

# Getting Started with mBot™

iOS and Android™

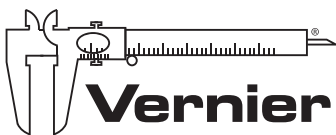
## Contents

About mBot  
Construct mBot  
mBot's Default Program  
Overview: Makeblock® and mBlock™ Blockly Apps  
Makeblock App: Play and Learn  
mBlock Blockly App: Programming mBot  
Additional Resources

## About mBot

mBot is an affordable STEM robot kit designed to help bring computer science education into the real world. mBot was developed 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.



PIONEERING TECHNOLOGIES  
FOR **STEM** SINCE 1981

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

[www.vernier.com](http://www.vernier.com)  
[engineering@vernier.com](mailto: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 or embedded in the Makeblock app.

- 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 included with mBot is used to control an mBot running the Default Program; the Default Program comes pre-loaded on mBot. The remote requires a CR2025 coin cell battery (not included).

## mBot's Default Program

mBot comes pre-loaded with a Default Program that has three distinct modes. Use the IR Remote Control to direct mBot when mBot is running the Default Program.

**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, using 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. mBot keeps track of the black line using the Line-Follower Sensor that is mounted on the front of the robot. 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.

### TROUBLESHOOTING: CONNECTING TO MBOT

#### Refresh the Connection

If you have trouble connecting to mBot in the iOS or Android apps, try refreshing the connection by turning mBot off and on with its Power Switch.

#### Return mBot to the Default Program

If you used mBlock software on a computer or Chromebook™ to upload programs to mBot, you may receive a message in the iOS or Android app such as "Device not supported, or firmware not detected" when you try to connect to your mBot. If you see such a warning, visit [vernier.com/til/4139](https://www.vernier.com/til/4139) for information about how to reload the Default Program on mBot.

For additional trouble shooting tips and FAQs, visit [vernier.com/mBot](https://www.vernier.com/mBot)

## Overview: Makeblock and mBlock Blockly Apps

This guide offers an introduction to two apps that can be used to control and program mBot: Makeblock and mBlock Blockly. These apps are free and are available for iOS and Android. The same apps are also needed for the activities in the iOS/Android version of the *Coding with mBot: Self-Driving Vehicles* module.

### Makeblock App

The Makeblock app contains many predefined actions, making it easy for students to direct mBot without having to write a program. With the Makeblock app, you can use your iOS or Android device as a remote control to direct mBot's actions and create simple paths that mBot follows autonomously.

### mBlock Blockly App

The mBlock Blockly app allows you to program your own actions for mBot. Programming in mBlock Blockly is similar to using Scratch 2.0, the graphical programming language developed by MIT's Lifelong Learning Lab.

#### Tip: mBlock app

The app, mBlock, is available for iOS, Android, Windows®, macOS®, and Chrome™. The apps have similar names but are quite different.

If you are using mBlock on computers or Chromebooks, a separate *Getting Started with mBot* guide is available. Visit [www.vernier.com/mbot](http://www.vernier.com/mbot)

## Makeblock App: Play and Learn

The Makeblock app can be used with several robots, including mBot and mBot Ranger. Using the Makeblock app, students can learn about mBot without coding.

### Getting Started with Makeblock and mBot

Before using the Makeblock app to control mBot, you must connect your mobile device to mBot via Bluetooth® wireless technology.

1. Make sure mBot is turned off.
2. Launch the Makeblock app and swipe through the different robot options. Select mBot to open the mBot control panel with menu options including Play, Code, and Build.
3. Turn on mBot with its onboard Power Switch.
4. Tap the Bluetooth connection icon in the control panel (see Figure 1). Hold your mobile device close to your mBot, and the robot will connect automatically. If you have trouble connecting mBot, try refreshing the connection by turning mBot off and back on again.

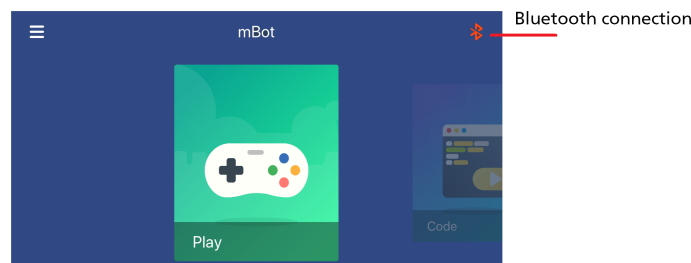


FIGURE 1

## Makeblock Control Panel for mBot

The Makeblock control panel for mBot has several options. Three are particularly useful when getting started:

**Play** Use your iOS or Android device as a remote control for mBot. Students explore Play mode in the first activities in the *Coding with mBot: Self-Driving Vehicles* module.

