

mBot Educational Robot Unit

Lessons and Resources for the mBot



Developed by:

Lisa Galpin and Laini Ferriell

lisa.galpin@tcnschools.org

lferriell@mvctc.com

Gaming Research Integration for Learning Laboratory™ (GRILL™) Summer 2019

TABLE OF CONTENTS

Table of Contents.....	2
1. mBbot	3
1.1. Introduction	3
1.2. Materials	3
1.3. Overview of Plan	3
1.3.1. Pacing	3
1.3.2. Student Handouts	4
1.4. mBot Project Lessons	4
1.4.1. Introduce Background Knowledge of “The Boring Stuff” (Day 1)	4
1.4.2. Introduce Background Knowledge of Sprite (Day 2-3)	5
1.4.3. Learning About Blocks (Day 4).....	5
1.4.4. Simple Device Programs (Day 5-9)	5
1.4.5. Sensor Device Programs (Day 10-12)	6
1.4.6. mBot Race Challenge (Day 13-15)	7
1.5. Rubric For Mbot Autonomous Rover	7
1.6. Resources	8

1. MBBOT

Primary Resources: mBlock software and mBot Educational Kit

1.1. INTRODUCTION

By using mBots, students can learn to program and create an autonomous rover. An educational mBot kit provides students with the opportunity to use the design process to learn about the basic parts of a robot and develop basic coding skills, in order to create an autonomous rover. Students also learn the basic thought process necessary in the field of computer science.

The mBlock software, a simple drag and drop block programming language, will be used to program the mBot. Students learn to combine various blocks to program the robot to perform different tasks. Once a student has mastered the block coding, the software can be switched to the Arduino programming language which is an easy transition to more complex programming languages.

1.2. MATERIALS

- Computer
- Student handouts
- Writing utensils
- mBot kit
- mBlock software
- Internet access

1.3. OVERVIEW OF PLAN

Unit Goal: Create and race an autonomous rover using a single mBot kit.

mBot Challenge Problem: Program your mBot to navigate a race course autonomously. The robot to navigate the furthest in three minutes wins the race.

Key Vocabulary: mBot, mBlock, sprite, device, block, code, script, ultrasonic, accuracy

1.3.1. PACING

- Day 1: Build background knowledge of robots
- Days 2-3: Background knowledge of sprites
- Day 4: Background knowledge on mBlocks
- Days 5-9: Simple programming of mBot
- Days 10-12: Sensor programming
- Days 13-15: Race Days

1.3.2. STUDENT HANDOUTS

- 1.4.1.A Sense, Plan, Act Diagram
- 1.4.1.B Robot or Not-a-Robot?
- 1.4.2 Create a Sprite Checklist
- 1.4.3 Guided Notes on mBlock Shapes
- 1.4.4 Simple Programming Checklist
- 1.4.5 Sensor Programming Checklist

1.4. MBOT PROJECT LESSONS

1.4.1. INTRODUCE BACKGROUND KNOWLEDGE OF “THE BORING STUFF” (DAY 1)

Resources for lesson: supplies for making a peanut butter and jelly sandwich, 1.4.1.A and 1.4.1.B Handouts

Goal: I can explain what makes a robot a robot: it is able to sense, plan, and act.

Assess Prior Knowledge: Challenge the students to tell you how to make a peanut butter and jelly sandwich. Do only what they instruct you to do. The goal is for the students to realize that they must tell you to do every single step in order for you to make an edible sandwich.

New Knowledge: Teach students the characteristics of a robot by going over student handout 1.4.1.A. Robots are able to sense, plan, and act. Show the video, *How Robots Sense, Plan, and Act* (<https://www.youtube.com/watch?v=tUs2nyDcls8>) and discuss what it means for a robot to sense, plan, and act. Next, show students the *Sense Think Act Board* video (<https://www.youtube.com/watch?v=Jy2u08vwox4>) and explain how the board works.

Finally, show students the video, *History of Robotics* (<https://www.youtube.com/watch?v=uoC2ZGRI8a8>). After the video, have a quick class discussion about what students learned. Suggested questions: What are robots? How do robots work? How are robots used in the real world? What is an autonomous rover?

