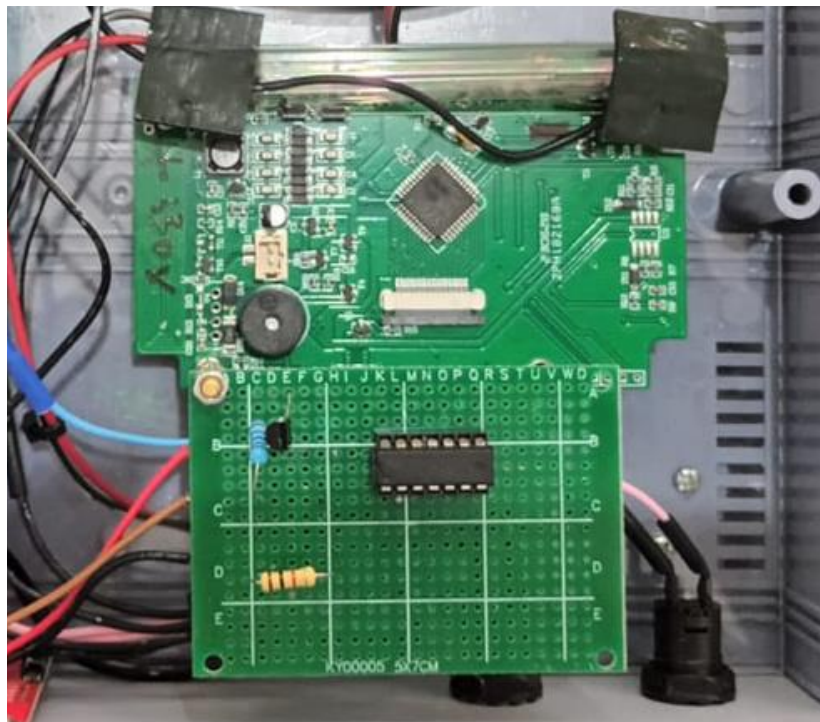
Pada tahap ini, rancangan PCB terdiri dari dua bagian utama. Bagian pertama adalah sistem dari sensor lingkungan ESP32 yang mencakup mikrokontroler ESP32 sebagai pusat pengendali dan pengolah data, serta koneksi ke berbagai sensor lingkungan, jalur komunikasi serial, dan input sinyal dari detektor GM seperti pada gambar 3.5.



Gambar 3. 5 PCB Monitor Radiasi dan Lingkungan

Bagian kedua adalah rangkaian Kit Detektor *Geiger Muller*, yang berfungsi sebagai unit deteksi radiasi terintegrasi dan menggantikan kebutuhan akan rangkaian *high voltage* eksternal yang biasa digunakan untuk tabung *Geiger-Müller*. Pada papan ini terdapat komponen-komponen utama seperti mikrokontroler SMD, transistor, resistor, kapasitor, serta buffer dan rangkaian pemroses sinyal. Ketika tabung GM mendeteksi partikel radiasi, sinyal pulsa kecil yang dihasilkan akan diperkuat melalui rangkaian transistor atau op-amp di PCB ini, kemudian dibersihkan atau dibentuk ulang oleh buffer agar stabil. Setelah itu, sinyal dikirim ke modul mikrokontroler eksternal, dalam hal ini ESP32, melalui koneksi kabel. Fungsi utama PCB ini adalah menjembatani

antara detektor fisik (tabung GM) dan sistem digital (ESP32), sehingga data deteksi radiasi bisa dibaca dan diproses secara digital untuk ditampilkan atau dikirim secara nirkabel. PCB ini juga dilengkapi dengan komponen SMD agar bentuknya lebih ringkas dan efisien daya. Biasanya daya untuk papan ini disuplai dari sumber 5V USB atau regulator eksternal, dan papan ini merupakan bagian penting dari keseluruhan sistem pendeteksian radiasi portabel.



Gambar 3. 6 PCB Kit Detektor *Geiger-Müller*

3.5.1.4 Output Display dan Spreadsheet Monitor Radiasi dan Lingkungan

Tahap kali ini yakni dilakukan perancangan antarmuka tampilan pada layar TFT (*Thin Film Transistor*) sebagai media output utama dari sistem monitoring radiasi berbasis mikrokontroler ESP32. Tampilan ini berfungsi sebagai penampil informasi mengenai parameter lingkungan serta tingkat paparan radiasi yang terdeteksi oleh Modul Surveymeter Kit. Desain antarmuka dirancang agar informatif dan mudah dipahami, dengan setiap elemen informasi yang

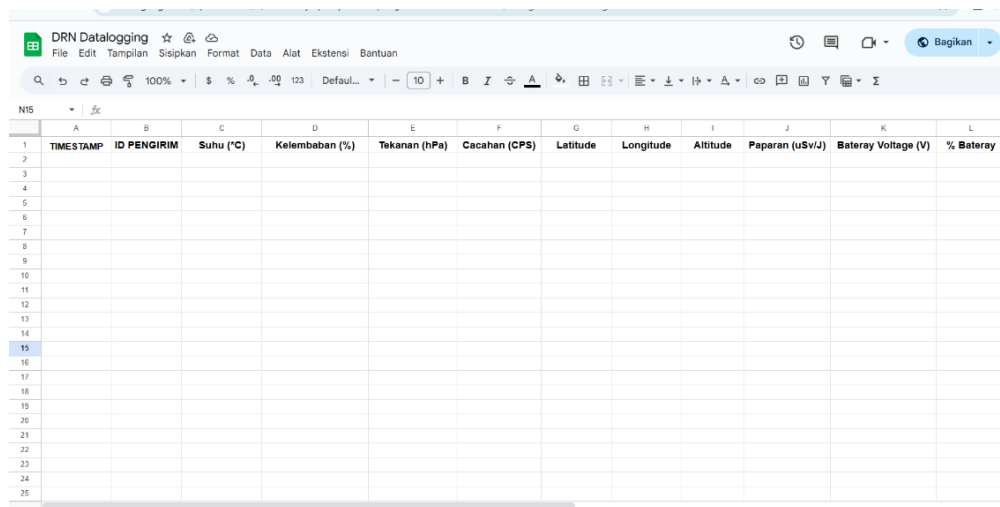
ditampilkan memiliki posisi yang telah ditentukan secara presisi melalui pemrograman pada Arduino IDE versi 1.8.9. Penentuan posisi tampilan ini diatur mennggunakan koordinat tertentu dengan fungsi seperti (x, y, z) yang mengatur letak teks pada layar.

MONITORING RADIASI			
Suhu :	*C	Paparan :	uSv/J
Kelembapan :	%	Cacahan :	CPM
Tekanan :	hPa	Baterai :	V
Lat :			
Long :			
Alt :			
ID Pengirim :		Jaringan :	
		IP Address :	
PAPARAN :			uSv/J

Gambar 3. 7 *Display* TFT Tampilan Monitor Radiasi dan Lingkungan

Gambar 3.7 menunjukkan parameter-parameter lingkungan yang terdiri dari suhu (°C), kelembapan (%), dan tekanan udara (hPa) ditampilkan secara berurutan di sisi kiri layar TFT 19488. Informasi geografis seperti latitude (Lat), longitude (Long), dan altitude (Alt) juga ditampilkan tepat di bawah parameter lingkungan tersebut, sehingga pengguna dapat mengetahui lokasi akurat perangkat saat mendeteksi radiasi. Di bagian kanan atas layar, ditampilkan parameter utama terkait radiasi, yaitu nilai paparan radiasi dalam satuan mikroSievert per jam ($\mu\text{Sv/h}$), nilai cacahan dalam satuan Count Per Minute (CPM), serta status tegangan baterai (V).

Seluruh tampilan ini diprogram sedemikian rupa agar setiap elemen muncul di lokasi yang konsisten dan tidak saling bertumpukan. Hal ini dicapai dengan mengatur koordinat x, y, dan z untuk setiap baris teks dan nilai menggunakan fungsi garis layar TFT penataan posisi dan warna ini tidak hanya bertujuan untuk estetika, tetapi juga mendukung keterbacaan data serta mempermudah pengguna dalam membaca dan menginterpretasikan informasi secara cepat.



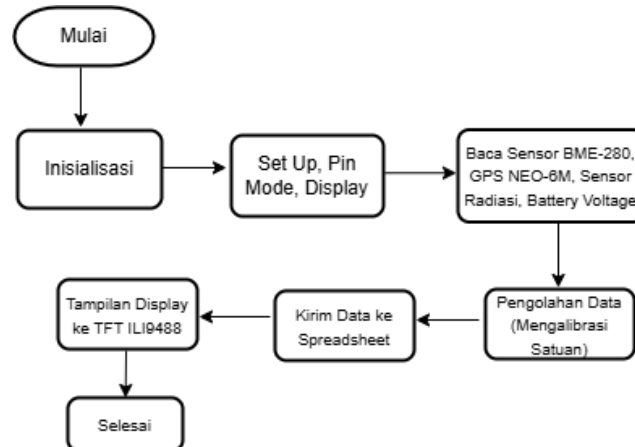
	A	B	C	D	E	F	G	H	I	J	K	L
	TIMESTAMP	ID PENGIRIM	Suhu (°C)	Kelembaban (%)	Tekanan (hPa)	Cacahan (CPS)	Latitude	Longitude	Altitude	Paparan (uSv/J)	Bateray Voltage (V)	% Bateray
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												
25												

Gambar 3. 8 Tampilan *Spreadsheet* Monitor Radiasi dan Lingkungan

Gambar 3.8 menunjukkan tampilan lembar kerja *Google Sheets* yang digunakan untuk mencatat data secara otomatis dari perangkat mikrokontroler, ESP32. Data yang ditampilkan pada sheet ini berasal dari berbagai sensor yang terpasang pada perangkat, seperti sensor suhu, kelembapan, tekanan udara, radiasi, serta modul GPS untuk pelacakan lokasi. Semua data ini dikirimkan secara real-time ke *Google Sheets* melalui integrasi dengan *Google Cloud*.

3.5.2 Perancangan Program Arduino 1.8.9

Perancangan perangkat monitor radiasi yang selanjutnya diolah menggunakan bahasa pemrograman yaitu menggunakan platform Arduino IDE 1.8.9, berikut diagram alir dari perancangan program :



Gambar 3. 9 Blok Diagram Perancangan Program

Tahap selanjutnya, dilakukan proses perancangan dan penulisan program menggunakan perangkat lunak Arduino IDE 1.8.9 yang bertujuan untuk mengontrol seluruh kerja dari perangkat keras sistem monitoring radiasi berbasis ESP32. Arduino IDE digunakan sebagai lingkungan pengembangan utama untuk memrogram mikrokontroler ESP32, karena menyediakan antarmuka yang sederhana namun lengkap, serta mendukung berbagai pustaka (*library*) untuk pengendalian sensor dan modul yang digunakan.

Proses perancangan program dimulai dari mendefinisikan fungsi utama yaitu `setup()` dan `loop()` sebagai struktur dasar dalam pemrograman arduino. Fungsi `setup()` digunakan untuk inisialisasi awal seperti konfigurasi pin input/output, pengaturan komunikasi serial, dan inisialisasi sensor. Sementara itu, fungsi `loop()` akan menjalankan instruksi secara berulang yang meliputi pembacaan sensor,

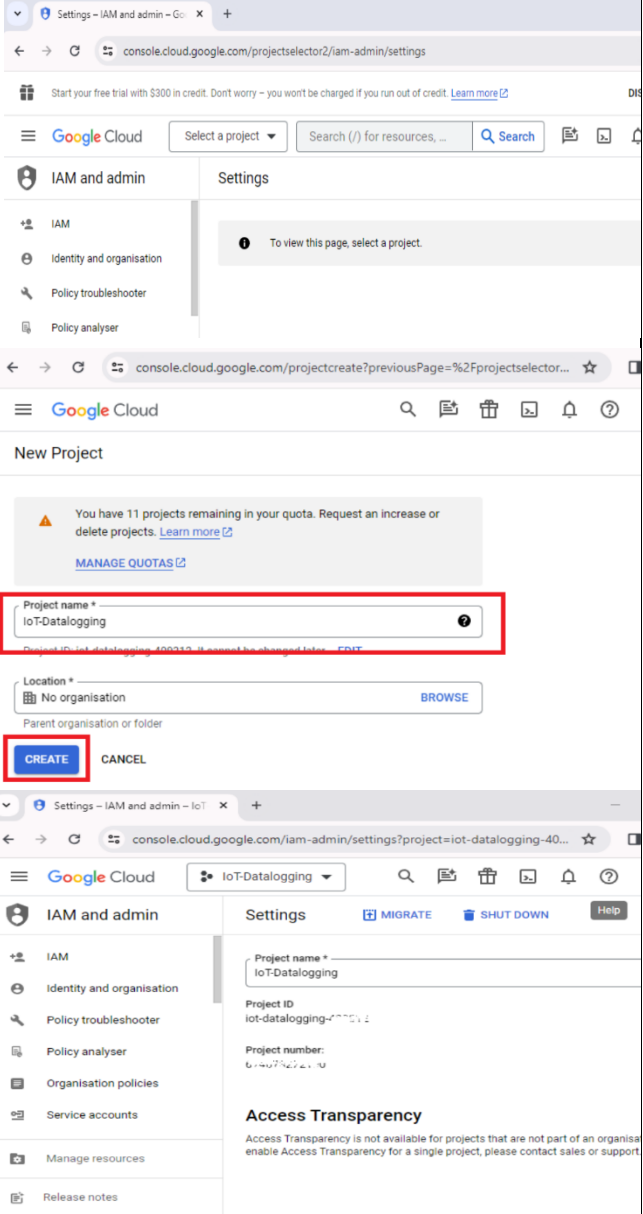
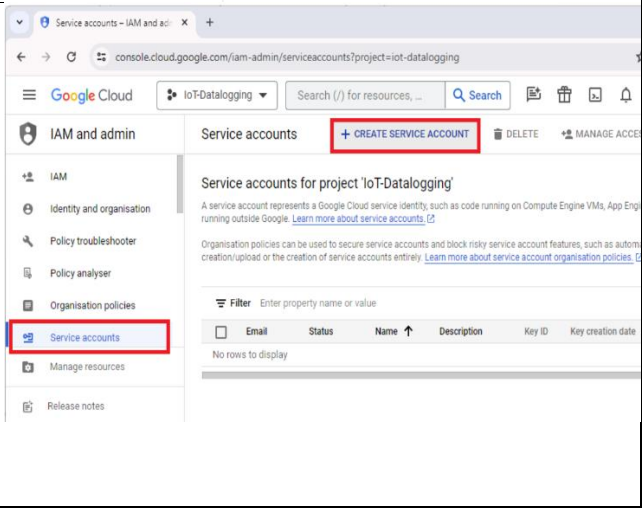
pengolahan data, dan pengiriman data ke server serta pembaruan tampilan pada layar TFT.

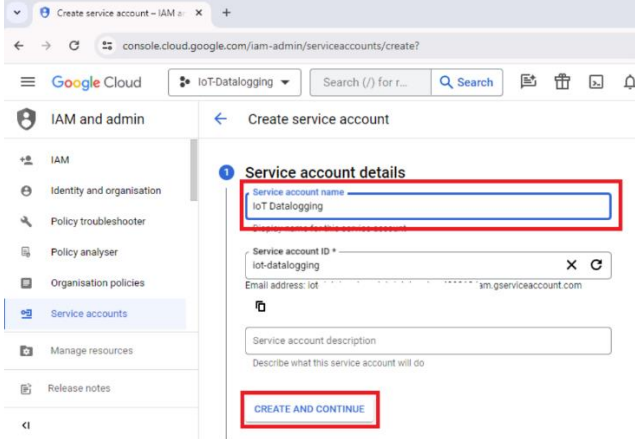
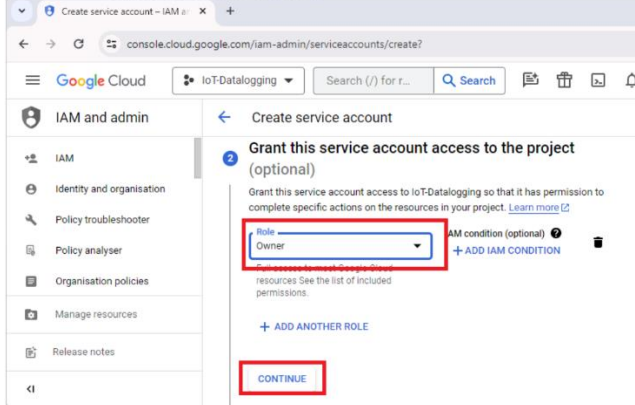
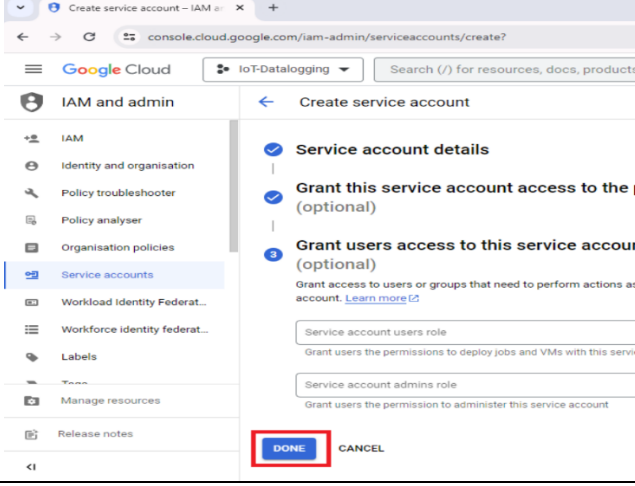
3.5.3 Perancangan Pengiriman Data ke *Google Sheet*

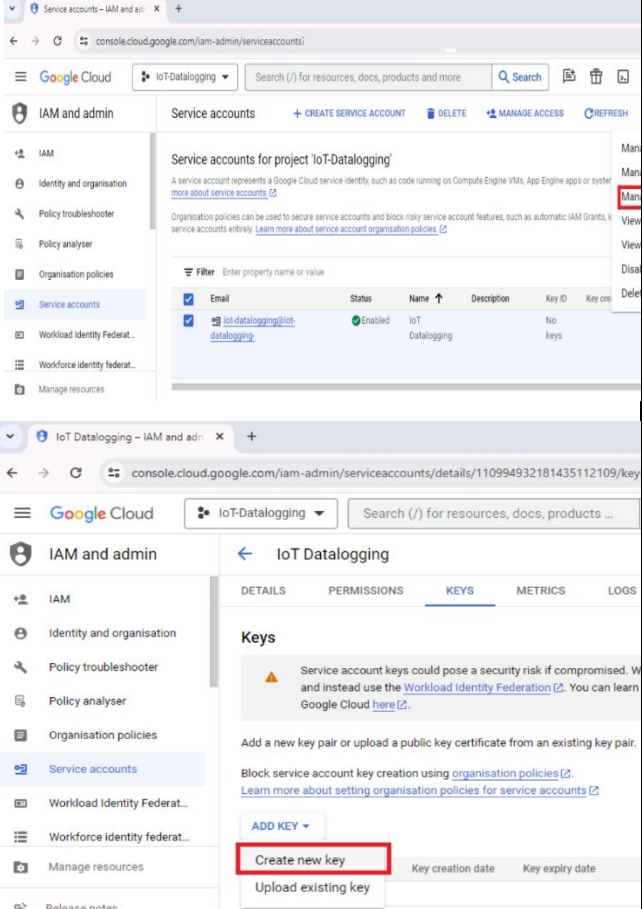
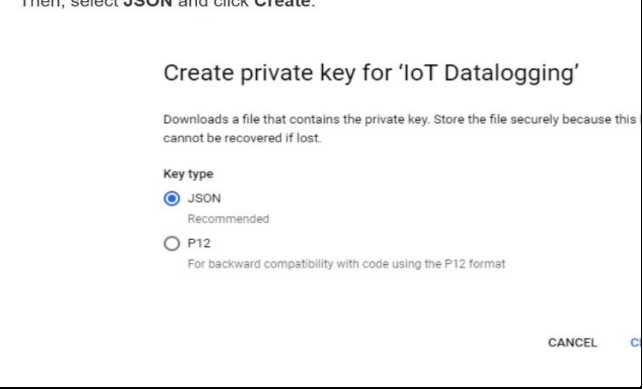
Berikut merupakan tahapan-tahapan dari pembuatan pengiriman data ke *google sheet*:

