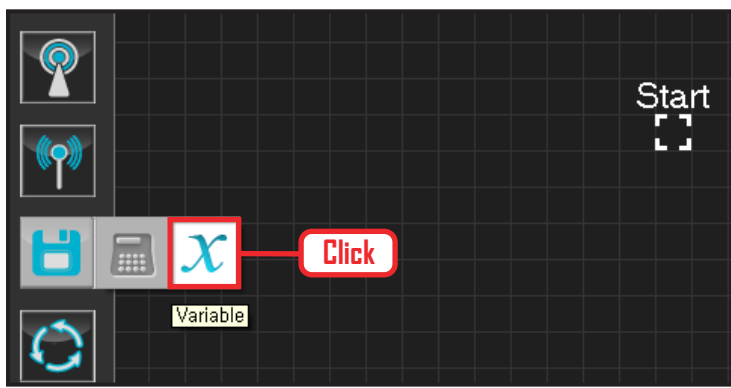


Analog Distance Sensor Example Step by Step

Example Description

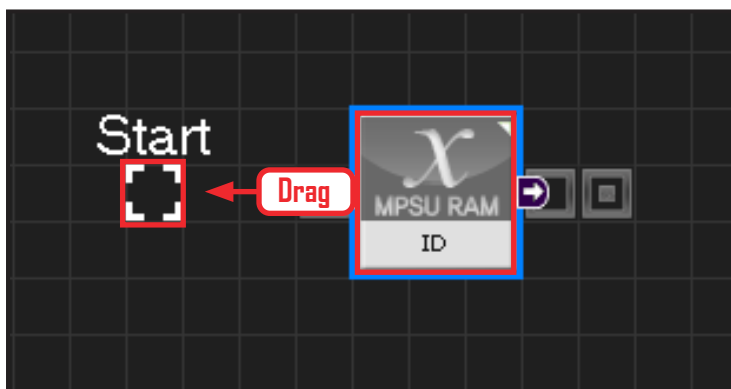
This example program is an obstacle avoidance program that uses analog sensor to make the robot avoid hitting an obstacle by turning to the left. Hovis Lite has two type of distance sensors, analog and digital. Digital sensor uses specific distance (10cm) as a reference and it can only determine if an object is within or beyond the reference range. On the other hand, analog sensor is capable of detecting an object within 6~80cm range. This example requires PSD sensor to be installed at ADC port #1 (left).



01 Assign Variable

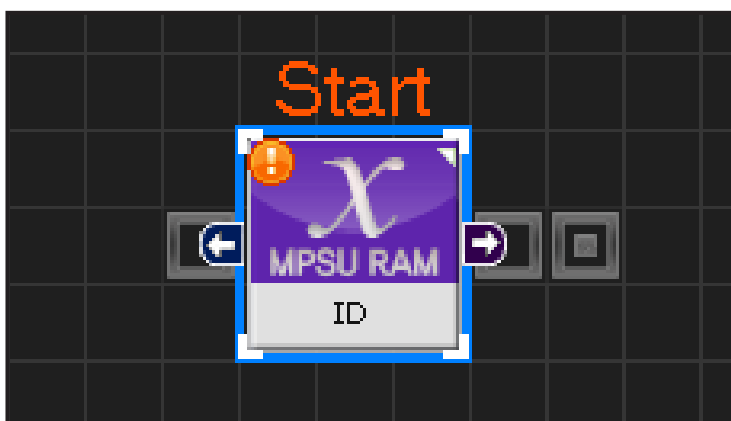
Operating the robot is same as operating the robot servo motor. Value has to be assigned so that servo will be able to operate.

Click Data › Variable module.



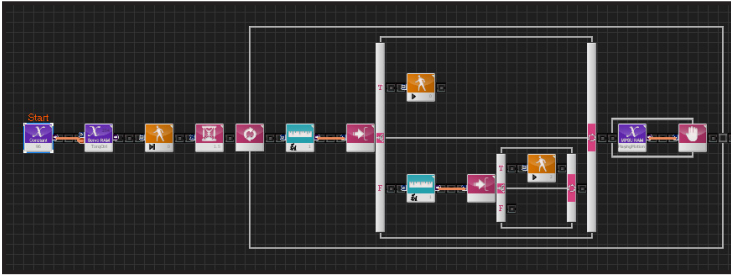
02 Start

Click and drag the connecting line located at left side of the module to the Start Point and dock



03 Start Programming

When the module and the Start Point is docked properly, module will become active and change color as seen in the photo to the left. This means programming has started..



