Smart Obstacle Avoidance

Time required: 60 minutes

By combining 90 degree turns and ultrasonic sensor readings, your robot can determine which way to go when it senses an obstacle. This program uses more blocks (functions) and a Boolean variable to track whether an obstacle has been detected or not.

A Boolean variable is either true or false. mBlock doesn't have Boolean variables, we use a 0 for false or 1 for true.

A Boolean variable (also known as a flag) keeps track of the state of the mBot, it allows the mBot to "remember" something for later use. This program will remember whether an obstacle has been detected or not.

Tutorial Assignment

1. Start mBlock.
2. Open **Calibrate Distance and Square**
3. Save the program as **Smart Obstacle Avoidance**.
4. Complete and test the program as pictured with the requirements listed.

Requirements

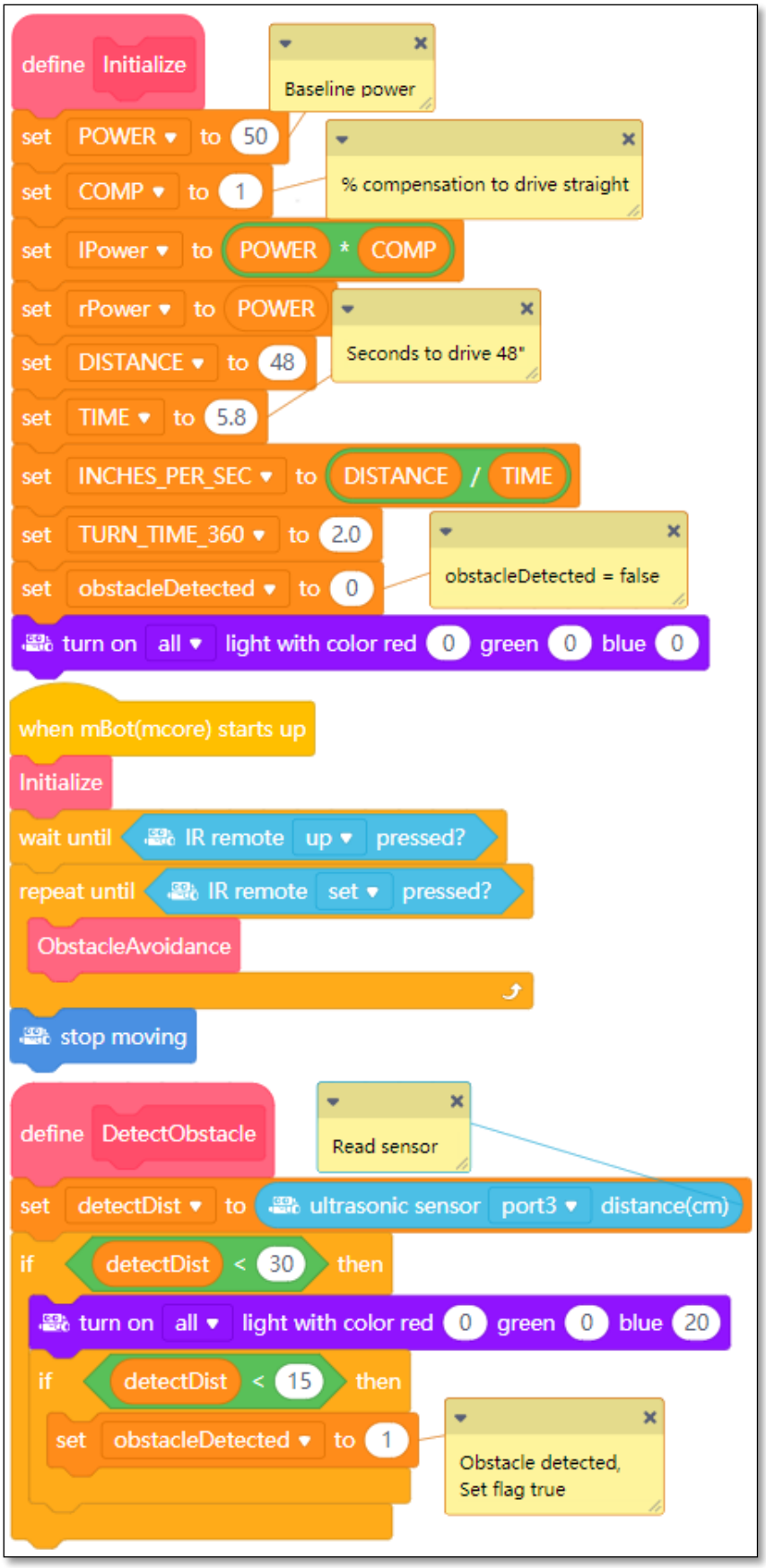
- When it detects an obstacle: turn right, take a sensor reading, turn left, take a sensor reading. Turn the robot in the direction that has the longest distance.
- Use the accurate turn and movement programs created earlier to make turns and movement more accurate.
- Test the obstacle avoidance with your foot.

Assignment

Start with your tutorial project and add the following.

- Setup a maze with available objects, see if your robot can navigate through the maze.
- When the robot moves forward, if there is an obstacle ahead (e.g. 50 cm away), the robot will be alerted and turn on an alarm light and/or very short sound.

- As an optional challenge: As the obstacle gets closer, the short alarm sound and light frequency will gradually accelerate until the robot turns away.


define Initialize
Baseline power
set POWER to 50
set COMP to 1
% compensation to drive straight
set IPower to POWER * COMP
set rPower to POWER
set DISTANCE to 48
Seconds to drive 48"
set TIME to 5.8
set INCHES_PER_SEC to DISTANCE / TIME
set TURN_TIME_360 to 2.0
set obstacleDetected to 0
obstacleDetected = false
turn on all light with color red 0 green 0 blue 0
when mBot(mcore) starts up
Initialize
wait until IR remote up pressed?
repeat until IR remote set pressed?
ObstacleAvoidance
stop moving
define DetectObstacle
Read sensor
set detectDist to ultrasonic sensor port3 distance(cm)
if detectDist < 30 then
turn on all light with color red 0 green 0 blue 20
if detectDist < 15 then
set obstacleDetected to 1
Obstacle detected,
Set flag true

define TurnRight degrees
set drvTime to degrees / 360 * TURN_TIME_360
left wheel turns at power lPower %, right wheel at power rPower * -1 %
wait drvTime seconds
stop moving
define TurnLeft degrees
set drvTime to degrees / 360 * TURN_TIME_360
left wheel turns at power lPower * -1 %, right wheel at power rPower %
wait drvTime seconds
stop moving
define ObstacleAvoidance
turn on all light with color red 0 green 20 blue 0
left wheel turns at power lPower %, right wheel at power rPower %
DetectObstacle
if obstacleDetected = 1 then
AvoidObstacle
define LookLeft
Find the distance to the obstacle on the left
TurnLeft 90
wait .2 seconds
set detectL to ultrasonic sensor port3 distance(cm)
define AvoidObstacle
turn on all light with color red 20 green 0 blue 0
LookLeft
LookRight
If L > R, Go left
if detectL > detectR then
TurnLeft 180
define LookRight
Find the distance to the obstacle on the right
TurnRight 180
wait .2 seconds
set detectR to ultrasonic sensor port3 distance(cm)
set obstacleDetected to 0
turn on all light with color red 0 green 20 blue 0
Obstacle avoided, set flag false

Assignment Submission

- **All students** → Attach finished programs to the assignment in Blackboard.
- **In class assignment submission** → Demonstrate in person.
- **Online submission** → A link to a YouTube video recording showing the assignment placed in the submission area in BlackBoard.