Tabel 3. 1 Tahapan Pengiriman Data ke *Google Sheet*

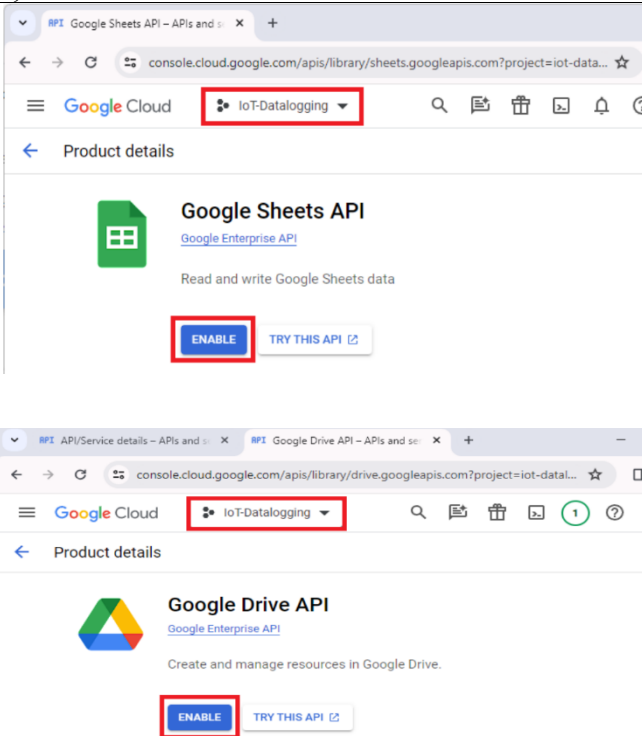
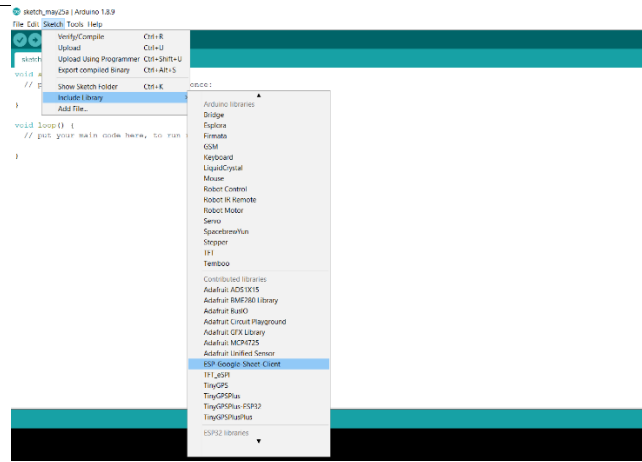
No	Tahapan	Gambar
1.	Pastikan memiliki akun google yang aktif	-
2.	Install Arduino IDE dan ditambahkan dukungan untuk ke ESP32	 Arduino IDE App

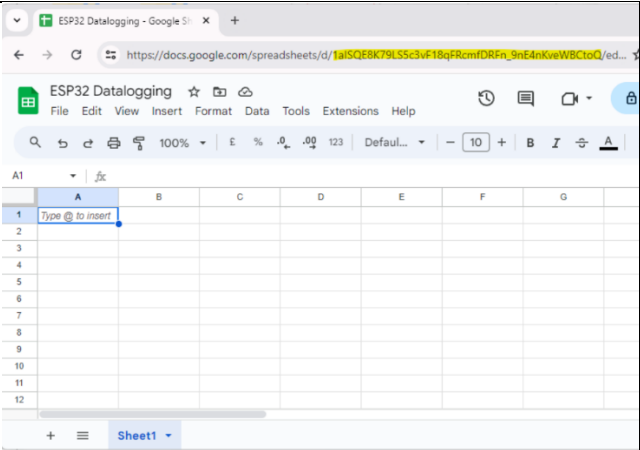
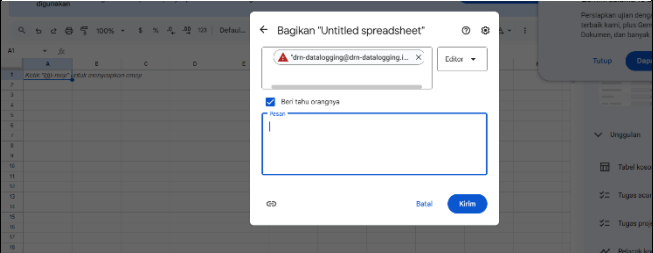
3.	- Akses Google Cloud Console - Buat Proyek Baru: Klik pada ikon proyek bagian atas dan pilih "New Project". Beri nama proyek, dan klik "Create" - Akun berhasil terbuat	
4.	Buat Layanan : - Navigasi ke "IAM & Admin" > "Service Accounts". - Klik "+ Create Service Account".	

5.	Isi nama akun layanan dan deskripsi (opsional), lalu klik <i>“Create and Continue”</i>	
6.	Pada bagian "Grant this service account access to project", pilih peran "Editor" untuk memberikan izin yang cukup. Klik "Continue".	
7.	Klik "Done" untuk menyelesaikan pembuatan akun layanan.	

8.	Buat Kunci JSON : - Pada daftar akun layanan, klik pada akun yang baru dibuat. - Pilih tab "Keys", lalu klik "Add Key" > "Create new key".	
9.	Pilih format "JSON" dan klik "Create". File kunci akan diunduh secara otomatis ke komputer. Simpan file ini dengan aman karena akan digunakan dalam kode Arduino.	
7.	Listing JSON yang sudah terdownload	<pre>{ "type": "service_account", "project_id": "drn-datalogging", "private_key_id": "00298957dd5a6717621ec9e3665eae257efad6b 2", "private_key": "-----BEGIN PRIVATE KEY-- --- \nMIEvgIBADANBgkqhkiG9w0BAQEFAAS CBKggwggSkAgEAAoIBAQCmHySkG6B5JR1 9\nhYqcAxWUxyWEs+5/R7it31IjvB3s/pIEDg MfwJlV8pU0AUKyMdvak9ltcY+F3747\nJjcv</pre>

		XS/X3tJXJfHeveSAE8r8AcJqbGqU/5PAvUjec av70ivJw7f0png6SrfzwmNL\n7PlvE2vVRRkui LzVi4YBdy6Jt5ItNoxBaoicZ+7qSlbi/DgWASg neEkStwlcS Y+0\nhMF1W80iFd67oLiRp/L+o4z Dy8ZOWQPdDBy4EFFFuEGUgocafqLIwSpq dhN2oxwP\nYq30BaCtPBCtNPXCnqBvlycLO rFJ98yHdVvsYjG6g060oj18sLJ/CGlQs+60sGI M\nACWM7n9RAgMBAAECggEABB9VDxw pu6KNJyFE6Ew3S+LemSVx71yom5VQJNeK kpKw\nAu5qFhlMdNKOm09wspquzwTl6itdm WqM/ktw0SRp3IczXGWc/p8NvZPq1J135UuI\ nIDxwvowJSTk0QGKIeFBQD1Urj0ltY9zYuU 30fqontzoovtE2hmzDnY5clvRO6f+c\noc+3+0 AfyhAn7T5f0Z9SQi0rjYoR0mi3Z0CVXbLbC RcOegm2eI35jPgrkYHTHjWn\nL4DMPomQ0 JNdEf6rduCPWlu6ar45jmsCGcz1UO3ID7xj6/ ZWVr6HtcfkHUBNpcCl\nQwdJmmsW+f1PIR TuQBRUPfEW9NB6u/4rcPPvBsLuMQKBgQD TMqMul8ODg0btCxaW\nnjum+h7LDL6Cez0ri 394Ic9NB1Z2aAP5mwYQb7QSeytizxjOoVkJQ M/8XNWGACfm9g\nVKywAk/xTuiqJX0Bypb /Qk8xWQb2WWHd0dO3oTSy1bFFFd0BoBuP StF0gPQZG4rA\n3OYjqI+w/k4qvLpkGoZlyja G/QKBgQDJXJaxOk1draVxEuDLpd/Cy/0UID jHjvFh\n2zZJnuhlUbPI8B9m+oST6pMk6Bjo3 YzeyUZZLWGL7skqJZgke5/AQnfRMh5JKCI w\nzvByASLxooax22sNEJ+ZDkAnvQAgD6T +5my/FnbBxvkaA20fCWtwVseaAuBFcNshG\nn Yn1GsWxr5QKBgQCzyL5q0lXoHp+I+FBxOB U40TFWwLt9lT7NZP8QuPCUcXDpAYGP\nn HZsFXr35IvpYBxhAa7cgTbend9y7IsBTA6HB byZ4Z9Ddzm3c0YW6o8rgWkX3/dy2\nxIDzJ7 LBrfFquxLY6F+tmY1Q/UHTo/OX/01S6+O Xt1Rujj9MCsx2ZmTTQKBgANl\nnJUFNu1o2 AFecwV+L5obxS44YUPzEYzuEJ7EsFb0Hm KFtswNydX3M+bgolarveV2U\nnRTRhuIBUgV 3NjngnAXWLaVvEnZAcNh9ejd1OOZmbZOw AyBhLTHWXMou6mxLvUJff\nnqLu8FDqGM VKbX+VeBNH00u8eq2RDpDLTnEsh2LMlAo GBAI9u0QBOJ8Ci04kwRpA5\nnm7RIhCDt2A 4BLiZHc1VEzVt/RA6sH/FF7Em61cQGrbTrZ MfZctOm4XcoVNxsqps\nnNV28P+/wsmmQtP 6lQbVDHauGJA2oU67uUzv+BAIRC3aaXo1Y LiLCOX4bINOBMd7d\nnQWkpav6dYh/z0d/fkx iGkQJJ\nn-----END PRIVATE KEY-----\n", "client_email": "drn-datalogging@drn- datalogging.iam.gserviceaccount.com", "client_id": "114480994319543046797",
--	--	---

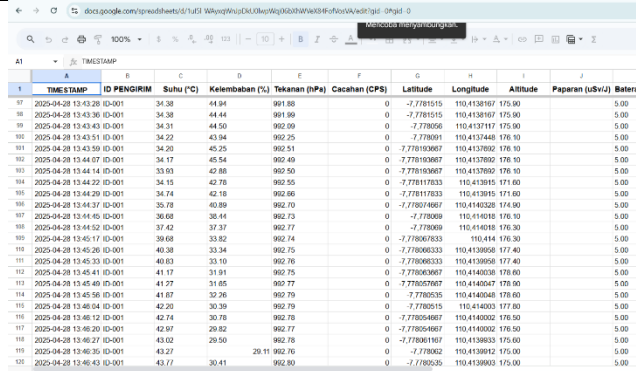
		<pre> "auth_uri": "https://accounts.google.com/o/oauth2/auth", "token_uri": "https://oauth2.googleapis.com/token", "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs", "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/drn-datalogging%40drn-datalogging.iam.gserviceaccount.com", "universe_domain": "googleapis.com" } </pre>
8.	Mengaktifkan API yang diperlukan : Klik link ini : https://console.cloud.google.com/apis/library/sheets.googleapis.com Buka Google Sheets API dan klik "Enable". Klik link ini : https://console.cloud.google.com/apis/library/drive.googleapis.com Buka Google Drive API dan klik "Enable".	 The screenshot shows two browser windows from the Google Cloud console. The top window displays the 'Google Sheets API' page with the 'ENABLE' button highlighted by a red box. The bottom window displays the 'Google Drive API' page, also with the 'ENABLE' button highlighted by a red box. Both pages show the API name, a brief description, and the 'ENABLE' button.
9.	Instalasi Library Arduino - Buka Arduino IDE. - Navigasi ke "Sketch" > "Include Library" > "Manage Libraries". - Pada kotak pencarian, ketik "ESP-Google-Sheet-Client". - Temukan library tersebut dan klik "Install".	 The screenshot shows the Arduino IDE interface. The 'Sketch' menu is open, and the 'Include Library' option is selected. This opens a 'Manage Libraries' window. In the search bar, the text 'ESP-Google-Sheet-Client' is entered. The search results list various libraries, and 'ESP-Google-Sheet-Client' is highlighted in blue.

10.	Membuat dan Mengatur Google Spreadsheet : - Buat Spreadsheet Baru: Buka <u>Google Sheets</u> dan buat spreadsheet baru. Beri nama sesuai keinginan. - Salin ID Spreadsheet: ID spreadsheet adalah bagian dari URL antara /d/ dan /edit.	
11.	Bagikan Spreadsheet dengan Akun Layanan: - Buka file JSON yang diunduh sebelumnya dan salin nilai dari client_email. - Kembali ke spreadsheet, klik tombol "Share" di kanan atas. - Tempelkan client_email pada kolom "Add people and groups" dan klik "Send".	Client email : "drn-datalogging@drn-datalogging.iam.gserviceaccount.com", 
12.	Menyusun Rangkaian ESP32 dengan Sensor dengan konfigurasi pin yang sesuai.	-
13.	Menulis dan Mengunggah Kode Arduino : - Buka Arduino IDE dan buat sketch baru. - Dimasukkan Kode: Digunakan contoh kode dari library	<pre>#include <Arduino.h> #include <WiFi.h> #include <Adafruit_Sensor.h> #include <Adafruit_BME280.h> #include "time.h" #include <ESP_Google_Sheet_Client.h> // For SD/SD_MMC mounting helper #include <GS_SDHelper.h></pre>

	"ESP-Google-Sheet-Client" yang telah diinstal. Sesuaikan bagian-bagian berikut: - WiFi Credentials: Masukkan SSID dan password jaringan WiFi Anda. - Service Account Credentials: Masukkan informasi dari file JSON, seperti client_email, private_key, dan project_id. - Spreadsheet ID: Dimasukkan ID spreadsheet yang telah Anda salin sebelumnya. - Diunggah kode : Disambungkan ESP32 ke komputer dan diunggah kode ke papan.	<pre> #define WIFI_SSID "REPLACE_WITH_YOUR_SSID" #define WIFI_PASSWORD "REPLACE_WITH_YOUR_PASSWORD" // Google Project ID #define PROJECT_ID "REPLACE_WITH_YOUR_PROJECT_ID" // Service Account's client email #define CLIENT_EMAIL "REPLACE_WITH_YOUR_CLIENT_EMAIL" // Service Account's private key const char PRIVATE_KEY[] PROGMEM = "-- ---BEGIN PRIVATE KEY-----\ REPLACE_WITH_YOUR_PRIVATE_KEY\n ----END PRIVATE KEY-----\n"; // The ID of the spreadsheet where you'll publish the data const char spreadsheetId[] = "YOUR_SPREADSHEET_ID"; // Timer variables unsigned long lastTime = 0; unsigned long timerDelay = 30000; // Token Callback function void tokenStatusCallback(TokenInfo info); // BME280 I2C Adafruit_BME280 bme; // Variables to hold sensor readings float temp; float hum; float pres; // NTP server to request epoch time const char* ntpServer = "pool.ntp.org"; // Variable to save current epoch time unsigned long epochTime; // Function that gets current epoch time unsigned long getTime() { time_t now; struct tm timeinfo; </pre>
--	--	--

		<pre> if (!getLocalTime(&timeinfo)) { //Serial.println("Failed to obtain time"); return(0); } time(&now); return now; } void setup(){ Serial.begin(115200); Serial.println(); Serial.println(); //Configure time configTime(0, 0, ntpServer); // Initialize BME280 sensor if (!bme.begin(0x76)) { Serial.println("Could not find a valid BME280 sensor, check wiring!"); while (1); } GSheet.printf("ESP Google Sheet Client v%s\n\n", ESP_GOOGLE_SHEET_CLIENT_VERSION) ; // Connect to Wi-Fi WiFi.setAutoReconnect(true); WiFi.begin(WIFI_SSID, WIFI_PASSWORD); Serial.print("Connecting to Wi-Fi"); while (WiFi.status() != WL_CONNECTED) { Serial.print("."); delay(1000); } Serial.println(); Serial.print("Connected with IP: "); Serial.println(WiFi.localIP()); Serial.println(); // Set the callback for Google API access token generation status (for debug only) </pre>
--	--	--

		<pre> GSheet.setTokenCallback(tokenStatusCallback) ; // Set the seconds to refresh the auth token before expire (60 to 3540, default is 300 seconds) GSheet.setPrerefreshSeconds(10 * 60); // Begin the access token generation for Google API authentication GSheet.begin(CLIENT_EMAIL, PROJECT_ID, PRIVATE_KEY); } void loop(){ // Call ready() repeatedly in loop for authentication checking and processing bool ready = GSheet.ready(); if (ready && millis() - lastTime > timerDelay){ lastTime = millis(); FirebaseJson response; Serial.println("\nAppend spreadsheet values..."); Serial.println("-----"); FirebaseJson valueRange; // New BME280 sensor readings temp = bme.readTemperature(); //temp = 1.8*bme.readTemperature() + 32; hum = bme.readHumidity(); pres = bme.readPressure()/100.0F; // Get timestamp epochTime = getTime(); valueRange.add("majorDimension", "COLUMNS"); valueRange.set("values/[0]/[0]", epochTime); valueRange.set("values/[1]/[0]", temp); valueRange.set("values/[2]/[0]", hum); valueRange.set("values/[3]/[0]", pres); </pre>
--	--	---

		<pre>// For Google Sheet API ref doc, go to https://developers.google.com/sheets/api/referen ce/rest/v4/spreadsheets.values/append // Append values to the spreadsheet bool success = GSheet.values.append(&response /* returned response */, spreadsheetId /* spreadsheet Id to append */, "Sheet1!A1" /* range to append */, &valueRange /* data range to append */); if (success){ response.toString(Serial, true); valueRange.clear(); } else{ Serial.println(GSheet.errorReason()); } Serial.println(); Serial.println(ESP.getFreeHeap()); } } void tokenStatusCallback(TokenInfo info){ if (info.status == token_status_error){ GSheet.printf("Token info: type = %s, status = %s\n", GSheet.getTokenType(info).c_str(), GSheet.getTokenStatus(info).c_str()); GSheet.printf("Token error: %s\n", GSheet.getTokenError(info).c_str()); } else{ GSheet.printf("Token info: type = %s, status = %s\n", GSheet.getTokenType(info).c_str(), GSheet.getTokenStatus(info).c_str()); } }</pre>
14.	Verifikasi Data di <i>Google Sheet</i>	

Setelah ESP32 berhasil terhubung ke WiFi dan mengirim data, buka *spreadsheet* untuk melihat data yang masuk. Data seperti suhu, kelembaban, tekanan, dan timestamp akan muncul sesuai dengan pengaturan dalam kode Arduino.

3.6 Tahap Pengambilan Data

Tahapan pengambilan data untuk perangkat monitoring radiasi dan lingkungan meliputi beberapa tahap yaitu :

3.6.1 Pengambilan Data Kalibrasi Pemantauan Radiasi

- a. Penempatan Sumber Radiasi: Sumber radiasi gamma ditempatkan pada posisi yang aman, detektor didekatkan dekat dengan sumber radiasi.
- b. Pencatatan Nilai Paparan Radiasi: Nilai paparan radiasi (dalam $\mu\text{Sv/J}$) dan cacahan per menit (CPM) dicatat menggunakan surveymeter standar.
- c. Pencatatan Nilai Paparan Radiasi: Nilai paparan radiasi dan cacahan per menit yang diukur oleh perangkat yang dibuat dicatat.
- d. Pencatatan Hasil Pengukuran: Semua hasil pengukuran dicatat pada tabel yang sesuai, termasuk selisih antara alat standar dan perangkat (penyimpangan absolut dan persentase).

3.6.2 Pengambilan Data Kalibrasi Lingkungan

- a. Persiapan Thermometer Jam Digital: Thermometer Jam Digital standar disiapkan untuk mengukur suhu dan kelembaban.
- b. Pencatatan Suhu dan Kelembaban: Suhu (dalam $^{\circ}\text{C}$) dan kelembaban (dalam %) dicatat menggunakan alat standar
- c. Pencatatan Suhu dan Kelembaban dengan Perangkat: Suhu dan kelembaban yang diukur oleh perangkat berbasis ESP32 dicatat.

- d. Perbandingan Hasil Pengukuran: Hasil dari alat standar dan perangkat yang dibuat dibandingkan, kemudian penyimpangan dihitung.
- e. Pengulangan Pengambilan Data: Pengambilan data diulang beberapa kali di lokasi yang berbeda (misalnya: dalam ruangan, luar ruangan) untuk menguji keakuratan dalam berbagai kondisi.

3.6.3 Pengambilan Data Kalibrasi Tegangan Baterai

- a. Penghubungan Perangkat ke Baterai: Perangkat dihubungkan ke baterai sebagai sumber daya.
- b. Pengukuran Tegangan Baterai dengan Voltmeter: Voltmeter standar digunakan untuk membaca tegangan baterai.
- c. Pencatatan Tegangan Baterai dengan Voltmeter: Tegangan baterai (dalam Volt) dicatat menggunakan voltmeter standar.
- d. Pencatatan Tegangan Baterai dengan Perangkat: Tegangan baterai yang diukur oleh perangkat berbasis ESP32 dicatat.
- e. Perbandingan Hasil Pengukuran Tegangan: Hasil dari voltmeter standar dan perangkat yang dibuat dibandingkan, kemudian penyimpangan dihitung.
- f. Pengulangan Pengambilan Data Tegangan: Pengambilan data diambil pada keadaan baterai masih penuh, setengah penuh, dan hampir habis untuk memastikan konsistensi.

3.6.4 Pengambilan Data Kalibrasi Posisi

- a. Penentuan Titik Lokasi: 5 titik lokasi berbeda untuk pengambilan data ditentukan.
- b. Verifikasi Data Referensi Lokasi: Setiap titik lokasi sudah memiliki data referensi dari Google Maps (latitude, longitude, dan altitude).

- c. Penggunaan Aplikasi Data Track Logger Handphone: Aplikasi data track logger digunakan untuk mendapatkan data posisi di lokasi tersebut.
- d. Penempatan Perangkat di Lokasi yang Sama: Perangkat ditempatkan di lokasi yang sama
- e. Pencatatan Data Posisi dari Perangkat: Data posisi (latitude, longitude, altitude) yang dihasilkan oleh perangkat dicatat.
- f. Pencatatan Data Posisi dari Track Logger dan Perangkat: Hasil pengukuran dari track logger dan perangkat yang dibuat dicatat ke dalam tabel.
- g. Perhitungan Penyimpangan Data Posisi: Penyimpangan untuk setiap parameter (*latitude, longitude, altitude*) dihitung.

3.7 Tahap Pengolahan Data

Tahapan pengambilan data untuk perangkat monitoring radiasi dan lingkungan meliputi beberapa tahap yaitu :

3.7.1 Pengolahan Data Kalibrasi Pemantauan Radiasi

Penyimpangan Presentase =

$$\left| \frac{\text{Nilai Alat Standar} - \text{Nilai Perangkat}}{\text{Nilai Alat Standar}} \right| \times 100\% \quad (3.1)$$

Akurasi Nilai =

$$Acc (\%) = 100 - error (\%) \quad (3.2)$$

$$CPM = N \times \left(\frac{60000}{T} \right) \quad (3.3)$$

$$Paparan = CPM / 151 \quad (3.4)$$

Keterangan :

- Nilai Alat Standar: Hasil pengukuran dari alat referensi.
- Nilai Perangkat: Hasil pengukuran dari perangkat yang dibuat.
- Penyimpangan dinyatakan dalam persen (%).

