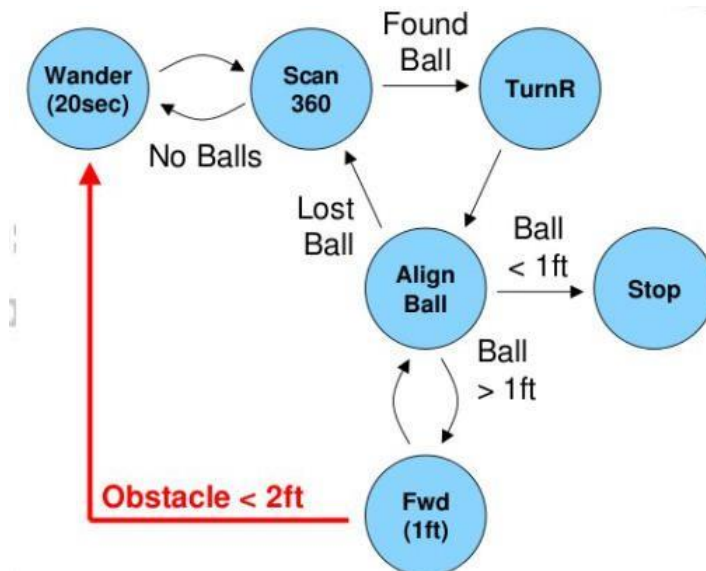


### A. Pseudo-code sistem control

Berikut ini Adalah bentuk desain diagram dan pseudocode(kode semu) pada robot



Gambar 1 Sistem kontrol high-level

Pseudo-code dalam struktur java:

```
wander(float duration_inSecond){
    float time_start = current_time();
    while((current_time-time_start) < duration_inSecond){
        switch(sensor_front){ //obstacle handler
            case "wall":
                if(sensor_right & sensor_left="wall"){
                    turn_right(180);
                }
                else if(sensor_right="wall" & sensor_left="nothing"){
                    turn_left(90);
                    follow_wall(right);
                }
                else if(sensor_right="nothing" & sensor_left="wall"){
                    turn_right(90);
                    follow_wall(left);
                }
                else{
                    turn_right(180);
                }
                break;

            case "no_obstacle":
                forward(1);
                break;
        }
    }
}
```

```
scan(float degree){
    float now = current_degree();
    while((current_degree()-now) < degree){
        cam.scan_ball();
        cam.turn_clockwise(45);
    }
}

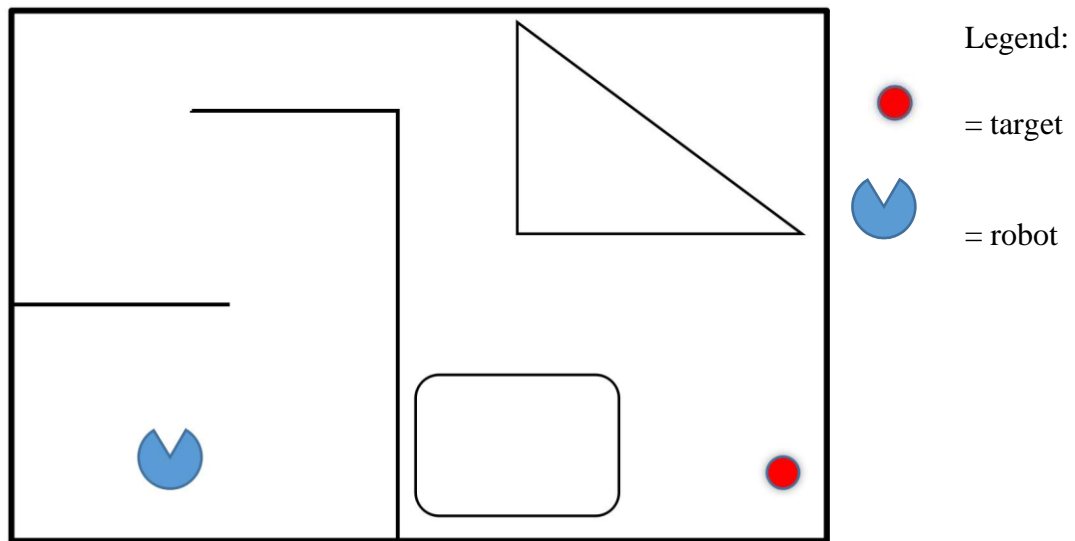
Main loop { //main function
    if(cam.stat=="ballNotFound"){
        wander(20);
        scan(360);
    }
    else{ //cam.stat == ballFound           while(cam.statAlign ==
FALSE){
        turn_right(15);
        scan(360);
    }
    if(ball_distance < 1){
        stop();
    }
    else{
        forward(1);
    }
}
}
```