04 Entire Program

Entire program using the analog sensor to make the robot avoid hitting an obstacle.

C-like Graphic

```

1 void main()
2 {
3     SERVO_TorqCtrl[254]
4     motionready( 0 )
5     delay( 1500 )
6     while( true )
7     {
8         if( ( MPSU_ADCType1 == 1 && MPSU_ADCVal1 >= 20 ) )
9         {
10            motion( 0 )
11        }
12        else
13        {
14            if( ( MPSU_ADCType1 == 1 && MPSU_ADCVal1 < 20 ) )
15            {
16                motion( 2 )
17            }
18            else
19            {
20            }

```

05 Viewing C-Like

Click the 'C-like' tab near the top right and task programming window will open as shown in the photo to the left. This is the task window of the entire program. Codes are very similar to the C language structure so studying the codes will help the user become familiar with the C language structure. Cursor will jump following the clicked module, making it easy to see the module changing to text.

1 Click Start

2 Select Constant

3 Input

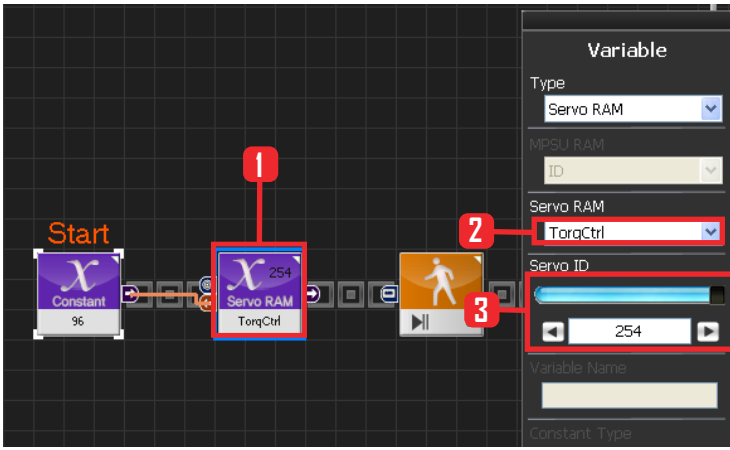
Variable
Type: Constant
MPSU RAM ID: [dropdown]
Servo RAM ID: [dropdown]
Servo ID: [input field]
Variable Name: [input field]
Constant Type: Bool Int
Constant Value: True False
[slider]
[input field: 96]

06 Setup Constant

This section allows the servo motor to operate on it's own.

Select Constant as the Variable Type. In properties, set constant value as 96.

When 96(0x60) is entered in the servo TorqControl register, servo becomes ready to operate. This value is sent to the torque value of the next moduel through the output connector.



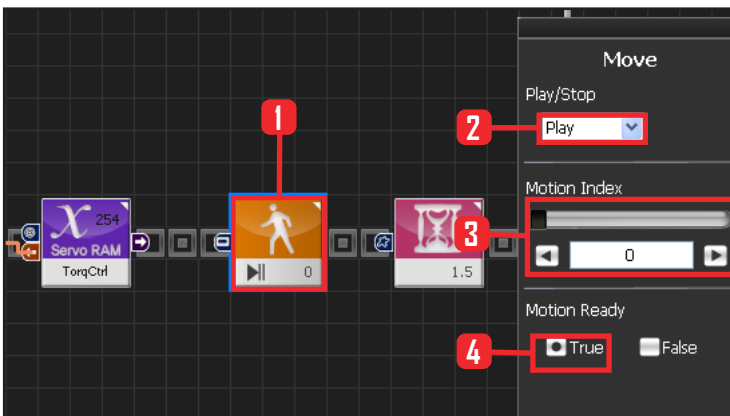
07 Apply to All Servos

This section applies contact value 96 to all servos.

Select Variable > Type : Servo RAM.

Select Servo RAM : TorqCtrl .

Set Servo ID : 254, 254 means it will be applied to all connected servos.



08 Motion Ready

Robot goes through a preparatory stage before starting the next motion. This preparatory stage allows the robot to move slowly to the the initial position of the motion to be run. This prevents stress or damage from sudden change in motion. IF Motion Ready is True prepare for next motion. Run next motion if False.