- N : jumlah cacahan yang terdeteksi dalam selang waktu tertentu (dalam satuan milidetik)
- T = waktu pengukuran dalam milidetik (disebut juga sebagai LOG_PERIOD)
- Acc (%) : Hasil akurasi dinyatakan dalam persen (%)

Tabel 3. 2 Pengambilan Data Kalibrasi Pemantauan Radiasi

Variasi	Surveymeter Standar		Perangkat yang dibuat		Penyimpangan (%)	Akurasi (%)
	Paparan (uSv/J)	Cacahan (CPM)	Paparan (uSv/J)	Cacahan (CPM)		
Dekat Sumber Radioaktif						

3.7.2 Pengolahan Data Kalibrasi Lingkungan

Penyimpangan Presentase =

$$\left| \frac{\text{Nilai Alat Standar} - \text{Nilai Perangkat}}{\text{Nilai Alat Standar}} \right| \times 100\% \quad (3.5)$$

Akurasi Nilai =

$$\text{Acc (\%)} = 100 - \text{error (\%)} \quad (3.6)$$

Keterangan :

- Nilai Alat Standar: Hasil pengukuran dari alat referensi.
- Nilai Perangkat: Hasil pengukuran dari perangkat yang dibuat.
- Penyimpangan dinyatakan dalam persen (%).
- Acc (%) : Hasil akurasi dinyatakan dalam persen (%)

Tabel 3. 3 Pengambilan Data Kalibrasi Lingkungan

Titik	Jam Digital	Perangkat yang dibuat	Penyimpangan	Akurasi
-------	-------------	-----------------------	--------------	---------

	Suhu *C	Kelembaban %	Suhu *C	Kelembaban %	Suhu *C	Kelembaban %	Suhu *C	Kelembaban %
(Luar Ruang)								
(Di dalam Ruang BerAC)								
(Di dalam Ruang Suhu)								

Untuk mengkalibrasi data tekanan, diambil melalui konversi data ketinggian dari GPS menjadi nilai tekanan udara menggunakan persamaan barometrik (Fathur, 2016), yang sebagaimana tercantum pada persamaan :

$$P = P_0 \left(1 - \frac{Lh}{T_0} \right)^{\frac{gM}{RL}} \quad (3.7)$$

Keterangan :

- P = Tekanan udara pada ketinggian
- P_0 = Tekanan pada permukaan laut
- L = Laju penurunan suhu
- h = Ketinggian dari permukaan laut
- T_0 = Suhu absolut di permukaan laut
- g = Percepatan Gravitasi
- M = Massa molar udara
- R = Konstanta gas universal

Tabel 3. 4 Pengambilan Data Tekanan

<i>Altitude</i> GPS NEO-6M	Tekanan BME-280	Hasil Konversi <i>Altitude</i> ke Tekanan	Penyimpangan (%)	Akurasi (%)

--	--	--	--	--

3.7.3 Pengolahan Data Tegangan Baterai

Penyimpangan Presentase =

$$\left| \frac{\text{Nilai Alat Standar} - \text{Nilai Perangkat}}{\text{Nilai Alat Standar}} \right| \times 100\% \quad (3.8)$$

Akurasi Nilai =

$$Acc (\%) = 100 - error (\%) \quad (3.9)$$

Keterangan :

- Nilai Alat Standar: Hasil pengukuran dari alat referensi.
- Nilai Perangkat: Hasil pengukuran dari perangkat yang dibuat.
- Penyimpangan dinyatakan dalam persen (%)
- Acc (%) : Hasil akurasi dinyatakan dalam persen (%)

Untuk mendapatkan data tegangan baterai yang dapat dibaca oleh mikrokontroler, digunakan rangkaian pembagi tegangan. Rangkaian ini bertujuan untuk menurunkan tegangan input dari baterai agar sesuai dengan batas maksimum tegangan ADC pada mikrokontroler. Sebagaimana dengan rumus yang digunakan:

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2} \quad (3.10)$$

Setelah ADC membaca V_{out} , diperlukan menghitung kembali tegangan sebenarnya V_{in} dari baterai menggunakan rumus pembalik:

$$V_{in} = V_{out} \frac{R_1 + R_2}{R_2} \quad (3.11)$$

Tabel 3.3 Pengambilan Data Kalibrasi Tegangan Baterai

Tabel 3. 5 Pengambilan Data Kalibrasi Tegangan Baterai

No	Voltmeter	Perangkat yang dibuat	
----	-----------	-----------------------	--

	Tegangan Volt	Tegangan Volt	Penyimpangan
Penuh			
Setengah Penuh			
Hampir Habis			

3.7.4 Pengolahan Data Kalibrasi Posisi

Penyimpangan Presentase =

$$\left| \frac{\text{Nilai Alat Standar} - \text{Nilai Perangkat}}{\text{Nilai Alat Standar}} \right| \times 100\% \quad (3.12)$$

$$Acc (\%) = 100 - error (\%) \quad (3.13)$$

Keterangan :

- Nilai Alat Standar: Hasil pengukuran dari alat referensi.
- Nilai Perangkat: Hasil pengukuran dari perangkat yang dibuat.
- Penyimpangan dinyatakan dalam persen (%).
- Acc (%) : Hasil akurasi dinyatakan dalam persen (%)

Tabel 3. 6 Pengambilan Data Kalibrasi Posisi

No	Google Maps			Perangkat yang dibuat			Penyimpangan		
	Alt	Lat	Long	Alt	Lat	Long	Alt	Antar Perangkat	Akurasi
1									
2									
3									
4									
5									

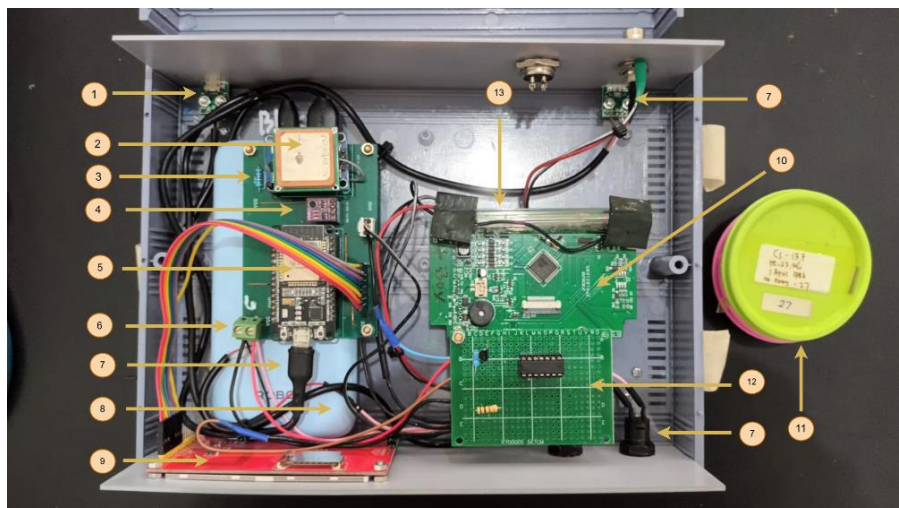
BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

4.1.1 Hasil Integrasi Modul dengan ESP32 WROOM DEVKIT V1

Pada tahap implementasi perangkat yang tertera pada gambar 4.1, seluruh komponen utama sistem dihubungkan secara langsung ke mikrokontroler ESP32 WROOM DEVKIT V1 sesuai dengan rancangan *schematic* yang tertera pada gambar 3.3. Rancangan *schematic* tersebut kemudian dirakit secara terintegrasi di dalam sebuah kotak pelindung. Perangkat ini dirancang agar seluruh komponen dapat bekerja secara sinergis dalam satu kesatuan sistem untuk mendeteksi, mengolah, serta menyajikan data radiasi dan parameter lingkungan. Modul utama berupa mikrokontroler ESP32 terpasang pada papan utama, yang juga terhubung dengan sensor BME-280 untuk membaca suhu, kelembapan dan tekanan, kemudian terdapat modul GPS NEO-6M yang berfungsi untuk memperoleh data Lokasi seperti; *latitude*, *longitude*, dan *altitude*.



Gambar 4. 1 Hasil Perancangan Sistem Perangkat Keras (*Hardware*)

Berikut merupakan keterangan dari komponen-komponen yang terdapat pada gambar 4.1

Tabel 4. 1 Keterangan komponen pada gambar 4.1

No	Keterangan
1	USB Port Powerbank
2	GPS NEO-6M
3	Resistor Pembagi Tegangan Battery Voltage
4	BME-280
5	ESP32 WROOM DEVKIT V1
6	Terminal Power On/Off
7	Mikro USB Port ESP32
8	Powerbank
9	TFT ILI9488
10	PCB Kit Detektor <i>Geiger-Müller</i>
11	Sumber Radioaktif Cesium-137
12	<i>Geiger Signal Conditioner</i>
13	Detektor <i>Geiger-Müller</i> 4011

Tahap selanjutnya, pada bagian sisi kanan sistem tampak terdapat kit detektor *Geiger-Müller* berwarna hijau yang berfungsi sebagai detektor radiasi. Modul ini dilengkapi dengan rangkaian pemroses sinyal internal yang mencakup sirkuit pengondisi sinyal (*signal conditioning*) dan *pulse amplifier* untuk mengolah sinyal deteksi dari tabung *Geiger-Müller* 4011. Hasil pengolahan ini kemudian dikirim ke mikrokontroler utama (ESP32) dalam bentuk sinyal digital, yang selanjutnya dihitung sebagai data cacahan per menit (CPM) serta dikonversi menjadi nilai paparan radiasi dalam satuan mikroSievert per jam ($\mu\text{Sv/h}$). output data dari detektor *Geiger-Müller* dikirimkan ke ESP32 melalui komunikasi serial UART untuk diproses lebih lanjut. Data gabungan dari seluruh sensor ini kemudian ditampilkan pada layar TFT ILI9488 yang terpasang dibagian depan perangkat, sehingga pengguna dapat memantau seluruh informasi secara langsung, kemudian disediakan tombol On/Off yang berfungsi sebagai saklar

yang dipasang di sisi luar kotak sistem. Sistem ini dilengkapi dengan sumber daya utama berupa baterai isi ulang, yang memungkinkan perangkat beroperasi secara portable tanpa ketergantungan pada sumber listrik eksternal.

Seluruh komponen yang telah diintegrasikan sesuai dengan fungsi dan jalur input masing-masing, sistem monitoring radiasi dan lingkungan ini dapat bekerja secara mandiri, menampilkan data secara langsung, serta siap untuk dikembangkan dalam platform IoT untuk pemantauan jarak jauh.

4.1.2 Hasil Perancangan Program Arduino IDE 1.8.9

Perancangan program untuk perangkat monitor ini dilakukan menggunakan Arduino IDE versi 1.8.9 sebagai lingkungan pengembangan perangkat lunak. Pemrograman difokuskan pada mikrokontroler ESP32 yang berfungsi sebagai pusat kendali sistem. Program ditulis dalam Bahasa C++ dan mengikuti struktur dasar pemrograman Arduino yang terdiri dari dua bagian utama, yaitu fungsi *setup()* dan *loop()*. Selain itu, *library* (kumpulan modul) yang menyederhanakan proses pemrograman disertakan dalam Arduino IDE. Tahapan dalam menggunakan program Arduino IDE adalah instalasi, kemudian konfigurasi perangkat keras dan perangkat lunak, lalu memilih model papan mikrokontroler yang sesuai dengan perangkat yang digunakan. Selanjutnya, menentukan port komunikasi yang akan digunakan oleh papan Arduino, lalu menulis kode program menggunakan bahasa pemrograman C/C++ setelah kode selesai disusun dan diverifikasi, program kemudian diunggah (*upload*) ke papan Arduino menggunakan kabel USB yang terhubung antara PC dan board, lalu output akan terlihat melalui serial monitor.

Langkah pertama dalam penulisan kode program adalah menyertakan sejumlah library yang diperlukan. Listing program berikut menunjukkan bagian awal kode yang memuat library yang digunakan:

```
-  
#include <Wire.h>  
#include <Adafruit_Sensor.h>  
#include <Adafruit_BME280.h>  
#include <TinyGPS++.h>  
#include <WiFi.h>  
#include <ESP_Google_Sheet_Client.h>  
#include <TFT_eSPI.h> // Library for TFT display
```

Gambar 4. 2 Insialisasi Library Arduino IDE 1.8.9

Setiap library memiliki fungsi tertentu yang mendukung komponen sistem, Library Wire.h yaitu library bawaan Arduino yang digunakan untuk komunikasi antar perangkat menggunakan protokol I2C. Selanjutnya, Adafruit_Sensor.h merupakan library standar dari Adafruit yang menyediakan antarmuka umum untuk berbagai jenis sensor. Kemudian Adafruit_BME280.h secara khusus digunakan untuk berkomunikasi dengan sensor lingkungan BME280. Library TinyGPS++.h berfungsi untuk mengolah data yang diterima dari modul GPS Data GPS mentah yang dikirim dalam format NMEA melalui komunikasi serial akan diuraikan oleh library ini menjadi informasi yang lebih berguna seperti Lokasi (Longitude, Latitude, Altitude), proses pengambilan data GPS menjadi lebih efisien dan akurat. Library Wifi.h merupakan library resmi dari ESP32 untuk mengatur koneksi jaringan nirkabel (Wi-Fi). Library ini memungkinkan mikrokontroler ESP32 terhubung ke jaringan internet. Dalam sistem ini, koneksi Wi-Fi sangat penting untuk mengirimkan data ke layanan penyimpanan daring seperti *google sheets*. Untuk kebutuhan pengiriman data ke spreadsheet online, digunakan library ESP_Google_Sheet_Client.h. Library ini menyediakan fungsi

untuk pengiriman data dari ESP32 langsung ke *Google Sheets* melalui koneksi internet. Dan terakhir, library `TFT_eSPI` digunakan untuk menampilkan data pada layar TFT (*Thin-Film Transistor*)

Tahap selanjutnya, adalah pendefinisian pin dan inisialisasi variabel yang akan digunakan dalam sistem. Listing program berikut menunjukkan beberapa set pin dan variabel yang akan digunakan:

```
08052025_GABUNGANSEMUA
// Define the pins
#define RX_PIN 15
#define TX_PIN 2
#define IN_PIN 4
#define LOG_PERIOD 5000 // 5 detik
const int batteryPin = 34; // Pin ADC untuk membaca tegangan
const float resistor1 = 51000; // Resistor 1 (51kΩ)
const float resistor2 = 100000; // Resistor 2 (100kΩ)
const int ADCmax = 4095; // Nilai maksimal ADC (12-bit pada ESP32)
const float Vref = 3.3; // Tegangan referensi (3.3V pada ESP32)
const float VmaxBattery = 5.0; // Tegangan maksimum baterai
const float VminBattery = 0; // Tegangan minimum baterai
float Vout; // Tegangan output dari pembagi tegangan
float Vin; // Tegangan input sebenarnya dari baterai
float batteryStatus; // Status baterai dalam persen
float batteryVoltage; // Status baterai dalam voltage hasil bacaan pembagi tegangan

volatile unsigned long counts = 0;
unsigned long previousCounts = 0;
unsigned long previousMillis = 0;
unsigned long cpm = 0;

// Initialize BME280, GPS, and TFT
Adafruit_BME280 bme; // BME280 sensor instance
TinyGPSPlus gps; // GPS instance
TFT_eSPI tft = TFT_eSPI(); // TFT instance
```

Gambar 4. 3 Inisialisasi PIN dan Variabel yang digunakan

Bagian awal dari program ini dimulai dengan perintah `#define` yang digunakan untuk menetapkan nilai konstan terhadap nama pin. Pin `RX_PIN` didefinisikan pada GPIO 15, `TX_PIN` pada GPIO 2, dan `IN_PIN` pada GPIO 4 untuk inputan sinyal detektor. Selain itu, terdapat konstanta `LOG_PERIOD` yang diatur sebesar 5000 milidetik atau 5 detik. Nilai ini menjadi interval waktu utama bagi loop program dalam melakukan logging atau pengambilan data secara berkala.

Selanjutnya, terdapat deklarasi variabel dan konstanta yang berkaitan dengan pengukuran tegangan baterai. Pin ADC yang digunakan untuk membaca tegangan baterai. Pin ADC yang digunakan untuk membaca tegangan adalah `batteryPin` dan

di definisikan pada GPIO 34. Dua resistor dengan nilai 51k Ω (resistor1) dan 100k Ω (resistor2) digunakan sebagai pembagi tegangan. Nilai maksimum ADC (ADCmax) ditetapkan sebesar 4095 karena esp32 memiliki ADC 12-bit. Tegangan referensi ADC (Vref) ditetapkan sebesar 3.3 V. Nilai VmaxBattery dan VminBattery masing-masing ditetapkan sebagai batas atas dan bawah tegangan baterai yaitu 5.0V dan 0V. Variabel-variabel seperti Vout, Vin, batteryStatus, dan batteryVoltage digunakan untuk menghitung dan menyimpan tegangan aktual baterai serta statusnya.

Variabel Counts, previousCounts, previousMillis, dan cpm dideklarasikan dengan tipe unsigned long untuk menyimpan nilai terkait perhitungan CPM (Counts Per Minute), yang kemungkinan digunakan dalam konteks sensor radiasi. Terakhir, program ini menginisialisasi objek : bme sebagai instance dari sensor BME280 untuk mengukur suhu, tekanan dan kelembapan; gps sebagai instance dari library TinyGPSPlus untuk menampilkan informasi pada layar TFT.

Bagian berikutnya, menangani pengaturan koneksi Wi-Fi dan mengirim data ke layanan *Google*, khususnya *Google Sheets* melalui *Google API*.

```
//-----WIFI TERTAUT-----
#define WIFI_SSID "Tenda_FA69F0"
#define WIFI_PASSWORD "bismillah"

//-----Google Project ID YANG DIKIRIM-----
// Google Project ID
#define PROJECT_ID "drn-datalogging"

// Service Account's client email
#define CLIENT_EMAIL "drn-datalogging@drn-datalogging.iam.gserviceaccount.com"

// Service Account's private key
const char PRIVATE_KEY[] PROGMEM = "-----BEGIN PRIVATE KEY-----\nmIIEvgIBADANBgkqhkiG9w0BAQEFAASCCKgwgwSkAgEAAoIBAQCmHysK66B5JR19\nehYqCAsWUxyWEs+5/R7it3IIjvB5C

// The ID of the spreadsheet where you'll publish the data
const char spreadsheetId[] = "1uISl-WAYxqWnJpDkU01wpWqj06bXhWVeX84FofVosVA";

//-----VARIABLE YANG DICARI-----
```

Gambar 4. 4 Pengiriman Data ke *Google Sheets*

Nilai SSID jaringan WiFi ditentukan dengan #define WIFI_SSID dan diberikan nama jaringan WiFi yang digunakan pada saat itu, sedangkan kata sandi WiFi diatur dengan #define WIFI_PASSWORD. Kemudian terdapat konfigurasi

untuk mengirim data ke layanan *Google*, khususnya *Google Sheets* melalui *Google API*, PROJECT_ID, CLIENT_EMAIL, dan PRIVATE_KEY terdapat pada akun JSON yang dibuat pada tabel (3.1). Selanjutnya, terdapat spreadsheetId, yaitu ID dari Google Spreadsheet tempat data akan dikirim dan disimpan.

Program berikutnya, mengatur dan mendapatkan waktu dari server NTP serta menyesuaikannya dengan zona waktu local, dalam hal ini zona waktu Indonesia bagian barat (WIB, UTC+7)

```
08052025_GABUNGANSEMUA
//-----NTP UNTUK WAKTU-----
const char* ntpServer = "pool.ntp.org";

// Function that gets current epoch time
unsigned long getTime() {
    time_t now;
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {
        return 0;
    }
    time(&now);
    return now;
}

//-----WAKTU +7-----
String getFormattedTime() {
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {
        return "Failed to obtain time";
    }

    // Add 7 hours to the current time
    timeinfo.tm_hour += 7;

    // Handle overflow if hours go over 23
    if (timeinfo.tm_hour >= 24) {
        timeinfo.tm_hour -= 24;
        timeinfo.tm_mday += 1; // Move to the next day

        // Handle day overflow by adjusting the month and year if needed
    }
}
```

Gambar 4. 5 Pengaturan Waktu dengan Zona yang sesuai

Deklarasikan `const char* ntpServer = "pool.ntp.org";` yang digunakan untuk menentukan alamat server NTP yang akan dihubungi. Server ini akan memberikan informasi waktu terkini yang disinkronkan melalui jaringan internet. Selanjutnya terdapat fungsi `getTime()`, yang bertugas mengambil waktu saat ini dalam format epoch.

Fungsi ini memanggil `getLocalTime(&timeinfo)` untuk mendapatkan struktur waktu lokal, dan jika berhasil, akan mengembalikan nilai waktu saat ini dalam bentuk unsigned long. Jika gagal, fungsi akan mengembalikan nilai 0. Kemudian,

terdapat fungsi `getFormattedTime()` yang digunakan untuk mendapatkan waktu dalam format string yang lebih mudah dibaca, seperti YYYY-MM-DD HH:MM:SS. Fungsi ini juga memanggil `getLocalTime(&timeinfo)` untuk mendapatkan waktu lokal, lalu menambahkan 7 jam (`timeinfo.tm_hour += 7`) agar sesuai dengan zona waktu WIB (UTC+7).

