LEGO® Mindstorms® EV3 Robotics Instructor Guide

Joanna M. Skluzacek, PhD

Associate Professor

Department of Youth Development

University of Wisconsin – Extension

Introduction to LEGO® Mindstorms® EV3

This curriculum packet is an introduction to LEGO® Mindstorms® EV3 robotic system with the v 2.1 programming software. The curriculum can assist even the most novice instructor in gaining experience and confidence in robotics and help advanced instructors to easily implement new challenges into their club or group setting.

The goals of this guide are to:

- Provide background and research on how robotics' projects help youth develop STEM (science, technology, engineering and math) content and life skills,
- Summarize essential information about LEGO® Mindstorms® EV3 robotic system,
- Assist in troubleshooting technology problems and,
- Provide detailed instructions and solutions for robotic challenges.

<u>Why Robotics</u>

A recent study by the President's Council of Advisors on Science and Technology found: "Economic forecasts point to a need for producing, over the next decade, approximately 1 million more college graduates in STEM fields than expected under current assumptions." The shortage of skilled STEM workers has been documented in several research papers (Bayer Corp., 2012; Fadigan & Hammrich, 2004; Grasso, Callahan, & Doucett, 2004; Smith, Heck, & Worker, 2012; Tran & Nathan, 2010) and government reports (Beatty & National Research Council, 2011; Committee on Prospering in the Global Economy of the 21st Century & Committee on Science Engineering and Public Policy, 2007; National Research Council, 2011; Nielsen & National Research Council, 2011). One proposed solution to this issue is to engage youth at a young age and throughout their formative years in STEM programs and projects, while inspiring them to seek out job opportunities within a STEM field. Teaching youth STEM through robotics allows the student to learn content knowledge of a subject area by applying the content in a real-world context. There are many examples of how robotic systems are used daily and this connection between educational programs and real-world skills enables students view the relevance in their educational programs.

Studies specific to LEGO® Mindstorms® suggest youth gain problem solving skills, creative thinking skills and an increase in scientific-technological knowledge (Barak & Zadok, 2009; Barker, 2007; Chen Yuan, 2012; Lindh & Holgersson, 2007; Mosley & Kline, 2006; Nagchaudhuri, 2002; Slangen, van Keulen, & Gravemeijer, 2011). Other studies show youth who worked with LEGO® Mindstorms® systems improved in-school performance in STEM related disciplines (Cejka, Rogers, & Portsmore, 2006; Panadero, Romaacuten, & Kloos, 2010). The inherently interdisciplinary nature of the Mindstorms® robotic systems encourages participants to use different types of skills and has proved to capture youth in long-term engagement which increases content mastery. Since the robot is created from LEGO® building blocks, youth have an almost infinite array of configurations they can build, inspiring creativity and increasing engagement. The incorporation of computer programming into the project allows participants to become more comfortable with technology and computer operations.

Alignment with 4-H Youth Development Outcomes

This curriculum is meant to serve as a resource for adult and youth leaders in robotics. The anticipated outcomes of using this resource are:

Adults will increase their comfort and confidence in leading STEM-based activities.

Adults will gain familiarity with the experiential learning model.

Youth will increase their involvement in 4-H STEM project opportunities.

Youth will demonstrate creativity, innovation and critical thinking skills.

> Youth can justify their decisions about their programming strategy.
>
> Youth can use information gained through trial-and-error programming to refine decision making skills.
>
> Youth can develop their own robotic challenges and express creativity and critical thinking.
>
> Since there is not one right way to complete a challenge, youth can be creative in their robotic program.

Youth will increase their ability to work collaboratively with others.

> Youth will work in pairs or teams to complete challenges.
>
> Youth will respect others opinions on how to solve problems.

Youth will be able to improve communication skills.

> Youth will be able to reflect on successes and failures in programming and describe the issues.
>
> Youth will use technology to help express ideas.
>
> Youth will learn how to deal with stress and trial-and-error in programming.