Application: Pass out student handout 1.4.1.B Robot or Not-a-Robot. Then have students move around the room looking at various items/pictures and classifying them: Robot or Not a Robot. Suggested pictures/items: vacuum, lawn mower, blender, coffee pot, Roomba, John Deere Tango E5 lawn mower, car, auto parking car, Mars Rover, automatic doors.

Generalization: Exit slip question- On a piece of paper have students jot down what makes a robot a robot. You are looking to see if they can tell you that robots are able to sense, plan, and act.

1.4.2. INTRODUCE BACKGROUND KNOWLEDGE OF SPRITE (DAY 2-3)

Resources for lesson: 1.4.2 Create a Sprite handout, computer

Goal: Students will learn how to create a sprite that can move around the screen and produce sound.

Assess Prior Knowledge: Put a picture of Pac Man on the board. Discuss with the students how Pac Man is considered a sprite. Talk with the students about how they think Pac Man moves and makes sound.

New Knowledge: Pass out 1.4.2 Create a Sprite handout. The purpose for this handout is for students to use tutorials to learn how to create a sprite, move a sprite, and program the sprite to make sound.

Application: 1.4.2 Create a Sprite handout. Students will follow directions for activity 3. In this activity, students will create their own sprite, make it move, and produce sound.

Generalization: Students will go around the room and will share their project.

Extension: Students can complete the tutorial, Paint Your Own Sprite.

1.4.3. LEARNING ABOUT BLOCKS (DAY 4)

Resources for lesson: 1.4.3 Guided notes on mBlock Shapes, computer, and writing utensil

Goal: I can explain the programming in a block of mBlock script.

Assess Prior Knowledge: mBlock programming blocks snap together like puzzle pieces. In small groups of 3-4, have students assemble various puzzles. To make it more challenging, do not give them the picture that they are trying to create. After students have completed the puzzle, share with the students the goal for the day and discuss how the mBlocks snap together like puzzle pieces to create a complete program.

New Knowledge: Pass out 1.4.3 Guided Notes on mBlock Shapes and explain to students the purpose of the assignment.

Application: At the end of the guided notes, students will read and write mBlock script.

Generalization: Exit Slip- Give each student a Post-it Note and have them write down one thing that stuck with them and one thing they still need to know. Students will put the completed Post-it Note on the board prior to leaving.

1.4.4. SIMPLE DEVICE PROGRAMS (DAY 5-9)

Resources for lesson: 1.4.4 Simple Device Program Checklist, computer, mBot

Teacher Goal: Students will realize their programs are not accurate. This will be a gateway for discussion about sensors.

Student Goal: I can write a script/code that will cause my mBot to move in a square and circle.

Assess Prior Knowledge: Display the quote on the board: “The expert in anything was once a beginner.” Discuss with the students what this quote means to them, and focus on how we have to start with simple programs in order to make complex programs.

New Knowledge: Pass out 1.4.4 Simple Device Program Checklist. The purpose for this handout is for students to use tutorials to learn how to write code to cause their mBot to move in a point circle, curve circle, and square.

Application: 1.4.4 Simple Device Program Checklist. Students will follow directions for the activities and apply them to their mBot. The mBot will move in a point circle, curve circle, and square.

Generalization: Exit Slip- Give each student a half sheet of paper and have them write down the main point from their discussion with the teacher. Students will hand them to the teacher as they leave the room.

1.4.5. SENSOR DEVICE PROGRAMS (DAY 10-12)

Resources for lesson: 1.4.5 Sensor Programming Checklist, computer, mBot

Goal: I can program the mBot to sense and navigate its environment its environment.

Assess Prior Knowledge: Prior to students arriving for the day, you will need to arrange your classroom so that there is an open space in the middle for the blindfolded navigation challenge. Ask for two student volunteers. Have them stand in the middle of the open space. Blindfold one student and tell them not to move until instructed to do so. Explain to the class that the two students are going to work together to safely navigate the blindfolded student across the room to the door. Remind the blindfolded student to listen carefully and only do what he/she is told to do. Give students time to complete the challenge. Monitor the activity closely for safety purposes so no one is injured.

At the completion of the activity, lead a class discussion about what happened. Here are some questions that could be asked: What worked and what didn't work? How are seeing people able to navigate a space? If you aren't able to see, what is another way one could navigate the

space? Go over the goal for the day and talk about how robots can be programmed to sense their environment and navigate the environment to perform a task.

