

a. Apakah perbedaan open-loop system dan close-loop system?

Open-loop system adalah sistem kendali yang bekerja berdasarkan input awal tanpa memperhatikan kondisi output yang dihasilkan. Sistem ini tidak memiliki mekanisme umpan balik, sehingga tidak dapat melakukan koreksi otomatis jika terjadi kesalahan atau penyimpangan dari hasil yang diinginkan. Contoh sederhana dari sistem open-loop adalah kipas angin yang menyala selama waktu tertentu tanpa mempertimbangkan suhu ruangan.

Sebaliknya, close-loop system adalah sistem kendali yang memiliki mekanisme umpan balik. Sistem ini mampu memantau output yang dihasilkan dan menggunakannya untuk menyesuaikan input atau aksi berikutnya. Dengan adanya feedback, sistem dapat melakukan koreksi secara otomatis untuk mencapai hasil yang lebih akurat dan stabil. Contoh dari sistem close-loop adalah AC otomatis yang menyesuaikan suhu berdasarkan pembacaan sensor suhu ruangan.

Perbedaan utama antara keduanya terletak pada keberadaan umpan balik dan kemampuan sistem untuk beradaptasi terhadap perubahan kondisi lingkungan.

b. Apakah sistem yang anda kerjakan dan simulasikan di atas termasuk open-loop system atau close-loop system? Justifikasi jawaban anda!

Sistem yang saya kerjakan termasuk dalam kategori close-loop system. Hal ini dapat dibuktikan dari cara kerja sistem yang melibatkan sensor sebagai input utama untuk menentukan aksi dari aktuator secara otomatis dan dinamis.

Dalam proyek ini, saya menggunakan dua jenis sensor, yaitu PIR sensor dan MPU6050 (accelerometer dan gyroscope). Sensor PIR berfungsi untuk mendeteksi gerakan manusia atau objek di sekitarnya, sedangkan MPU6050 digunakan untuk membaca nilai orientasi dan percepatan pada sumbu X, Y, dan Z. Data dari kedua sensor ini kemudian diproses oleh mikrokontroler ESP32 untuk mengatur posisi lima buah servo motor.

Setiap kali sensor mendeteksi perubahan kondisi, baik berupa gerakan dari PIR maupun perubahan orientasi dari MPU6050, mikrokontroler akan mengeksekusi perintah untuk mengubah posisi servo motor sesuai dengan logika yang telah diprogram. Sebagai contoh, jika PIR mendeteksi adanya gerakan, maka semua servo akan bergerak ke posisi tertentu dan kemudian kembali ke posisi awal. Jika MPU6050 mendeteksi perubahan nilai roll, pitch, atau yaw, maka servo akan menyesuaikan posisinya berdasarkan nilai tersebut.

Karena output dari sistem (posisi servo) bergantung langsung pada input dari sensor dan sistem terus menyesuaikan aksi berdasarkan perubahan kondisi lingkungan, maka ini menunjukkan adanya mekanisme umpan balik. Mekanisme ini merupakan ciri utama dari sistem loop tertutup atau close-loop system.

Sebaliknya, jika sistem hanya menggerakkan servo berdasarkan waktu atau perintah tetap tanpa memperhatikan data sensor, maka itu baru disebut open-loop. Namun dalam proyek ini, karena pengambilan keputusan dilakukan berdasarkan data sensor secara real-time, maka sistem yang saya buat jelas termasuk close-loop system.

c. Jelaskan fungsi masing-masing sensor yang digunakan pada sistem di atas!

Dalam proyek ini, terdapat dua jenis sensor yang digunakan, yaitu sensor PIR dan sensor MPU6050. Masing-masing memiliki fungsi spesifik yang saling melengkapi dalam sistem kendali berbasis mikrokontroler.

1. Sensor PIR (Passive Infrared Sensor) Sensor PIR berfungsi untuk mendeteksi gerakan objek hidup, seperti manusia, berdasarkan perubahan radiasi inframerah yang dipancarkan oleh tubuh. Ketika seseorang bergerak di dekat sensor, PIR akan mengeluarkan sinyal digital (HIGH) yang menandakan adanya gerakan. Dalam sistem ini, sinyal tersebut digunakan sebagai pemicu untuk menggerakkan semua servo motor secara bersamaan, sebagai bentuk respons terhadap deteksi gerakan.
2. Sensor MPU6050 (Accelerometer dan Gyroscope) MPU6050 adalah sensor gabungan yang terdiri dari akselerometer dan giroskop tiga sumbu. Sensor ini mampu membaca percepatan (acceleration) dan rotasi (angular velocity) pada sumbu X, Y, dan Z. Dalam proyek ini, data dari MPU6050 digunakan untuk menentukan posisi atau sudut dari masing-masing servo motor. Nilai roll, pitch, dan yaw yang dihasilkan oleh sensor akan diubah menjadi sudut servo, sehingga servo dapat menyesuaikan posisinya secara dinamis sesuai orientasi yang terdeteksi.

