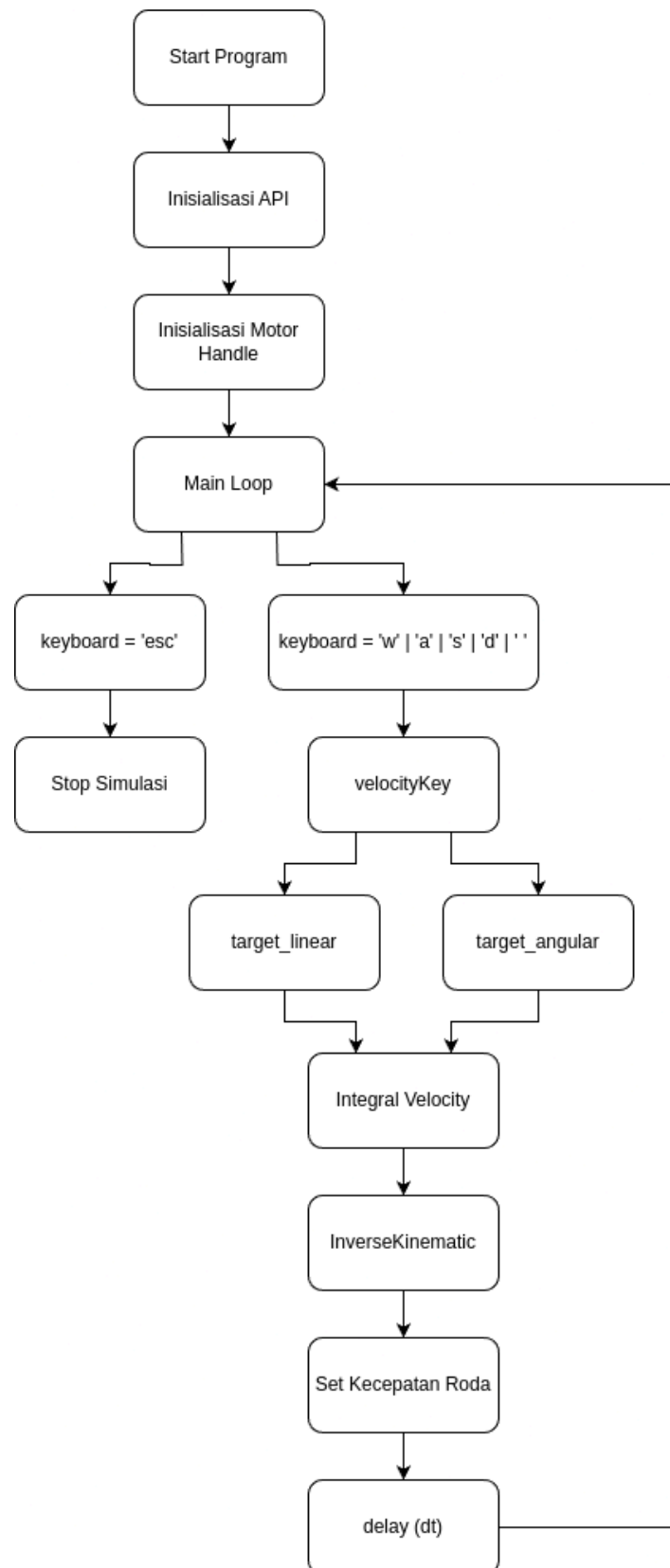


Nama : Hernanda Achmad Priyatna

NRP : 5022221114

Laporan Program Soccer Robot



Nama : Hernanda Achmad Priyatna

NRP : 5022221114

Flowchart di atas menggambarkan alur kerja program untuk mengendalikan robot secara manual menggunakan input dari keyboard. Program ini membaca perintah dari pengguna dan menerjemahkannya menjadi kecepatan linear dan angular, yang kemudian dikonversi menjadi kecepatan roda melalui inverse kinematics. Selain itu, program menerapkan mekanisme pembatasan kecepatan untuk memastikan pergerakan yang halus dan realistis. Berikut adalah penjelasan rinci mengenai cara kerja program:

1. Inisialisasi

- Program dimulai dengan mencetak "Program started".
- Membuat koneksi ke CoppeliaSim melalui RemoteAPIClient.
- Menginisialisasi variabel seperti target_linear, target angular, current_linear, dan current angular yang akan digunakan dalam perhitungan kecepatan.
- Mengambil handle motor kanan dan kiri robot menggunakan fungsi getMotorsHandle().

2. Loop Utama (Main Loop)

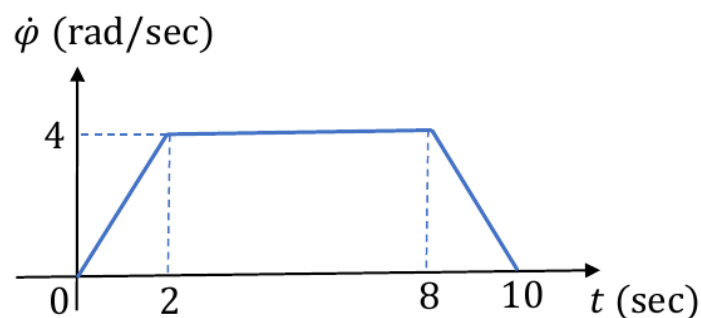
Program berjalan dalam loop utama hingga tombol "esc" ditekan.

2.1 Pengecekan Input Keyboard

- Jika "esc" ditekan, program keluar dari loop dan berhenti.
- Jika tombol 'w', 'a', 's', 'd', atau ' ' ditekan, program:
 - Memeriksa apakah tombol yang ditekan berbeda dari tombol sebelumnya untuk menghindari input berulang.
 - Menyimpan waktu saat tombol pertama kali ditekan (time_hysteresis_start).
 - Menentukan nilai target kecepatan linear dan angular menggunakan fungsi velocityKey().

2.2 Perhitungan Integral Velocity (Gerak Lurus Berubah Beraturan - GLBB)

$$V_t = V_t + V_0 + a dt$$



- Program menggunakan rumus GLBB untuk menyesuaikan kecepatan secara bertahap agar tidak berubah secara tiba-tiba.
- Kecepatan diperbarui berdasarkan percepatan maksimum yang diperbolehkan:
 - Jika target lebih besar, nilai kecepatan bertambah dengan batasan percepatan.
 - Jika target lebih kecil, nilai kecepatan berkurang dengan batasan percepatan.

Nama : Hernanda Achmad Priyatna

NRP : 5022221114

- Kecepatan dibatasi agar tidak melebihi nilai maksimum yang telah ditentukan (CONST_MAX_LINEAR dan CONST_MAX_ANGULAR).

2.3 Konversi Kecepatan Linear ke Kecepatan Roda (Inverse Kinematic)

- Fungsi `inverseKinematic()` digunakan untuk mengubah kecepatan linear dan angular menjadi kecepatan masing-masing roda berdasarkan model differential drive.
- Kecepatan roda kanan dan kiri kemudian dikirim ke simulator menggunakan `setRobotMotion()`.

2.4 Delay (Update Periodik)

- Program menunggu selama `CONST_DT` detik sebelum mengulang proses.
- Ini memastikan data diperbarui secara periodik agar simulasi berjalan stabil.

3. Menghentikan Simulasi

- Jika tombol "esc" ditekan, program keluar dari loop utama.
- Simulasi dihentikan menggunakan `sim.stopSimulation()`.
- Pesan "Program ended" dicetak sebagai tanda bahwa program selesai.