**Code** Open the mBlock Blockly app or be directed to the app store.

**Build** View step-by-step animated instructions for how to build mBot.

## Play with mBot

In Play mode, you can explore some of the characteristics of mBot. Tap Play in the control panel (see Figure 1) to view the options.

### Drive

Use your iOS/Android device as a remote control device to drive mBot forward and backward and turn right or left. You can also turn on the headlights, beep the horn, run fast, spin around, and rotate back and forth.

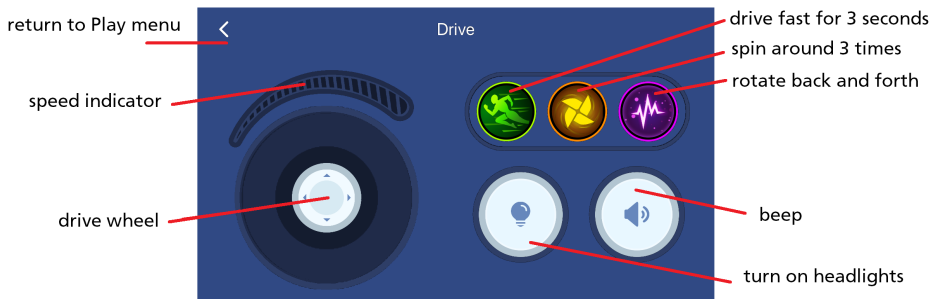


FIGURE 2

### Draw and Run

This mode is a favorite. Draw a line on your mobile device, and mBot will drive in this path.

### Musician

In this mode your mobile device becomes a musical keyboard. As you play a song, you'll hear the sound coming from mBot. This mode includes several pre-recorded songs, including "Happy Birthday" and "Twinkle, Twinkle Little Star."

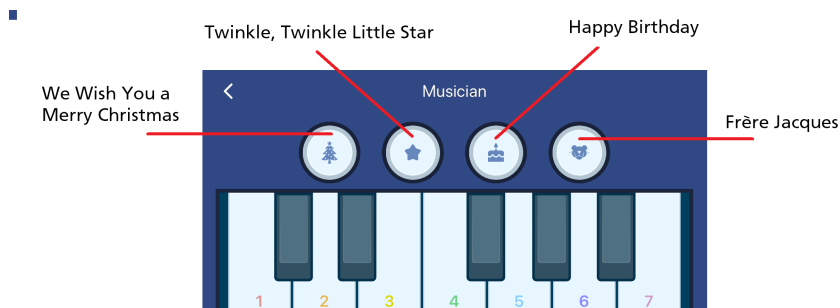


FIGURE 3

### Voice Control

Use simple voice commands to control mBot's motion.

## SWITCHING BETWEEN APPS

The Makeblock and mBlock Blockly apps require a unique Bluetooth wireless connection between your mBot and your mobile device. You should always turn off your mBot before opening either of the apps to ensure you are making a fresh connection.

## mBlock Blockly App: Programming mBot

Coding with the mBlock Blockly app is very similar to coding with Scratch; both programs use block-style coding elements, which are organized by function. The primary difference between Scratch and mBlock Blockly is that you control your robot in the real world with mBlock rather than manipulating a sprite on the sprite stage, as you do in Scratch.

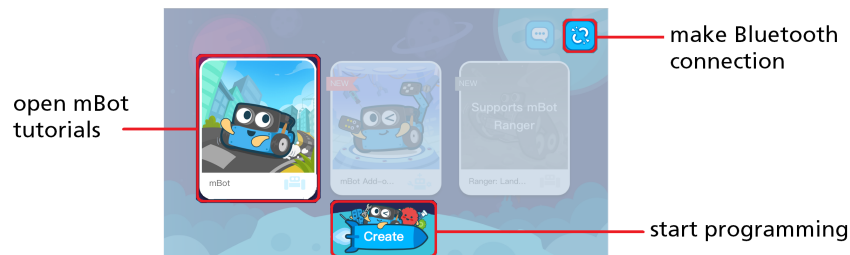


FIGURE 4

## Getting Started with mBlock Blockly and mBot

Before using mBlock Blockly with mBot, you must connect mBot to your mobile device via Bluetooth.

