

- a. Perbedaan open-loop system dan close-loop system:
 1. Open-loop System : tipe sistem yang mana output tidak memengaruhi input secara langsung. Jadi output tidak memberikan feedback dan tidak memiliki kontrol kepada input dari sistem. Contohnya adalah ketika lampu menyala saat saklar ditekan.
 2. Close-loop System : Kebalikan dari open-loop system dimana output memengaruhi input. Jadi output memberikan feedback untuk mengoreksi input secara otomatis. Contohnya pada mesin cuci otomatis yang memiliki sensor untuk mendeteksi kelembaban dan ketika mencapai ukuran kering tertentu mesin cuci akan otomatis berhenti.
- b. Sistem ini adalah closed-loop system karena sistem membaca sensor MPU6050 (membaca gyroscope) untuk mengetahui nilai roll, pitch, dan yaw nya. Dari nilai-nilai ini kemudian servo akan bergerak secara langsung menyesuaikan perubahan posisi dari sensor. Hal yang sama juga terjadi pada PIR sensor.
- c. MPU6050 = Mengukur acceleration dan gyroscope. Pada project ini lebih digunakan ke mengukur orientasi gyroscope nya (roll, pitch, yaw), yang kemudian pergerakannya akan diikuti servo.
PIR = Mendeteksi keberadaan gerakan objek melalui radiasi inframerah yang pasif sesuai namanya. Dari gerakan ini semua servo bergerak ke posisi acak lalu kembali ke posisi awal.
- d. Servo 1 : PIN 17 = Output, untuk mengontrol sudut servo 1 berkebalikan dengan sensor
Servo 2 : PIN 16 = Output, untuk mengontrol sudut servo 2 berkebalikan dengan sensor
Servo 3 : PIN 25 = Output, untuk mengontrol sudut servo 3 searah dengan sensor
Servo 4 : PIN 26 = Output, untuk mengontrol sudut servo 4 searah dengan sensor
Servo 5 : PIN 13 = Output, untuk mengontrol sudut servo 5 mengikuti gerakan yaw kemudian kembali ke awal
PIR : PIN 2 = Input, untuk menerima sinyal HIGH ketika ada gerakan terdeteksi
MPU6050 : PIN 21 22 = SDA, SCL, untuk komunikasi data antara mikro dengan sensor ini.
- e. Begitu sistem dinyalakan, ESP32 akan langsung mengarahkan semua servo ke posisi tengah di sudut 90 derajat. Ini semacam titik awal sebelum sistem mulai bergerak misalnya seperti ketika seseorang berdiri tegak dulu sebelum melangkah. Tujuan awal ini supaya semua gerakan berikutnya lebih terkontrol dan punya acuan yang jelas.

Setelah posisi awal siap, sensor MPU mulai “merasakan” keadaan di sekitarnya. Sensor ini berfungsi layaknya indra keseimbangan, yang bisa mengetahui arah dan kemiringan. Ia terus membaca apakah sistem miring ke kanan atau kiri (roll), condong ke depan atau belakang (pitch), atau sedang berputar ke kanan atau ke kiri (yaw). Bersamaan dengan itu, sensor PIR juga aktif. PIR punya peran berbeda. Dia bukan digunakan untuk membaca arah, melainkan untuk mendeteksi apakah ada gerakan di sekitar.

Setiap data yang diterima dari kedua sensor tersebut akan diolah oleh ESP32. Mikrokontroler ini bisa diibaratkan sebagai “otak” dari sistem. Setelah membaca data, ESP32 mengubahnya menjadi instruksi untuk 5 motor servo yang ada. Misalnya, ketika MPU mendeteksi kemiringan ke kanan sebesar 30 derajat, ESP32 akan memberi perintah pada servo agar bergerak mengikuti perubahan tersebut. Servo 1 dan 2 akan bergerak berlawanan arah dengan roll, servo 3 dan 4 mengikuti arah pitch, sedangkan servo 5 merespons yaw. Ada aturan tambahan untuk yaw, yaitu jika tidak ada perubahan dalam waktu satu detik, servo yaw otomatis kembali ke posisi tengah, seperti kepala seseorang yang kembali tegak setelah menoleh sebentar.

Apabila sensor PIR mendeteksi adanya pergerakan di sekitar, ESP32 juga akan merespons. Semua servo akan diarahkan ke posisi khusus yang telah ditentukan sebelumnya, berhenti sejenak, lalu kembali ke posisi semula di sudut 90 derajat. Setelah aksi ini selesai, sistem langsung kembali ke siklus pembacaan sensor.

Siklus ini terjadi terus-menerus, MPU membaca kondisi, ESP32 mengolah informasi tersebut, dan servo bergerak sesuai instruksi. Begitu servo bergerak, MPU akan kembali membaca posisi dan siklusnya berulang. Kalau diibaratkan tubuh manusia MPU berperan seperti indera yang merasakan posisi, ESP32 sebagai otak yang mengambil keputusan, dan servo adalah otot yang bergerak mengikuti perintah. Saat tidak ada input baru, servo yaw otomatis kembali ke tengah, siap menerima gerakan berikutnya.