Bagian selanjutnya, setting `setup()` yang dijalankan sekali saat ESP32 dinyalakan.

```

09052025_GABUNGANSEMUA
//-----
void setup()
{
    // Initialize serial communication for debugging
    Serial.begin(115200);

    //-----POSISI TFT-----
    tft.begin();
    tft.setRotation(3); // Landscape
    tft.fillRect(TFT_BLACK);

    //-----GARIS FRAME-----
    tft.drawRect(0, 0, 476, 319, TFT_BLUE);
    tft.drawRect(3, 3, 470, 313, TFT_RED);
    tft.drawLine(3, 65, 470, 65, TFT_RED); // Top center line
    tft.fillRect(8, 0, 458, 53, TFT_NAVY); // Blue background for the title
    tft.setCursor(140, 25, 4); // Title position
    tft.setTextColor(TFT_YELLOW, TFT_NAVY);
    tft.println("Monitoring Radiasi");

    //-----GARIS MERAH-----
    tft.drawLine(3, 250, 470, 250, TFT_RED);
    tft.fillRect(8, 258, 458, 53, TFT_BLACK);

    // Main frame for sensor data
    tft.fillRect(8, 70, 458, 180, TFT_BLACK);

    //-----Display TAMPILAN TFT-----

    tft.setCursor(30, 90, 2);
    tft.setTextColor(TFT_YELLOW, TFT_BLACK);
    tft.print("Suhu ");
    tft.setCursor(100, 90, 2);
    tft.print(" ");
    tft.setCursor(180, 90, 2);
    tft.println("°C");

    tft.setCursor(30, 110, 2);
    tft.setTextColor(TFT_YELLOW, TFT_BLACK);
    tft.print("Kelembapan ");
}

```

Gambar 4. 6 *Setting Setup* untuk Tampilan *Display*

Bagian `setup()` dalam program ini mencakup beberapa proses penting yang berkaitan dengan tampilan visual di layar TFT dan pengaturan awal sistem. Dimulai dari inisialisasi serial untuk memulai komunikasi dengan kecepatan 115200 bps, kemudian menginisialisasi layar tft dengan `tft.begin()`, untuk mengatur pengaturan posisi menggunakan `tft.setCursor`, dan untuk menampilkan label data sensor yaitu `tft.print`.

Tahap berikutnya, inisialisasi tiap masing-masing modul dan jaringan untuk memastikan sensor, komunikasi, dan jaringan siap digunakan

Pada tahap ini, masing-masing sensor diinisialisasi pada void setup() sebelum diproses dan dimasukkan ke dalam sistem loop().

```
08062025_GABUNGANSEMUA
//-----
// Initialize BME280
if (!bme.begin(0x76)) { // Change the I2C address if needed
  tft.setCursor(50, 50, 2);
  tft.setTextColor(TFT_RED, TFT_BLACK);
  tft.println("BME280 not found");
  while (1); // Stop if BME280 is not detected
}
//-----
// Initialize GPS
Serial2.begin(9600, SERIAL_8N1, RX_PIN, TX_PIN);
//-----
// Initialize frequency measurement pin
//pinMode(inputPin, INPUT_PULLUP); // Set pin D4 sebagai input dengan pull-up internal
//pinMode(IN_PIN, INPUT);

Serial.println("Booting...");
Serial.println("Measuring CPM setiap 5 detik...");

pinMode(IN_PIN, INPUT);
attachInterrupt(digitalPinToInterrupt(IN_PIN), ISR_impulse, FALLING);
//-----
// Initialize Wi-Fi
WiFi.setAutoReconnect(true);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
  delay(1000);
}

Serial.println();
Serial.println("Connected!");
Serial.print("IP address: ");
Serial.println(WiFi.localIP());
delay(1000);

Serial.println();
Serial.print("Connected to SSID: ");
```

Gambar 4. 7 Inisialisasi masing-masing sensor ke Setup

Tahapan selanjutnya, merupakan tahapan yang paling penting karna proses tersebut yang nanti nya akan melakukan pembacaan sensor dan pemrosesan data secara berkala yaitu fungsi loop(). Berikut merupakan beberapa fungsi loop() yang digunakan untuk masing-masing modul:

```
08062025_GABUNGANSEMUA
void loop()
{
  //----- Read BME280 data-----
  temperature = bme.readTemperature();
  humidity = bme.readHumidity();
  pressure = bme.readPressure() / 100.0F; // Convert to hPa

  //-----Read GPS data if available-----
  while (Serial2.available() > 0) {
    gps.encode(Serial2.read());
  }

  if (gps.location.isUpdated()) {
    latitude = gps.location.lat();
    longitude = gps.location.lng();
    altitude = gps.altitude.meters();
  }

  //-----FREKUENSI COUNTER-----
  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= LOG_PERIOD) {
    previousMillis = currentMillis;

    unsigned long countsThisPeriod = counts - previousCounts;
    previousCounts = counts;

    cpm = countsThisPeriod * (60000 / LOG_PERIOD); // konversi ke CPM
    frequency = cpm;
    Serial.print("Cacah (5 detik terakhir): ");
    Serial.println(countsThisPeriod);
    Serial.print("CPM: ");
```

Gambar 4. 8 Fungsi loop dan rumus dari masing-masing sensor

Fungsi loop() yang digunakan kali ini dimulai dari pembacaan data dari sensor BME-280 yang hasil tekanan nya akan dikonversi dari Pa ke hPa, kemudian setelah membaca data Sensor BME-280, selanjutnya membaca data

dari GPS NEO-6M data yang diterima di-*encode* diproses oleh TinyGPS++ (`gps.encode()`) yang dimana `gps.location.lat()` berfungsi sebagai pembaca nilai *latitude*, `gps.location.lng()` digunakan untuk membaca data *longitude*, dan `gps.altitude.meters()` digunakan untuk membaca data ketinggian atau *altitude*, kemudian setelah sensor NEO-6M hasil nilai nya telah terbaca, dilanjutkan perhitungan CPM untuk perhitungan dari sinyal radiasi. `unsigned long currentMillis = millis();` digunakan untuk mengambil waktu saat ini dalam milidetik, kemudian data baru akan diambil setiap 5 detik (`LOG_PERIOD`), untuk menghitung jumlah pulsa yang masuk dalam periode waktu dan mengonversi jumlah hitungan ke dalam *counts per minute* digunakan persamaan pada gambar (4.8)

Tahap selanjutnya yaitu menghitung *Battery Voltage*, sistem menghitung tegangan baterai dengan membaca nilai analog dari pin ADC, kemudian mengonversinya menjadi tegangan sebenarnya menggunakan rumus pembagi tegangan pada persamaan (3.8), konversi V_{in} ke nilai presentase (0-100%) berdasarkan nilai minimum dan maksimum tegangan baterai, misalnya $V_{minbattery} = 3.0\text{ V}$ menandai baterai sudah habis, $V_{maxbattery} = 4.9\text{ V}$ menandakan baterai penuh yang rumus persamaannya tertera pada gambar (4.9), kemudian untuk mengonversi paparan radiasi berdasarkan frekuensi

melalui persamaan pada gambar (4.9) yang dimana hasil dari cacahan tersebut dibagi dengan 151 CPM yang setara dengan 1 $\mu\text{Sv/h}$.

```
//-----MENGHITUNG BATTERY VOLTAGE-----
// Membaca nilai analog dari pin ADC
int analogValue = analogRead(batteryPin);
float Vout = analogValue * (Vref / ADCmax); // Menghitung Vout dari ADC

// Menghitung tegangan input sebenarnya (Vin) dari pembagi tegangan
float Vin = Vout * ((resistor1 + resistor2) / resistor2);
Vin += 0.02; //agar menjadi 5.00 V
batteryVoltage = round(Vin * 100.0) / 100.0;

// Membulatkan persen ke 1 desimal
batteryStatus = round(batteryStatus * 10.0) / 10.0;

// Menghitung persentase status baterai
batteryStatus = ((Vin - VminBattery) / (VmaxBattery - VminBattery)) * 100.0;

// Membatasi nilai status baterai antara 0% hingga 100%
if (batteryStatus > 100.0) {
    batteryStatus = 100.0;
} else if (batteryStatus < 0.0) {
    batteryStatus = 0.0;
}

//-----RADIASI SERIAL RANDOM-----
// Hitung radiasi berdasarkan frekuensi (CPS ke  $\mu\text{Sv/h}$ )
// Generate random radiation exposure
float radiationExposure = frequency / 151.0 * 1; // 151 cpm = 1 usv /h
```

Gambar 4. 9 Listing Persamaan *Battery Voltage* dan Konversi Paparan

Tahap berikutnya, yaitu menampilkan hasil pengukuran sensor ke layar TFT (*Thin-Film Transistor*). Pertama-tama, posisi kursor diatur menggunakan `tft.setCursor(x, y, z);`, di mana x dan y menentukan posisi koordinat pada layar, dan z adalah ukuran teks. Kemudian warna teks diatur menggunakan `tft.setTextColor`. Setiap data dicetak di koordinat tertentu pada layar TFT sesuai dengan `tft.setCursor()`. Hasil pengolahan dari masing-masing sensor ditampilkan dan dicetak ke layar TFT menggunakan `tft.println()` yang tertera pada gambar (4.10)

```
//-----TAMPILAN TFT HASIL-----
// Update temperature value
tft.setCursor(120, 90, 2);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.print(temperature); // Print new temperature value

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(120, 110, 2);
tft.println(humidity);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(120, 130, 2);
tft.println(pressure);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(145, 160, 2);
tft.println(latitude, 8);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(145, 180, 2);
tft.println(longitude, 8);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(145, 200, 2);
tft.println(altitude);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(320, 90, 2);
tft.println(radiationExposure);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(320, 110, 2);
tft.println(frequency);
```

Gambar 4. 10 Set Tampilan TFT

Tahap selanjutnya, mengunggah data ke *Google Sheets* setiap 5 detik. Pertama-tama, sistem memeriksa apakah waktu saat ini sudah melebihi batas waktu pengiriman terakhir (*lastTime*), dan jika iya, maka proses *upload* dimulai. Data yang dikirim mencakup waktu dalam format yang sudah disesuaikan (+7 jam), ID perangkat (*deviceId*), serta semua data sensor: suhu (*temperature*), kelembaban (*humidity*), tekanan (*pressure*), koordinat GPS (*latitude*, *longitude*), ketinggian (*altitude*), frekuensi (*frequency*), tegangan baterai (*batteryVoltage*), dan paparan radiasi (*radiationExposure*). Semua data disusun dalam *valueRange* dan dikirim menggunakan metode *.append()* ke lembar kerja *Google Sheets* dengan nama "Sheet1!A1".

```
08052025_GABUNGANSEMUA

// Upload data to Google Sheets every 5 seconds
if (millis() - lastTime > timerDelay) {
    lastTime = millis();

    FirebaseJson response;
    FirebaseJson valueRange;

    // Get current epoch time and adjust for timezone (+7 hours)
    unsigned long epochTime = time(multiplier);
    unsigned long adjustedTime = epochTime + 7 * 3600;

    // Format adjusted time as a string
    struct tm *ptm = gmtime((time_t *) &adjustedTime);
    char timeString[30];
    strftime(timeString, 30, "%Y-%m-%d %H:%M:%S", ptm);

    // Prepare the data to upload to Google Sheets
    valueRange.set("values/0/0", String(deviceId)); // Fixed line
    valueRange.set("values/0/1", String(getFormattedTime())); // Add formatted time
    valueRange.set("values/0/2", String(temperature));
    valueRange.set("values/0/3", String(humidity));
    valueRange.set("values/0/4", String(pressure));
    valueRange.set("values/0/5", double(latitude));
    valueRange.set("values/0/6", double(longitude));
    valueRange.set("values/0/7", double(altitude));
    valueRange.set("values/0/8", String(frequency));
    valueRange.set("values/0/9", String(batteryVoltage));
    valueRange.set("values/0/10", String(batteryStatus));
    valueRange.set("values/0/11", String(radiationExposure)); // Update with actual radiation exposure value

    // Append values to the spreadsheet
    bool success = GSheet.values.append(response, spreadsheetId, "Sheet1!A1", valueRange);
    if (success) {
        response.toString(Serial, true);
        valueRange.clear();
    } else {
        Serial.println(GSheet.errorReason());
    }
}

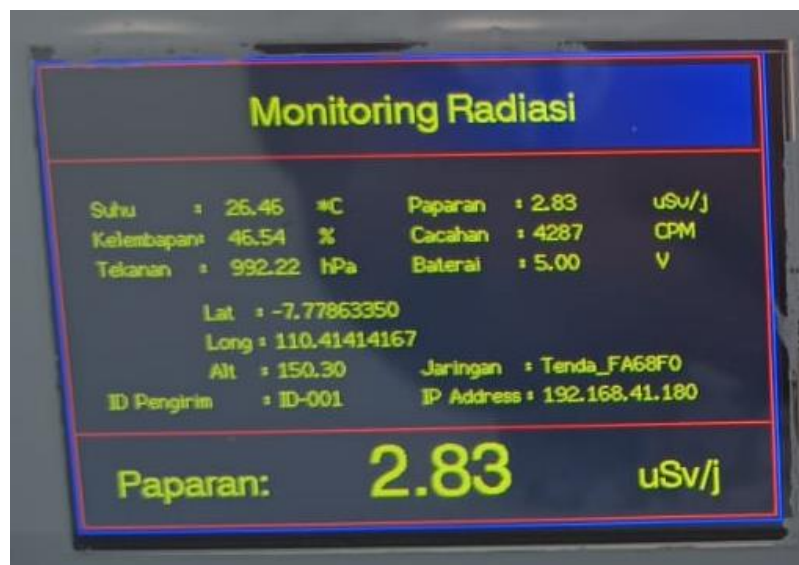
delay(5000); // Delay for 5 seconds before next update
```

Gambar 4. 11 Listing Data Logging ke *Spreadsheet*

Pada penelitian kali ini, hasil data akan dibaca melalui *spreadsheet* dan display dari TFT ILI9488, yang dibuat melalui *google cloud* dan data log akan tersimpan melalui *google drive*, yang dimana program tersebut di inisialisasi terlebih dahulu melalui Arduino IDE.

4.1.3 Hasil Tampilan Display TFT ILI9488 dan *Spreadsheet*

Pada tahap pengujian perangkat, dilakukan observasi terhadap hasil keluaran data yang ditampilkan melalui layar TFT ILI9488. Gambar 4.3 menunjukkan tampilan utama dari antarmuka alat pemantauan radiasi berbasis mikrokontroler yang telah dirancang dan direalisasikan. Layar ini menampilkan berbagai parameter lingkungan serta informasi terkait radiasi, jaringan, dan status item.



Gambar 4. 12 Hasil *Display* Tampilan TFT

Tampilan display terbagi menjadi beberapa bagian utama. Di bagian paling atas, terdapat judul “Monitoring Radiasi” yang menjadi indikator fungsi utama dari perangkat. Selanjutnya, dibagian kiri bawah dari judul “Monitoring Radiasi” ditampilkan data dari bacaan sensor BME-280 berupa suhu, kelembapan, dan tekanan udara. Bagian tengah dibawah dari data Sensor BME-280 ditampilkan bagian informasi geospasial berupa koordinat lintang dan bujur, serta ketinggian yang diambil melalui modul GPS NEO-6M. Di bagian kanan atas ditampilkan dari informasi data berupa paparan, cacahan, dan status baterai, di bagian kanan bawah terdapat informasi data dari jaringan dan IP Address yang dipakai dan di

bagian paling penting di paling bawah, menunjukkan informasi hasil dari pengukuran paparan radiasi.

Kemudian hasil dari spreadsheet ditampilkan, Seluruh data ini dicatat secara berkala dan ditandai dengan timestamp yang menunjukkan waktu pengambilan data.

	A	B	C	D	E	F	G	H	I	J	K	L
1	TIMESTAMP	ID PENGIRIM	Suhu (°C)	Kelembaban (%)	Tekanan (hPa)	Cacahan (CPS)	Latitude	Longitude	Altitude	Paparan (uSv/J)	Bateray Voltage (V)	% Bateray
57	2025-04-28 13:43:28 ID-001	34.38	44.94	991.88	0	-7.7781515	110.4138167	175.90	5.00	100.00		
58	2025-04-28 13:43:38 ID-001	34.38	44.44	991.99	0	-7.7781515	110.4138167	175.90	5.00	100.00		
59	2025-04-28 13:43:43 ID-001	34.31	44.50	992.09	0	-7.7780556	110.4137117	175.90	5.00	100.00		
190	2025-04-28 13:43:51 ID-001	34.22	43.94	992.25	0	-7.778091	110.4137448	176.10	5.00	100.00		
191	2025-04-28 13:43:59 ID-001	34.20	45.25	992.51	0	-7.778193667	110.4137692	176.10	5.00	100.00		
192	2025-04-28 13:44:07 ID-001	34.17	45.54	992.49	0	-7.778193667	110.4137692	176.10	5.00	100.00		
193	2025-04-28 13:44:14 ID-001	33.93	42.88	992.50	0	-7.778193667	110.4137692	176.10	5.00	100.00		
194	2025-04-28 13:44:22 ID-001	34.15	42.78	992.55	0	-7.778117833	110.413915	171.60	5.00	100.00		
195	2025-04-28 13:44:29 ID-001	34.74	42.18	992.66	0	-7.778117833	110.413915	171.60	5.00	100.00		
196	2025-04-28 13:44:37 ID-001	35.78	40.89	992.70	0	-7.778074667	110.4140328	174.90	5.00	100.00		
197	2025-04-28 13:44:45 ID-001	36.68	38.44	992.73	0	-7.778069	110.414018	176.10	5.00	100.00		
198	2025-04-28 13:44:52 ID-001	37.42	37.37	992.77	0	-7.778069	110.414018	176.30	5.00	100.00		
199	2025-04-28 13:45:17 ID-001	39.68	33.82	992.74	0	-7.778067833	110.414	176.30	5.00	100.00		
110	2025-04-28 13:45:26 ID-001	40.38	33.34	992.75	0	-7.778066333	110.4139958	177.40	5.00	100.00		
111	2025-04-28 13:45:33 ID-001	40.83	33.10	992.76	0	-7.778066333	110.4139958	177.40	5.00	100.00		
112	2025-04-28 13:45:41 ID-001	41.17	31.91	992.75	0	-7.778063667	110.4140038	178.60	5.00	100.00		
113	2025-04-28 13:45:49 ID-001	41.27	31.65	992.77	0	-7.778057667	110.4140047	178.90	5.00	100.00		
114	2025-04-28 13:45:56 ID-001	41.87	32.26	992.79	0	-7.7780535	110.4140048	178.60	5.00	100.00		
115	2025-04-28 13:46:04 ID-001	42.20	30.39	992.79	0	-7.7780515	110.414003	177.80	5.00	100.00		
116	2025-04-28 13:46:12 ID-001	42.74	30.78	992.78	0	-7.778054667	110.4140002	176.50	5.00	100.00		
117	2025-04-28 13:46:20 ID-001	42.97	29.82	992.77	0	-7.778054667	110.4140002	176.50	5.00	100.00		
118	2025-04-28 13:46:27 ID-001	43.02	29.50	992.78	0	-7.778061167	110.4139933	175.60	5.00	100.00		
119	2025-04-28 13:46:35 ID-001	43.27	29.11	992.76	0	-7.778062	110.4139912	175.00	5.00	100.00		
120	2025-04-28 13:46:43 ID-001	43.77	30.41	992.80	0	-7.7780535	110.4139903	175.00	5.00	100.00		

Gambar 4. 13 Hasil Display Tampilan *Spreadsheet*

Dengan adanya tampilan data yang lengkap dan informatif pada layar TFT dan *spreadsheet* ini, pengguna dapat melakukan interpretasi hasil secara langsung tanpa memerlukan perangkat tambahan. Penggunaan warna terang seperti kuning pada nilai “Paparan” juga membantu menarik perhatian terhadap parameter utama yang berfokus pada sistem monitoring ini.

4.1.4 Hasil Data Kalibrasi

4.1.4.1 Hasil Data Kalibrasi Paparan Radiasi

Langkah pertama dalam proses kalibrasi paparan radiasi adalah merekam data hitungan dari perangkat yang telah dibuat, untuk cacahan dinyatakan dalam satuan count per minute atau CPM, untuk paparan radiasi dinyatakan dalam satuan mikrosievert per jam secara berkelanjutan selama interval tertentu, biasanya antara 5 hingga 10 menit, untuk mendapatkan nilai rata-rata yang stabil. Kemudian nilai detektor yang ditangkap dicatat dengan

detektor diarahkan ke sumber radiasi, setelah data di dapat bandingkan dengan alat surveymeter sekunder yang telah terukur dengan variasi yang sama. Hitung menggunakan rumus pada persamaan (diisi persamaan berapa), kemudian nilai dari penyimpangan eror ditentukan dari pembagian nilai selisih pengukuran perangkat yang dibuat dengan hasil pengukuran dari surveymeter sekunder, dan dikalikan 100 menggunakan persamaan rumus (Paleri, 2015) yang dimana diuraikan pada persamaan (3.1), kemudian nilai akurasi didapatkan dari hasil nilai 100 dikurangkan dengan nilai penyimpangan eror.

