

Getting Started with mBot™

Computer and Chromebook™

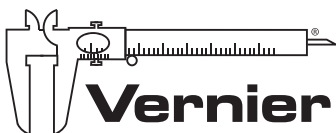
Contents

About mBot
Construct mBot
mBot's Default Program
mBlock Software
Programming mBot: 3 Easy Steps
First mBot Program: Simple Obstacle Avoidance
Additional Resources

About mBot

mBot is an affordable STEM robot kit designed to help bring computer science education into the real world. mBot is made by Makeblock®, a leading DIY-robotics-construction and STEM-education platform for students, educators, and hobbyists. At the heart of mBot is the mCore, a microcontroller based on the Arduino™ Uno. mCore has four sensor ports and two motor ports. It also includes two RGB LEDs, a Buzzer, a Light Sensor, an IR transmitter and receiver, and an Onboard Button. Two of mBot's sensors, the Ultrasonic Sensor and the Line-Follower Sensor, are connected to mCore's sensor ports.

mBot comes loaded with a Default Program that has three modes: Mode A–Manual Mode, Mode B–Obstacle Avoidance, and Mode–C Line Following. In addition to these three modes, mBot can run a program uploaded directly to the robot via a USB cable. You can write these programs using mBlock software running on a Windows® or macOS® computer or a Chromebook.



PIONEERING TECHNOLOGIES
FOR **STEM** SINCE 1981

Vernier Software & Technology
13979 SW Millikan Way
Beaverton, OR 97005-2886

www.vernier.com
engineering@vernier.com

Vernier and caliper design are our registered trademarks. mBot and mBlock are trademarks of Makeblock. All other marks not owned by us that appear herein are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by us.

Released: 6.2018

Construct mBot

To construct mBot, follow the building instructions included with mBot and keep these tips in mind:

- Construction typically requires 30 minutes.
- The only tool required is the included screwdriver.
- To power mBot, you need either four (4) AA batteries or the optional 3.7 V rechargeable battery; batteries are not included with mBot.
- The IR Remote Control requires a CR2025 coin cell battery (not included). The remote is used to direct mBot.

mBot's Default Program

mBot comes pre-loaded with a Default Program that has three distinct modes.

Mode A—Manual Drive In this mode, you can drive the robot forward or backward and turn it right or left by pressing the arrow keys on the IR Remote Control. Numbered keys correspond to sounds. In Mode A, the LEDs are white.

Mode B—Obstacle Avoidance In this mode, mBot drives forward until it detects an obstacle, such as a wall or chair, with the Ultrasonic Sensor. Then, mBot backs up, turns, and resumes driving forward. In Mode B, the LEDs are green.

Mode C—Line Following In this mode, mBot will follow a black line while driving forward. It keeps track of the black line using the Line-Follower Sensor, mounted to the front of the mBot. In Mode C, the LEDs are blue.

mBot starts off in Mode A. There are two ways to change modes:

- Select the mode by pressing the **A**, **B**, or **C** button on the IR Remote Control. Make sure there is a battery in the remote!
- Press the Onboard Button on mBot.

RETURN MBOT TO THE DEFAULT PROGRAM

You can always use the mBlock software to return mBot to the Default Program, even after you have uploaded programs to mBot:

1. Turn on mBot and connect mBot to a computer or Chromebook with a USB cable (included with mBot).
2. Open the mBlock software. To connect to mBot, choose Serial Port from the Connect menu and select the COM (USB) port to which mBot is connected (usually the last on the list).
3. Choose Reset Default Program > mBot from the Connect menu. mBlock will upload the Default Program to mBot. You can then disconnect mBot and use the modes of the Default Program.

mBlock Software

mBot is compatible with a variety of programming software platforms, including the Arduino IDE. In this guide, you will use mBlock software, created specifically for mBot.

There are two versions of mBlock software; one version works on computers and Chromebooks and one version works on iOS and Android™ devices. While the two versions have the same title, they are noticeably different. A separate *Getting Started with mBot* guide is available for iOS/Android. See www.vernier.com/mbot

Download mBlock Software

You can download mBlock for computers/Chromebook from the Vernier website:

www.vernier.com/mblock

Install Arduino Drivers

The first time you use mBlock, you need to install the Arduino drivers on mBot. To do this, connect mBot to your computer or Chromebook with a USB cable. Choose Install Arduino Driver from the Connect menu in mBlock. This will install the software driver necessary for communicating between the computer or Chromebook and mBot.

