

BAB II

DASAR TEORI

2.1 Tinjauan Pustaka

Dalam penyusunan penelitian ini, penulis melakukan kajian literatur dari berbagai referensi yang relevan. Ringkasan hasil kajian tersebut disajikan pada Tabel 2.1. Penelitian-penelitian terdahulu yang dianalisis dijadikan sebagai dasar acuan sekaligus sumber inspirasi dalam pengembangan sistem yang dirancang.

Tabel 2.1 Tinjauan Pustaka

No.	Penulis	Judul	Keterangan	Perbaikan
1.	Puji Astuti	Rancang Bangun Alat Monitor Aktivitas Senam dengan Sensor Getar ADXL335 dan Sensor Arah GY-273 berbasis Aplikasi Android untuk Penunjang Evaluasi Pembelajaran Olahraga	Penelitian ini mengembangkan sebuah sistem pemantauan aktivitas senam berbasis mikrokontroler ESP32 yang memanfaatkan sensor ADXL335 dan GY-273 untuk mendeteksi getaran serta arah pergerakan. Data yang diperoleh dari sensor kemudian dikirimkan ke aplikasi Android, yang berfungsi menampilkan hasil pemantauan dalam	Sistem tersebut masih memiliki keterbatasan karena belum dilengkapi dengan fitur evaluasi yang mampu menghasilkan nilai akhir dari aktivitas senam secara otomatis.

No.	Penulis	Judul	Keterangan	Perbaikan
			bentuk grafik secara <i>real-time</i> [10]	
2.	Ari Bawono Putranto, Naurah Elfrida Anargya, Ahmad Ridlo Hanifudin Tahier, dan Fakhruddin Mangkusasmito	Design and Development of an Automated Gymnastics Movement Evaluation Device	Penelitian ini mengembangkan sistem penilaian gerakan senam otomatis menggunakan <i>accelerometer</i> ADXL335 dan magnetometer HMC5883L yang terintegrasi dengan ESP32-C3 Mini sebagai <i>transmitter</i> dan Wemos D1 Mini ESP32 sebagai <i>receiver</i> . Data diproses menggunakan <i>downsampling</i> , <i>average pooling</i> , normalisasi Min-Max, dan dibandingkan dengan data master untuk menghasilkan nilai gerakan. Pengujian pada pemanasan SKJ 2012 menunjukkan	Penelitian masih menggunakan ADXL335 dan HMC5883L secara terpisah sehingga data gerakan terbatas pada percepatan dan arah orientasi. Pengujian juga masih difokuskan pada pemanasan SKJ 2012 dengan penilaian berbasis <i>error</i> , <i>accuracy</i> , dan pembobotan parameter. Pengembangan dapat dilakukan dengan menggunakan sensor inersia yang mampu membaca percepatan dan kecepatan sudut

No.	Penulis	Judul	Keterangan	Perbaikan
			bahwa gerakan yang sesuai dengan referensi memperoleh skor lebih tinggi dibandingkan gerakan yang tidak sesuai. [14]	secara bersamaan serta memperluas pengujian pada rangkaian gerakan senam yang lebih lengkap.
3.	Reza Arinda Pratama, Dahnia Syauqy, dan Agung Setia Budi	Sistem Monitoring Gerakan <i>Sit-up</i> berbasis Data Sensor <i>Accelerometer</i> dan <i>Gyroscope</i> menggunakan Algoritma K-Nearest Neighbor pada NodeMCU ESP8266	Penelitian ini mengembangkan sistem pemantauan gerakan sit-up yang bertujuan untuk mengurangi subjektivitas penilaian sekaligus meminimalkan risiko cedera. Sistem tersebut memanfaatkan sensor MPU6050, yang merupakan kombinasi antara <i>accelerometer</i> dan <i>gyroscope</i> , untuk mendeteksi tahapan gerakan sit-up secara real-time. Data yang diperoleh kemudian diolah menggunakan	Perancangan perangkat keras masih dapat dioptimalkan agar menghasilkan alat yang lebih ringan dan ringkas, sehingga meningkatkan kenyamanan saat digunakan. Di sisi lain, ketergantungan aplikasi pada koneksi internet serta <i>server</i> Blynk yang rentan terhadap gangguan jaringan menjadi salah satu

No.	Penulis	Judul	Keterangan	Perbaikan
			algoritma K-Nearest Neighbor (KNN) untuk mengklasifikasikan gerakan serta menghitung jumlah repetisi secara otomatis [11].	keterbatasan, sehingga pengembangan aplikasi yang mampu beroperasi secara <i>offline</i> atau memanfaatkan basis data lokal perlu dipertimbangkan sebagai solusi alternatif.

Berdasarkan Tabel 2.1, penelitian terdahulu menunjukkan bahwa sensor *accelerometer* dan *gyroscope* telah banyak digunakan untuk mendeteksi dan memantau aktivitas gerakan tubuh. Beberapa penelitian juga telah menerapkan mikrokontroler ESP32 sebagai perangkat pemrosesan dan komunikasi data. Namun, masih terdapat keterbatasan pada jenis sensor, metode pengolahan data, cakupan gerakan, maupun sistem penilaian yang digunakan. Oleh karena itu, penelitian ini dikembangkan dengan menggunakan sensor MPU-6050 yang mampu membaca *accelerometer* dan *gyroscope* secara bersamaan, serta menerapkan pengolahan data untuk membandingkan pola gerakan peserta terhadap data master hingga menghasilkan skor akhir.

2.2 Senam Irama

Senam irama merupakan salah satu bentuk aktivitas jasmani yang dilakukan dengan memadukan gerakan tubuh dan irama musik, sehingga menghasilkan gerakan yang selaras, teratur, dan berkesinambungan. Dalam pembelajaran pendidikan jasmani, senam irama tidak hanya berfungsi sebagai sarana kebugaran, tetapi juga untuk mengembangkan keluwesan, koordinasi, keseimbangan, serta kemampuan mengikuti tempo gerak. Secara umum, prinsip utama senam irama

meliputi kesesuaian antara gerakan dan irama, fleksibilitas tubuh, serta kontinuitas gerakan yang dilakukan secara runtut dan tidak terputus [15], [16].

Pelaksanaan senam irama pada umumnya dibagi menjadi tiga tahap, yaitu pemanasan, gerakan inti, dan pendinginan. Tahap pemanasan bertujuan untuk mempersiapkan otot, sendi, serta sistem pernapasan agar tubuh siap menerima aktivitas dengan intensitas yang lebih tinggi. Selanjutnya, pada tahap inti dilakukan rangkaian gerakan utama sesuai jenis senam yang digunakan. Tahap terakhir, yaitu pendinginan, berfungsi untuk membantu tubuh kembali ke kondisi normal serta mengurangi risiko cedera setelah melakukan aktivitas fisik. Pembagian tahapan ini umum digunakan dalam berbagai jenis senam karena dapat meningkatkan efektivitas latihan secara bertahap [17].

Berdasarkan intensitas gerakannya, senam dapat dibedakan menjadi *low impact* dan *high impact*. Gerakan *low impact* memiliki intensitas yang relatif ringan tanpa lompatan atau hentakan yang kuat, sehingga beban terhadap sendi lebih kecil. Sebaliknya, gerakan *high impact* memiliki tempo yang lebih cepat serta melibatkan gerakan yang lebih dinamis, seperti lompatan dan hentakan, sehingga membutuhkan kontrol tubuh yang lebih baik [17].

Dalam penelitian ini, jenis senam yang digunakan sebagai objek pengujian adalah Senam Kesegaran Jasmani (SKJ) 2021. Rangkaian gerakan SKJ 2021 terdiri atas tahapan pemanasan, gerakan inti, dan pendinginan yang dilakukan mengikuti pola gerakan serta irama musik yang telah ditentukan. Karakteristik gerakan yang berulang dan mengikuti irama tersebut memungkinkan data gerakan peserta direkam dan dibandingkan dengan data gerakan master yang digunakan sebagai referensi.

Pengukuran dalam penelitian ini difokuskan pada pola gerakan yang dapat terbaca dari satu titik sensor di area panggul. Parameter yang diamati meliputi perubahan percepatan, intensitas langkah dan hentakan, serta perubahan kecepatan sudut ketika peserta melakukan rotasi atau perubahan gerakan. Pendekatan tersebut disesuaikan dengan kemampuan *accelerometer* dan *gyroscope* pada sensor MPU-6050 serta batasan posisi sensor yang digunakan dalam penelitian.