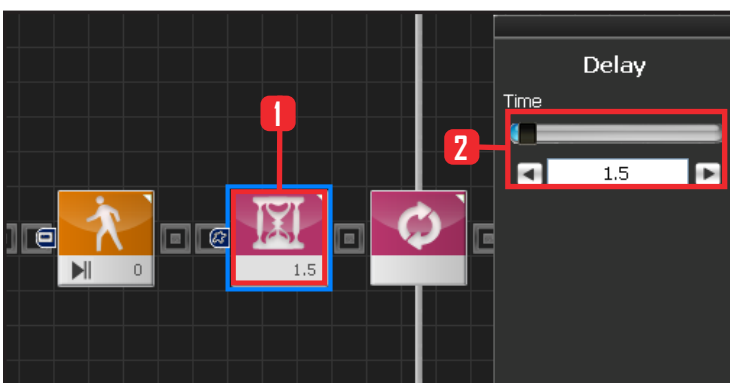
Select Motion > Move module.

Select Play/Stop : Play .

Set Motion Index : 0 . walk forward

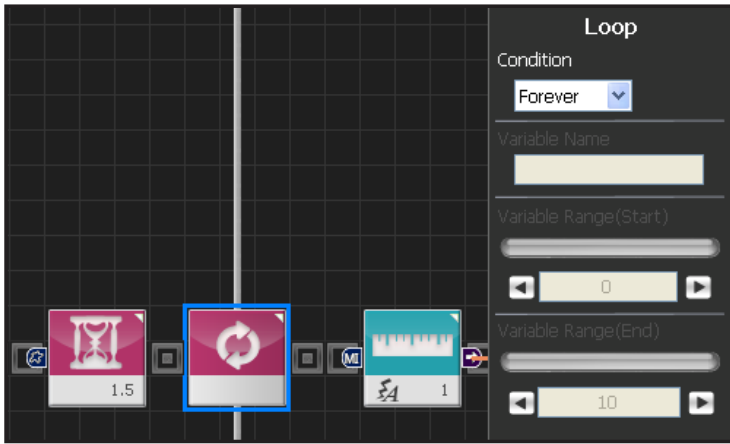
Select Motion Ready : True .

Motion Ready Stage



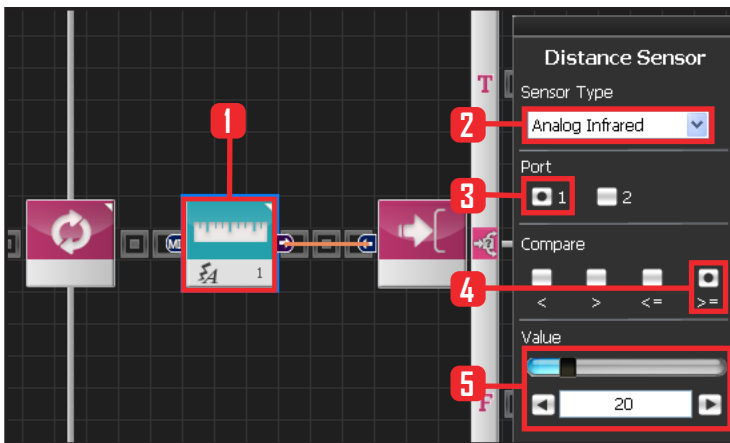
09 Delay

Set delay to 1,5s to prevent next step from starting before Motion Ready ends.



10 Loop

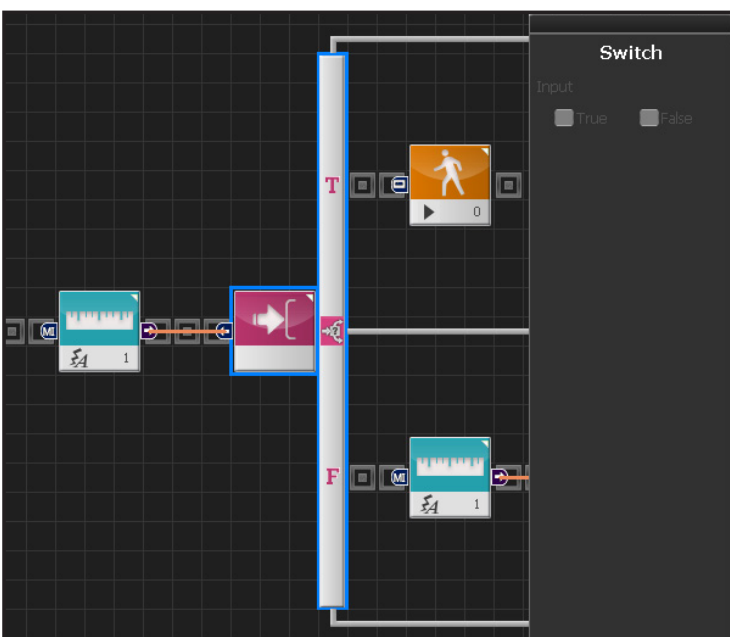
Select Loop: Forever
Infinite loop.