## B. Sensor yang digunakan

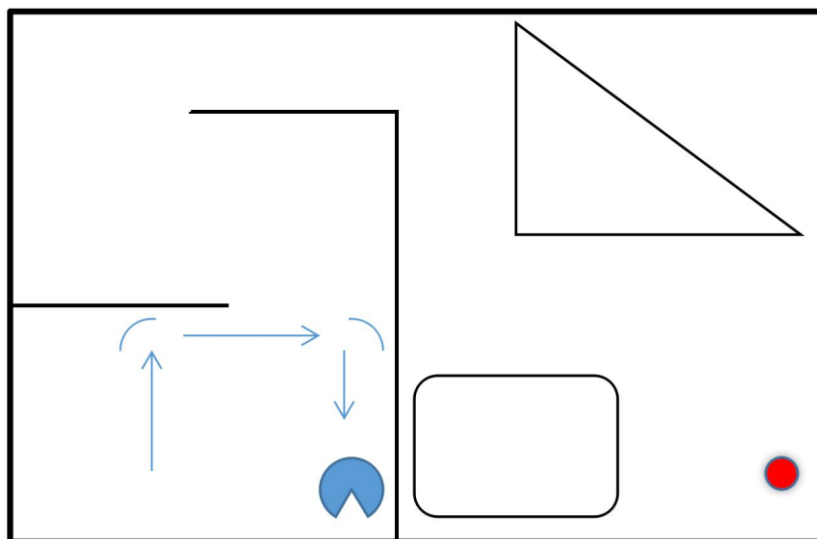
Sensor yang digunakan untuk mendeteksi halangan atau obstacle adalah sensor infrared. Sensor akan mengirimkan sinyal jika ke robot jika ada halangan di depan agar robot tidak tertabrak halangan. Sedangkan untuk mendeteksi bola berwarna merah menggunakan sensor TCS 3200. Sensor ini mampu mendeteksi warna Red-Green-Blue (RGB). Warna yang dikirim akan sesuai dengan warna sinyal yang diterima oleh sensor. Robot juga dilengkapi kemampuan untuk mengenali citra atau objek melalui tangkapan gambar

### C. Simulasi high level control

Playground:

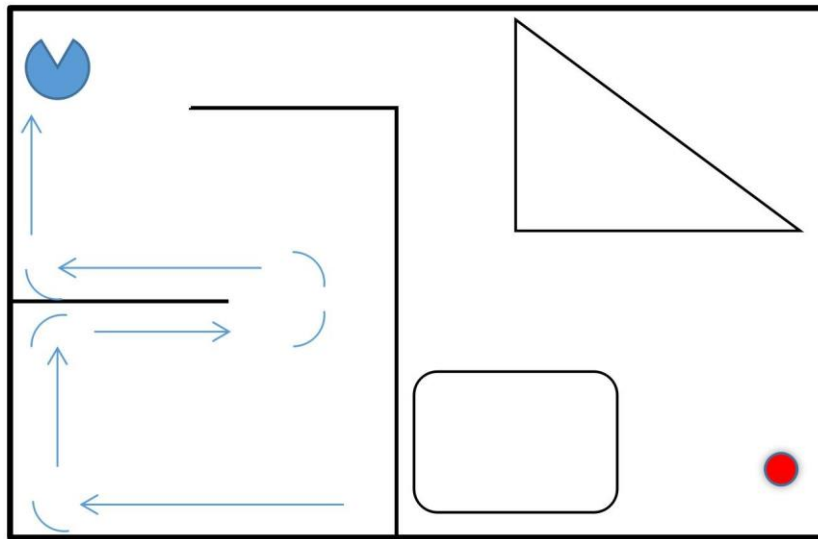


Robot menjalankan fungsi scan 360 derajat. cam.stat = “ballNotFound”, Wander 20 sekon.



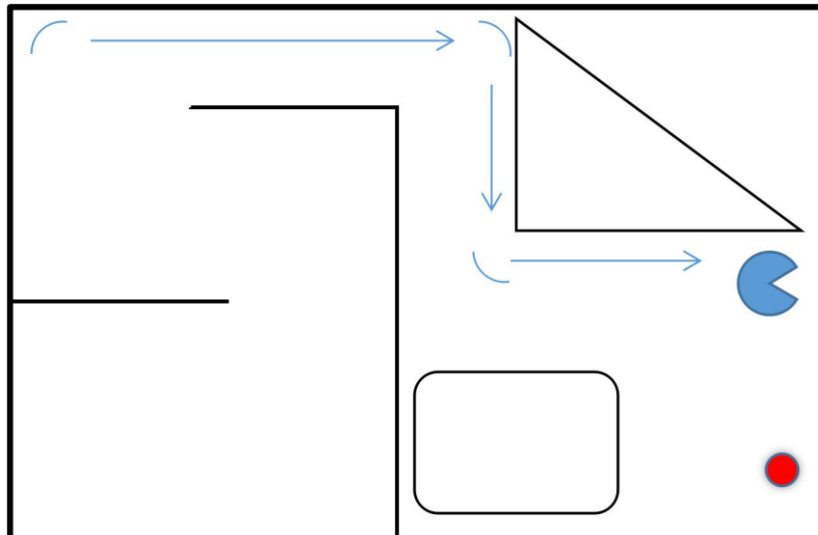
Karena robot tidak mendeteksi adanya obstacle pada sensor depan, robot akan maju 1 ft. Robot mendeteksi adanya obstacle pada hanya pada sensor depan, turn\_right 90 derajat. Lalu maju 1 ft karena tidak ada halangan pada sensor depan. Turn\_right pada dinding selanjutnya. Dan maju 1 ft karena tidak ada obstacle.