Kombinasi kedua sensor ini memungkinkan sistem untuk merespons baik terhadap gerakan eksternal maupun perubahan orientasi, menjadikannya sistem yang adaptif dan interaktif.

d. Jelaskan alasan, fungsi, dan arah tuju koneksi setiap pin ESP32 yang dimanfaatkan dari skematik (poin 5 Informasi Pengerjaan) yang telah anda buat!

Pin GPIO 13 hingga GPIO 18 Pin ini digunakan sebagai output digital untuk mengontrol lima buah servo motor. Setiap servo terhubung ke satu pin GPIO yang akan

mengirimkan sinyal PWM (Pulse Width Modulation) untuk menentukan sudut posisi servo. Fungsi utama pin ini adalah sebagai pengendali aktuator berdasarkan data sensor.

Pin GPIO 19 dan GPIO 21 Kedua pin ini digunakan untuk komunikasi I2C dengan sensor MPU6050. GPIO 19 berfungsi sebagai pin SCL (Serial Clock Line), sedangkan GPIO 21 berfungsi sebagai pin SDA (Serial Data Line). I2C adalah protokol komunikasi serial yang memungkinkan ESP32 membaca data dari sensor MPU6050 secara efisien.

Pin GPIO 22 Pin ini digunakan sebagai input digital dari sensor PIR. Ketika PIR mendeteksi gerakan, pin ini akan menerima sinyal HIGH, yang kemudian diproses oleh ESP32 untuk mengaktifkan servo motor.

Pin GND dan 3.3V Pin GND digunakan sebagai ground untuk semua perangkat yang terhubung, sedangkan pin 3.3V digunakan sebagai sumber daya untuk sensor dan modul yang membutuhkan tegangan rendah. Koneksi ini penting untuk memastikan semua komponen bekerja stabil dan aman.

Alasan pemilihan pin didasarkan pada kemampuan masing-masing pin ESP32, seperti dukungan terhadap sinyal PWM untuk servo dan protokol I2C untuk sensor MPU6050. Penataan pin juga mempertimbangkan kemudahan routing dan efisiensi dalam skematik.

e. Dalam suatu rapat monitoring, anda diminta untuk menjelaskan kode yang anda buat ke rekan kerja tim anda yang berbeda divisi dengan anda. Buatlah penjelasan yang mudah dipahami untuk menjelaskan alur bagaimana sistem yang anda program bekerja berdasarkan eksekusi kode yang telah dibuat hingga ke eksekusi yang dilakukan oleh mikrokontroler, sensor, dan aktuator yang ada!

1. Inisialisasi Sistem Program dimulai dengan mengimpor library yang diperlukan, seperti Wire.h untuk komunikasi I2C, Adafruit_MPU6050.h untuk sensor MPU6050, dan Servo.h untuk mengontrol servo motor. Kemudian, pin-pin ESP32 yang digunakan untuk servo, sensor PIR, dan komunikasi I2C diatur sesuai fungsinya.
2. Setup Awal Pada bagian setup(), komunikasi serial diaktifkan untuk keperluan debugging. Sensor MPU6050 diinisialisasi dan dicek apakah terdeteksi dengan benar. Servo motor juga di-attach ke masing-masing pin GPIO, dan sensor PIR disiapkan sebagai input digital.
3. Loop Utama (Pengulangan) Di dalam fungsi loop(), ESP32 secara terus-menerus membaca data dari sensor MPU6050, yaitu nilai roll, pitch, dan yaw. Nilai-nilai ini

kemudian dikonversi menjadi sudut servo dan dikirimkan ke masing-masing servo motor agar mereka bergerak sesuai orientasi yang terdeteksi.

Selain itu, ESP32 juga memantau status sensor PIR. Jika PIR mendeteksi gerakan (sinyal HIGH), maka semua servo akan bergerak ke posisi tertentu sebagai respons terhadap gerakan tersebut. Setelah itu, servo kembali ke posisi awal.

4. Eksekusi Aktuator Servo motor sebagai aktuator akan menerima sinyal PWM dari ESP32 dan bergerak sesuai sudut yang ditentukan. Gerakan servo ini merupakan output dari sistem yang dikendalikan berdasarkan input sensor.
5. Umpan Balik dan Responsif Sistem bekerja secara real-time dan responsif terhadap perubahan lingkungan. Sensor terus membaca data baru, dan mikrokontroler menyesuaikan aksi servo berdasarkan data tersebut. Ini menunjukkan bahwa sistem memiliki loop tertutup, di mana output selalu dikontrol berdasarkan input sensor.
6. *Monitoring dan Debugging* Data dari sensor dan status servo dapat ditampilkan melalui Serial Monitor, sehingga pengguna dapat memantau kinerja sistem dan melakukan evaluasi jika diperlukan.

Secara keseluruhan, sistem ini dirancang untuk bekerja secara otomatis dan adaptif, dengan ESP32 sebagai pusat kendali, sensor sebagai input, dan servo sebagai output. Alur kerja yang terstruktur dan responsif menjadikan sistem ini efektif untuk aplikasi interaktif berbasis mikrokontroler.