11 Setup Analog Sensor

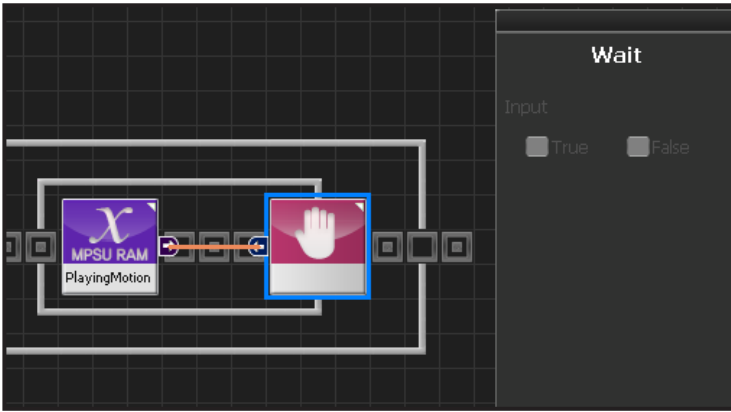
Setup with 20cm as standard.

Select Sensor > Distance Sensor module.
Select Sensor Type : Analog Infrared
Select Port : 1.
Select Compare : >= . True if equal to or greater than standard.
Set Value : 20 , 20cm .



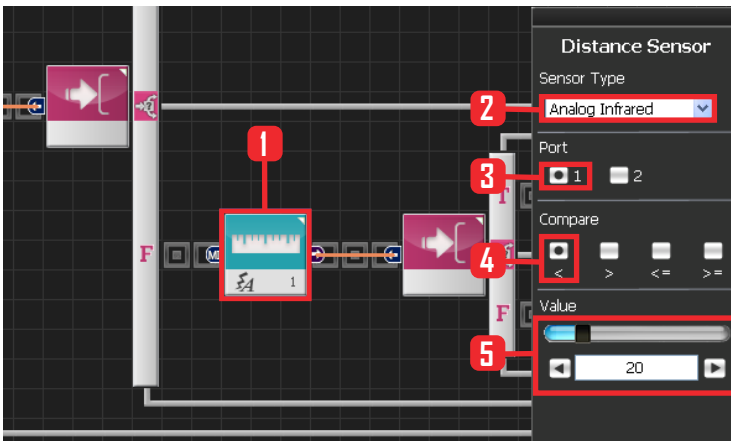
12 If Conditional Statement

Proceed if True or go to the next conditional statement if False.



13 Forward

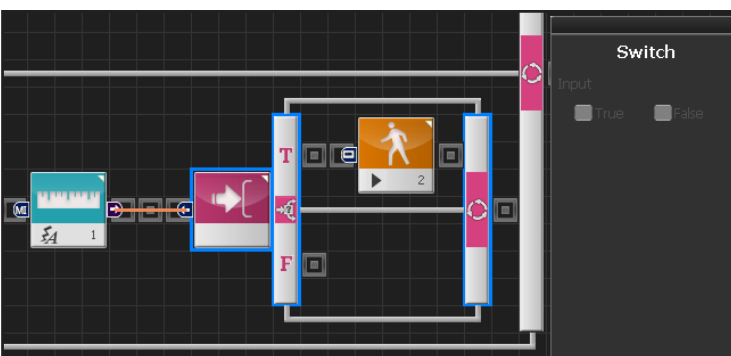
Robot will move forward since the distance is farther than 20cm.
When False is selected as Motion Ready value, robot will proceed with forward motion.