New Knowledge: Pass out the 1.4.5 Sensor Programming Checklist to students. Go over the checklist and expectations. Give students time to ask questions and begin working.

Application: Students will complete two challenges using the ultrasonic and line following sensors.

Generalization: Students will complete their reflection questions at the end of the second challenge.

1.4.6. MBOT RACE CHALLENGE (DAY 13-15)

Goal: To program an autonomous rover to win a race against other autonomous rovers.

Prior to students coming into the classroom: The teacher will create a small obstacle course for the mBot race.

Day 1: Teacher will put students into groups. Next, the students will observe the race track and start planning out how they want to program their robot. Here are a couple of questions for the students to think about: Which sensors should be used? What would be the fastest path to program?

Day 2: Students will program their mBot and use the obstacle course to test and retest.

Day 3: Students will race against other students.

1.5. RUBRIC FOR MBOT AUTONOMOUS ROVER

Category	Need Improvement 1	Getting There 2	Accomplished 3	Score
Collaboration	Ignores and distracts others. Shows no understanding of project and has a negative attitude during work time. Argues with others and does not ask or answer any questions.	Listens respectfully and follows directions. Shows understanding of the project but sometimes argues with others. Asks and answers questions.	Listens respectfully and engages in discussion. Shows understanding of project and has a positive attitude during work time. Never argues with others. Asks and answers questions and provides	

			evidence to support answers.	
Programming	Programs are ineffective, incomplete, or incomprehensible. Basically, they don't work.	Programs are effective and can be understood.	Programs are efficient, streamlined, effective, and easily understood.	
Race Navigation	The rover is not able to navigate the course OR there are more than 5 human interactions.	The rover is able to navigate the course with 3-5 human interactions.	Rover is able to navigate the course with 1-2 human interactions.	
Reflection	Student only explains one concept: either what would be kept the same or what would be changed. Also, does not provide evidence for response.	Student explains what would be kept the same and what would be changed, but does not provide evidence for response.	Student explains what would be kept the same and what would be changed. Provides evidence for response.	
Comments:				

1.6. RESOURCES

Mbot Resources

- Construction Manual – http://cdnlab.makeblock.com/mBot%20V1.1Building-Manual-EN_D1.2.2_KD010086000%20Print_CC.pdf
- Quick Start Guide – http://cdnlab.makeblock.com/mBot%20V1.1_Quick-Start-Guide-EN_V1.0_D1.1.1_KD010087000_Print_PDF.pdf
- Getting Started: Simple Programming and Sensor Programming – <http://learn.makeblock.com/en/mbot-programming/>

- Programming resources: Create a Sprite, and Block Shapes – <http://www.mblock.cc/doc/en/start-with-scratch-programming/start-with-scratch-programming.html>
- Programming enrichment resource: Paint Your Own Sprite – <http://www.mblock.cc/doc/en/start-with-scratch-programming/draw-a-sprite.html>

Video Resources

- *History of Robotics* – <https://www.youtube.com/watch?v=u0C2ZGRI8a8>
- *How Robots Sense, Plan, and Act* – <https://www.youtube.com/watch?v=tUs2nyDcls8>
- *Sense Think Act Board* video – <https://www.youtube.com/watch?v=Jy2u08vwox4>

Note

All images included in the following handouts: Guided Notes on mBlock Shapes, Simple Device Program Checklist, and Sensor Programming Checklist are screenshots by Elaina Ferriell and Lisa Galpin taken from Makeblock Co., Ltd © 2013-2019.

<p>SENSE</p> 	<hr/> <p><i>Gather information from surroundings using sensors.</i></p> <hr/>
<p>PLAN</p> 	<hr/> <p><i>Use gathered information with programming to make a plan.</i></p> <hr/>
<p>ACT</p> 	<hr/> <p><i>Able to carry out the plan and complete a task.</i></p> <hr/>

Robot or Not-A-Robot?

Name: _____ Date: _____

Directions: Visit the various stations around the room and decide whether the image or item is a ROBOT or NOT-A-ROBOT.

Remember that robots are able to sense, plan, and act!!!