2.3 ESP 32

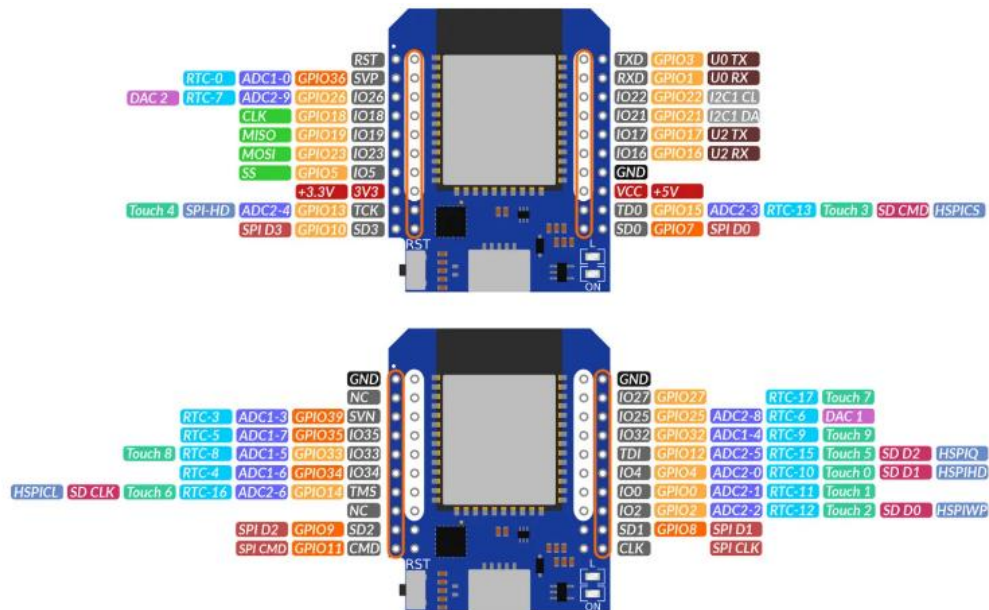
ESP32 merupakan mikrokontroler berbasis *System-on-Chip* (SoC) yang dikembangkan oleh Espressif Systems dan dirancang untuk mendukung berbagai aplikasi *Internet of Things* (IoT). Perangkat ini telah terintegrasi dengan fitur komunikasi nirkabel berupa Wi-Fi dan Bluetooth dalam satu chip, sehingga mampu melakukan pertukaran data secara efisien tanpa memerlukan modul tambahan [18].

Secara umum, ESP32 memiliki kemampuan pemrosesan yang cukup tinggi dengan dukungan prosesor, memori internal, serta berbagai periferal seperti ADC, DAC, UART, SPI, dan I2C. Selain itu, ESP32 juga mendukung mode hemat daya (*low power mode*) yang membuatnya cocok digunakan pada sistem berbasis sensor maupun perangkat portabel.

ESP32 memiliki fleksibilitas tinggi dalam penggunaannya karena tersedia dalam berbagai varian modul dengan karakteristik yang berbeda-beda. Beberapa jenis ESP32 yang umum digunakan di antaranya adalah ESP32 D1 mini. Setiap varian tersebut memiliki keunggulan tersendiri, baik dari segi arsitektur prosesor, konsumsi daya, maupun fitur komunikasi yang dimiliki.

2.3.2 ESP32 D1 Mini

ESP32 D1 Mini merupakan papan pengembangan berbasis mikrokontroler ESP32 yang memiliki kemampuan pemrosesan serta komunikasi nirkabel Wi-Fi. Dukungan Wi-Fi tersebut memungkinkan perangkat digunakan dalam mode *station* maupun *access point*, sehingga dapat diterapkan pada sistem komunikasi antarperangkat dalam jaringan lokal [19]. Selain komunikasi nirkabel, ESP32 juga menyediakan berbagai antarmuka seperti GPIO, ADC, DAC, UART, SPI, dan I2C yang dapat digunakan untuk menghubungkan sensor maupun perangkat pendukung lainnya.



Gambar 2.1 ESP32 D1 Mini

Gambar 2.1 menunjukkan konfigurasi pin pada *board* Wemos D1 Mini ESP32. *Board* ini menyediakan sejumlah pin *General Purpose Input Output* (GPIO) yang dapat difungsikan sebagai masukan maupun keluaran digital serta mendukung fitur *external interrupt*. Selain itu, beberapa pin pada *board* ini juga dapat digunakan untuk fungsi tambahan, seperti PWM, ADC, DAC, I2C, SPI, UART, dan sensor sentuh kapasitif. Pada *board* Wemos D1 Mini ESP32, terdapat 26 GPIO ESP32 yang diekspos melalui *header board* sehingga dapat dihubungkan dengan sensor, *display*, *push button*, maupun komponen pendukung lainnya [20].

Fitur Analog to Digital Converter (ADC) pada ESP32 dibagi menjadi dua kelompok, yaitu ADC1 dan ADC2. Kanal ADC1 yang terhubung pada GPIO32 hingga GPIO39 masih dapat digunakan meskipun modul Wi-Fi sedang aktif. Sementara itu, ADC2 terletak pada GPIO0, GPIO2, GPIO4, GPIO12 hingga GPIO15, serta GPIO25 sampai GPIO27. Akan tetapi, kanal ADC2 berbagi perangkat keras dengan Wi-Fi, sehingga penggunaannya perlu diperhatikan apabila sistem menggunakan komunikasi Wi-Fi secara bersamaan. ESP32 juga menyediakan dua buah kanal Digital to Analog Converter (DAC) beresolusi 8-bit yang masing-masing berada pada GPIO25 dan GPIO26, sehingga memungkinkan

mikrokontroler menghasilkan keluaran berupa sinyal analog [20].

Pada komunikasi I2C, konfigurasi standar menggunakan GPIO21 sebagai jalur data (SDA) dan GPIO22 sebagai jalur *clock* (SCL). Namun demikian, konfigurasi tersebut masih dapat disesuaikan melalui perangkat lunak sesuai dengan kebutuhan sistem. Untuk komunikasi serial UART, pin GPIO1 dan GPIO3 umumnya digunakan sebagai jalur transmisi dan penerimaan data pada UART0, sedangkan UART2 dapat menggunakan GPIO17 dan GPIO16. Selain itu, komunikasi SPI dapat dilakukan melalui dua bus yang tersedia, yaitu VSPI dan HSPI. Bus VSPI memanfaatkan GPIO23 sebagai MOSI, GPIO19 sebagai MISO, GPIO18 sebagai SCK, dan GPIO5 sebagai pin *Chip Select*. Adapun bus HSPI menggunakan GPIO13 sebagai MOSI, GPIO12 sebagai MISO, GPIO14 sebagai SCK, serta GPIO15 sebagai *Chip Select*.

Beberapa pin pada ESP32 mempunyai fungsi khusus yang berkaitan dengan proses *booting*, *debugging* JTAG, akses memori flash, maupun komunikasi USB. Oleh karena itu, penggunaan pin-pin tersebut perlu diperhatikan agar tidak mengganggu proses kerja sistem secara keseluruhan. Pada penelitian ini, Wemos D1 Mini ESP32 digunakan sebagai pusat pengendali pada modul *receiver* yang bertugas menghubungkan berbagai perangkat pendukung, seperti LCD, *push button*, dan komponen lainnya untuk mendukung proses pengolahan data.

Berdasarkan spesifikasi yang ditunjukkan pada Tabel 2.2, Wemos D1 Mini ESP32 menggunakan prosesor dual-core Xtensa LX6 32-bit yang mampu bekerja pada frekuensi maksimum 240 MHz. Dari sisi memori, mikrokontroler ini dibekali SRAM sebesar 520 KB, ROM sebesar 448 KB, serta *flash memory* berkapasitas 4 MB yang digunakan untuk penyimpanan program dan data. Selain kemampuan pemrosesan yang cukup tinggi, ESP32 juga telah dilengkapi dengan teknologi komunikasi nirkabel berupa Wi-Fi IEEE 802.11 b/g/n dan Bluetooth versi 4.2 BR/EDR. Dukungan fitur tersebut memungkinkan ESP32 untuk diimplementasikan pada berbagai aplikasi *Internet of Things* yang memerlukan komunikasi data secara nirkabel.