20 seconds elapsed, robot menjalankan fungsi scan 360 derajat. cam.stat = “ballNotFound”, wander 20 sekon.



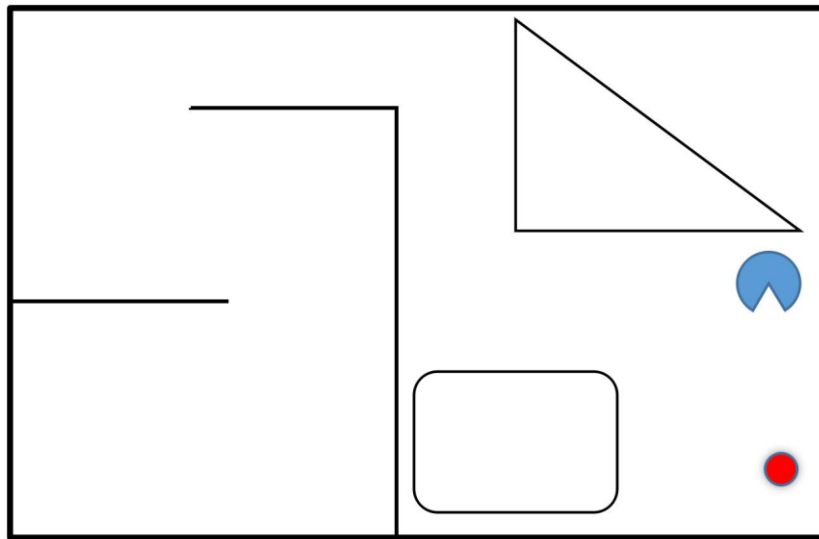
Robot mendeteksi adanya obstacle pada sensor depan dan kiri, yang menyebabkan robot menjalankan fungsi `turn_right(90)` dan `follow_wall(left)`. Robot akan terus mengikuti dinding pada kiri robot menggunakan sensor kiri sebagai basis untuk navigasi.

20 seconds elapsed, robot menjalankan fungsi `scan 360 derajat`. `cam.stat = "ballNotFound"`, wander 20 sekon.

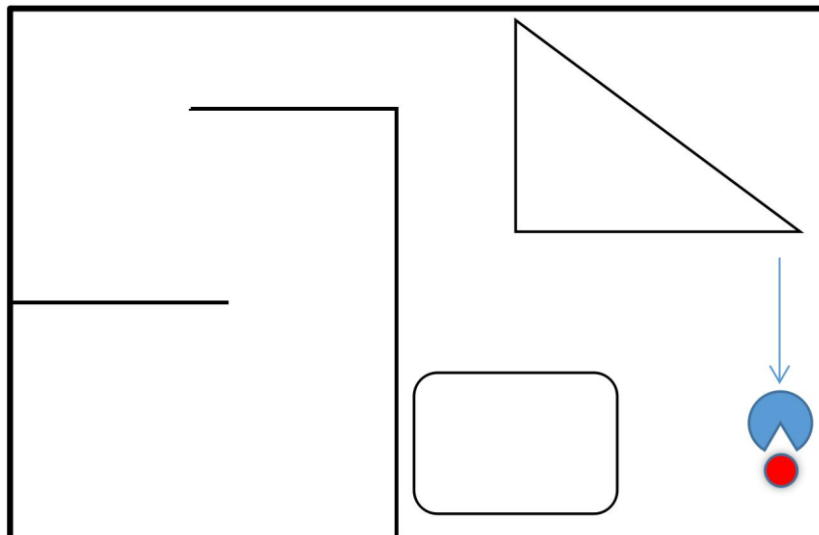


Robot mendeteksi obstacle pada sensor depan dan kiri, fungsi `turn_right` dan `follow_wall(right)` dijalankan.

20 seconds elapsed, robot menjalankan fungsi `scan 360 derajat`. `cam.stat = "ballFound"`. `cam.statAlign = false`.



Robot menjalankan fungsi `turn_right(15)` hingga `cam.statAlign = true`.



Setelah `cam.statAlign = true`, robot akan memeriksa jarak bola menggunakan algoritma robo-vision. Karena jarak  $> 1$  ft, robot akan maju 1 ft. Setelah jarak bola dengan robot  $< 1$  ft, robot berhenti.