Berikut adalah data kalibrasi perangkat yang dibuat dan surveymeter standar yang telah dimasukkan ke dalam tahap analisis dari hasil kalibrasi :

Tabel 4. 2 Data Kalibrasi Kit Detektor *Geiger Muller* dengan Surveymeter Standar

Variasi	Surveymeter Standar		Perangkat yang dibuat		Penyimpangan (%)		Akurasi (%)	
	Paparan (uSv/J)	Cacahan (CPM)	Paparan (uSv/J)	Cacahan (CPM)	Paparan (uSv/J)	Cacahan (CPM)	Paparan (uSv/J)	Cacahan (CPM)
Didekatkan dengan sumber radioaktif	2.02	305	1.67	252	17.32	17.37	82.67	82.62



Gambar 4. 14 Hasil paparan dan cacahan dari display TFT



Gambar 4. 15 paparan dan cacahan dari surveymeter standar

4.1.4.2 Hasil Data Kalibrasi Lingkungan

Proses kalibrasi dari data lingkungan kali ini yakni, dengan menempatkan perangkat yang telah dibuat dengan dibandingkan alat termometer digital, data diambil dengan variasi dalam keadaan di luar ruangan, di dalam ruangan ber AC, dan di dalam suhu ruang. Alat yang diuji ditempatkan pada keadaan yang sama dalam keadaan berdampingan. Penempatan ini dimaksudkan untuk memastikan bahwa kedua alat mengalami kondisi lingkungan yang identik sehingga pembacaan perangkat yang dibuat dapat dibandingkan secara objektif.

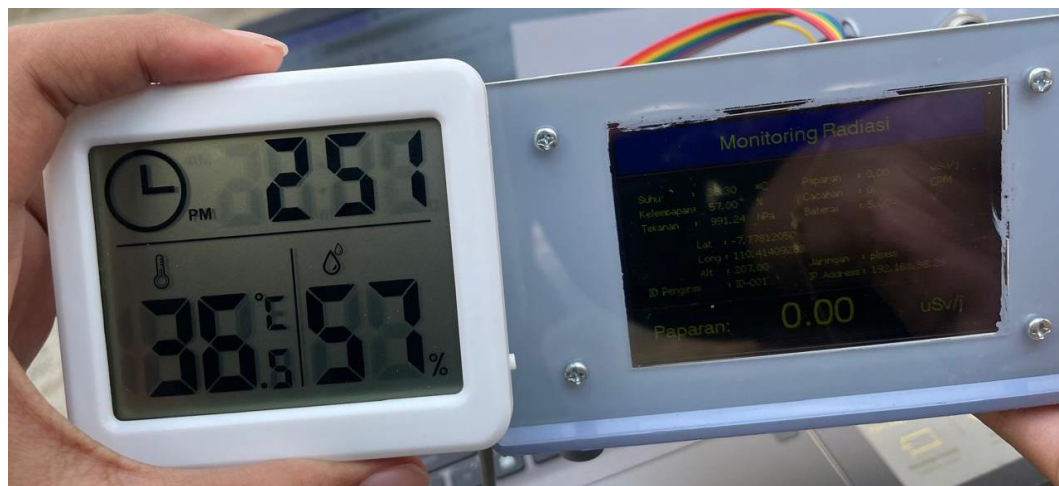
Saat proses pengambilan data dilakukan, alat dibiarkan terlebih dahulu selama kurang lebih 10-15 menit untuk mencapai kestabilan termal. Setelah kondisi stabil, pengukuran suhu dilakukan secara berkala dengan perbedaan kondisi keadaan ruangan. Setelah data suhu, kelembapan dan tekanan terbaca, nilai dari penyimpangan eror ditentukan dari pembagian selisih pengukuran perangkat yang dibuat dengan hasil pengukuran dari alat termometer digital

dan dikalikan 100 menggunakan persamaan rumus (Paleri, 2015) yang dimana diuraikan pada persamaan (3.2), kemudian nilai akurasi didapatkan dari hasil nilai 100 dikurangkan dengan nilai penyimpangan eror.

Berikut adalah data kalibrasi sensor BME-280 dan alat thermometer jam digital yang telah dimasukkan ke dalam tahap analisis dari hasil kalibrasi :

Tabel 4. 3 Hasil data kalibrasi Sensor BME-280 dengan Termometer Jam Digital

Titik	Jam Digital		Perangkat yang dibuat		Penyimpangan (%)		Akurasi (%)	
	Suhu (*C)	Kelembaban (%)	Suhu (*C)	Kelembaban (%)	Suhu	Kelembaban	Suhu	Kelembaban
(Luar Ruangan)	36.50	57.00	36.30	57.00	0.55	0	99.45	100
(Di dalam Ruangan BerAC)	27.50	50.00	27.38	49.87	0.43	0.26	99.57	99.74
(Di dalam Suhu Ruang)	34.40	62.00	34.00	62.36	1.16	0.58	98.84	99.42



Gambar 4. 16 Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di luar ruangan



Gambar 4. 17 Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di ruangan bersuhu dingin (berAC)

Dalam situasi dimana tidak tersedia alat pembanding tekanan udara yang akurat, pendekatan umum yang digunakan adalah mengonversi data ketinggian dari GPS menjadi nilai tekanan udara menggunakan persamaan barometrik. Nilai tekanan hasil konversi ini kemudian digunakan sebagai referensi untuk mengkalibrasi atau memvalidasi tekanan pada Sensor BME 280 pada perangkat yang dikembangkan (Fakhtur, 2016).

Tabel 4. 4 Hasil data kalibrasi Tekanan

Altitude GPS-NEO 6M	Tekanan BME-280 (hPa)	Hasil Konversi Altitude ke Tekanan (hPa)	Penyimpangan (%)	Akurasi (%)
164.7	991.24	998.18	0.70	99.30
162.1	991.53	998.19	0.67	99.33
164.7	990.89	997.39	0.65	99.35

4.1.4.3 Hasil Data Kalibrasi Tegangan Baterai

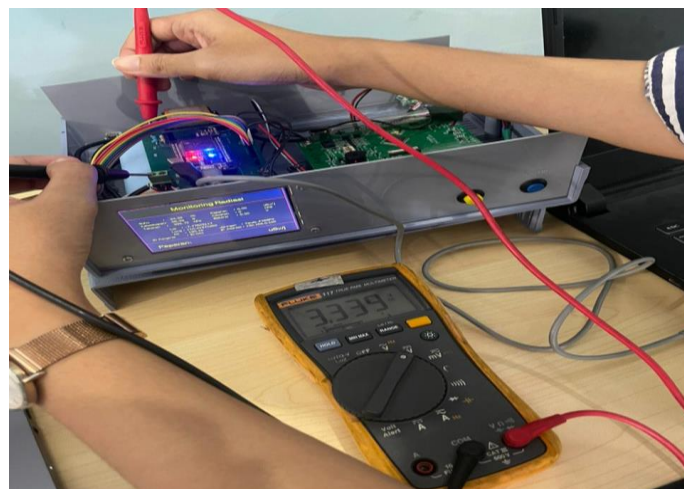
Kalibrasi data tegangan baterai dilakukan untuk mengevaluasi tingkat akurasi pembacaan tegangan oleh perangkat yang dirancang, dengan membandingkan hasilnya perangkat yang dibuat terhadap alat ukur standar yaitu voltmeter digital. Proses pengambilan data dilakukan dengan menggunakan tiga

variasi kondisi status powerbank, yaitu saat baterai dalam kondisi full, setengah terpakai, dan hampir habis. Masing-masing kondisi diuji dengan mengukur tegangan secara langsung menggunakan voltmeter sebagai referensi, kemudian dibandingkan dengan pembacaan tegangan dari perangkat yang dirancang.

Berikut adalah data kalibrasi tegangan baterai dan voltmeter yang telah dimasukkan ke dalam tahap analisis dari hasil kalibrasi :

Tabel 4. 5 Hasil Konversi Tegangan Baterai dengan *Voltmeter*

Variasi	Voltmeter	Perangkat yang dibuat	Penyimpangan (%)	Akurasi (%)
	Tegangan Volt	Tegangan Volt		
Penuh	5.00 V	3.3 V	0	100
Setengah Penuh	5.00 V	3.3 V	0	100
Hampir Habis	5.00 V	3.3 V	0	100



Gambar 4. 18 Pembuktian Kalibrasi Tegangan Baterai dengan *Voltmeter*

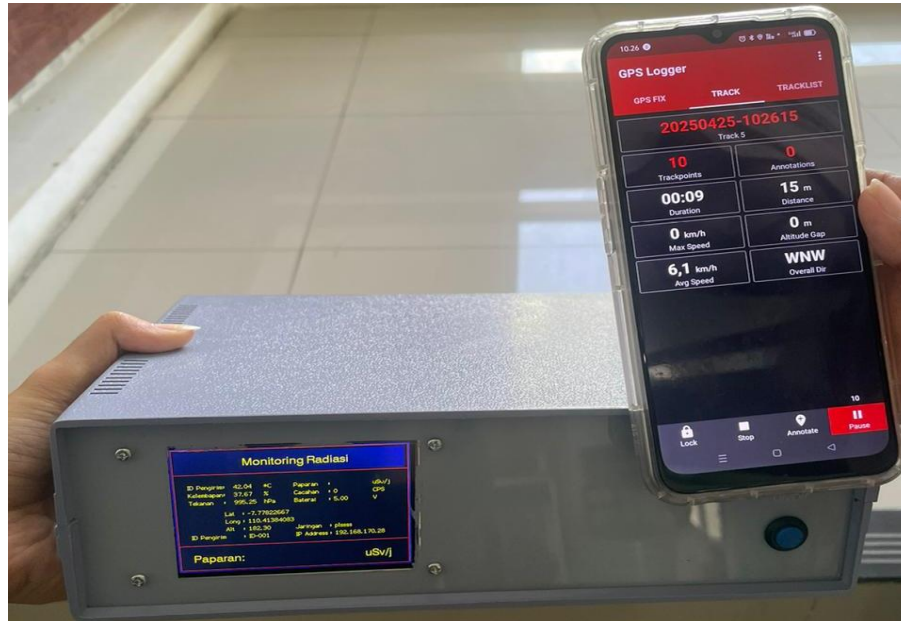
4.1.4.4 Hasil Data Kalibrasi Posisi

Proses kalibrasi sensor GPS NEO-6M dilakukan untuk memastikan tingkat akurasi koordinat yang dihasilkan oleh perangkat yang dirancang, dengan cara membandingkannya terhadap data referensi dari alat pembanding yaitu Track Logger Handphone. Pengambilan data dilakukan pada beberapa titik lokasi

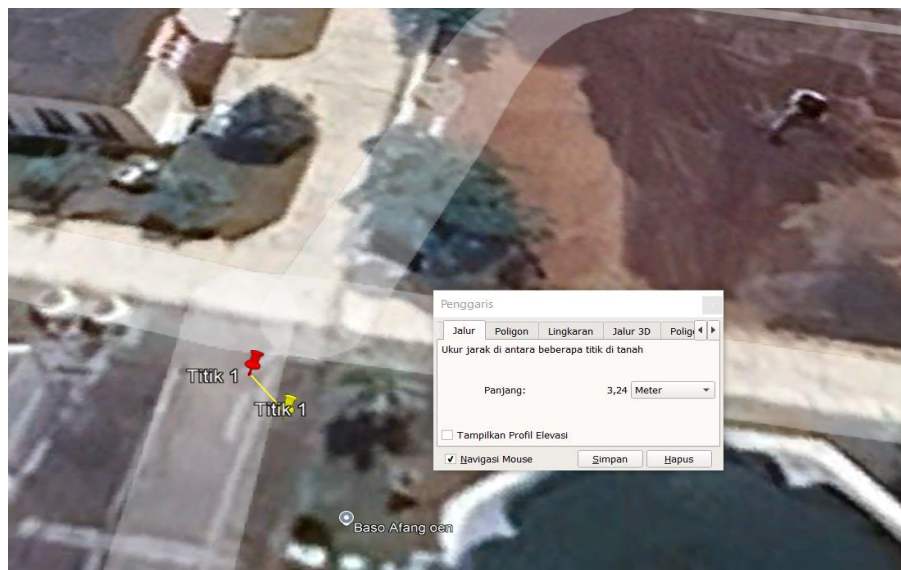
berbeda di area lingkungan reaktor. Kalibrasi ini bertujuan untuk mengetahui seberapa dekat data posisi (*latitude*, *longitude*, dan *latitude*) yang dihasilkan oleh sensor NEO-6M terhadap data dari track logger handphone yang telah teruji keakuratannya.

Tabel 4. 6 Hasil Data Kalibrasi *Track Logger* dengan Sensor NEO-6M

Titik	Data Track Logger			Perangkat yang dibuat			Penyimpangan		
	Alt	Lat	Long	Alt	Lat	Long	Alt	Antar Perangkat	Akurasi
1	185,9	- 7,7778 5833	110,4141 0667	173,5	- 7,77 7831	110,414 09	6.67	3.24 m	1,5 m
2	173,3	- 7,7779 35	110,4139 1833	171,9	- 7,77 7939	110,413 927	0.80	1.06 m	1,7 m
3	178,7	- 7,7781 9167	110,4140 5333	177,9	- 7,77 8201	110,414 023	0.44	3.43 m	1,8 m
4	167	- 7,7783 65	110,4140 2333	191,7	- 7,77 8336	110,414 004	14.7	3.95 m	1,6 m
5	163,1	- 7,7783 7333	110,4136 3167	188,7	- 7,77 8408	110,413 65	15.6 9	4.46 m	1,8 m



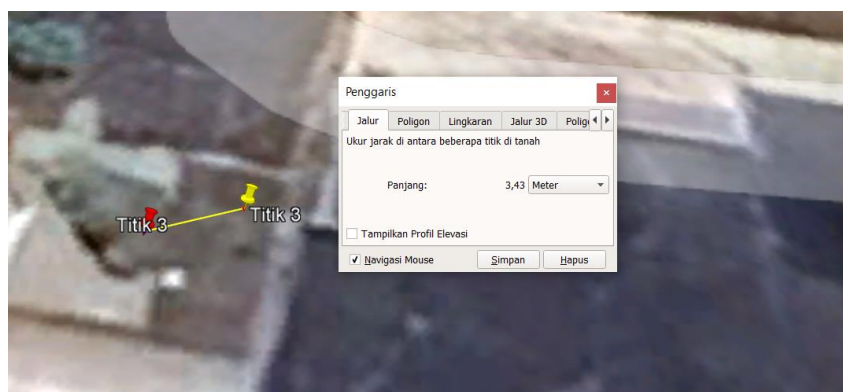
Gambar 4. 20 Dokumentasi Kalibrasi Data Posisi



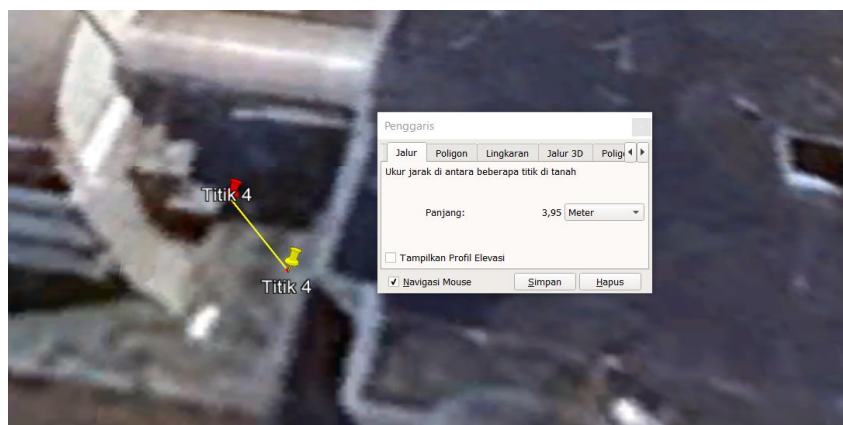
Gambar 4. 19 Pembuktian Antar Jarak Pada Titik 1



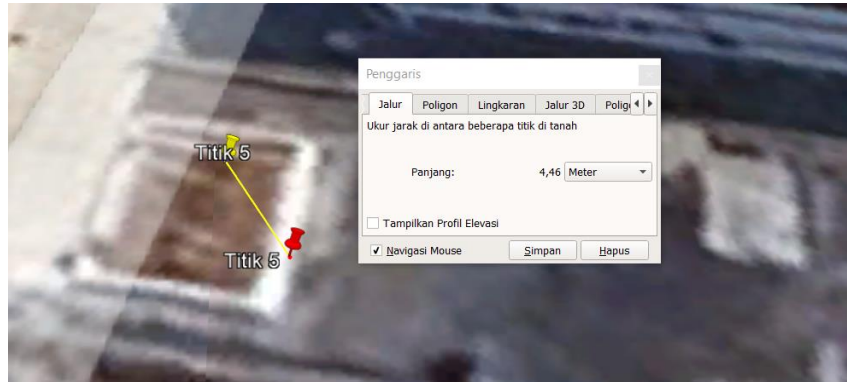
Gambar 4. 23 Pembuktian Rentang Titik 2 Pada Antar Perangkat



Gambar 4. 22 Pembuktian Rentang Titik 3 Pada Antar Perangkat



Gambar 4. 21 Pembuktian Rentang Titik 4 Pada Antar Perangkat



Gambar 4. 24 Pembuktian Rentang Titik 5 Pada Antar Perangkat

4.2 Pembahasan

Berdasarkan tabel 4.1, telah dilakukan proses kalibrasi perangkat detektor radiasi buatan terhadap surveymeter standar dengan mendekatkannya pada sumber radioaktif. Dalam pengambilan data ini, perangkat dirancang untuk menghitung jumlah cacahan (*counts*) dalam periode waktu tertentu, yaitu selama 5 detik, kemudian hasil tersebut dikonversi menjadi satuan CPM (*Counts Per Minute*) dengan persamaan (3.3)

Dengan periode waktu dalam menghitung jumlah pulsa sebesar 5000 ms, maka setiap hasil perhitungan cacahan selama 5 detik dikalikan dengan faktor 12 untuk mendapatkan estimasi jumlah cacahan per menit. Untuk hasil paparan radiasi jenis tabung *Geiger-Müller* 4011 memiliki konversi indeks 151, yang berarti hasil dari 151 CPM adalah 1 uSv/J, yang dimana hasil dari cacahan yang didapat dibagi dengan hasil konversi tersebut yaitu 151 CPM yang menghasilkan hasil dari paparan radiasi tersebut.

Pada kondisi pengukuran saat didekatkan dengan sumber radioaktif, surveymeter standar mencatat paparan sebesar 1.67 uSv/J dengan 252 CPM, sedangkan perangkat buatan mencatat paparan sebesar 2.04 uSv/J dan 308 CPM. Dari hasil tersebut, dapat dihitung presentase penyimpangan antara perangkat

dengan alat standar, yaitu 18.13% untuk nilai paparan dan 18.18% untuk cacahan. Akurasi perangkat terhadap alat standar juga dihitung, yaitu sebesar 81.87% untuk paparan dan 81.82% untuk cacahan. Hasil ini menunjukkan bahwa perangkat buatan memiliki tingkat akurasi yang cukup baik terhadap alat standar, meskipun terdapat deviasi tertentu yang dapat disebabkan oleh faktor kalibrasi sensor, waktu pembacaan, atau sensitivitas detektor terhadap partikel.

Tabel 4.2 menunjukkan perbandingan data hasil kalibrasi antara Sensor BME-280 dengan Termometer Digital. Data tersebut diambil dengan keadaan variasi dalam keadaan di luar ruangan, di dalam ruangan ber-AC, dan pada keadaan suhu ruang biasa. Alat dibiarkan terlebih dahulu selama kurang lebih 10-15 menit untuk mencapai kestabilan termal. Alat yang diuji ditempatkan pada keadaan yang sama dalam keadaan berdampingan. Penempatan ini dimaksudkan untuk memastikan bahwa kedua alat mengalami kondisi lingkungan yang identik sehingga pembacaan perangkat yang dibuat dapat dibandingkan secara objektif.

Pada keadaan di luar ruangan, hasil yang didapatkan pada Sensor BME-280 menghasilkan suhu sebesar 36.30°C dengan presentase kelembapan sebesar 57.00%, yang dibandingkan dengan alat sekunder berupa thermometer jam digital yang menghasilkan nilai suhu sebesar 36.50°C dan kelembapan sebesar 57.00 %, hasil dari penyimpangan kedua alat tersebut menghasilkan penyimpangan suhu sebesar 0.55% dan penyimpangan kelembapan sebesar 0%, dengan akurasi mencapai 99.45% untuk suhu, dan 100% untuk kelembapan.

Dalam kondisi ruangan berpendingin udara (AC), sensor BME-280 mencatat suhu sebesar 27.38°C dengan presentase kelembapan sebesar 49,87%, yang dibandingkan alat sekunder berupa thermometer jam digital menghasilkan nilai

suhu sebesar 27.50°C dan kelembapan sebesar 50.00%, hasil dari kedua alat tersebut menghasilkan penyimpangan suhu sebesar 0.43% dan penyimpangan kelembapan sebesar 0.26%, dengan akurasi mencapai 99.57% untuk suhu, dan 99.74% untuk kelembapan.

Sementara itu, pada ruangan bersuhu normal, suhu yang terukur oleh Sensor BME-280 adalah 34.00°C dengan presentase kelembapan sebesar 62.36%, yang dibandingkan thermometer digital menghasilkan nilai suhu sebesar 34.40°C dengan kelembapan sebesar 62.00%. hasil dari kedua alat tersebut menghasilkan nilai suhu sebesar 1.16% dan kelembapan 0.58% dengan akurasi mencapai 98.84% untuk suhu dan 99.42 % untuk kelembapan.