Tabel 2.2 Spesifikasi ESP32 D1 [21]

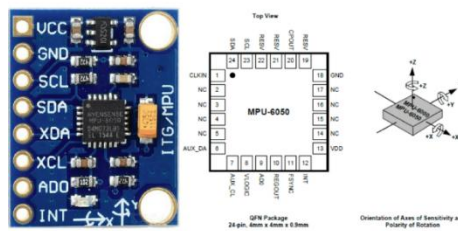
Subjek	Spesifikasi
Processor	Xtensa Dual-Core 32-bit LX6
Memori	448KB ROM, 520KB SRAM, 4MB Flash
Clock Speed	240 MHz
Protokol Nirkabel	Wi-Fi 802.11b/g/n dan Bluetooth V4.2 BR/EDR
Tegangan Operasi	3.0 ~ 3.6 V

Berdasarkan Tabel 2.2, ESP32 D1 Mini memiliki prosesor dual-core 32-bit dengan frekuensi kerja hingga 240 MHz serta dilengkapi memori internal dan komunikasi Wi-Fi. Kemampuan tersebut mendukung kebutuhan sistem dalam melakukan pembacaan data sensor, penyimpanan sementara, serta komunikasi data secara nirkabel. Pada penelitian ini, fitur Wi-Fi dimanfaatkan untuk komunikasi antara modul *transmitter* dan *receiver*, sedangkan antarmuka I2C dan UART digunakan untuk komunikasi dengan sensor dan perangkat pendukung lainnya.

2.4 Sensor MPU-6050

Sensor inersia merupakan sensor yang digunakan untuk mengukur perubahan gerak suatu objek berdasarkan percepatan dan kecepatan sudut. Pada penelitian ini digunakan MPU-6050 sebagai *Inertial Measurement Unit* (IMU) 6-axis yang terdiri atas *accelerometer* tiga sumbu X, Y, dan Z serta *gyroscope* tiga sumbu X, Y, dan Z. *Accelerometer* digunakan untuk membaca percepatan yang timbul akibat perubahan gerakan, langkah, dan hentakan, sedangkan *gyroscope* digunakan untuk membaca kecepatan sudut ketika sensor mengalami gerakan rotasi. Kombinasi kedua jenis pengukuran tersebut digunakan karena aktivitas senam tidak hanya menghasilkan gerakan translasi, tetapi juga perubahan rotasi tubuh. MPU-6050 adalah modul IMU (*Inertial Measurement Unit*) yang menggabungkan akselerometer tiga sumbu dan giroskop tiga sumbu dalam satu paket serta menyediakan Digital Motion Processor (DMP) yang dapat melakukan *sensor fusion* dasar di dalam chip. Kemampuan rentang pengukuran akselerometer sampai ± 16 g dan *gyroscope* sampai $\pm 2000^\circ/\text{s}$ serta akses data melalui antarmuka I²C membuat MPU-6050 praktis untuk aplikasi pemantauan gerakan tubuh, pengukuran sudut, dan deteksi rotasi. Dalam konteks penilaian gerakan senam,

MPU-6050 dapat digunakan untuk menangkap parameter seperti percepatan linier dan kecepatan sudut yang kemudian diolah untuk mengenali pola gerakan atau menghitung sudut-sudut sendi secara real-time[22]. Pada penelitian ini, penggunaan *accelerometer* dan *gyroscope* pada MPU-6050 bertujuan untuk memperoleh dua informasi gerakan yang saling melengkapi. *Accelerometer* digunakan untuk mendeteksi perubahan percepatan, intensitas langkah, dan hentakan gerakan, sedangkan *gyroscope* digunakan untuk mendeteksi perubahan kecepatan sudut atau rotasi sensor. Penggunaan kedua elemen pengukuran tersebut diperlukan karena *accelerometer* lebih merepresentasikan perubahan gerak translasi, sedangkan *gyroscope* lebih peka terhadap perubahan rotasi. Dengan demikian, kombinasi keduanya dapat memberikan informasi gerakan yang lebih lengkap dibandingkan apabila hanya menggunakan salah satu jenis pengukuran.



Gambar 2.2 Sensor MPU-6050

Berdasarkan gambar 2.2 di atas, modul MPU-6050 memiliki beberapa pin utama yang mendukung proses komunikasi dan operasional sensor. Pin VCC berfungsi sebagai sumber tegangan yang umumnya menggunakan 3,3V atau 5V (tergantung modul), sedangkan pin GND sebagai ground. Untuk komunikasi data, sensor ini menggunakan protokol I2C melalui pin SDA (data) dan SCL (*clock*), sehingga memudahkan integrasi dengan mikrokontroler. Selain itu, terdapat pin XDA dan XCL yang berfungsi sebagai jalur komunikasi I2C tambahan untuk menghubungkan sensor eksternal lain. Pin AD0 digunakan untuk menentukan alamat I2C sensor sehingga memungkinkan penggunaan lebih dari satu perangkat dalam satu jalur komunikasi. Sensor ini juga dilengkapi dengan pin INT yang berfungsi sebagai *output* interrupt untuk memberikan sinyal ke mikrokontroler ketika data siap dibaca atau terjadi kondisi tertentu. Dengan konfigurasi pin tersebut, MPU-6050 menjadi sensor yang fleksibel dan mudah diimplementasikan dalam berbagai sistem berbasis mikrokontroler.

Tabel 2.3 Spesifikasi MPU 6050 [23]

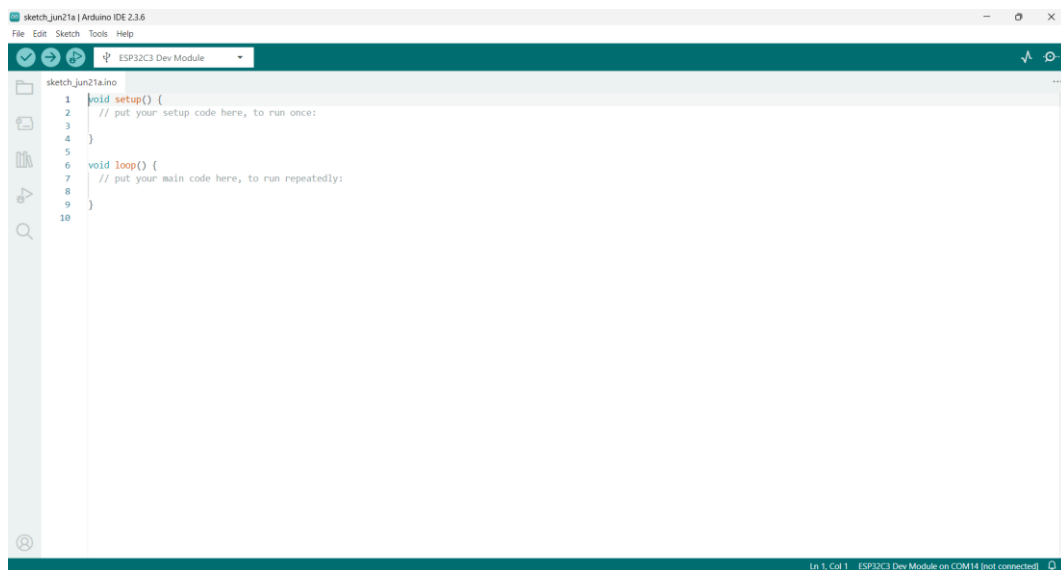
Spesifikasi	Keterangan
Jenis Sensor	6-axis IMU
<i>Accelerometer</i>	3 sumbu X, Y, Z
<i>Gyroscope</i>	3 sumbu X, Y, Z
Rentang <i>Accelerometer</i>	± 2 g, ± 4 g, ± 8 g, ± 16 g
Rentang <i>Gyroscope</i>	± 250 , ± 500 , ± 1000 , ± 2000 °/s
<i>Output Data</i>	Digital
Antarmuka Komunikasi	I2C
Jalur Komunikasi	XDA dan XCL