14 Motion Near The Wall

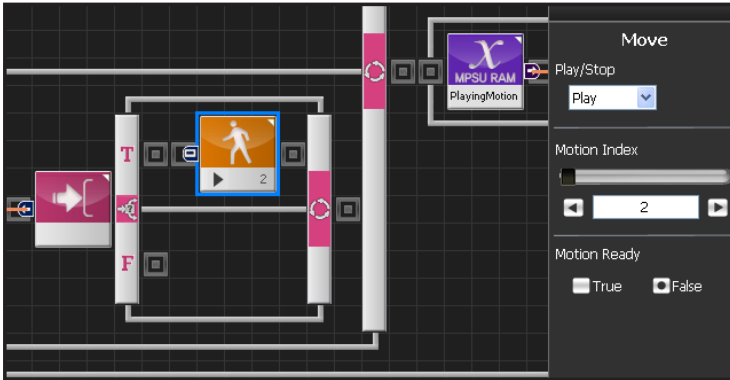
Robot will make a left turn if it detects an obstacle within 20cm.

Select Sensor > Distance Sensor module.
Select Sensor Type : Analog Infrared .
Select Port : 1.
Select Compare : < . True if less than standard.
Value : 20 . 20cm .



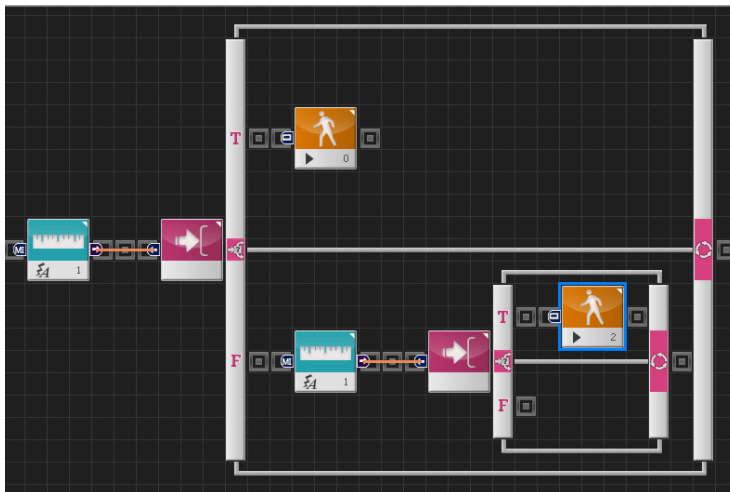
15 If Conditional Statement

Run statement within True if less than 20cm from the obstacle.



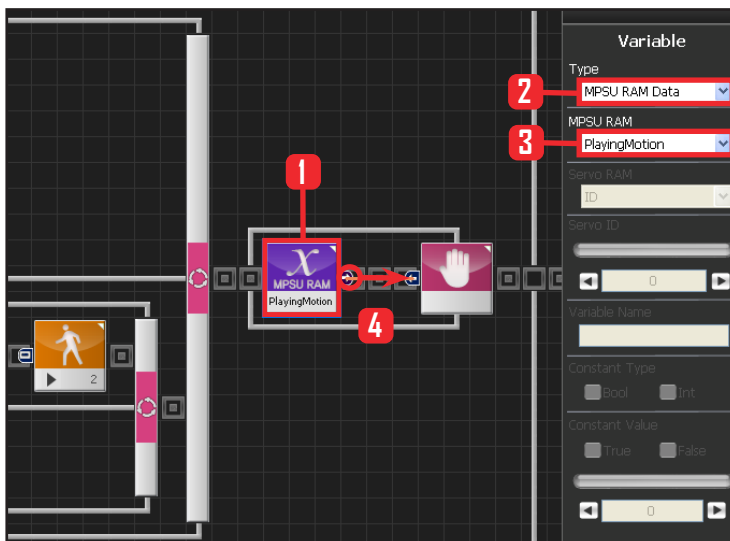
16 Left Turn

Robot motion #2 makes the robot change direction to the left. Robot will run the left turn motion if the Motion Ready value is False.



17 Motion According to Distance

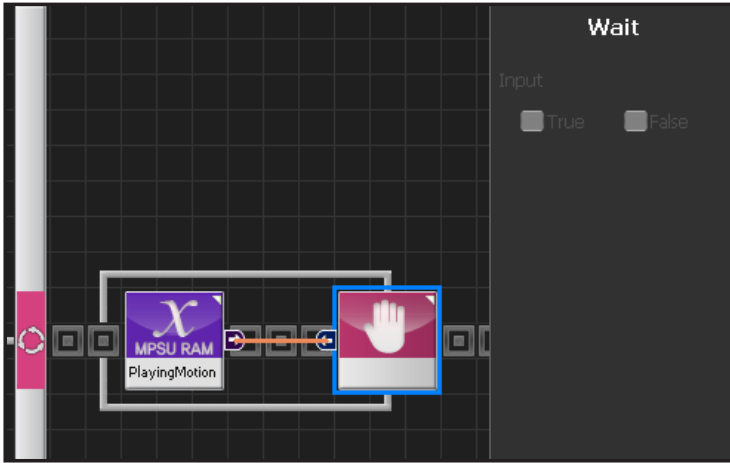
If the distance to the obstacle is greater than 20cm, robot will keep moving forward. If the distance is less than 20cm, robot will make a left turn.