Image/Item	Robot or Not-a-Robot

Create a Sprite Checklist

Goal: I can create a sprite that can move around the screen and produce sound.

Learning Intentions

Today I will: follow an mBlock tutorial.

So that: I can learn how to write script/code for my sprite.

I will know I have this when: I can write a script/code that will cause my sprite to move and make sound.

Visit the following link: <http://www.mblock.cc/doc/en/start-with-scratch-programming/start-with-scratch-programming.html>

Directions: Please complete each activity on the checklist.

Activity 1: Create a Sprite and add movement.

- Go to Start with Scratch Website: <http://www.mblock.cc/doc/en/start-with-scratch-programming/start-with-scratch-programming.html>
- Start with Scratch Programming will be on the top of your screen. Click the Next button on the lower right corner of the box.
- The titles Create Your Sprite and Play With It will be at the top.
- Open mBot software.
- Follow the tutorial by reading each step and then completing it on your own mBot software.
- Share what you created with a peer. Peer's initials: _____

Activity 2: Add Sound to Sprite.

- Go to Start with Scratch Website: <http://www.mblock.cc/doc/en/start-with-scratch-programming/start-with-scratch-programming.html>
- On the left side of the screen, click Add Some Sound. The title Add Some Sound will be at the top.
- Open mBot software.
- Follow the tutorial by reading each step and then completing it on your own mBot software.
- Share what you created with a peer. Peer's initials: _____

Activity 3: Challenge

- Open mBot software.
- Using the learning from activity 1 and activity 2, create a sprite with movement and sound.
- Conference with teacher. Teacher's initials: _____

Guided Notes on mBlock Shapes

Goal: I can explain the programming in a block of mBlock script.

Learning Intention

Today I will: take guided notes over mBlock shapes and their functions.

So that: I can learn how to put blocks together to write code.

I will know I have this when: I can write and explain what the code/script means.

Directions: Visit the following link: <http://www.mblock.cc/doc/en/block-reference/block-reference.html>. Complete the notes as you read through the information.

Block Reference

Scratch programming blocks come in different _____, which can be connected to each other _____ like a jigsaw puzzle.

_____ are created by a series of _____ blocks. Each _____ type has its own shape and a specially shaped slot. There are five shapes of blocks:

- a. _____
- b. _____
- c. _____
- d. _____
- e. _____

A _____ Block is a block that _____ the script, and is _____ placed on _____ of other blocks. Draw a hat block in the space provided:

A _____ Block is a rectangular block that can fit _____ or _____ other blocks, with a notch at the top and a bump on the bottom. Stack blocks to perform the _____ commands, making up the _____ of all the programming blocks. Draw a stack block in the space provided:

A _____ Block is a block that contains a _____, either numbers or strings. Reporter blocks can fit _____ a value is needed, but _____ stand alone. A Report Block can also fit into another Reporter Block if there is a specific slot. Draw a Report Block in the space provided:

A _____ Block contains a _____, which can be either " _____ " or " _____ ". Boolean Blocks fit in the corresponding _____ slots on other blocks, so they _____ stand alone either. Draw a Boolean Block in the space provided:

A _____ Block _____ a script or project. Cap Blocks have a smooth bottom, so no other blocks can be placed _____ them, but blocks can be placed _____ a Cap Block. Draw a Cap Block in the space provided:

So what do these blocks mean???

Event Blocks: click the event tab to learn more.

Color? _____

Main function? _____

Complete the table below by locating the block and explaining its function.

mBlock	Function
	
	
	

Motion Blocks: click the motion tab to learn more.

Color? _____

Main function? _____

Complete the table below by locating the block and explaining its function.

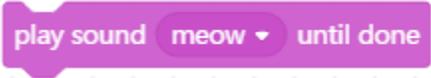
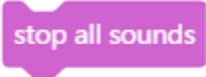
mBlock	Function
	
	
	

Sound Blocks: click the sound tab to learn more.

Color? _____

Main function? _____

Complete the table below by locating the block and explaining its function.

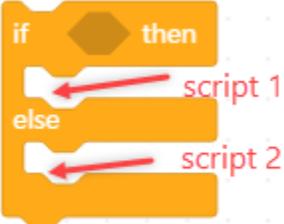
mBlock	Function
	
	
	

Control Blocks: click the control tab to learn more.