Berdasarkan Tabel 2.3, Sensor MPU-6050 membaca data fisik berupa percepatan dan kecepatan sudut, kemudian mengubahnya menjadi data digital menggunakan ADC (*Analog-to-Digital Converter*) internal 16-bit. Dengan demikian, keluaran sensor yang dibaca berupa data digital hasil konversi ADC 16-bit yang selanjutnya diproses oleh mikrokontroler ESP32. Pada program yang digunakan dalam penelitian, pembacaan sensor dilakukan menggunakan *library* MPU-6050 sehingga nilai *accelerometer* telah dikonversi ke satuan m/s^2 dan nilai *gyroscope* ke rad/s . Nilai hasil pembacaan tersebut kemudian dikalikan dengan faktor 1000 dan disimpan dalam bentuk bilangan integer untuk mempermudah proses penyimpanan dan pengiriman data. Oleh karena itu, nilai yang digunakan sebagai data mentah pada penelitian merupakan nilai hasil penskalaan pembacaan sensor, bukan nilai ADC mentah secara langsung. Setelah melalui proses normalisasi Min-Max, nilai data berada pada rentang 0 sampai 100 dan tidak lagi dinyatakan dalam satuan fisik. Pada penelitian ini tidak digunakan nilai set point tetap untuk menentukan benar atau salahnya suatu gerakan, karena nilai sensor berubah secara dinamis mengikuti rangkaian gerakan senam. Penilaian dilakukan berdasarkan perbandingan pola data peserta uji terhadap data master. Pada kondisi diam, pembacaan *gyroscope* diharapkan cenderung mendekati nol dan *accelerometer* relatif lebih stabil,

sedangkan pada saat terjadi langkah, hentakan, maupun rotasi, nilai sensor akan mengalami perubahan sesuai karakteristik gerakan.

2.6 Arduino IDE

Arduino IDE merupakan perangkat lunak yang digunakan untuk menulis, menyusun, memeriksa, dan mengunggah program ke papan mikrokontroler Arduino maupun *board* lain yang kompatibel. Sebagai sebuah Integrated Development Environment (IDE), perangkat lunak ini menyediakan lingkungan kerja yang memudahkan pengguna dalam membuat kode program, mengompilasinya menjadi bahasa mesin, serta memindahkannya ke memori mikrokontroler agar dapat dijalankan sesuai fungsi yang dirancang. Melalui Arduino IDE, proses pengembangan sistem berbasis mikrokontroler menjadi lebih sederhana karena seluruh tahapan utama pemrograman tersedia dalam satu aplikasi.[24]



Gambar 2.3 Tampilan Arduino IDE [24]

Pada tampilan Arduino IDE, pengguna dapat menuliskan program yang disebut *sketch*. File *sketch* ini umumnya disimpan dengan ekstensi *.ino*. Selain *editor* kode, Arduino IDE juga menyediakan berbagai fitur pendukung, seperti pemilihan jenis *board*, pengaturan port komunikasi, dan akses ke pustaka (*library*) yang diperlukan sesuai kebutuhan rancangan. Sebelum program dikirim ke *board*, Arduino IDE akan melakukan proses verifikasi atau kompilasi untuk memastikan kode telah ditulis dengan benar. Apabila ditemukan kesalahan sintaks maupun

kesalahan logika dasar pada penulisan program, sistem akan menampilkan pesan *error* sehingga pengguna dapat segera melakukan perbaikan. Fitur ini sangat membantu karena mempercepat proses pengembangan, mengurangi kesalahan pemrograman, dan mempermudah pengujian sistem secara bertahap.

2.7 Baterai Lithium Polymer 103040

Baterai Lithium Polymer 103040 merupakan baterai sekunder yang dapat diisi ulang dan termasuk dalam keluarga baterai berbasis lithium. Baterai berbasis lithium banyak digunakan pada perangkat elektronik portabel karena memiliki kemampuan penyimpanan energi yang baik, tegangan kerja yang relatif stabil, serta densitas energi yang lebih tinggi dibandingkan beberapa jenis baterai isi ulang konvensional. Selain itu, perkembangan teknologi baterai lithium juga mengarah pada penggunaan elektrolit berbasis polimer atau padat untuk meningkatkan fleksibilitas desain dan keamanan baterai [25]. Karakteristik tersebut membuat baterai Lithium Polymer sesuai digunakan pada perangkat portabel yang membutuhkan sumber daya ringkas, ringan, dan mudah ditempatkan pada ruang terbatas.

Pada penelitian ini, baterai Lithium Polymer 103040 digunakan sebagai sumber daya pada modul *transmitter* untuk menyuplai kebutuhan energi mikrokontroler ESP32 D1 Mini dan sensor MPU-6050 selama sistem beroperasi. Baterai ini memiliki tegangan nominal sebesar 3,7 V dengan kapasitas 1200 mAh. Tegangan nominal tersebut sesuai untuk perangkat elektronik berdaya rendah, sedangkan kapasitas 1200 mAh memungkinkan sistem bekerja secara portabel tanpa harus selalu terhubung dengan sumber listrik eksternal. Dengan bentuk fisik yang pipih dan ukuran yang relatif kecil, baterai ini memudahkan proses penempatan komponen pada modul *transmitter* agar tetap ringkas dan nyaman digunakan.



Gambar 2.4 Baterai Lithium Polymer 103040[26]

Tabel 2.4 Baterai Lithium Polymer 103040[27]

Subjek	Spesifikasi
Jenis baterai	Lithium Polymer
Kode ukuran	103040
Tegangan nominal	3,7 V
Kapasitas	1200 mAh
Energi	4,44 Wh
Bentuk fisik	Pipih/prismatik

Berdasarkan Tabel 2.4, baterai Lithium Polymer 103040 memiliki tegangan nominal 3,7 V dengan kapasitas 1200 mAh. Nilai kapasitas tersebut menunjukkan kemampuan baterai dalam menyuplai arus listrik selama periode tertentu sesuai dengan kebutuhan beban pada rangkaian. Energi baterai sebesar 4,44 Wh diperoleh dari hasil perkalian tegangan nominal 3,7 V dengan kapasitas 1,2 Ah. Dalam penggunaannya, baterai lithium perlu dilengkapi dengan sistem pengisian dan proteksi yang baik agar tidak terjadi kondisi pengisian berlebih, pengosongan berlebih, maupun arus berlebih yang dapat menurunkan umur pakai dan keamanan baterai [28]. Oleh karena itu, penggunaan rangkaian proteksi atau modul pengisian baterai menjadi bagian penting dalam perancangan sistem agar baterai dapat bekerja secara aman, stabil, dan sesuai kebutuhan perangkat.

2.8 SMPS

Switch-Mode Power supply (SMPS) merupakan jenis catu daya yang berfungsi untuk mengubah tegangan listrik masukan menjadi tegangan DC yang stabil sesuai kebutuhan perangkat elektronik, dengan efisiensi yang lebih tinggi dibandingkan catu daya linear karena menggunakan teknik pensaklaran (*switching*) pada frekuensi tinggi. Proses kerjanya diawali dengan penyearahan tegangan AC menggunakan rangkaian *bridge diode* sehingga menghasilkan tegangan DC, kemudian dilanjutkan dengan proses penyaringan untuk mengurangi *ripple* dan *noise* agar tegangan menjadi lebih halus. Selanjutnya, SMPS mengatur penyaluran daya ke beban melalui mekanisme *switching* yang mengontrol lebar pulsa tegangan keluaran sesuai kebutuhan sistem, sehingga mampu menjaga kestabilan tegangan meskipun terjadi perubahan pada beban maupun sumber listrik. Dalam proses ini, komponen seperti induktor dan kapasitor berperan sebagai penyimpan dan penstabil energi untuk mempertahankan kualitas keluaran[29]. Dengan karakteristik tersebut, SMPS banyak digunakan pada perangkat elektronik modern, termasuk dalam perancangan alat penilaian gerakan olahraga, karena mampu menyediakan suplai daya yang stabil, efisien, dan mendukung desain sistem yang ringkas serta portabel.