1. Make sure mBot is turned off.
2. Launch the mBlock Blockly app.
3. Turn on mBot with its onboard Power Switch.
4. Tap the Bluetooth connection icon in the control panel (see Figure 4). Hold your mobile device close to your mBot and the robot will connect automatically. If you have trouble connecting mBot, try refreshing the connection by turning mBot off and on.

## MBOT TUTORIALS

If you or your students are new to coding, we recommend starting with the mBot tutorials.

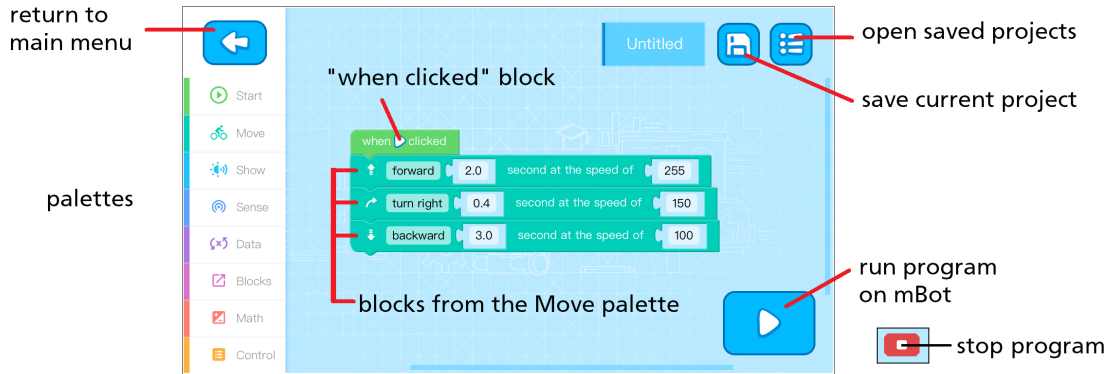
Select mBot from the main screen (see Figure 4) to access more than 60 guides covering topics ranging from the basics of using mBot's built-in sensors to more advanced programming commands.

Consider assigning specific tutorials to students and giving students time to develop familiarity with mBot's movement and its sensor before moving on to programming.


## Programming mBot with mBlock Blockly

When you are ready to write your own programs for mBot, select Create from the main screen (see Figure 4). This opens the programming platform (see Figure 5).

In the left sidebar, there are eight palettes containing blocks. To start, tap a palette and explore the list of available blocks. To add a block to the programming area, tap or drag the block. Once blocks are in the programming area, drag them together. As the blocks get close together, they snap together. Always start your program with a "when clicked" block from the Start palette.




**FIGURE 5** The programming platform in mBlock Blockly for iOS and Android



As you are coding, save your project by tapping Save, . Give your project a unique name, and save your project frequently. Saved projects are available each time you open mBlock Blockly.

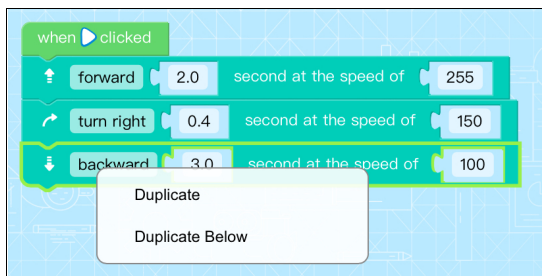
You can quickly edit your program:

**Duplicate a block** Tap and hold a block. Choose to duplicate the specific block or the block and everything below the block.

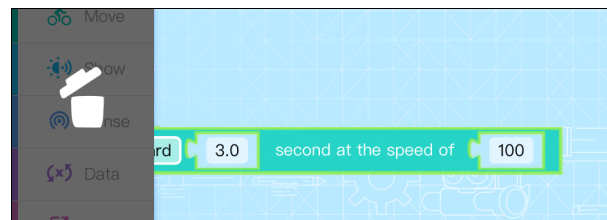
**Delete a block** Drag a block toward the left sidebar. The sidebar turns into a trash can; drop the block into the trash can. Note that when you pull a block out of the program, it will also carry the code that is below the block. To delete a single block, first separate the block (and code below) from the program. Then, tap and drag the specific block to the trash can, deleting the block. Finally, re-insert the remaining code into the program.