Kalibrasi tekanan pada sensor BME-280 dalam penelitian kali ini dilakukan dengan membandingkan hasil pembacaannya terhadap tekanan udara yang diperoleh melalui nilai konversi ketinggian (*altitude*) dari modul GPS NEO-6M. Dalam kondisi di mana alat ukur tekanan yang terstandar tidak tersedia, metode ini merupakan pendekatan umum yang valid secara ilmiah untuk memperoleh nilai tekanan udara berdasarkan model atmosfer standar. Metode konversi ini memanfaatkan persamaan barometrik, yang menggambarkan hubungan antara tekanan dan ketinggian (NASA, 2022). Tekanan udara menurun seiring bertambahnya ketinggian, dan hubungan ini dapat di hitung menggunakan persamaan (3.7) yang dimana pada penelitian kali ini menghasilkan konversi dari GPS menunjukkan nilai tekanan di luar ruangan sebesar 998.18 hPa pada ketinggian 164.7 m, pada saat di ruangan berpendingin udara (AC) menunjukkan nilai tekanan sebesar 998.19 hPa pada ketinggian 162.1 m, dan pada saat di ruangan bersuhu normal, menunjukkan nilai tekanan sebesar 997.39 hPa pada ketinggian

164.7 m. Sementara itu, sensor BME-280 menghasilkan pembacaan tekanan sebesar 9921.24 hPa pada saat diluar ruangan, 991.53 hPa pada saat di ruangan berpendingin (AC), dan 990.89 hPa pada saat ruangan bersuhu normal.

Dari hasil ini, dihitung penyimpangan antara nilai referensi hasil konversi dari pembacaan sensor. Nilai penyimpangan berada pada kisaran 0.65%-0.70% dan akurasi sensor mencapai 99.30%-99.35% yang tergolong baik untuk perangkat berbasis mikrokontroler. Penelitian Fakhtur (2016) menyatakan bahwa konversi ketinggian dari GPS menjadi tekanan udara adalah metode valid untuk kalibrasi awal sensor tekanan, terutama dalam penelitian berbasis mikrokontroler dan *Internet of Things* (IoT).

Hasil menandakan bahwa Sensor BME-280 menunjukkan korelasi yang kuat terhadap alat pembanding, dengan tingkat deviasi rendah yang dapat ditoleransi pada berbagai lingkungan. Penelitian dari Rachmawati et al (2014) mencatat bahwa BME-280 dapat diintegrasikan dalam sistem monitoring berbasis mikrokontroler untuk aplikasi indoor maupun outdoor. Berdasarkan *datasheet* resmi dari (Bosch, 2018) memiliki spesifikasi suhu dengan akurasi sebesar ± 1 °C, kelembapan dengan akurasi akurasi: $\pm 3\%$ RH, dan tekanan dengan akurasi ± 1 hPa.

Tabel 4.5 menunjukkan perbandingan data hasil kalibrasi tegangan baterai dari ESP32 dan *voltmeter*, data diambil melalui beberapa variasi yaitu pada saat baterai dalam keadaan penuh, setengah penuh, dan hampir habis. Dalam proses pengukuran tegangan baterai menggunakan mikrokontroler ESP32, diperoleh hasil yang menunjukkan tegangan sebesar 3.3 V dari pembacaan ADC internal. Namun, hasil akhir yang ditampilkan sebesar 5.00 V. Hal ini terjadi bukan karena kesalahan pengukuran, melainkan disebabkan oleh penggunaan rangkaian pembagi tegangan

pada persamaan (3.10) yang berfungsi menurunkan tegangan dari baterai agar tetap berada dalam batas aman yang dapat dibaca oleh ADC ESP32, yaitu maksimum 3.3 V. Rangkaian pembagi tegangan yang dipakai yaitu 51 k Ω (R1) dan 100 k Ω (R2). Persamaan (3.11) memungkinkan sistem untuk menampilkan nilai V_{in} aktual (5.0 V) meskipun input ke ESP32 adalah 3.3 V, karena hasil pembacaan ADC dikonversi kembali berdasarkan konfigurasi resistor. ADC 12-bit ESP32 menghasilkan nilai dari 0 hingga 4095, dan dikonversi nilai ADC ke tegangan menggunakan persamaan (3.10)

Powerbank modern tidak menyalurkan tegangan baterai internal secara langsung ke perangkat, melainkan menggunakan rangkaian pengatur tegangan (*step-up converter*) yang menjaga agar tegangan output tetap stabil pada 5.0 V, terlepas dari kondisi kapasitas baterai di dalam nya. Dengan demikian, meskipun sel baterai internal mengalami penurunan tegangan dari 4.2 V (penuh) hingga mendekati 3.3 V (hampir habis), tegangan output tetap dijaga pada nominal 5 V selama kapasitas baterai masih mencukupi. Output ini baru akan turun atau mati total ketika baterai telah melewati batas bawah (*Cut-off Voltage*) yang ditentukan oleh sistem proteksi powerbank (Texas, 2023).

Hasil ini sesuai dengan penelitian oleh Nanda et al (2020) yang menggunakan pembagi tegangan untuk membaca output baterai ke mikrokontroler dan menunjukkan bahwa sistem ADC yang dikombinasikan dengan pembagi tegangan untuk membaca output baterai ke mikrokontroler dan menunjukkan bahwa sistem ADC yang dikombinasikan dengan pemabagi tegangan dapat memberikan pembacaan akurat terhadap tegangan aktual selama sistem dikalibrasi dengan benar. Penelitian serupa juga dilakukan oleh (Kurniawan, 2011), yang

mengimplementasikan pembagi tegangan untuk membaca tegangan AC dan DC menggunakan mikrokontroler, serta menghasilkan pembacaan yang stabil dan sesuai dengan alat ukur eksternal.

Berdasarkan hasil pengujian, data kalibrasi tegangan baterai menunjukkan hasil pembacaan sebesar 5.00 V, yang juga dimana keluaran dari ADC ESP32 yaitu 3.3 V, menunjukkan sistem memiliki akurasi yang tinggi dengan penyimpangan sebesar 0 %. Meskipun demikian, secara teoritis, setiap sistem ADC memiliki toleransi kesalahan akibat faktor seperti resolusi ADC, toleransi resistor dalam pembagi tegangan, serta fluktuasi tegangan referensi. Oleh karena itu, hasil ini dapat dikatakan mendekati akurat 100%, dan layak digunakan untuk kalibrasi tegangan baterai.

Berdasarkan Tabel 4.6 dilakukan pengujian kalibrasi antara sensor GPS NEO-6M dengan *Track Logger Handphone* sebagai pembanding. Parameter yang diuji meliputi nilai altitude, latitude, dan longitude pada lima titik pengamatan yang berbeda di area sekitar Reaktor Kartini. Sensor GPS NEO-6M digunakan sebagai perangkat utama, sedangkan track logger handphone digunakan sebagai acuan pembanding yang umum digunakan dalam keseharian.

Diketahui bahwa nilai akurasi GPS NEO-6M berada pada rentang antara 1,5 meter hingga 1,8 meter. Akurasi ini merupakan estimasi deviasi posisi terhadap lokasi sebenarnya berdasarkan kekuatan sinyal dan jumlah satelit yang diterima oleh modul GPS. Sementara itu, hasil penyimpangan antar titik menunjukkan nilai yang bervariasi, mulai dari 1,06 meter pada titik kedua hingga 4,46 meter pada titik kelima.

Pada titik kedua tercatat penyimpangan horizontal hanya sebesar 1,06 meter dengan akurasi 1,7 meter. Hal ini menunjukkan bahwa modul GPS bekerja secara optimal dan posisi yang diperoleh sangat mendekati data acuan. Namun, pada titik kelima, meskipun GPS mencatat akurasi internal sebesar 1,8 meter, posisi aktual GPS ternyata meleset sejauh 4,46 meter dari titik referensi. Kondisi ini menunjukkan bahwa nilai akurasi internal tidak selalu mencerminkan hasil posisi yang akurat, karena dapat dipengaruhi oleh kondisi lingkungan, interferensi sinyal, jumlah satelit yang tertangkap, serta keberadaan penghalang fisik seperti bangunan atau pepohonan.

Selain itu, penyimpangan juga terjadi pada parameter altitude (ketinggian). Nilai altitude yang dicatat oleh GPS NEO-6M menunjukkan adanya perbedaan terhadap nilai pembanding dari track logger. Sebagai contoh, pada titik pertama, nilai altitude yang tercatat adalah 185,9 meter, sedangkan nilai pembanding dari handphone adalah 173,5 meter, menghasilkan selisih 12,4 meter. Begitu pula pada titik keempat, terdapat selisih 14,7 meter antara kedua perangkat. Perbedaan ini menunjukkan bahwa modul GPS NEO-6M cenderung memiliki deviasi yang lebih besar dalam pengukuran ketinggian dibandingkan posisi horizontal (*latitude* dan *longitude*). Hal ini umum terjadi karena sinyal GPS untuk altitude lebih rentan terhadap gangguan dan biasanya memiliki akurasi yang lebih rendah dibandingkan koordinat horizontal.

Perbandingan hasil ini juga dikuatkan oleh penelitian terdahulu. Yunita et al. (2024) dalam penelitiannya menyebutkan bahwa modul Ublox NEO-6M memiliki tingkat akurasi rata-rata antara 1,87 hingga 3,27 meter, dengan deviasi maksimum mencapai 6,5 meter berdasarkan pengujian di ruang terbuka. Sementara itu, Utami

et al. (2025) menjelaskan bahwa penggunaan modul GPS NEO-6M untuk pelacakan posisi kapal menunjukkan hasil yang cukup akurat dan responsif secara real-time, namun tetap memiliki penyimpangan tergantung kondisi medan.

Dengan demikian, dapat disimpulkan bahwa meskipun GPS NEO-6M memiliki kemampuan akurasi yang baik secara teori, dalam praktiknya tetap terjadi penyimpangan antar titik yang fluktuatif, baik pada koordinat horizontal maupun nilai ketinggian, tergantung pada kondisi eksternal. Hal ini penting diperhatikan dalam implementasi sistem yang mengandalkan data lokasi presisi, khususnya pada aplikasi pelacakan berbasis GPS.

4.3 Keterkaitan Hasil Penelitian Dalam Perspektif Islam

Penelitian ini adalah untuk mengkaji akurasi dan keandalan dari detektor *Geiger-Müller*, sensor BME-280 dan GPS NEO-6M dalam sistem monitoring berbasis mikrokontroler, khususnya dalam mengukur parameter dari radiasi dan lingkungan seperti suhu, kelembapan, tekanan, serta posisi. Melalui proses kalibrasi dengan alat pembanding dan metode konversi ilmiah seperti persamaan barometrik, diharapkan sistem yang dibangun mampu menghasilkan data yang mendekati kondisi aktual di lingkungan sekitarnya.

Sebagaimana yang dijelaskan pada surat Al-Hijr ayat 19 :

وَالْأَرْضَ مَدَدْنَاهَا وَأَلْقَيْنَا فِيهَا رَوَاسِيَ وَأَنْبَتْنَا فِيهَا مِنْ كُلِّ شَيْءٍ مَّوْزُونٍ

Artinya : “Kami telah menghamparkan bumi, memancangkan padanya gunung-gunung, dan menumbuhkan di sana segala sesuatu menurut ukuran(-nya).”

Kata “مَوْزُونٍ” dalam ayat tersebut secara bahasa berarti sesuatu yang diukur secara presisi dan seimbang. Dalam konteks ini, hasil pengukuran dari sistem yang dirancang dapat dikatakan telah berhasil mendekati ukuran aktual alam yang telah

ditetapkan Allah, sebagaimana ditunjukkan oleh kesesuaian data dengan alat pembanding. Hasil penelitian ini menunjukkan bahwa ketika alat ukur yang digunakan telah dikembangkan dan dikalibrasi dengan baik, maka ia mampu membaca “ukuran-ukuran” tersebut secara akurat. Ini menjadi bukti bahwa alam bekerja berdasarkan hukum yang teratur dan dapat didekati dengan metode ilmiah, sebagaimana nilai “مُؤَرُونَ” yang terkandung dalam ayat tersebut.

Ayat ini menegaskan bahwa Allah menciptakan bumi dengan keseimbangan dan ukuran yang tepat. Dalam Tafsir (Katsir, 2000) dijelaskan bahwa segala sesuatu yang tumbuh di bumi diciptakan dengan ukuran dan kadar yang telah ditentukan oleh Allah. Hal ini mencerminkan keteraturan dan keseimbangan dalam ciptaan-Nya.

Penelitian ini tentang Rancang Bangun Perangkat Monitor Radiasi Gamma Berbasis Mikrokontroler ESP32 Menggunakan Detektor *Geiger-Müller* untuk Pemantauan Radiasi dan Lingkungan Terpadu sejalan dengan konsep ini. Dengan merancang sistem yang mampu memantau radiasi dan parameter lingkungan secara akurat, saya mencerminkan upaya manusia untuk memahami dan menjaga keseimbangan alam yang telah ditetapkan oleh Allah. Sistem yang saya kembangkan bertujuan untuk mendeteksi dan mengukur radiasi gamma serta parameter lingkungan lainnya dengan presisi, yang merupakan bagian dari menjaga dan memahami keteraturan ciptaan-Nya.

Dengan demikian, penelitian ini tidak hanya berkontribusi pada ilmu pengetahuan dan teknologi, tetapi juga mencerminkan pemahaman dan penghargaan terhadap keseimbangan dan keteraturan yang telah Allah tetapkan di alam semesta.

Di sisi lain, kehadiran teknologi monitoring ini juga memiliki nilai fungsional dalam konteks menjaga lingkungan dari kerusakan.

Yang sebagaimana tercantum pada surat Ar-Rum ayat 41 :

ظَهَرَ الْفَسَادُ فِي الْبَرِّ وَالْبَحْرِ بِمَا كَسَبَتْ أَيْدِي النَّاسِ لِيُذِيقَهُمْ بَعْضَ الَّذِي عَمِلُوا لَعَلَّهُمْ
يَرْجِعُونَ

Artinya : *"Telah tampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia; Allah menghendaki agar mereka merasakan sebagian dari (akibat) perbuatan mereka, agar mereka kembali (ke jalan yang benar)."*

Kata "الْفَسَادُ" (*al-fasād*) bermakna kerusakan yang muncul di daratan dan lautan akibat ulah manusia. (Shihab, 2002) menjelaskan bahwa kerusakan tersebut mencakup kerusakan ekologis dan lingkungan, baik yang tampak secara fisik maupun sistemik. Dalam konteks penelitian ini, paparan radiasi yang tidak terkontrol dapat menjadi salah satu bentuk kerusakan lingkungan yang berbahaya, karena dapat merusak ekosistem dan mengganggu kesehatan manusia.

Ayat tersebut menunjukkan bahwa manusia sering menjadi penyebab utama kerusakan lingkungan akibat aktivitas yang tidak dikendalikan, termasuk potensi bahaya dari paparan radiasi. Saya mengembangkan penelitian ini sebagai upaya ilmiah untuk meminimalkan kerusakan tersebut melalui teknologi monitoring yang dapat mendeteksi keberadaan radiasi gamma serta parameter lingkungan lainnya secara akurat. Dengan adanya data yang mendekati valid dan akurat, manusia dapat melakukan tindakan pencegahan dan pengendalian, yang pada akhirnya mendukung prinsip islam dalam menjaga bumi dari kerusakan yang lebih besar.

Dengan demikian, penelitian ini tidak hanya memberikan kontribusi terhadap pengembangan teknologi monitoring lingkungan dan radiasi berbasis mikrokontroler, tetapi juga mencerminkan pelaksanaan nilai-nilai islam dalam

menjaga kelestarian alam. Upaya ini sejalan dengan perintah Allah SWT agar manusia tidak melakukan kerusakan di muka bumi, serta menjadi wujud nyata dari amanah sebagai khalifah dalam memelihara dan mengelola ciptaan-Nya secara bijak dan bertanggung jawab.

Penelitian ini juga dapat dikaji dari sudut pandang Maqashid Syariah, yaitu tujuan-tujuan utama dari syariat Islam yang bertujuan menjaga dan mewujudkan kemaslahatan bagi manusia. Terdapat lima pokok utama dalam Maqashid Syariah, yaitu: *Hifzh al-Din* (menjaga agama), *Hifzh al-Nafs* (menjaga jiwa), *Hifzh al-'Aql* (menjaga akal), *Hifzh al-Mal* (menjaga harta), dan *Hifzh al-Nasl* (menjaga keturunan).

Jika dianalisis dari kelima aspek tersebut, maka hasil penelitian ini dapat dijabarkan sebagai berikut: Pertama, dari sisi *Hifzh al-Din* (menjaga agama), hasil penelitian ini menunjukkan bentuk tanggung jawab spiritual dalam menjaga keseimbangan alam ciptaan Allah SWT. Sistem monitoring yang dirancang mencerminkan pelaksanaan amanah sebagai *khalifah* di bumi untuk merawat dan tidak merusak lingkungan. Hal ini sejalan dengan QS. Ar-Rum ayat 41 yang menjelaskan bahwa kerusakan di darat dan laut timbul akibat ulah manusia. Dengan sistem ini, manusia diberi alat untuk memantau dan mencegah kerusakan, sebagai bagian dari ibadah dan pengamalan ajaran agama Islam.

Kedua, pada aspek *Hifzh al-Nafs* (menjaga jiwa), hasil penelitian menunjukkan bahwa sistem yang dibangun mampu mendeteksi paparan radiasi gamma secara akurat. Keakuratan ini penting dalam memberikan peringatan dini terhadap potensi bahaya kesehatan, seperti kanker atau kerusakan organ akibat paparan radiasi.

Dengan adanya sistem monitoring ini, manusia dapat mengambil langkah pencegahan untuk menjaga keselamatan dan kesehatan jiwa secara lebih efektif.

Ketiga, dalam perspektif *Hifzh al-'Aql* (menjaga akal), penelitian ini adalah wujud pemanfaatan akal dan potensi intelektual manusia dalam bidang teknologi dan sains. Dengan memanfaatkan teknologi mikrokontroler ESP32, sensor Geiger-Müller, serta integrasi dengan berbagai parameter lingkungan, sistem ini membantu manusia memahami dan mengelola alam secara rasional. Hal ini sejalan dengan nilai Islam yang mendorong pencarian ilmu dan pengembangan teknologi demi kemaslahatan bersama.

Keempat, dalam konteks *Hifzh al-Mal* (menjaga harta), sistem ini berkontribusi dalam mencegah kerusakan aset dan sumber daya akibat paparan radiasi atau perubahan lingkungan yang ekstrem. Dengan data pemantauan suhu, kelembapan, tekanan udara, dan ketinggian, sistem ini dapat digunakan dalam sektor industri, pertanian, atau pemukiman untuk menghindari kerugian materiil akibat lingkungan yang tidak stabil. Ini membuktikan bahwa penelitian ini mendukung perlindungan terhadap harta benda secara langsung.

Kelima, dari sisi *Hifzh al-Nasl* (menjaga keturunan), sistem ini turut serta dalam upaya perlindungan terhadap generasi mendatang. Paparan radiasi dalam jangka panjang dapat berdampak pada reproduksi dan kesehatan anak-anak. Oleh karena itu, sistem monitoring ini memberikan data penting untuk pengendalian lingkungan, yang pada akhirnya berdampak pada keberlanjutan kesehatan masyarakat dan keturunan yang akan datang.

Dengan demikian, penelitian ini secara utuh tidak hanya berkontribusi pada pengembangan ilmu pengetahuan dan teknologi, tetapi juga mencerminkan

penerapan nilai-nilai Islam dalam menjaga kemaslahatan hidup manusia dan lingkungan sesuai dengan prinsip Maqashid Syariah.

BAB V

PENUTUP

5.1 Kesimpulan

1. Telah berhasil dilakukan perancangan sistem perangkat monitor radiasi gamma berbasis mikrokontroler ESP32 dengan menggunakan detektor *Geiger-Müller* yang mampu mendeteksi intensitas radiasi dan lingkungan terpadu
2. Perangkat ini mampu menampilkan hasil pemantauan radiasi dan lingkungan pada layar serta dapat mengirimkan data secara nirkabel melalui konektivitas Wi-Fi, mendukung sistem pemantauan radiasi dan lingkungan terpadu
3. Hasil pengujian menunjukkan bahwa perangkat bekerja secara stabil, dan memiliki Tingkat akurasi yang cukup baik untuk pemantauan radiasi dan lingkungan
4. Dengan integrasi data lingkungan lainnya, perangkat ini layak digunakan sebagai sistem awal (*early warning system*) dan media edukasi dalam pemantauan radiasi yang bersifat portable dan ekonomis.

5.2 Saran

Pengembangan lebih lanjut dari sistem ini dapat mencakup penyimpanan data berbasis cloud agar pemantauan dapat dilakukan secara jarak jauh dan terintegrasi dalam sistem *Internet of Things* (IoT). Selain itu, penambahan fitur alarm atau notifikasi otomatis saat deteksi radiasi melebihi ambang batas akan meningkatkan aspek keamanan pengguna. Kalibrasi berkala juga disarankan untuk menjaga akurasi alat, terutama jika digunakan di lingkungan dengan variabel lingkungan

ekstrem. Penelitian ini juga dapat dijadikan dasar untuk pengembangan sistem pemantauan lingkungan terpadu di area industri, laboratorium nuklir, atau kawasan rawan radiasi lainnya, dengan tetap mengedepankan prinsip ketelitian dan akurasi sesuai standar ilmiah dan nilai-nilai spiritual.