Gambar 2.5 *Power supply* 5V-20A [30]

Tabel 2.5 Spesifikasi SMPS 5V-20A [30]

Subjek	Spesifikasi
Tegangan <i>Input</i>	110 – 220V AC
Tegangan <i>Output</i>	5V DC
Arus <i>Output</i>	20A

Subjek	Spesifikasi
Daya <i>Output</i>	100W
Dimensi	19 x 8 x 3 cm

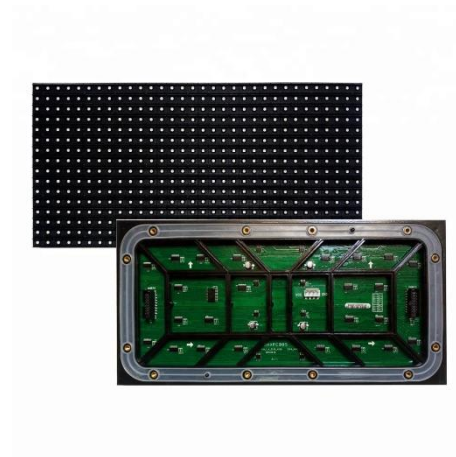
Berdasarkan Tabel 2.5, SMPS ini menerima tegangan *input* dalam rentang 110 hingga 220V AC. Tegangan *output* yang dihasilkan adalah 5V DC dengan arus maksimum mencapai 20A. Selain itu, daya *output* yang dapat dihasilkan berkisar 100W. Pemanfaatan SMPS 5V-20A dalam alat penilaian olahraga senam memiliki tujuan untuk menyuplai tegangan yang dibutuhkan oleh komponen-komponen pada modul panel *display* P10.

2.9 Display LED Dot Matriks P10

Display LED Matrix P10 merupakan panel LED yang memiliki jarak antar piksel (*pixel pitch*) sebesar 10 mm dan digunakan sebagai media keluaran untuk menampilkan informasi hasil penilaian. Pada penelitian ini digunakan panel LED Matrix P10 *single color*, sehingga panel menghasilkan tampilan dalam satu warna sesuai dengan jenis LED yang digunakan. Beberapa panel dapat disusun menjadi satu kesatuan tampilan agar informasi berupa status sistem maupun nilai akhir peserta dapat terlihat dengan lebih jelas.

Komunikasi antara mikrokontroler ESP32 pengendali *display* dengan panel LED Matrix P10 dilakukan menggunakan antarmuka HUB12. Antarmuka tersebut menyediakan jalur data dan jalur kontrol yang digunakan untuk mengatur proses penampilan informasi pada panel. Jalur kontrol digunakan untuk pemilihan baris, pemberian sinyal *clock*, *latch* atau *strobe*, serta *output enable* sesuai dengan konfigurasi *scan* panel yang digunakan. Selain jalur data dan kontrol, ground panel, ESP32, dan sumber catu daya dihubungkan pada referensi ground yang sama agar proses pengiriman sinyal dapat berlangsung dengan baik.

Panel LED memperoleh catu daya 5 V secara langsung dari SMPS melalui terminal catu daya panel. Jalur HUB12 digunakan untuk membawa sinyal data dan kontrol, bukan sebagai jalur utama penyedia daya panel. Dengan demikian, arus yang dibutuhkan panel tidak disuplai melalui ESP32.



Gambar 2.6 *Display* LED Dot Matriks P10 [31]

Pada Gambar 2.6 menunjukkan LED Matriks P10 yang merupakan kumpulan LED tersusun dalam kolom dan baris. Beberapa panel LED dot matriks dapat dirangkai membentuk tampilan yang lebih luas atau layar, dengan konfigurasi serial untuk memperpanjang baris dan paralel untuk menambah kolom.

Tabel 2.6 Spesifikasi *Display* Dot Matriks P10 [32]

Subjek	Spesifikasi
<i>LED Color</i>	Biru
<i>LED Type</i>	SMD
<i>Module Size</i>	320 mm × 160 mm
<i>Resolution Module</i>	32 × 16 dots
<i>Pixel pitch</i>	10 mm
<i>Scan mode</i>	1/4S
<i>Drive Mode</i>	Constant Voltage 5V/ Power supply 5V
<i>HUB Type</i>	Hub 12

Berdasarkan Tabel 2.6, panel LED yang digunakan berwarna biru dengan tipe SMD (*Surface Mounted Device*). Panel memiliki dimensi 320 mm × 160 mm dengan resolusi 32 × 16 titik (*dots*)[33]. Jarak antar titik LED (*pixel pitch*) sebesar 10 mm. Sistem *display* bekerja dengan mode *scan* 1/4S, di mana satu dari empat baris LED diaktifkan secara bergantian untuk menghemat daya dan mengurangi interferensi. *Display* ini beroperasi pada tegangan konstan sebesar 5V. Konektor HUB 12 digunakan sebagai jalur komunikasi data modul *display* LED dan mikrokontroler.

2.10 Router dan *Access point* Wi-Fi (ESP32)

Access point merupakan perangkat atau sistem yang berfungsi menyediakan jaringan Wi-Fi agar perangkat lain dapat terhubung secara nirkabel. Pada jaringan Wi-Fi konvensional, fungsi *access point* umumnya dilakukan oleh router. Namun, pada sistem berbasis mikrokontroler, fungsi tersebut dapat dijalankan oleh ESP32 melalui mode SoftAP atau software *access point*. Mode SoftAP memungkinkan ESP32 membuat jaringan Wi-Fi sendiri, sehingga perangkat lain dapat terhubung langsung ke ESP32 tanpa memerlukan router tambahan.

ESP32 merupakan mikrokontroler yang telah dilengkapi dengan fitur komunikasi nirkabel Wi-Fi. Berdasarkan datasheet ESP32, perangkat ini mendukung Wi-Fi 802.11 b/g/n pada frekuensi 2,4 GHz serta mendukung mode *Station* dan SoftAP [34]. Mode *Station* digunakan ketika ESP32 terhubung ke jaringan Wi-Fi yang sudah tersedia, sedangkan mode SoftAP digunakan ketika ESP32 berperan sebagai penyedia jaringan Wi-Fi. Dengan adanya mode SoftAP, ESP32 dapat digunakan sebagai pusat jaringan lokal yang menerima koneksi dari perangkat lain.

Pada mode *access point*, ESP32 membuat nama jaringan atau Service Set Identifier (SSID) dan kata sandi yang dapat digunakan oleh perangkat *client* untuk terhubung ke jaringan tersebut. Dalam pemrograman Arduino IDE, konfigurasi ESP32 sebagai *access point* dapat dilakukan menggunakan fungsi `WiFi.softAP(ssid, password)`. Selain itu, alamat IP *access point* juga dapat dikonfigurasi menggunakan fungsi `WiFi.softAPConfig()`, sehingga alamat IP ESP32 dapat ditentukan secara tetap [34]. Pengaturan alamat IP tetap diperlukan

agar perangkat *client* dapat mengirimkan data ke alamat *server* yang sama setiap kali sistem dijalankan.

Pada penelitian ini, ESP32 D1 Mini pada modul *receiver* digunakan sebagai *access point* sekaligus *server*. Modul *receiver* membuat jaringan Wi-Fi lokal yang dapat diakses oleh ESP32 D1 Mini pada modul *transmitter*. Setelah jaringan terbentuk, modul *transmitter* berperan sebagai *client* yang menghubungkan diri ke jaringan Wi-Fi *receiver*. Data hasil pembacaan sensor dari modul *transmitter* kemudian dikirimkan menuju modul *receiver* melalui jaringan Wi-Fi lokal tersebut. Konsep ini sejalan dengan pemanfaatan ESP32 sebagai modul Wi-Fi yang dapat digunakan untuk membangun sistem komunikasi berbasis *client* dan *server* pada perangkat IoT .

Penggunaan ESP32 sebagai *access point* memberikan keuntungan karena sistem tidak bergantung pada router eksternal maupun jaringan internet. Hal ini membuat komunikasi antara modul *transmitter* dan *receiver* dapat berjalan secara mandiri dalam satu jaringan lokal. Selain itu, konfigurasi ini juga lebih sederhana karena modul *receiver* menjadi pusat komunikasi yang menerima data dari *transmitter*, kemudian meneruskan hasil pengolahan data ke media *output* seperti *display* LED dot matriks P10 dan aplikasi Android.