Note If you are using mBlock on a computer running macOS 10.12 (Sierra) or newer, you need to update the Arduino driver. See www.vernier.com/til/4124

Programming in mBlock

mBlock is based on Scratch 2.0, the graphical programming language developed by MIT's Lifelong Learning Lab. If you are familiar with Scratch, you will find mBlock easy to use. However, even if this is your first time using a graphical programming language, you can pick it up quickly.

In mBlock, you can write programs to do many things:

- Move Sprites on the Stage to tell a story.
- Design your own video games or music videos.
- Drive mBot through a maze.
- Direct mBot to avoid obstacles or follow a black line.


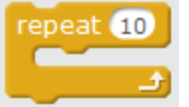
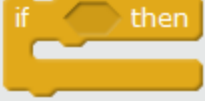
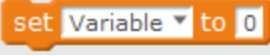




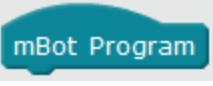

The mBlock window is divided into sections or panes:

- Stage
- Sprites
- Scripts
- Code
- Arduino (available only in Arduino Mode)

Programs that involve mBlock's Stage and Sprites can use commands in all of the Scripts categories (e.g., Motion or Looks). Coding with mBlock's Stage and Sprites is just like coding with these commands in Scratch.

Palettes and Blocks

When programming in mBlock, you will build your program out of blocks. Blocks are organized into different palettes, which are color coded. Most of the programs you write for mBot will use the block from the blue-green Robots palette.

Palette	Description and use	Example blocks
Control	Guide the flow of your code	  
Data&Blocks	Create variables or define your own commands/blocks	 
Operators	Math and logic commands	  
Robots	mBot-specific commands	 

Programming mBot: 3 Easy Steps

The general process for programming mBot is

1. Open mBlock software and start Arduino mode.
2. Write your mBot program.
3. Upload the program to mBot and try it out.

Step 1 Open mBlock and Start Arduino Mode

To start coding, open mBlock and start Arduino mode by choosing Arduino mode from the Edit menu.

When you are in Arduino mode, a new pane appears on the right side of the mBlock window; in this pane, mBlock translates your mBlock code into Arduino code (see Figure 1).