18 Motion Movement Check

Loop refers to continuous repetition. It takes time for the actual motion to complete after Move command has been issued, but loop with single move module will continue to run and give motion command even while the previous motion is still running. The lag in actual motion will result in difference between the number of motion commands given by the move module and the number of actual motions. To correct this difference, loop will need to wait for the motion to complete before repeating the process. 'Playing Motion' is found within Variable > MPSU RAM Data. 'Playing Motion' is a variable that checks whether the motion is in process. Loop will wait for the current motion to end if 'wait' is added to the 'Playing Motion'.

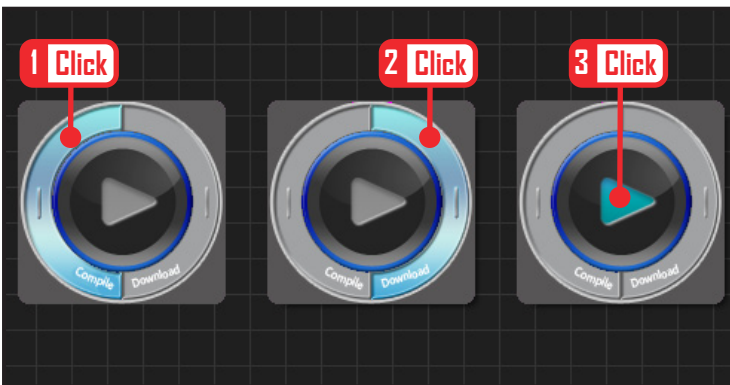
- Select Data > Variable Module.
- Select Type : MPSU RAM Data
- Select MPSU RAM : Playing Motion
- Add Wait module to the output connector.



19 Wait

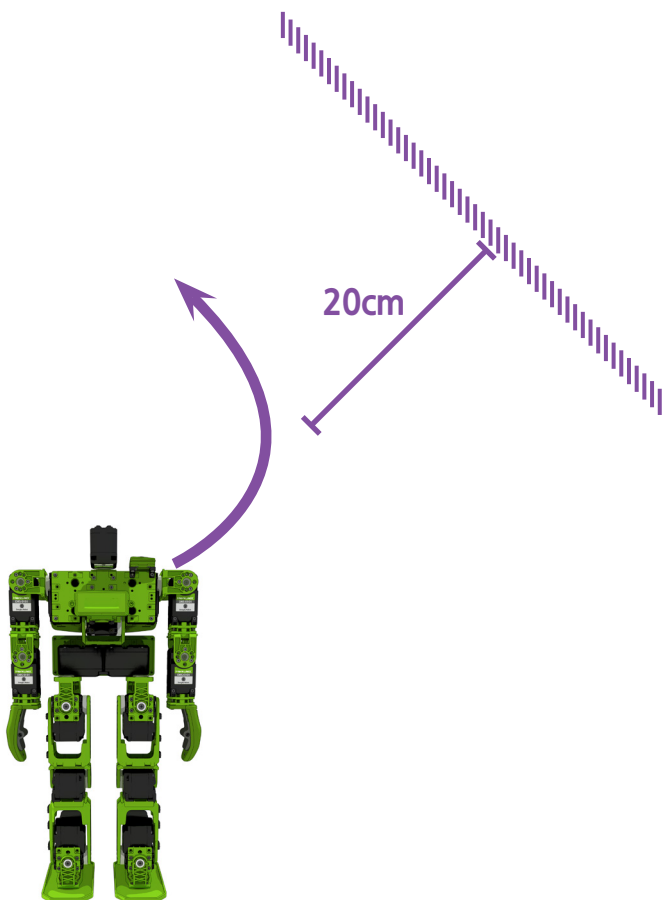
Wait until the motion ends.

Go to the beginning and repeat when motion ends.



20 Compile, Download, Run

Click 'Compile'. Click 'download' on the right if there is no compilation error. Download to robot, Click 'Run' button (Arrow button) after the download..



21 Robot Motion

Robot will walk forward and then make a left turn if it detects an obstacle within 20cm distance.