Dalam sistem penilaian aktivitas senam otomatis, *access point* ESP32 berperan sebagai penghubung komunikasi nirkabel antara modul *transmitter* dan modul *receiver*. Modul *transmitter* yang membawa sensor MPU-6050 mengirimkan data gerakan peserta ke modul *receiver* melalui jaringan Wi-Fi yang dibuat oleh ESP32 *receiver*. Selanjutnya, modul *receiver* menerima, mengolah, dan menampilkan data hasil penilaian. Dengan demikian, *access point* ESP32 menjadi bagian penting dalam sistem komunikasi data tanpa memerlukan perangkat jaringan tambahan.

2.11 Saklar

Saklar merupakan komponen yang digunakan untuk menghubungkan maupun memutus aliran arus atau tegangan listrik dalam suatu rangkaian elektronik [35]. Ketika saklar berada pada kondisi off (*open circuit*), aliran arus listrik akan terputus sehingga tidak dapat mengalir. Sebaliknya, pada kondisi on (*close circuit*),

kontak logam di dalam saklar saling terhubung sehingga memungkinkan arus listrik mengalir melalui rangkaian tersebut.



Gambar 2.7 Saklar [36]

Tabel 2.7 Spesifikasi Saklar [37]

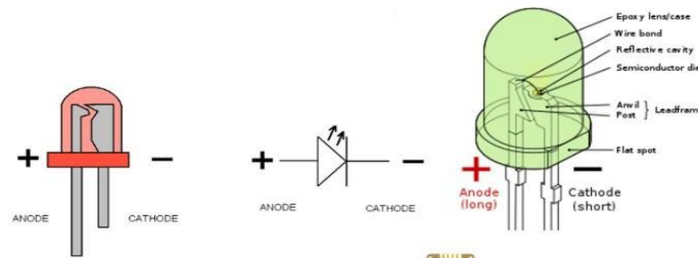
Subjek	Spesifikasi
Model	KCD11-101
Jumlah Pin	2 Pin
Arus maksimum	3A
Tegangan Maksimum	250V
Dimensi	10 x 15 mm

Pada perancangan ini, saklar digunakan sebagai pengendali untuk mengaktifkan maupun menonaktifkan modul *receiver* dan *transmitter*. Berdasarkan Tabel 2.7, jenis saklar yang digunakan adalah *rocker switch* tipe KCD11-1013A dengan kemampuan bekerja pada tegangan maksimum 250V dan arus hingga 3A. Saklar ini memiliki dua terminal koneksi serta ukuran fisik sekitar 10×15 mm.

2.12 LED (Light Emitting Diode) dalam sistem

Light Emitting Diode (LED) merupakan komponen elektronika berbahan semikonduktor yang dapat menghasilkan cahaya saat dialiri arus listrik [38]. LED termasuk jenis dioda yang memiliki dua terminal, yaitu anoda (positif) dan katoda (negatif), sebagaimana ditunjukkan pada simbol pada Gambar 2.8, sehingga arus listrik hanya dapat mengalir dalam satu arah. Cahaya pada LED akan muncul ketika

diberikan tegangan maju (*forward bias*), yaitu saat arus mengalir dari anoda menuju katoda [39].



Gambar 2.8 LED (Light Emitting Diode) dalam Sistem

Tabel 2.8 Spesifikasi LED [40]

Subjek	Spesifikasi
Diameter	5mm
Warna Cahaya	Merah
<i>Forward voltage</i>	1.9V – 2.1V Arus Operasi
Diameter	5mm

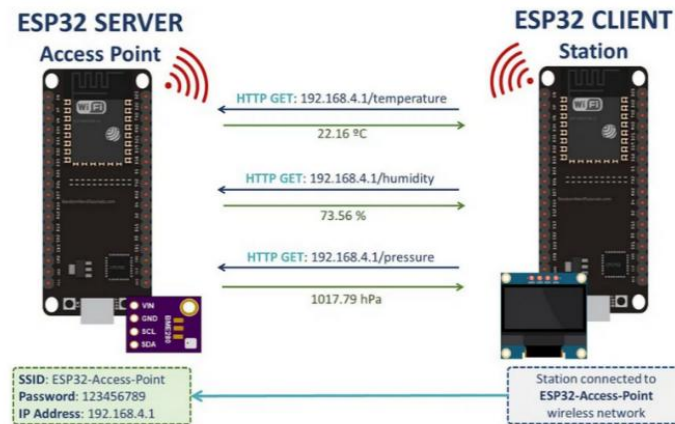
Berdasarkan Tabel 2.8, LED yang digunakan memiliki ukuran diameter 5 mm dengan warna pancaran merah. Nilai tegangan maju (*forward voltage*) yang dibutuhkan berada pada kisaran 1,9V hingga 2,1V, dengan arus kerja sebesar 20 mA. Pada perancangan ini, beberapa LED dimanfaatkan sebagai indikator visual untuk menunjukkan kondisi atau status sistem.

2.13 Komunikasi *Client Server* ESP32 WiFi

Komunikasi *client-server* merupakan pola pertukaran data ketika perangkat *client* menginisiasi koneksi atau mengirimkan permintaan, sedangkan perangkat *server* menyediakan layanan, menerima data, melakukan pemrosesan, dan mengirimkan respons. Dalam sistem tertanam, ESP32 dapat berperan sebagai *client* maupun *server* karena telah dilengkapi antarmuka Wi-Fi pada frekuensi 2,4 GHz.

Penerapan jaringan Wi-Fi pada sistem mikrokontroler memungkinkan beberapa perangkat saling bertukar data melalui jaringan lokal dan mendukung pengelolaan informasi secara terpusat [52]. Kinerja komunikasi ESP32 dipengaruhi

oleh jarak komunikasi, kondisi propagasi, kekuatan sinyal, penghalang fisik, serta kepadatan kanal pada frekuensi 2,4 GHz [53].



Gambar 2.9 Arsitektur *Client-server* [41]

Arsitektur komunikasi terdiri atas ESP32 *client*, *access point* atau router, dan ESP32 *server*. ESP32 *client* membaca data sensor atau nilai hasil pengolahan, kemudian mengirimkannya kepada ESP32 *server* melalui jaringan Wi-Fi lokal. ESP32 *server* menerima dan memvalidasi data sebelum memperbarui informasi pada panel *display* P10.

Tabel 2.9 *Client-server*

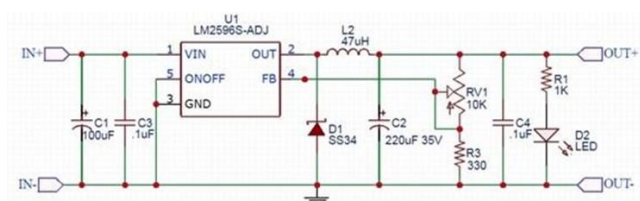
Perangkat	Peran
ESP32 <i>client</i>	Menginisiasi koneksi dan mengirim paket data sensor atau nilai.
<i>Access point</i> /router	Menyediakan SSID dan meneruskan lalu lintas data pada jaringan lokal.
ESP32 <i>server</i>	Mendengarkan port, menerima, memvalidasi, dan mengolah data.
<i>Display</i> P10	Menampilkan nilai atau informasi yang telah diproses <i>server</i> .
<i>Request</i>	Pesan atau paket data yang dikirim oleh <i>client</i> .
<i>Response</i> /ACK	Konfirmasi status penerimaan atau pemrosesan dari <i>server</i> .

Berdasarkan Tabel 2.9, komunikasi *client-server* pada sistem melibatkan perangkat *client*, *access point*, *server*, dan perangkat keluaran. ESP32 *client* menginisiasi koneksi serta mengirimkan data, sedangkan ESP32 *server* menerima dan memproses data yang masuk melalui jaringan Wi-Fi lokal. Proses komunikasi juga melibatkan mekanisme *request* dan *response* sebagai bentuk pertukaran informasi antara *client* dan *server*. Pada penelitian ini, arsitektur tersebut digunakan untuk menghubungkan modul *transmitter*, *receiver*, aplikasi Android, dan bagian *display* dalam proses pengiriman data serta hasil penilaian.