DAFTAR PUSTAKA

- Ahmed, A., Khan, M., & Ali, R. (2021). Efficient environmental monitoring using the BME280 sensor for IoT applications. *Journal of Sensors and IoT Applications*, 10(2), 95–108.
- Anonim. (2020). Panduan pemantauan radiasi di fasilitas nuklir. Badan Tenaga Nuklir Nasional (BATAN).
- Arifin, A., Suryani, I., & Ramadhan, R. (2019). Analisis penggunaan Arduino IDE dalam pembelajaran sistem elektronika pada SMK di Indonesia. *Jurnal Pendidikan Teknik Elektronika*.
- Babak, S. (2014). Radiation monitoring of environment using unmanned aerial complex. *The Advanced Science Journal*, 12, 41–44.
- Budianto, M. (2019). Peran teknologi dalam pemantauan cesium-137 di lingkungan. *Jurnal Teknologi dan Inovasi*, 18(1), 20–28.
- Drosdowski, P., & Müller, M. (2019). Optimizing embedded system design with Arduino. Elsevier.
- Habib, A. (2016). Pengembangan sistem monitoring jarak jauh radiasi gamma secara realtime berbasis web server. *Jurnal Teknologi Informasi*, 5.
- Hart, D. (2018). Development and optimization of gamma radiation monitoring systems.
- Himawan. (2021). Rancang bangun sistem monitoring kualitas udara menggunakan ESP32 dan protokol MQTT. UIN Syarif Hidayatullah.
- Iskandar, A. (2015). Pengukuran radiasi gamma untuk keperluan pemantauan lingkungan. *Jurnal Sains Nuklir Indonesia*, 8(2), 34–42.
- Jones, R. (2019). Introduction to radiation detection and measurement. Springer.
- Kamal. (2023). Implementasi aplikasi Arduino IDE pada mata kuliah sistem digital. *Jurnal Pendidikan dan Teknologi*.
- Knoll, G. F. (2010). Radiation detection and measurement (4th ed.). John Wiley & Sons.
- Knoll, G. F. (2020). Radiation detection and measurement. Wiley.
- Kumar, R., Singh, S., & Sharma, M. (2020). Advancements in IoT-based environmental monitoring systems: Applications and challenges. *Journal of Environmental Monitoring and Management*, 14(3), 200–212.

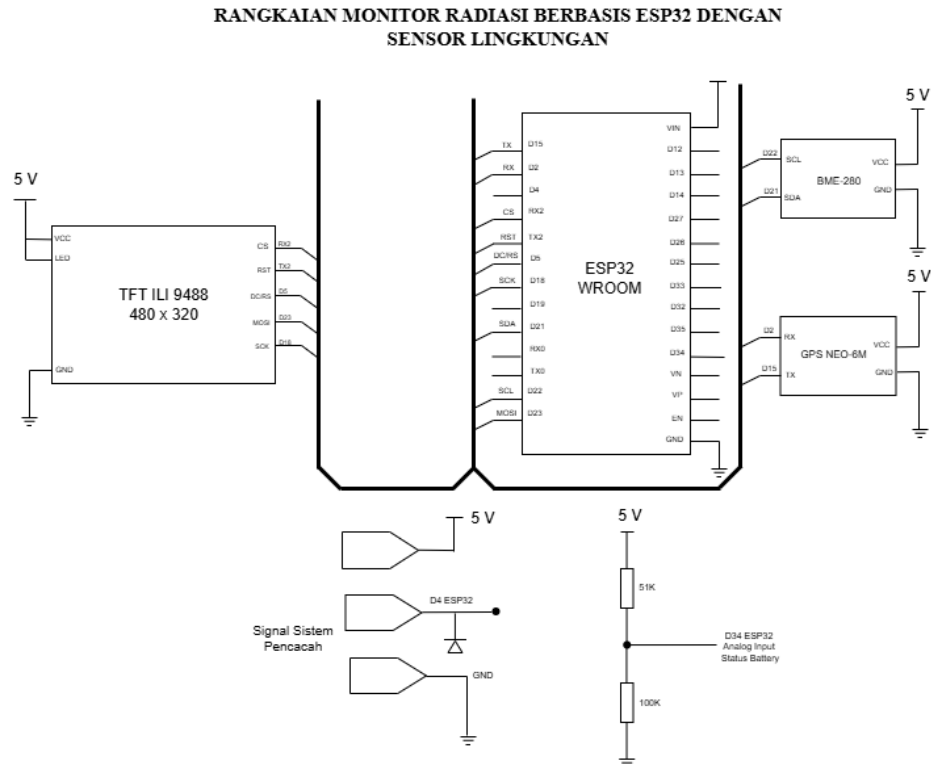
- Leo, J. (2016). Prinsip kerja Arduino dan penggunaannya sebagai alat untuk belajar dan penelitian. *Jurnal Internasional Kontrol, Otomasi, Komunikasi dan Sistem*, 1(2).
- Mahda. (2022). Pemanfaatan modul Geiger-Müller untuk mendeteksi radiasi pada pengolahan limbah B3 rumah sakit. *Jurnal Eltek*, 95–102.
- Miller, A., & Collins, J. (2022). Advanced monitoring systems for radiation protection: Environmental and health safety. *Health Physics Journal*, 59(4), 305–320.
- Muttaqin, et al. (2024). Pemantauan kualitas udara berbasis IoT dengan sensor DHT11 dan sensor MQ135. *Jurnal Pengelolaan Laboratorium Pendidikan*, 102–115.
- Nasution, R. (2014). Pemantauan radiasi gamma cesium-137 pasca-kecelakaan nuklir. *Jurnal Teknologi Nuklir*, 5(1), 10–18.
- Nugroho, T. S., & Setyadi, W. S. (2005). Desain dan perancangan area monitor berbasis mikrokontroler menggunakan detektor Geiger-Müller. *Prosiding Pertemuan dan Presentasi Ilmiah Teknologi Akselerator dan Aplikasinya*, 7. BATAN.
- Patel, M., & Kumar, R. (2019). Arduino: A flexible and accessible platform for interactive embedded systems design in education. *International Journal of Engineering Education*, 1365–1375.
- Purnama, E. (2017). Studi kontaminasi radiasi cesium-137 di sekitar area Chernobyl. *Jurnal Lingkungan dan Kesehatan*, 20(3), 70–75.
- Prasetyo, H., & Andriani, M. (2020). Analisis komponen pada catu daya tegangan tinggi untuk aplikasi industri. *Jurnal Rekayasa Elektrika*, 15(1), 75–80.
- Rahman, F., Fauzi, I., & Ardianto, A. (2019). Pengaruh pembaruan sistem Arduino IDE terhadap efisiensi pengembangan proyek embedded di Indonesia. *Jurnal Teknologi dan Sistem Informasi*, 4(2), 200–207.
- Sari, D. (2018). Pencemaran radiasi cesium-137 di Indonesia. *Jurnal Ilmu Lingkungan*, 14(4), 100–108.
- Sayono, & Sujitno, B. A. T. (2010). Pengaruh tekanan gas isian argon-etanol dan argon-brom terhadap unjuk kerja detektor Geiger-Mueller. *Jurnal Iptek Nuklir Ganendra*, 13(2), 64–76.
- Setiawan, D. (2016). Penggunaan detektor Geiger-Müller untuk pemantauan radiasi gamma. *Jurnal Teknik Nuklir*, 10(3), 29–37.

- Shobari, I., Hidayat, R., & Sumaryono, S. (2014). Pemantauan paparan radiasi lingkungan terpadu dengan komunikasi GSM/GPRS. *Jurnal Forum Nuklir*, 8(2), 93–100.
- Singh, P., & Kumar, S. (2020). Gamma radiation and its applications in nuclear reactor monitoring. *Journal of Nuclear Science and Engineering*, 45(3), 201–215.
- Sprawls, P. (2016). *Energy and radiation*.
- Sutapa, I., Wijaya, R., & Hartono, S. (2016). Pengenalan radiasi dan jenis-jenisnya dalam pengukuran radiasi lingkungan. *Jurnal Fisika dan Teknologi*, 5(3), 112–120.
- Taştan, S., & Gökozan, H. (2019). The integration of environmental sensor data and positioning data for radiation monitoring systems. *Journal of Environmental Monitoring*, 21(5), 347–355.
- Tero, T. (2019). *Arduino projects for engineers: Fundamentals of open source hardware development*. Springer.
- Wahyudi, T. (2017). *Teknologi deteksi radiasi: Prinsip dan aplikasi*. Bandung: ITB Press.
- WHO. (2020). *Electromagnetic fields and public health*. World Health Organization Report.
- Widodo, H. (2014). Dampak kesehatan akibat paparan cesium-137: Studi kasus Fukushima. *Jurnal Kesehatan Masyarakat*, 12(2), 45–52.
- Widodo, T. (2021). Efek biologis radiasi gamma. *Jurnal Kesehatan Lingkungan*, 10(4), 56–63.
- Wijaya, A., Santoso, B., & Prasetyo, E. (2019). Understanding radiation and its effects: A review on radiation monitoring systems. *Journal of Radiation Safety and Technology*, 8(2), 45–59.
- Yang, W., & Chen, Y. (2018). High voltage power supplies: Fundamentals, technology, and applications. *IEEE Transactions on Industrial Electronics*, 65(9), 7412–7422.
- Zhao, H., & Li, Y. (2022). Principles and applications of Geiger-Müller counters in ionizing radiation detection. *Radiation Detection and Measurement Journal*, 32(2), 123–135.

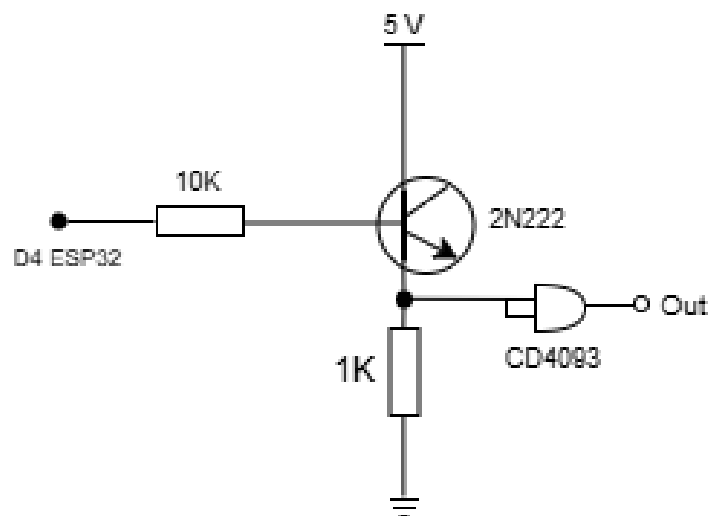
LAMPIRAN

Lampiran 1. 1

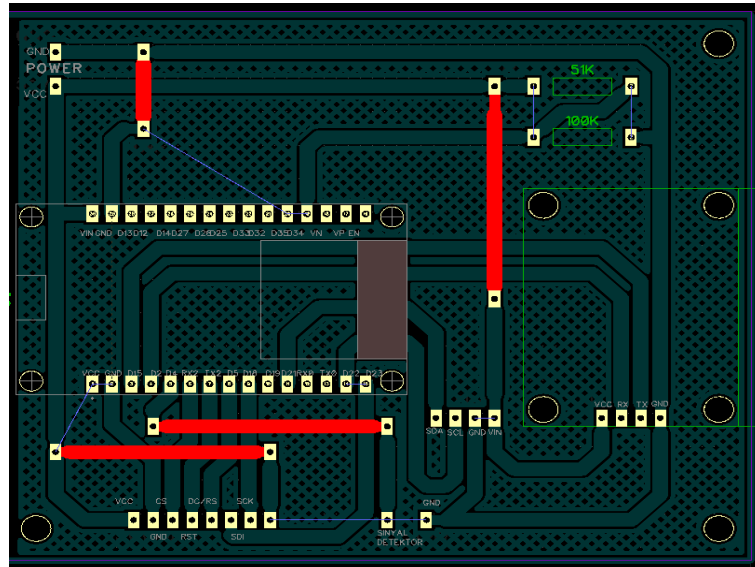
Gambar Perancangan Perangkat Keras



Schematic Rangkaian Monitor Radiasi dan Lingkungan



Schematic Signal Conditioning Kit Detektor Geiger Muller



MONITORING RADIASI

Suhu : *C

Paparan : uSv/J

Kelembapan : %

Cacahan : CPM

Tekanan : hPa

Baterai : V

Lat :

Long :

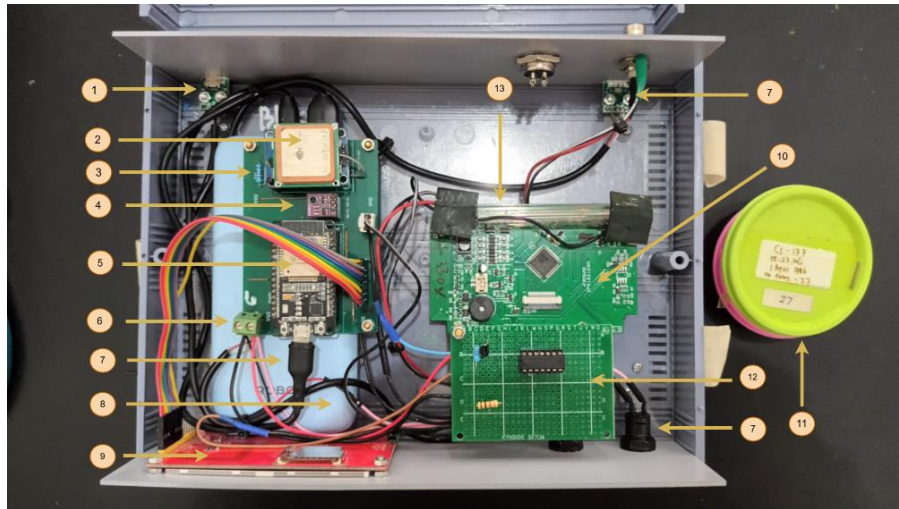
Alt :

ID Pengirim :

Jaringan : IP Address :

PAPARAN : uSv/J

Lampiran 1. 2 Gambar Hasil Perancangan Perangkat Keras



Hasil Perancangan Sistem Perangkat Keras (*Hardware*)



Hasil *Display* Tampilan TFT

	A	B	C	D	E	F	G	H	I	J	K	L
1	TIMESTAMP	ID PENGIRIM	Suhu (°C)	Kelembaban (%)	Tekanan (hPa)	Cacahan (CPS)	Latitude	Longitude	Altitude	Paparan (uSv/J)	Baterai Voltage (V)	% Baterai
97	2025-04-28 13:43:28 ID-001		34.38	44.94	991.88	0	-7.7781515	110.4138167	175.90	5.00	100.00	
98	2025-04-28 13:43:36 ID-001		34.38	44.44	991.99	0	-7.7781515	110.4138167	175.90	5.00	100.00	
99	2025-04-28 13:43:43 ID-001		34.31	44.50	992.09	0	-7.778056	110.4137117	175.90	5.00	100.00	
100	2025-04-28 13:43:51 ID-001		34.22	43.94	992.25	0	-7.778091	110.4137448	176.10	5.00	100.00	
101	2025-04-28 13:43:59 ID-001		34.20	45.25	992.51	0	-7.778193667	110.4137692	176.10	5.00	100.00	
102	2025-04-28 13:44:07 ID-001		34.17	45.54	992.49	0	-7.778193667	110.4137692	176.10	5.00	100.00	
103	2025-04-28 13:44:14 ID-001		33.93	42.88	992.50	0	-7.778193667	110.4137692	176.10	5.00	100.00	
104	2025-04-28 13:44:22 ID-001		34.15	42.78	992.55	0	-7.778117833	110.413915	171.60	5.00	100.00	
105	2025-04-28 13:44:29 ID-001		34.74	42.18	992.66	0	-7.778117833	110.413915	171.60	5.00	100.00	
106	2025-04-28 13:44:37 ID-001		35.78	40.89	992.70	0	-7.778074667	110.4140328	174.90	5.00	100.00	
107	2025-04-28 13:44:45 ID-001		36.68	38.44	992.73	0	-7.778069	110.414018	176.10	5.00	100.00	
108	2025-04-28 13:44:52 ID-001		37.42	37.37	992.77	0	-7.778069	110.414018	176.30	5.00	100.00	
109	2025-04-28 13:45:17 ID-001		39.68	33.82	992.74	0	-7.778067833	110.414	176.30	5.00	100.00	
110	2025-04-28 13:45:26 ID-001		40.38	33.34	992.75	0	-7.778066333	110.4139958	177.40	5.00	100.00	
111	2025-04-28 13:45:33 ID-001		40.83	33.10	992.76	0	-7.778066333	110.4139958	177.40	5.00	100.00	
112	2025-04-28 13:45:41 ID-001		41.17	31.91	992.75	0	-7.778063667	110.4140038	178.60	5.00	100.00	
113	2025-04-28 13:45:49 ID-001		41.27	31.65	992.77	0	-7.778057667	110.4140047	178.90	5.00	100.00	
114	2025-04-28 13:45:56 ID-001		41.87	32.26	992.79	0	-7.7780535	110.4140048	178.60	5.00	100.00	
115	2025-04-28 13:46:04 ID-001		42.20	30.39	992.79	0	-7.7780515	110.414003	177.80	5.00	100.00	
116	2025-04-28 13:46:12 ID-001		42.74	30.78	992.78	0	-7.778054667	110.4140002	176.50	5.00	100.00	
117	2025-04-28 13:46:20 ID-001		42.97	29.82	992.77	0	-7.778054667	110.4140002	176.50	5.00	100.00	
118	2025-04-28 13:46:27 ID-001		43.02	29.50	992.78	0	-7.778061167	110.4139933	175.60	5.00	100.00	
119	2025-04-28 13:46:35 ID-001		43.27		29.11 992.76	0	-7.778062	110.4139912	175.00	5.00	100.00	
120	2025-04-28 13:46:43 ID-001		43.77	30.41	992.80	0	-7.7780535	110.4139903	175.00	5.00	100.00	

Hasil *Display* Tampilan *Spreadsheet*



Hasil Paparan dan Cacahan dari display TFT



Paparan dan Cacahan dari Surveymeter Standar



Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di Luar Ruangan



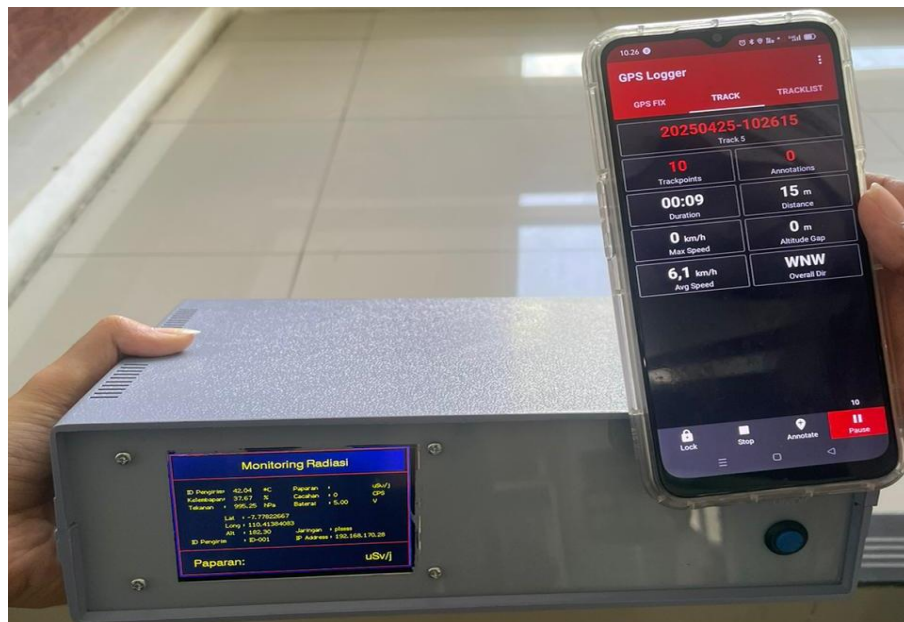
Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di ruangan bersuhu dingin (berAC)



Pembuktian Kalibrasi Sensor BME-280 dan Termometer Digital di ruangan bersuhu normal



Pembuktian Kalibrasi Tegangan Baterai dengan *Voltmeter*



Dokumentasi Kalibrasi Posisi

Lampiran 1. 3 Listing Program Arduino IDE 1.8.9

```
#include <Wire.h>

#include <Adafruit_Sensor.h>

#include <Adafruit_BME280.h>

#include <TinyGPS++.h>

#include <WiFi.h>

#include <ESP_Google_Sheet_Client.h>

#include <TFT_eSPI.h> // Library for TFT display


// Define the pins

#define RX_PIN 15

#define TX_PIN 2

#define IN_PIN 4

#define LOG_PERIOD 5000 // 5 detik

const int batteryPin = 34; // Pin ADC untuk membaca tegangan

const float resistor1 = 51000; // Resistor 1 (51k $\Omega$ )

const float resistor2 = 100000; // Resistor 2 (100k $\Omega$ )

const int ADCmax = 4095; // Nilai maksimal ADC (12-bit pada ESP32)

const float Vref = 3.3; // Tegangan referensi (3.3V pada ESP32)

const float VmaxBattery = 5.0; // Tegangan maksimum baterai

const float VminBattery = 0; // Tegangan minimum baterai

float Vout; // Tegangan output dari pembagi tegangan

float Vin; // Tegangan input sebenarnya dari baterai

float batteryStatus; // Status baterai dalam persen

float batteryVoltage; // Status baterai dalam voltage hasil bacaan pembagi tegangan
```

```

volatile unsigned long counts = 0;

unsigned long previousCounts = 0;

unsigned long previousMillis = 0;

unsigned long cpm = 0;


// Initialize BME280, GPS, and TFT

Adafruit_BME280 bme; // BME280 sensor instance

TinyGPSPlus gps;    // GPS instance

TFT_eSPI tft = TFT_eSPI(); // TFT instance


//-----WIFI TERTAUT-----
-----

#define WIFI_SSID "Tenda_FA68F0"

#define WIFI_PASSWORD "bismillah"


//-----Google Project ID YANG
DIKIRIM-----

// Google Project ID

#define PROJECT_ID "drn-datalogging"


// Service Account's client email

#define CLIENT_EMAIL "drn-datalogging@drn-
datalogging.iam.gserviceaccount.com"


// Service Account's private key

const char PRIVATE_KEY[] PROGMEM = "-----BEGIN PRIVATE KEY-----
\nMIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAKgwggSkAgEAAoIBAQCmH
ySkG6B5JR19\nhYqcAxWUxyWEs+5/R7it31IjvB3s/pIEDgMfwJIV8pU0AUKy
Mdvak9ltcY+F3747\nJjcvXS/X3tJXJfHeveSAE8r8AcJqbGqU/5PAvUjecav70ivJ

