Simple Movement

Time required: 60 minutes

Please read all the directions carefully before beginning the assignment.

- Comment your code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.

Understanding

Demonstrate understanding of:

delay, setup, loop, if statement

Knowledge Points

This program shows how to control the motors. The program doesn't execute until the remote button up is pressed.

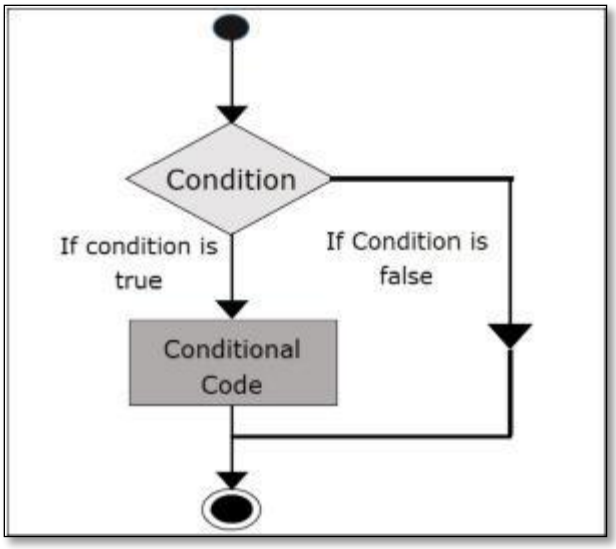
An **if** statement is a control structure. The **if** statement checks for a condition and executes the following statement or set of statements if the condition is 'true'.

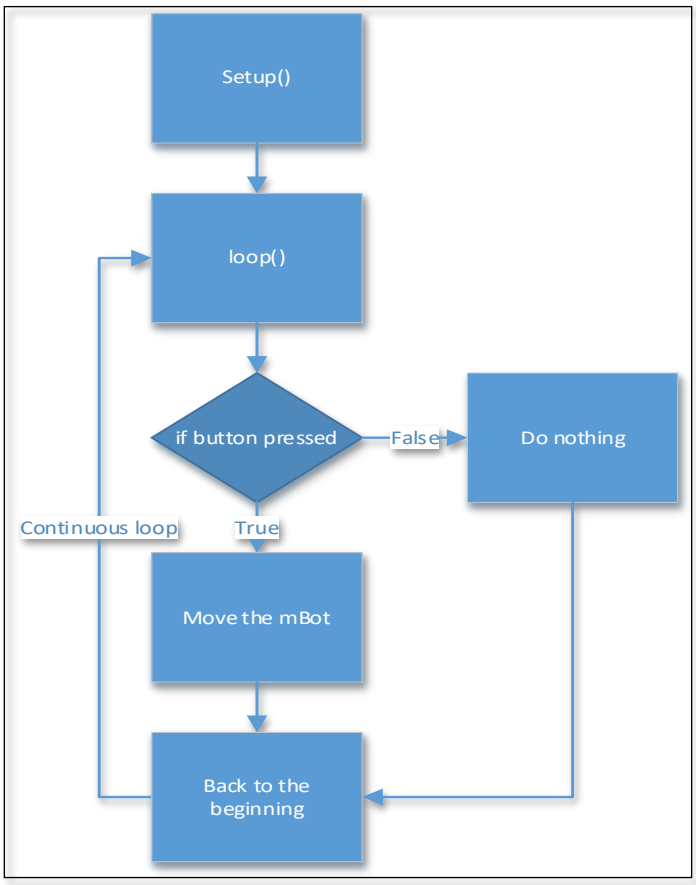
The == is the equal to comparison operator. Equal to (==)returns true if the value on the left is equal to the value on the right, otherwise it returns false. This is also called a Boolean condition.

Syntax

```
if (condition == true) {
  // do stuff if the condition is true
}
```

This program uses an **if** control structure as shown in the diagrams.





Requirements

Complete and successfully run the program as shown.

Tutorial Assignment

1. Start the Arduino IDE. Create a new sketch called **SimpleMovement**.

2. Complete and test the program as shown.

```
/**
  @file    SimpleMovement.ino
  @author  William A Loring
  @version V1.0.1
  @date revised 02/05/2020  created: 12/10/2016
  @Description: Sample code for mBot movement
*/
#include <MeMCore.h>  // Include mBot library
MeIR ir;              // Setup IR Remote object

// Create motor control objects
MeDCMotor MotorL(M1); // MotorL is Left Motor
MeDCMotor MotorR(M2); // MotorR is Right Motor

void setup() {
  ir.begin(); // Begin listening for the ir remote
}

// Loop until the up remote button is pressed
void loop() {
  // If the up remote button is pressed, the mBot moves!
  if (ir.keyPressed(IR_BUTTON_UP)) {
    // motor.run() speed range is 255 to -255, 0 is stop, 127 is 50%
    // Move forward with 50% motor speed
    MotorL.run(-127); // MotorL (Left)  forward is -negative
    MotorR.run(127);  // MotorR (Right) forward is +positive
    delay(1000);      // Delay in milliseconds, motor keeps running

    // Move backward with 127 actual motor speed, which is 50%
    MotorL.run(127);  // MotorL (Left)  backward is +positive
    MotorR.run(-127); // MotorR (Right) backward is -negative
    delay(1000);

    MotorL.stop();    // Stop MotorL
    MotorR.stop();    // Stop MotorR
  }
}
```

Assignment

Start with your tutorial project and add the following.

1. Add different movements to the program.

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.