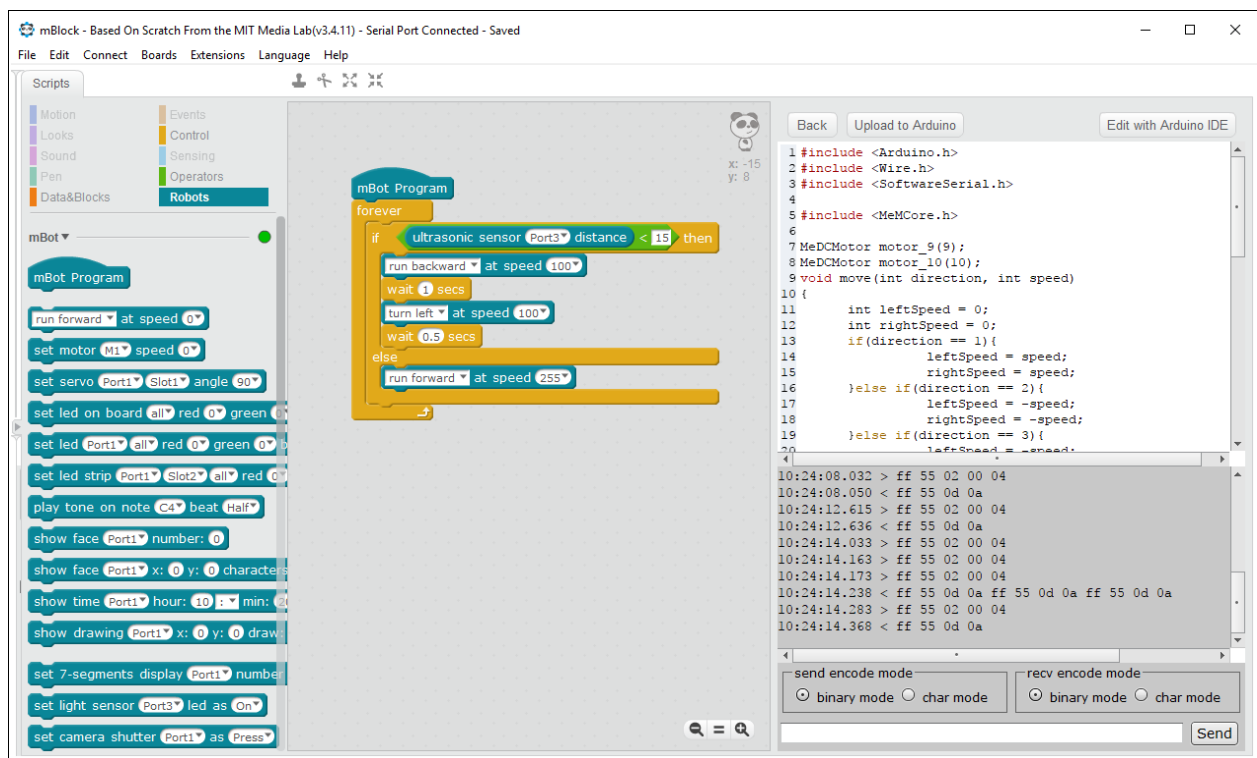


FIGURE 1 mBlock in Arduino Mode. Note that mBot is connected to mBlock: “Serial Port Connected” appears in the mBlock window title and the status circle in the Robots palette is green.

Note that not all commands and blocks are available for use in Arduino mode. Because your program will be uploaded and run on mBot, commands that utilize the Scratch stage, such as the “motion” and “looks” blocks, are not allowed and will be grayed out. In some cases, there are similar blocks in the Robot section that you can use in place of commands/blocks in the grayed out sections; for instance, the timer blocks in the Sensing palette (grayed out) are also in the Robots palette.

Step 2 Write Your mBot Program

Creating a program for mBot is simple: drag and drop commands or blocks into the code pane (the middle pane in mBlock). When you create programs in mBlock for mBot, always start with an “mBot Program” hat block (found in the Robots palette).

In Arduino mode, as you move blocks in the code pane (middle pane), mBlock translates the code into C, the native language of the Arduino micro-controller that is inside mBot. This C code appears in the Arduino pane of the mBlock window. This code is there so you can see what the C programming language is like; you do not need to understand it or do anything with it.

Step 3 Upload and Test your Program

When you are ready to test your program on mBot, it is time to connect to mBot and upload the program.

- a. Use a USB cable (included with mBot) to connect mBot to your computer or Chromebook.
- b. To connect to mBot, choose Serial Port from the Connect menu and select the serial port to which mBot is connected.
- c. Click the Upload to Arduino button in the Arduino pane. During the uploading process, you will see various communication text in the pane beneath the Arduino code.

Once the upload is finished, you will see a message telling you the upload is complete. The program will start running as soon as the program has uploaded.

Note: You can restart your program at any time by power-cycling mBot or pressing the Reset button on mBot.

TIP: IDENTIFY MBOT'S SERIAL PORT

If you are not sure which Serial port mBot is connected to, do the following:

1. Turn off mBot. Look at the list of Serial ports available in the Connect menu in mBlock.
2. Dismiss the Connect menu by hovering or clicking in a different part of the window. Turn on mBot.
3. Look at the list of serial ports available in the Connect menu. mBlock will have updated the list of available serial ports; you will notice that a new port is available. This is mBot's port.

First mBot Program: Simple Obstacle Avoidance

The sample code in Figure 2 is for a simple obstacle-avoidance routine. mBot waits until the Onboard button is pressed, then drives forward at full speed. If the Ultrasonic Sensor detects something closer than 15 cm, mBot backs up slowly for 1 second, then turns left. If the obstacle is no longer detected, mBot resumes driving forward; otherwise, mBot backs up and turns left again.