ESP32 *server* mendengarkan koneksi yang masuk, menerima dan memvalidasi format data, memperbarui informasi pada *display* P10, kemudian mengirimkan *acknowledgment* atau status pemrosesan kepada *client*. Mekanisme *request-response* digunakan agar perangkat pengirim dapat mengetahui apakah data telah diterima dan diproses dengan benar. ESP32 mendukung integrasi pemrosesan data, antarmuka perangkat, dan komunikasi jaringan dalam satu mikrokontroler [54].

2.14 Modul DC-DC Step-down converter LM2596

Modul penurun tegangan atau *Step-down converter* adalah komponen elektronika yang berfungsi untuk mempertahankan kestabilan tegangan keluaran meskipun terjadi perubahan pada tegangan masukan [42]. Salah satu jenisnya adalah *step-down regulator* atau *buck converter*, yaitu rangkaian yang digunakan untuk menurunkan tegangan DC dari nilai yang lebih tinggi menjadi lebih rendah. Contoh IC yang termasuk dalam kategori regulator *switching* tipe *step-down* adalah IC LM2596.

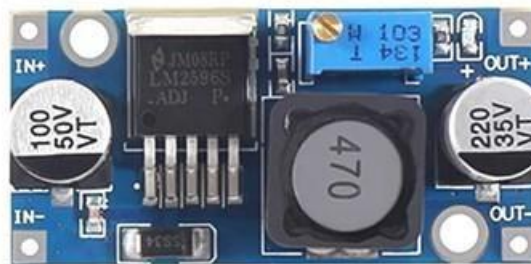


Gambar 2.10 Rangkaian Schematic LM2596 [43]

Pada Gambar 2.10 ditampilkan rangkaian *buck converter* yang menggunakan IC LM2596S-ADJ untuk menurunkan tegangan DC dari terminal

input (IN+) menjadi tegangan *output* (OUT+). Tegangan masukan dialirkan ke pin Vin melalui kapasitor C1 dan C3 yang berperan sebagai *filter* untuk mereduksi *noise*. IC LM2596S-ADJ bekerja dengan metode *switching* untuk mengatur tegangan keluaran, di mana arus dialirkan menuju induktor L2 dan dioda D1 guna menyimpan serta menyalurkan energi secara bergantian.

Kapasitor C2 digunakan untuk menstabilkan tegangan *output* dengan menyaring gangguan yang muncul. Pengaturan tegangan keluaran dilakukan melalui jalur umpan balik ke pin FB (*feedback*) menggunakan rangkaian pembagi tegangan yang terdiri dari potensiometer RV1 dan resistor R3, sehingga nilai *output* dapat disesuaikan dengan kebutuhan. Selain itu, LED bersama resistor R1 berfungsi sebagai indikator bahwa *output* dalam kondisi aktif, sedangkan kapasitor C4 digunakan sebagai *filter* tambahan untuk mengurangi *ripple* pada tegangan keluaran.



Gambar 2.11 Regulator Tegangan LM2596 [44]

Tabel 2.10 Spesifikasi LM2596 [44]

Subjek	Spesifikasi
<i>Input Voltage</i>	3.2V – 35V
<i>Output Voltage</i>	1.25V – 35V (<i>Adjustable</i>)
<i>Output Current</i>	Rated current is 2.5A, maximum 3A (<i>Additional heatsink is required</i>)
<i>Conversion Efficiency</i>	92%
<i>Operating Temperature</i>	–45 °C to +85 °C

Berdasarkan Tabel 2.10, modul DC-DC converter ini mampu menerima tegangan masukan pada rentang 3,2V hingga 35V. Tegangan keluarannya dapat

diatur dari 1,25V sampai 35V sesuai dengan kebutuhan perangkat yang digunakan. Arus keluaran yang direkomendasikan sebesar 2,5A, namun modul ini masih dapat menangani arus maksimum hingga 3A dengan bantuan *heatsink* untuk mencegah terjadinya panas berlebih. Selain itu, tingkat efisiensi konversi yang dimiliki modul ini dapat mencapai sekitar 92%.

Pada perancangan ini digunakan modul DC-DC converter tipe LM2596, seperti yang ditunjukkan pada Gambar 2.11. Modul ini berfungsi untuk menghasilkan tegangan keluaran yang stabil sesuai kebutuhan komponen pada sistem *transmitter* dan *receiver*, dengan cara menurunkan tegangan dari sumber baterai ke level yang dibutuhkan. LM2596 memiliki empat pin utama, yaitu IN+ sebagai terminal Vcc yang terhubung ke kutub positif baterai, IN- sebagai ground yang terhubung ke kutub negatif baterai, OUT+ sebagai keluaran tegangan positif yang disalurkan ke rangkaian atau komponen, serta OUT- yang dihubungkan ke ground pada rangkaian distribusi daya atau komponen yang digunakan.

2.15 Metode *Downsampling* dengan Metode *Average pooling*

Downsampling merupakan proses pengurangan jumlah sampel yang bertujuan untuk memperkecil data yang harus diproses dengan tetap mempertahankan pola utama dari data tersebut. Pada penelitian ini, proses *downsampling* dilakukan menggunakan metode *average pooling*. Metode ini bekerja dengan membagi data sensor yang berurutan ke dalam beberapa jendela atau kelompok dengan ukuran tertentu, kemudian seluruh nilai pada setiap jendela dihitung nilai rata-ratanya untuk menghasilkan satu data baru. Dengan demikian, beberapa sampel data mentah dapat direpresentasikan oleh satu nilai hasil rata-rata sehingga jumlah data yang digunakan pada tahap pengolahan berikutnya menjadi lebih sedikit[45]. Proses *downsampling* menggunakan *average pooling* dapat dinyatakan dengan Persamaan (2.1).

$$y_i = \frac{1}{n} \sum_{k=1}^n x_k \quad (2.1)$$

dengan x_k sebagai data asli pada satu jendela, (n) sebagai jumlah data dalam jendela, dan y_i sebagai satu data hasil *downsampling* yang mewakili jendela tersebut. Pada penelitian ini, nilai ditentukan berdasarkan jumlah data yang akan

dirata-ratakan dalam setiap kelompok. Semakin besar nilai , semakin banyak data yang digabungkan menjadi satu nilai sehingga jumlah data hasil *downsampling* menjadi semakin sedikit.

Downsampling dengan metode *average pooling* diterapkan secara terpisah pada data *accelerometer* sumbu X, Y, dan Z serta *gyroscope* sumbu X, Y, dan Z. Selain mengurangi jumlah data, penggunaan nilai rata-rata pada setiap jendela juga dapat mengurangi pengaruh fluktuasi kecil yang terdapat pada data sensor. Hasil proses ini berupa data yang lebih ringkas sebelum dilanjutkan ke tahap *moving average*, normalisasi Min-Max, dan perhitungan RMSE.

Pemilihan ukuran jendela atau faktor *downsampling* perlu disesuaikan dengan karakteristik data gerakan. Nilai faktor yang terlalu besar dapat menyebabkan perubahan gerakan yang berlangsung dalam waktu singkat menjadi kurang terlihat, sedangkan nilai yang terlalu kecil hanya memberikan sedikit pengurangan terhadap jumlah data. Oleh karena itu, faktor *downsampling* ditentukan agar jumlah data dapat dikurangi tanpa menghilangkan pola utama dari gerakan yang direkam.

2.16 Moving average

Moving average merupakan metode *smoothing* yang digunakan untuk mengurangi fluktuasi pada data dengan menghitung nilai rata-rata dari sejumlah data yang berdekatan. Pada metode ini digunakan ukuran *window* tertentu yang bergerak mengikuti urutan data.

Persamaan *moving average* dapat dituliskan sebagai:

$$MA_t = \frac{1}{N} \sum_{i=0}^{N-1} x_{t-i} \quad (2.2)$$

dengan sebagai nilai hasil *moving average*, sebagai data yang diproses, dan sebagai ukuran *window*.

Pada penelitian ini, *moving average* diterapkan setelah proses *downsampling* pada data *accelerometer* dan *gyroscope* sumbu X, Y, dan Z. Ukuran *window* yang digunakan adalah 10 data, sehingga setiap nilai keluaran diperoleh dari rata-rata

maksimal 10 data yang berdekatan. Proses ini bertujuan untuk membuat data lebih halus sebelum masuk ke tahap normalisasi Min-Max.

