

1. Sistem kontrol yang beroperasi tanpa umpan balik dikenal sebagai sistem loop terbuka. Ini menunjukkan bahwa sistem tidak memantau atau mengubah keluarannya sebagai respons terhadap hasil tersebut. Tanpa memahami apakah hasilnya sesuai dengan yang diinginkan, mikrokontroler hanya mengarahkan aktuator. Teknologi semacam ini lebih mudah digunakan, tetapi juga kurang presisi dan lebih rentan terhadap interferensi. Di sisi lain, sistem loop tertutup adalah sistem yang menggunakan umpan balik sensor untuk memantau keluarannya. Keluaran dikoreksi dan disesuaikan agar sesuai dengan target yang diinginkan menggunakan data dari sensor. Mekanisme ini meningkatkan akurasi, stabilitas, dan kapasitas sistem untuk beradaptasi dengan perubahan lingkungan.
2. Sistem yang digunakan dan dimodelkan untuk penelitian ini tergolong sistem loop tertutup. Hal ini dikarenakan sistem mendeteksi perilaku aktuator, khususnya motor servo, menggunakan sensor MPU6050 dan PIR sebagai sumber umpan balik. Sensor PIR mendeteksi pergerakan di sekitar robot, sementara sensor MPU6050 membaca data akselerasi dan rotasi robot. Mikrokontroler ESP32 memproses data dari kedua sensor untuk secara otomatis mengubah posisi dan gerakan motor servo. Dengan proses umpan balik yang terus-menerus menghubungkan sensor, mikrokontroler, dan aktuator, sistem ini dapat mengubah pergerakannya sesuai dengan kondisi yang diamati, sehingga tergolong sistem loop tertutup.
3. Sensor MPU6050 dan PIR merupakan sensor yang digunakan dalam sistem ini, dan masing-masing memiliki fungsi penting. Sensor MPU6050 mengukur percepatan linear dan kecepatan sudut suatu gerakan dengan menggabungkan kemampuan akselerometer dan giroskop. Orientasi, posisi, dan keseimbangan sistem robot ditentukan menggunakan data dari sensor ini. Di sisi lain, PIR (Passive Infrared Sensor) menggunakan radiasi inframerah yang dilepaskan oleh tubuh manusia atau hewan untuk mendeteksi pergerakan benda hidup di seluruh sistem. Untuk memulai tindakan tertentu, seperti mengoperasikan motor servo, sensor PIR mengirimkan sinyal logika tinggi ke mikrokontroler ketika mendeteksi gerakan.
4. Setiap pin pada ESP32 memiliki fungsi dan arah koneksi yang telah disesuaikan dengan kebutuhan sistem. Pin GPIO 21 dan GPIO 22 digunakan untuk komunikasi I²C dengan sensor MPU6050, di mana GPIO 21 berfungsi sebagai jalur data (SDA) dan GPIO 22 sebagai jalur clock (SCL). Sensor PIR dihubungkan ke pin GPIO 14 sebagai input digital yang mendeteksi gerakan di lingkungan sekitar. Sementara itu, empat buah servo motor dihubungkan secara berurutan ke pin GPIO 18, 19, 23, dan 25 sebagai keluaran PWM untuk mengontrol sudut pergerakan masing-masing servo. Semua komponen mendapatkan suplai daya dari pin 3.3V atau 5V pada ESP32 serta terhubung ke jalur GND agar sistem memiliki referensi tegangan yang sama dan stabil. Arah koneksi tersebut disusun agar data dari sensor masuk ke ESP32 sebagai input, kemudian ESP32 memproses data tersebut dan mengirimkan sinyal kontrol ke servo motor sebagai output.
5. Alur kerja sistem dimulai ketika ESP32 melakukan inisialisasi terhadap semua komponen, termasuk komunikasi I²C untuk sensor MPU6050, konfigurasi input untuk sensor PIR, serta inisialisasi pin PWM untuk mengendalikan servo motor. Setelah sistem aktif, ESP32 akan terus membaca data dari sensor MPU6050 untuk mengetahui nilai percepatan dan rotasi, serta memantau sinyal dari sensor PIR untuk mendeteksi adanya gerakan. Data yang

diperoleh kemudian diproses oleh ESP32 untuk menentukan perintah yang harus dikirim ke servo motor. Jika sensor PIR mendeteksi gerakan, atau jika terjadi perubahan orientasi dari data MPU6050, maka ESP32 akan mengirimkan sinyal PWM ke servo untuk mengatur sudut gerakannya sesuai dengan hasil perhitungan. Proses ini berlangsung terus menerus secara real-time sehingga sistem mampu menyesuaikan diri terhadap perubahan lingkungan. Dengan demikian, sistem ini menunjukkan karakteristik close-loop control, di mana sensor memberikan umpan balik yang memengaruhi tindakan aktuator, dan seluruh komponen bekerja secara terintegrasi di bawah kendali ESP32.