```

```
w7f0png6SrfzwmNL\n7PlvE2vVRRkuiLzVi4YBdy6Jt5ItNoxBaoicZ+7qSlbi/Dg
WASgneEkStwlcY+0\nhMF1W80iFd67oLiRp/L+o4zDy8ZOWQPDdBy4EFFFu
EGUgocafqLIwSpqdhN2oxwP\nYq30BaCtPBCTNPXCnqBvlycLORfJ98yHdVvs
YjG6g060oj18sLJ/CGlQs+60sGIM\nACWM7n9RAgMBAAECggEAB9VDxw
pu6KNJyFE6Ew3S+LemSVx71yom5VQJNeKkpKw\nAu5qFhlMdNKOm09wsp
quzwTi6itdmWqM/ktw0SRp3IczXGWc/p8NvZPq1J135UuI\nIDxwvowJSTk0Q
GKIeFBQD1Urj0ltY9zYuU30fqontzoovtE2hmzDnY5clvRO6f+c\nnoc+3+0AfyhA
n7T5f0Z9SQi0rjYoR0mi3Z0CVXbLbCRcOegm2eI35jPgrkYHThjWn\nL4DMP
omQ0JNDef6rduCPWlu6ar45jmsCGcz1UO3ID7xj6/ZWVr6HtcfkHUBNpcCl\nQ
wdJmmsW+f1PIRTuQBRUPfEW9NB6u/4rcPPvBsIuMQKBgQDTMqMul8ODg
0btCxaW\njum+h7LDL6Cez0ri394Ic9NB1Z2aAP5mwYQb7QSeYtizxjOoVkQM
/8XNWGACfm9g\nVKywAk/xTuiqJX0Bypb/Qk8xWQb2WWHd0dO3oTSy1bF
FFd0BoBuPStF0gPQZG4rA\n3OYjqI+w/k4qvLpkGoZlYjaG/QKBgQDJXJaxOk
1draVxEuDLpd/Cy/0UIDjHjvFh\n2zZJnuhlUbPI8B9m+oST6pMk6Bjo3YzeyUZ
ZLWGL7skqJZgke5/AQnfRMh5JKCIw\nnzvByASLxooax22sNEJ+ZDkAnvQAg
D6T+5my/FnbBxvkA20fCWtwVseaAuBFcNshG\nYn1GsWxr5QKBgQCzyL5q0
lXoHp+I+FBxOBU40TFWwLt9IT7NZP8QuPCUcXDpAYGP\nhZsFXr35IvpYB
xhAa7cgTbend9y7IsBTA6HBbyZ4Z9DdzM3c0YW6o8rgWkX3/dy2\nnxIDzJ7LBr
ffQuXLY6F+tmmy1Q/UHTo/OX/01S6+OXt1Rujj9MCsx2ZmTTQKBgANl\nnJU
FNu1o2AFecwV+L5obxS44YUPzEYzuEJ7EsFb0HmKFtswnYdX3M+bgolarve
V2U\nnRTRhuIBUgV3NjngnAXWLaVvEnZAcNh9ejd1OOZmbZOWAyBhLTHW
XMOu6mxLvUJff\nnqLu8FDqGMVKbX+VeBNH00u8eq2RDpDLTnEsh2LMlAo
GBAI9u0QBOJ8Ci04kwRpA5\nnm7RIhCDt2A4BLiZHc1VEzVt/RA6sH/FF7Em
61cQGrbTrZMfZctOm4XcoVNxsqps\nNV28P+/wsmmQtP6lQbVDHauGJA2o
U67uUzv+BAIRC3aaXo1YLiLCOX4bINOBMd7d\nnQWkpav6dYh/z0d/fkxiGkQ
JJ\n-----END PRIVATE KEY-----\n";
```

```
// The ID of the spreadsheet where you'll publish the data
```

```
const char spreadsheetId[] = "1uI5l-
WAYxqWnJpDkU0lwpWqj06bXhWVeX84FofVosVA";
```

```
//-----VARIABEL YANG DICARI-----
-----
```

```
unsigned long lastTime = 0;
```

```
unsigned long timerDelay = 3000;
```

```
// Sensor variables
```

```

float temperature, humidity, pressure;

double latitude, longitude;

float altitude;

int frequency = 0;

float radiationExposure;


// Sender ID declaration

const char* deviceId = "ID-001";

unsigned long epochTime = time(nullptr);


void ISR_impulse() {
    counts++;
}


//-----NTP UNTUK WAKTU-----
-----

const char* ntpServer = "pool.ntp.org";


// Function that gets current epoch time

unsigned long getTime() {
    time_t now;
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {
        return 0;
    }
    time(&now);
    return now;
}

```

```

//-----WAKTU +7-----
-----

String getFormattedTime() {
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {
        return "Failed to obtain time";
    }

    // Add 7 hours to the current time
    timeinfo.tm_hour += 7;

    // Handle overflow if hours go over 23
    if (timeinfo.tm_hour >= 24) {
        timeinfo.tm_hour -= 24;
        timeinfo.tm_mday += 1; // Move to the next day

        // Handle day overflow by adjusting the month and year if needed
        if (timeinfo.tm_mday > 31 && (timeinfo.tm_mon == 0 || timeinfo.tm_mon
== 2 || timeinfo.tm_mon == 4 || timeinfo.tm_mon == 6 || timeinfo.tm_mon == 7 ||
timeinfo.tm_mon == 9 || timeinfo.tm_mon == 11)) {
            timeinfo.tm_mday = 1;
            timeinfo.tm_mon += 1;

        } else if (timeinfo.tm_mday > 30 && (timeinfo.tm_mon == 3 ||
timeinfo.tm_mon == 5 || timeinfo.tm_mon == 8 || timeinfo.tm_mon == 10)) {
            timeinfo.tm_mday = 1;
            timeinfo.tm_mon += 1;

        } else if (timeinfo.tm_mon == 1) { // February
            // Leap year check

```

```

        bool isLeapYear = (timeinfo.tm_year + 1900) % 4 == 0 &&
        ((timeinfo.tm_year + 1900) % 100 != 0 || (timeinfo.tm_year + 1900) % 400 == 0);

        int daysInFeb = isLeapYear ? 29 : 28;

        if (timeinfo.tm_mday > daysInFeb) {

            timeinfo.tm_mday = 1;

            timeinfo.tm_mon += 1;

        }
    }
}

```

```

// Handle year overflow

if (timeinfo.tm_mon > 11) {

    timeinfo.tm_mon = 0;

    timeinfo.tm_year += 1;

}

}

```

```

char timeStringBuff[25]; // Buffer to hold the time string

strftime(timeStringBuff, sizeof(timeStringBuff), "%Y-%m-%d %H:%M:%S",
&timeinfo); // Format: YYYY-MM-DD HH:MM:SS

return String(timeStringBuff);

}

```

```

//=====
=====
=====
=====

```

```

void setup()

{

    // Initialize serial communication for debugging

    Serial.begin(115200);

```

//-----POSISI TFT-----

tft.begin();
tft.setRotation(3); // Landscape
tft.fillScreen(TFT_BLACK);

//-----GARIS FRAME-----

tft.drawRect(0, 0, 476, 319, TFT_BLUE);
tft.drawRect(3, 3, 470, 313, TFT_RED);
tft.drawLine(3, 65, 470, 65, TFT_RED); // Top center line
tft.fillRect(8, 8, 458, 53, TFT_NAVY); // Blue background for the title
tft.setCursor(140, 25, 4); // Title position
tft.setTextColor(TFT_YELLOW, TFT_NAVY);
tft.println("Monitoring Radiasi");

//-----GARIS MERAH-----

tft.drawLine(3, 250, 470, 250, TFT_RED);
tft.fillRect(8, 258, 458, 53, TFT_BLACK);

// Main frame for sensor data
tft.fillRect(8, 70, 458, 180, TFT_BLACK);

//-----Display TAMPILAN TFT-----

tft.setCursor(30, 90, 2);

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Suhu ");  
tft.setCursor(100, 90, 2);  
tft.print(": ");  
tft.setCursor(180, 90, 2);  
tft.println("*C");
```

```
tft.setCursor(30, 110, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Kelembapan ");  
tft.setCursor(100, 110, 2);  
tft.print(": ");  
tft.setCursor(180, 110, 2);  
tft.println("%");
```

```
tft.setCursor(30, 130, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Tekanan ");  
tft.setCursor(100, 130, 2);  
tft.print(": ");  
tft.setCursor(180, 130, 2);  
tft.println("hPa");
```

```
tft.setCursor(100, 160, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Lat ");  
tft.setCursor(135, 160, 2);  
tft.print(": ");
```

```
tft.setCursor(100, 180, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Long ");  
tft.setCursor(135, 180, 2);  
tft.print(": ");
```

```
tft.setCursor(100, 200, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Alt ");  
tft.setCursor(135, 200, 2);  
tft.print(": ");
```

```
tft.setCursor(240, 200, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Jaringan");  
tft.setCursor(310, 200, 2);  
tft.print(": ");
```

```
tft.setCursor(240, 220, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("IP Address");  
tft.setCursor(310, 220, 2);  
tft.print(": ");
```

```
tft.setCursor(240, 90, 2);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.print("Paparan ");
```

```

tft.setCursor(310, 90, 2);
tft.print(": ");
tft.setCursor(400, 90, 2);
tft.println("uSv/j");

tft.setCursor(30, 220, 2);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.print("ID Pengirim ");
tft.setCursor(135, 220, 2);
tft.print(": ");

tft.setCursor(240, 110, 2);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.print("Cacahan ");
tft.setCursor(310, 110, 2);
tft.println(": ");
tft.setCursor(400, 110, 2);
tft.println("CPM");

tft.setCursor(240, 130, 2);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.print("Baterai ");
tft.setCursor(310, 130, 2);
tft.print(": ");
tft.setCursor(400, 130, 2);
tft.println("V");

tft.setCursor(30, 275, 4);

```

```

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.println("Paparan:");
tft.setCursor(275, 275, 4);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(380, 275, 4);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.println("uSv/j");

//-----

// Initialize BME280
if (!bme.begin(0x76)) { // Change the I2C address if needed
  tft.setCursor(50, 50, 2);
  tft.setTextColor(TFT_RED, TFT_BLACK);
  tft.println("BME280 not found");
  while (1); // Stop if BME280 is not detected
}

//-----

// Initialize GPS
Serial2.begin(9600, SERIAL_8N1, RX_PIN, TX_PIN);

//-----

// Initialize frequency measurement pin
//pinMode(inputPin, INPUT_PULLUP); // Set pin D4 sebagai input dengan pull-
up internal
//pinMode(IN_PIN, INPUT);

Serial.println("Booting...");

Serial.println("Measuring CPM setiap 5 detik...");

```

```

pinMode(IN_PIN, INPUT);

attachInterrupt(digitalPinToInterrupt(IN_PIN), ISR_impulse, FALLING);

//-----

// Initialize Wi-Fi
WiFi.setAutoReconnect(true);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(1000);
}

Serial.println();
Serial.println("Connected!");
Serial.print("IP address: ");
Serial.println(WiFi.localIP());
delay(1000);

Serial.println();
Serial.print("Connected to SSID: ");
Serial.print("Jaringan: ");
Serial.println(WiFi.SSID());
delay(1000);

// Initialize Google Sheets
GSheet.begin(CLIENT_EMAIL, PROJECT_ID, PRIVATE_KEY);
}

```

```

//=====
=====

void loop()
{
    //----- Read BME280 data-----
    -----

    temperature = bme.readTemperature();
    humidity = bme.readHumidity();
    pressure = bme.readPressure() / 100.0F; // Convert to hPa

    //-----Read GPS data if available-----
    -----

    while (Serial2.available() > 0) {
        gps.encode(Serial2.read());
    }

    if (gps.location.isUpdated()) {
        latitude = gps.location.lat();
        longitude = gps.location.lng();
        altitude = gps.altitude.meters();
    }

    //-----FREKUENSI COUNTER-----
    -----

    unsigned long currentMillis = millis();

    if (currentMillis - previousMillis >= LOG_PERIOD) {
        previousMillis = currentMillis;
    }
}

```

```

    unsigned long countsThisPeriod = counts - previousCounts;
    previousCounts = counts;

    cpm = countsThisPeriod * (60000 / LOG_PERIOD); // konversi ke CPM
    frequency = cpm;
    Serial.print("Cacah (5 detik terakhir): ");
    Serial.println(countsThisPeriod);
    Serial.print("CPM: ");
    Serial.println(cpm);
}

//-----MENGHITUNG BATTERY
//VOLTAGE-----

// Membaca nilai analog dari pin ADC
int analogValue = analogRead(batteryPin);
float Vout = analogValue * (Vref / ADCmax); // Menghitung Vout dari ADC

// Menghitung tegangan input sebenarnya (Vin) dari pembagi tegangan
float Vin = Vout * ((resistor1 + resistor2) / resistor2);
Vin += 0.02; //agar menjadi 5.00 V
batteryVoltage = round(Vin * 100.0) / 100.0;

// Membulatkan persen ke 1 desimal
batteryStatus = round(batteryStatus * 10.0) / 10.0;

// Menghitung persentase status baterai
batteryStatus = ((Vin - VminBattery) / (VmaxBattery - VminBattery)) * 100.0;

```

```

// Membatasi nilai status baterai antara 0% hingga 100%
if (batteryStatus > 100.0) {
    batteryStatus = 100.0;
} else if (batteryStatus < 0.0) {
    batteryStatus = 0.0;
}

//-----RADIASI SERIAL RANDOM-----
-----

// Hitung radiasi berdasarkan frekuensi (CPS ke  $\mu\text{Sv/j}$ )
// Generate random radiation exposure
float radiationExposure = frequency / 151.0 * 1; // 151 cpm = 1 usv /h

//-----TAMPILAN TFT HASIL-----
-----

// Update temperature value
tft.setCursor(120, 90, 2);
tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.print(temperature); // Print new temperature value

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(120, 110, 2);
tft.println(humidity);

tft.setTextColor(TFT_YELLOW, TFT_BLACK);
tft.setCursor(120, 130, 2);
tft.println(pressure);

```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(145, 160, 2);  
tft.println(latitude, 8);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(145, 180, 2);  
tft.println(longitude, 8);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(145, 200, 2);  
tft.println(altitude);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(320, 90, 2);  
tft.println(radiationExposure);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(320, 110, 2);  
tft.println(frequency);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(320, 130, 2);  
tft.println(batteryVoltage);
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(320, 220, 2);  
tft.println(WiFi.localIP());
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(320, 200, 2);  
tft.println(WiFi.SSID());
```

```
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.setCursor(145, 220, 2);  
tft.println(deviceId);
```

```
tft.setCursor(200, 260, 6);  
tft.setTextColor(TFT_YELLOW, TFT_BLACK);  
tft.println(radiationExposure);
```

```
//-----SERIAL PRINT MONITOR-----  
-----
```

```
Serial.println("ID pengirim: ");  
Serial.println(deviceId);
```

```
Serial.println("Radiasi: ");  
Serial.println(radiationExposure);  
Serial.println("uSv/j");
```

```
Serial.print("Suhu: ");  
Serial.print(bme.readTemperature());  
Serial.println(" *C");
```

```
Serial.print("Kelembapan: ");
```

```
Serial.print(bme.readHumidity());
```

```
Serial.println(" %");
```

```
Serial.print("Tekanan: ");
```

```
Serial.print(bme.readPressure() / 100.0F);
```

```
Serial.println(" hPa");
```

```
Serial.print("Cacahan: ");
```

```
Serial.print(frequency);
```

```
Serial.println(" ");
```

```
Serial.print("Latitude: ");
```

```
Serial.println(gps.location.lat(), 8);
```

```
Serial.print("Longitude: ");
```

```
Serial.println(gps.location.lng(), 8);
```

```
Serial.print("Altitude: ");
```

```
Serial.println(gps.altitude.meters());
```

```
Serial.print("Timestamp: ");
```

```
Serial.println(getFormattedTime());
```

```
Serial.print(Vout, 3);
```

```
Serial.print("V | Vin (Baterai): ");
```

```
Serial.print("Tegangan Baterai: ");
```

```
Serial.print(batteryVoltage);
```

```
Serial.print("V, Status Baterai: ");
```

```

Serial.print(batteryStatus);

Serial.println("%");


Serial.println();

//-----KIRIM DATA-----
-----

// Upload data to Google Sheets every 5 seconds
if (millis() - lastTime > timerDelay) {
    lastTime = millis();

    FirebaseJson response;
    FirebaseJson valueRange;

    // Get current epoch time and adjust for timezone (+7 hours)
    unsigned long epochTime = time(nullptr);
    unsigned long adjustedTime = epochTime + 7 * 3600;

    // Format adjusted time as a string
    struct tm *ptm = gmtime((time_t *)&adjustedTime);
    char timeString[30];
    strftime(timeString, 30, "%Y-%m-%d %H:%M:%S", ptm);

    // Prepare the data to upload to Google Sheets
    valueRange.set("values/[0]/[1]", String(deviceId)); // Fixed line
    valueRange.set("values/[0]/[0]", String(getFormattedTime())); // Add
formatted time
    valueRange.set("values/[0]/[2]", String(temperature));
    valueRange.set("values/[0]/[3]", String(humidity));

```

```

    valueRange.set("values/[0]/[4]", String(pressure));
    valueRange.set("values/[0]/[6]", double(latitude));
    valueRange.set("values/[0]/[7]", double(longitude));
    valueRange.set("values/[0]/[8]", String(altitude));
    valueRange.set("values/[0]/[5]", String(frequency));
    valueRange.set("values/[0]/[10]", String(batteryVoltage));
    valueRange.set("values/[0]/[11]", String(batteryStatus));

    valueRange.set("values/[0]/[9]", String(radiationExposure)); // Update with
actual radiation exposure value

    // Append values to the spreadsheet

    bool success = GSheet.values.append(&response, spreadsheetId, "Sheet1!A1",
&valueRange);
    if (success) {
        response.toString(Serial, true);
        valueRange.clear();
    } else {
        Serial.println(GSheet.errorReason());
    }
}

delay(5000); // Delay for 5 seconds before next update
}

```



KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
Jalan Gajayana Nomor 50, Telepon (0341) 551354, Fax. (0341) 572533
Website: <http://www.uin-malang.ac.id> Email: info@uin-malang.ac.id

JURNAL BIMBINGAN SKRIPSI/TESIS/DISERTASI

IDENTITAS MAHASISWA

NIM : 210604110069
Nama : RATNA RAHAYU
Fakultas : SAINS DAN TEKNOLOGI
Jurusan : FISIKA
Dosen Pembimbing 1 : FARID SAMSU HANANTO, S.Si., M.T.
Dosen Pembimbing 2 : MUBASYIROH, S.S., M.Pd.I
Judul Skripsi/Tesis/Disertasi : RANCANG BANGUN PERANGKAT MONITOR RADIASI GAMMA BERBASIS MIKROKONTROLER ESP32 MENGGUNAKAN DETEKTOR GEIGER-MÜLLER UNTUK PEMANTAUAN RADIASI DAN LINGKUNGAN TERPADU

IDENTITAS BIMBINGAN

No	Tanggal Bimbingan	Nama Pembimbing	Deskripsi Proses Bimbingan	Tahun Akademik	Status
	12 Maret 2025	FARID SAMSU HANANTO, S.Si., M.T.	BAB I PENDAHULUAN	Genap 2024/2025	Sudah Dikoreksi
	09 April 2025	FARID SAMSU HANANTO, S.Si., M.T.	BAB II TINJAUAN PUSTAKA	Ganjil 2024/2025	Sudah Dikoreksi
	23 April 2025	FARID SAMSU HANANTO, S.Si., M.T.	BAB III METODE PENELITIAN	Genap 2024/2025	Sudah Dikoreksi
	14 Mei 2025	FARID SAMSU HANANTO, S.Si., M.T.	BIMBINGAN DATA HASIL PENELITIAN	Genap 2024/2025	Sudah Dikoreksi
	20 Mei 2025	MUBASYIROH, S.S., M.Pd.I	BIMBINGAN REVISI BAB I DAN KONSULTASI INTEGRASI BAB IV	Genap 2024/2025	Sudah Dikoreksi
	23 Mei 2025	MUBASYIROH, S.S., M.Pd.I	Perbaikan Revisi BAB IV Keterkaitan Hasil penelitian dalam perspektif islam	Genap 2024/2025	Sudah Dikoreksi
	25 Mei 2025	MUBASYIROH, S.S., M.Pd.I	Perbaikan Revisi BAB IV hasil penelitian dan keterkaitan perspektif islam	Genap 2024/2025	Sudah Dikoreksi
	26 Mei 2025	FARID SAMSU HANANTO, S.Si., M.T.	KONSULTASI BAB IV	Genap 2024/2025	Sudah Dikoreksi

Telah disetujui
Untuk mengajukan ujian Skripsi/Tesis/Desertasi

Malang, 30 JUNI 2025
Dosen Pembimbing 1

Dosen Pembimbing 2

MUBASYIROH, S.S., M.Pd.I

FARID SAMSU HANANTO, S.Si., M.T.