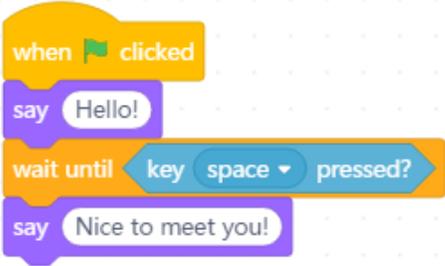
Color? _____

Main function? _____

Complete the table below by locating the block and explaining its function.

mBlock	Function
	
	
	
	

Read the script/code and write down what it means.

Script/Code	Meaning
 <p>The code consists of two blocks on a grid background. The first block is yellow with a semi-circle on the left, containing the text "when right arrow key pressed". The second block is blue with a semi-circle on the left, containing a circular arrow icon, the number "15" in a white circle, and the text "degrees".</p>	
 <p>The code consists of four blocks on a grid background. The first block is yellow with a semi-circle on the left, containing a green flag icon and the text "clicked". The second block is purple with a semi-circle on the left, containing the text "say Hello!". The third block is orange with a semi-circle on the left, containing the text "wait until", a blue circle with a white arrow, the text "key space", and the text "pressed?". The fourth block is purple with a semi-circle on the left, containing the text "say Nice to meet you!".</p>	

Write the script/code for the function below:

When the up arrow key is pressed, move 25 steps, wait 3 seconds, say "Go, North!"

Empty space for writing the script/code.

Simple Device Program Checklist

Goal: I can code the mBot to move in a circle and move in a square.

Learning Intentions

Today I will: develop my own program.

So that: I can learn how to write script/code for my mBot.

I will know I have this when: I can write a script/code that will cause my mBot to move in a square and circle.

Directions: Please complete each activity on the checklist.

Activity 1: Write code for mBot to move in a circle. We will learn about two types of circles: a point circle and a curve circle.

Point Circle

- Open mBlock Software
- Title program point circle
- Pick an event block (when mBot starts up)



- Pick 1 of the action blocks (turn right at power ____ % for ____secs
 - You will need to input the amount of power and the length of time. (Think about how much you want the mBot to turn to make a complete circle.)



- Use the USB cable and connect the cable to your mBot and the computer.

- Turn upload mode on (located on the bottom of the screen of the mBlock software).
- Click connect.
- Click upload.
- Run program with mBot.
- If the mBot did not make a point circle, please go back and adjust the power and seconds in your blocks.
- You did it! The mBot makes a point circle. Share what you created with a peer. Peer's initials: _____

Curve Circle

- Open mBlock Software
- Title program curve circle
- Pick an event block (when mBot starts up)



- Pick 1 of the action blocks (left wheel turns at power ____ % right wheel at power ____%)
 - You will need to input the amount of power per wheel. (Think about how much you want each wheel to control.)



- Use the USB cable and connect the cable to your mBot and the computer.
- Turn upload mode on (located on the bottom of the screen of the mBlock software).
- Click connect.
- Click upload.
- Run program with mBot.

- If the mBot did not make a curve circle, please go back and adjust the powers.
- You did it! The mBot makes a curve circle. Share what you created with a peer. Peer's initials: _____

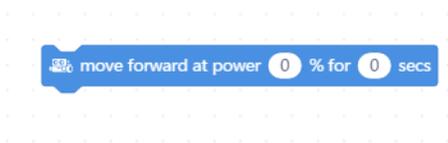
Activity 2: Write code for mBot to move in a square.

- Open mBlock Software
- Title program square

- Pick an event block (when mBot starts up)



- Pick 1 action blocks (move forward at power ____ % for ____secs)
 - You will need to put in the amount of power and the length of time. (Think about how fast and how long you want the mBot to go for the first command.)



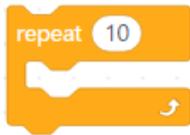
- Pick 1 control block (wait ____ secs)
 - You will need to insert the amount of time for the mBot to wait.



- Pick 1 action blocks (turn right at power ____ % for ____secs)
 - You will need to put in the amount of power and the length of time. (Think about how much you want the mBot to turn.)