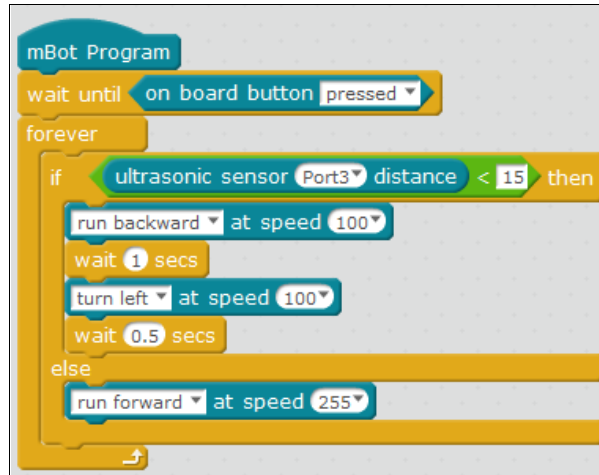


FIGURE 2 Obstacle Avoidance program

Build the Obstacle Avoidance program in mBlock and upload the code to mBot. Does mBot move as expected? Experiment with the code by doing the following:

- Increase or decrease the driving or turning speed.
- Increase or decrease the “wait” durations following the “run backward” or “turn left” blocks.
- Change the distance at which an obstacle is detected from 15 cm to greater distance.
- Add commands so that sometimes mBot turns left to avoid an obstacle and other times it turns right.

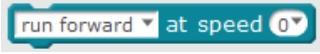
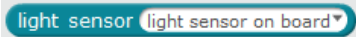


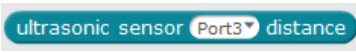

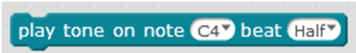

Improving your Program

As you change your code in the middle pane in mBlock, the corresponding code in the Arduino pane updates automatically. When you are ready to test out your new program, follow these steps:

1. Use the USB cable to connect mBot to your computer or Chromebook.
2. To connect to mBot in mBlock, choose Serial Port from the Connect menu and select the appropriate serial port.
3. Click the Upload to Arduino button in the Arduino pane. When the upload is complete, disconnect the USB cable and test your new program.

Using mBot Sensors and Outputs in Your Code

mBot has a variety of sensors and output devices that can be incorporated into your mBot program. The Robots palette includes blocks for each of these devices.

Sensor/ Output Device	Block	Notes
Motors		<ul style="list-style-type: none"> Connected to M1 (left motor) and M2 (right motor) Choose direction (e.g., run forward or turn right) and speed (0–255) Use "wait" block to define how long motors run
Light Sensor		<ul style="list-style-type: none"> Built into mCore Reports light level on a scale of 0–1000
IR receiver		<ul style="list-style-type: none"> Built into mCore Conditional statement to recognize when buttons on the IR remote are pressed
Line-Follower Sensor		<ul style="list-style-type: none"> Sensor module, attached to Port 2 via RJ25 cable Reports a value (0–3) corresponding to where the sensor is relative to a line
Ultrasonic Sensor		<ul style="list-style-type: none"> Sensor module, attached to Port 3 via RJ25 cable Reports the distance (in centimeters) to the closest object in front of the
RGB LEDs		<ul style="list-style-type: none"> Two RGB LEDs are built into mCore Choose RGB values (each 0 to 255) to set color for each LED
Buzzer		<ul style="list-style-type: none"> Built into mCore Choose note (e.g., C4) and beat (e.g., half)
Onboard Button		<ul style="list-style-type: none"> Built into mCore Boolean reporter (true/false) When combined with a "wait until" block, can be very useful for starting mBot programs

Additional Resources

Activities and technical support can be found on Vernier web page for mBot (www.vernier.com/mBot):

- *Coding with mBot: Self-Driving Vehicles*, a STEM module with nine student activities and detailed Instructor Information (order code: MBOT-MSDV-E)
- mBot Troubleshooting and FAQs

These two books offer an in-depth dive into coding with mBot and include a variety of activities:

- *mBot for Makers: Conceive, Construct, and Code Your Own Robots at Home or in the Classroom* by Andrew Carle and Rick Schertle
- *A Gentle Introduction to Robotics: Volume 1: mBlock and the mBot* by Charles McKnight