2.17 Min-Max Normalization

Normalisasi adalah suatu proses yang digunakan untuk mengubah skala data sehingga nilai atribut berada dalam rentang tertentu yang telah ditentukan [46]. Salah satu metode yang umum digunakan adalah *Min-Max Normalization*, yaitu teknik transformasi linier yang menjaga perbandingan relatif antar nilai sebelum dan sesudah proses normalisasi, sehingga distribusi data menjadi lebih proporsional. Penerapan metode ini bertujuan untuk mempersempit rentang nilai minimum dan maksimum, sehingga dapat meningkatkan efisiensi serta mempercepat proses pembelajaran pada tahap klasifikasi[47].

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \times 100 \quad (2.3)$$

2.18 RMSE

Root Mean Square Error (RMSE) merupakan metode yang digunakan untuk mengukur besar perbedaan antara dua kumpulan data berdasarkan selisih pada setiap pasangan data. Pada penelitian ini, RMSE digunakan untuk membandingkan data peserta uji terhadap data master setelah kedua data melalui proses *downsampling* menggunakan *average pooling*, *moving average*, dan normalisasi Min-Max. Semakin kecil nilai RMSE menunjukkan bahwa data peserta semakin mendekati pola data master, sedangkan nilai RMSE yang semakin besar menunjukkan perbedaan yang semakin besar.

Persamaan RMSE yang digunakan dapat dituliskan sebagai:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2} \quad (2.4)$$

dengan *RMSE* sebagai nilai *Root Mean Square Error*, x_i sebagai data peserta uji pada data ke- i , y_i sebagai data master pada data ke- i , dan N sebagai jumlah data yang dibandingkan. Nilai RMSE yang semakin kecil menunjukkan bahwa perbedaan antara data peserta uji dengan data master semakin kecil. Sebaliknya,

nilai RMSE yang semakin besar menunjukkan bahwa terdapat perbedaan yang semakin besar antara pola data peserta uji dan data master.

Perhitungan RMSE dilakukan secara terpisah terhadap enam komponen sensor, yaitu *accelerometer X*, *accelerometer Y*, *accelerometer Z*, *gyroscope X*, *gyroscope Y*, dan *gyroscope Z*. Selanjutnya, keenam nilai RMSE tersebut dirata-ratakan untuk memperoleh satu nilai RMSE rata-rata yang mewakili keseluruhan perbedaan antara data peserta uji dan data master.

$$RMSE_{avg} = \frac{RMSE_{AX} + RMSE_{AY} + RMSE_{AZ} + RMSE_{GX} + RMSE_{GY} + RMSE_{GZ}}{6} \quad (2.5)$$

dengan $RMSE_{AX}$, $RMSE_{AY}$, dan $RMSE_{AZ}$ masing-masing merupakan nilai RMSE *accelerometer* sumbu X, Y, dan Z, sedangkan $RMSE_{GX}$, $RMSE_{GY}$, dan $RMSE_{GZ}$ merupakan nilai RMSE *gyroscope* sumbu X, Y, dan Z. Nilai $RMSE_{avg}$ digunakan sebagai nilai kesalahan keseluruhan dalam proses penentuan skor.

Skor akhir diperoleh dengan mengurangi nilai RMSE rata-rata dari nilai maksimum 100. Dengan persamaan tersebut, semakin kecil nilai RMSE rata-rata maka semakin tinggi skor yang diperoleh karena pola data peserta uji semakin mendekati data master. Sebaliknya, semakin besar nilai RMSE rata-rata maka skor akan semakin rendah. Persamaan skor tersebut merupakan metode konversi yang ditetapkan pada sistem penelitian untuk mempermudah interpretasi hasil perbandingan, sehingga bukan merupakan rumus baku dari metode RMSE maupun standar penilaian resmi senam.

$$Score = 100 - RMSE_{\{avg\}} \quad (2.6)$$

2.19 Fire Monkey

FireMonkey merupakan sebuah platform *runtime* sekaligus *framework* pengembangan aplikasi yang digunakan pada Embarcadero C++ Builder, Delphi, dan Appmethod untuk sistem operasi Android. Platform ini dikenal sebagai *framework* aplikasi pertama yang memanfaatkan teknologi Graphic Processing Unit (GPU) dalam proses *rendering*. FireMonkey menyediakan pustaka antarmuka grafis lintas platform yang memungkinkan pengembang membangun aplikasi

dengan tampilan pengguna (*user interface*) yang dapat dijalankan pada berbagai sistem operasi seperti *Windows*, *macOS*, *iOS*, dan *Android*.

Dalam implementasinya, FireMonkey memanfaatkan akselerasi perangkat keras melalui GPU apabila tersedia, sehingga dapat meningkatkan performa tampilan grafis. Dukungan ini mencakup penggunaan Direct2D pada *Windows Vista*, *Windows 7*, dan *Windows 8*, OpenGL pada *macOS*, serta OpenGL ES pada *iOS* dan *Android*. Sementara itu, pada sistem operasi yang tidak mendukung Direct2D, seperti *Windows XP*, proses *rendering* dilakukan menggunakan GDI+.

Aplikasi yang dikembangkan menggunakan FireMonkey terbagi menjadi dua kategori utama, yaitu HD dan 3D. Aplikasi HD mengacu pada aplikasi dua dimensi yang bersifat konvensional, namun tetap menggunakan basis grafis vektor sehingga memungkinkan penskalaan tampilan tanpa menurunkan kualitas visual. Di sisi lain, kategori 3D menyediakan dukungan untuk pengembangan antarmuka berbasis tiga dimensi yang umumnya digunakan untuk kebutuhan visualisasi. Kedua jenis ini dapat digunakan secara bersamaan dalam satu aplikasi sesuai kebutuhan.

Selain sebagai pustaka antarmuka, FireMonkey juga berfungsi sebagai *framework* pengembangan perangkat lunak yang lengkap, dengan berbagai fitur yang sebanding dengan *Visual Component Library* (VCL). Perbedaan utama antara FireMonkey dan VCL terletak pada kemampuannya yang lintas platform, penggunaan grafis berbasis vektor, fleksibilitas komponen visual yang dapat diturunkan satu sama lain, dukungan terhadap *styling*, serta kemampuan dalam menerapkan efek visual dan animasi pada setiap komponen.

Karena sifatnya yang *multiplatform*, satu basis kode sumber dapat digunakan untuk berbagai sistem operasi yang didukung. FireMonkey juga mendukung pengembangan aplikasi pada arsitektur 32-bit dan 64-bit untuk *Windows*, serta 32-bit untuk *macOS*, *iOS*, dan *Android*. Sejak pertama kali diperkenalkan pada versi XE2, FireMonkey terus mengalami pengembangan yang signifikan. Saat ini, integrasi dengan *macOS* telah berjalan dengan baik di dalam IDE, sementara perangkat Mac hanya diperlukan pada tahap distribusi akhir. Selain

itu, berbagai fitur tambahan seperti sensor, *gesture*, dan GPS telah disediakan untuk mendukung pengembangan aplikasi *mobile* secara optimal [48].

2.20 Android

Android merupakan sebuah platform yang bersifat komprehensif karena dirancang sebagai satu kesatuan perangkat lunak yang lengkap untuk perangkat *mobile*. Konsep ini memberikan kemudahan bagi berbagai pihak, baik pengembang, pengguna, maupun vendor. Bagi pengembang, Android telah menyediakan berbagai tools dan *framework* yang diperlukan dalam proses pembuatan aplikasi *mobile* berbasis Android.

Untuk mulai mengembangkan aplikasi, pengembang cukup menggunakan Android SDK (*Software Development Kit*) yang telah menyediakan fasilitas lengkap, termasuk *emulator* sebagai simulasi perangkat *smartphone*. Dengan adanya *emulator* ini, proses pengujian aplikasi dapat dilakukan tanpa harus menggunakan perangkat keras secara langsung. Namun demikian, *emulator* memiliki keterbatasan, terutama dalam hal simulasi sensor perangkat keras seperti *gyroscope*, kompas, temperatur, dan sensor lainnya yang hanya dapat berfungsi secara optimal pada perangkat *smartphone* asli [48].