When you are ready to test your program, tap Run, . The program will be sent to mBot via Bluetooth and will begin running.

After tapping , notice that the icon changes to , which can be used to stop the program.




**FIGURE 6** Tap and hold a block to duplicate



**FIGURE 7** Drag a block toward the palettes to delete

## First mBot Program: Simple Obstacle Avoidance

The sample code in Figure 8 is a simple obstacle-avoidance program. The program starts when you tap . mBot drives forward at a speed of 150 (out of 255). If the Ultrasonic Sensor detects an obstacle, mBot backs up slowly for a second, then turns left for half a second. If the obstacle is no longer detected, mBot resumes driving forward; otherwise, mBot backs up and turns left again.

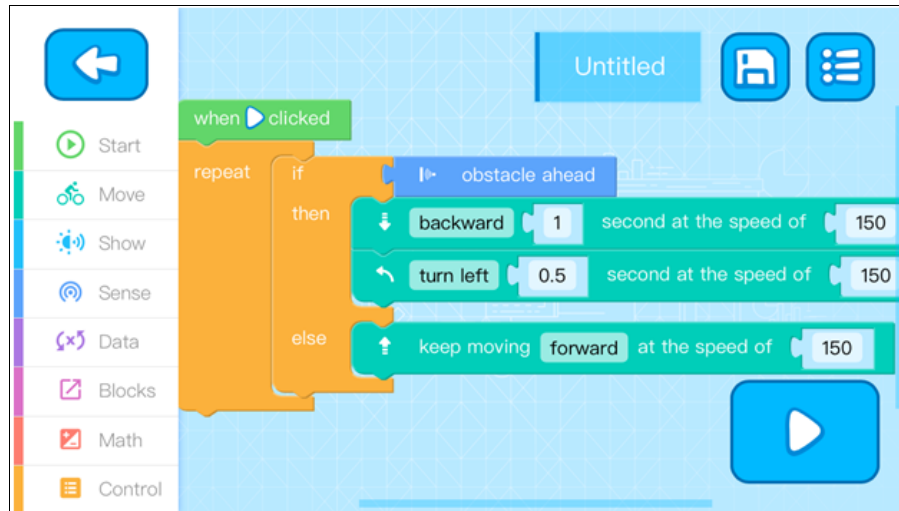


FIGURE 8 Obstacle Avoidance program

Build the Obstacle Avoidance program in the mBlock Blockly app and send the program to mBot. Does mBot move as expected? If not, examine the code to see if anything is missing. Continue troubleshooting your code until mBot can successfully move around an obstacle in its path.

### Develop Your Coding Skills

Now you can experiment the code by adjusting blocks or adding new blocks and increasing the functionality of your program. Try the following or come up with your own ideas:

- 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 to so mBot sometimes turns right rather than left to avoid an obstacle.



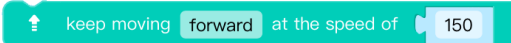


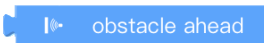


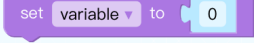
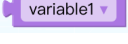
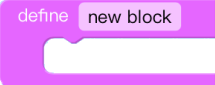
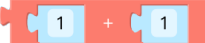

## Additional Resources

In the Vernier STEM module, *Coding with mBot: Self-Driving Vehicles*, students learn to program mBot to perform many of the actions that are required of autonomous vehicles. *Coding with mBot: Self-Driving Vehicles* is available for Windows, macOS, Chrome, iOS, and Android. Activities are available for students from elementary to high school. More information is available at [www.vernier.com/mBot](http://www.vernier.com/mBot)

*mBot for Makers: Conceive, Construct, and Code Your Own Robots at Home or in the Classroom*, by Andrew Carle and Rick Schertle, also offers an in-depth exploration of coding with mBot.

## Reference Table: mBlock Blockly Palettes

The mBlock Blockly palettes give you the power to increase the complexity of your programs. For example, incorporate blocks from the Sense palette to use the Ultrasonic Sensor to detect objects that are blocking the way.

Palette	Description and use	Example blocks
Start	Initiate a program. Every mBot program must start with a "when clicked" block.	
Move	Control motors and motion	 
Show	Control the Buzzer and LEDs	 
Sense	Read from sensors such as the Ultrasonic Sensor (measures distance to an object)	 
Data	Create and define variables	  
Blocks	Create and define your own blocks	
Math	Incorporate math and logic commands	 
Control	Guide the flow of the code	