A blue Scratch block with a motor icon on the left. The text inside the block reads "turn right at power 50 % for 1 secs".

- Pick 1 control block (repeat). Put this block around every block except for the event block.



- Use the UBS cable and connect the cable to your mBot and the computer.
- Turn upload mode on (located on the bottom of the screen of the mBlock software).
- Click connect.
- Click upload.
- Run program with mBot.
- If the mBot did not make a square, please go back and adjust the power and seconds in your blocks.
- You did it! The mBot makes a square. Share what you created with a peer.
Peer's initials: _____

Activity 3: Discussion with Teacher

- Reflect on the program for a square. What did you notice? Do you think your mBot will be accurate every time the program is ran?

Sensor Programming Checklist

Goal: I can program the mBot to sense its environment and navigate it as well.

Learning Intention

Today I will: take guided notes and practice using sensors.

So that: I can learn how sensors work together with programming to create a robot that can respond to its environment.

I will know I have this when: I can write a script that allows the robot to use the ultrasonic sensor to detect distance and the light sensor to follow a black line.

Directions: Complete the checklist below by following the steps below. Be sure to conference with the teacher at the checkpoints.

Activity 1: Learning to use the ultrasonic sensor

☐ Visit the following link <http://learn.makeblock.com/en/robot-kits/mbot/the-ultrasonic-sensor/> and complete the guided notes below:

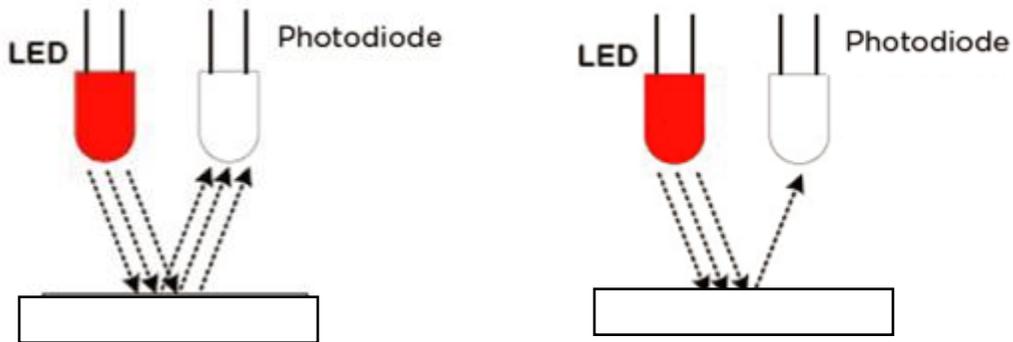
- The ultrasonic sensor measures _____. One of the “_____” transmits a _____, and the other waits for the _____ of the _____ to return. From the time this process takes place, the _____ of the object from the sensor can be _____. The ultrasonic sensor has a range of _____.
- I want to get the robot to go forward until it is close to an object, then turn away from the object and go off in a new direction. In the code below, the robot will turn if an object is within _____, otherwise it will keep going _____.



- Challenge: Can you write a program that avoids an object and stops?
- Conference with the teacher. Be prepared to demonstrate your robot and share your script. _____

Activity 2: Learning to use the line follower sensor

- Visit the following link <http://learn.makeblock.com/en/robot-kits/mbot/the-line-follower/> and complete the guided notes below:
 - The line follower has _____ sensors which can detect a _____ surface (within the range of 1-2cm). It works by emitting IR (_____) light and recording how much is _____ back. If _____ is reflected back, it can be deduced it is _____ to a _____ surface. If a _____ is reflected back, it can be deduced that the surface is _____, or the sensor is not near a surface.
 - Label the images **light surface** or **dark surface** based on the amount of light being reflected back to the photodiode.



- In mBlock there are _____ blocks associated with the line follower:



- The first block will return a number between _____ and _____ based on the following values:

Sensor 1 (Left)	Sensor 2 (Right)	Returned Value
		0
		1
		2
		3

○ The second block will return either _____ or _____.

- Challenge: Can you write a program so the robot follows a line?
- Conference with the teacher. Be prepared to demonstrate your robot and share your script. _____

Reflection:

What challenges did you face and how did you overcome them?

How did programming sensors make your robot more accurate?