Alignment with 4-H Science Abilities, 4-H Science Process Skills and STEM Content Expertise

The 4-H Science Abilities are divided into overarching process skills. These seven process skills are: observing, questioning, hypothesizing, predicting, investigating, interpreting, and communicating. Several of these skills are described in the previous section. In addition to life skill development outcomes, youth who use robotics as a platform for learning also gain abilities specific to STEM content. For instance, youth will observe how the robot performs, question why the robot performed as it did and troubleshoot the program using the information gathered. Specifically, participants will improve their basic computer operation skills (save and open files, cut and paste tools); gain experience with application of basic math functions (addition, subtraction, etc.); apply geometry to calculate turn radius and wheel rotation distance; and use units of measurement along with measurement estimation and comparison.

How to Use This Curriculum Guide

It is highly recommended to review the video resources found on the 4-H EV3 robotics webpage: https://fyi.uwex.edu/wi4hstem/lego-ev3/ as these materials will enable you to gain the basics of the

EV3 system, such as turning on the robot and locating saved files. This curriculum guide *does not* discuss building the robot, turning on the robot, finding saved files nor does it provide a software tutorial.

This manual is designed to be used with the LEGO® Mindstorms® EV3 base set robot and the LEGO® Mindstorms® EV3 education software purchased from the LEGO® online store (http://www.lego.com). There are other robots available through other vendors, but some of these kits do not have the same components as the base set LEGO® kit. The software also comes in a free "home edition" download. All the challenges presented in this curriculum guide can be performed with either the software edition.

The LEGO® Mindstorms® EV3 robotic kits are available for check-out through the University of Wisconsin-Extension Media Library for a nominal fee. Many county extension offices also have robotic kits and laptops available for lending. It is recommended to have one robot and computer (loaded with the software) per two youth participants.

The lessons are created to be used with the "basic robot" configuration. This means the robot is built according to the instruction manual that comes with the base set kit. The building instruction manual provides stepwise procedures for building, along with length scale keys so youth can easily find the right sized part by comparing it to the picture.

The curriculum provided in this guide is for the instructor; however, the guide may be edited for youth participants depending on the teaching strategy, objectives and venue of the program. For example, the instructor can give the participants the challenge instructions as text only or a full programming guide. If participants are given the text only, the instructor will need to have more direct teaching time to describe the programming blocks and the details of downloading the program from the computer to the robot.

The full pictorial programming guide can be given to youth directly, however; this in only recommended if the instructor has very little time for the session, for instance, if the program is a marketing event for a robotics camp or club project. It might also be helpful to give youth the full programming guide for the first activity as part of a longer session. In this instance, the instructor should take time for the group to discuss each programming block and the variables associated with the programming, since the youth will not develop this understanding by using the complete pictorial guide alone.

The complete pictorial guides provide examples of solutions to complete the challenges. It is important to remember that there are multiple ways to successfully complete challenges and youth are encouraged to be creative with their programming. It is also important to recognize that the numbers for duration, etc. provided within the complete pictorial guide are estimates. The robot will perform differently under different conditions. Examples of performance variables are described in the Distance and Turns Instructor Guide.

<u>Troubleshooting Tips</u>

Make sure the **battery** is charged. The rechargeable battery is reliable and does last a long time (at least 15 hours) when charged, however; if the robot has been left unused for a long period of time or exposed to cold weather during transport or shipping the battery might be discharged.

Having **AA batteries** on hand may be helpful if the battery is not working properly.

When using the **ultrasonic sensor** remind the youth to not have their hands in front of the sensor when they press the button to start the program. The sensor will sense their hand and the robot will immediately travel backwards.

There are two ways to build the **light sensor** in the building manual. In one configuration the light sensor is pointing straight forward and in the other configuration the light sensor is pointing down. Make sure to select the correct configuration for the task.

Resources

This curriculum guide is meant to serve as an introduction to robotics and provide a few robotic challenges to get instructors started using this engaging learning environment. However, LEGO® Mindstorms® robotic systems are widely used and there are several other places to find teaching resources. One important resource is a tutorial entitled "**Robot Educator**" that is embedded into the EV3 software.

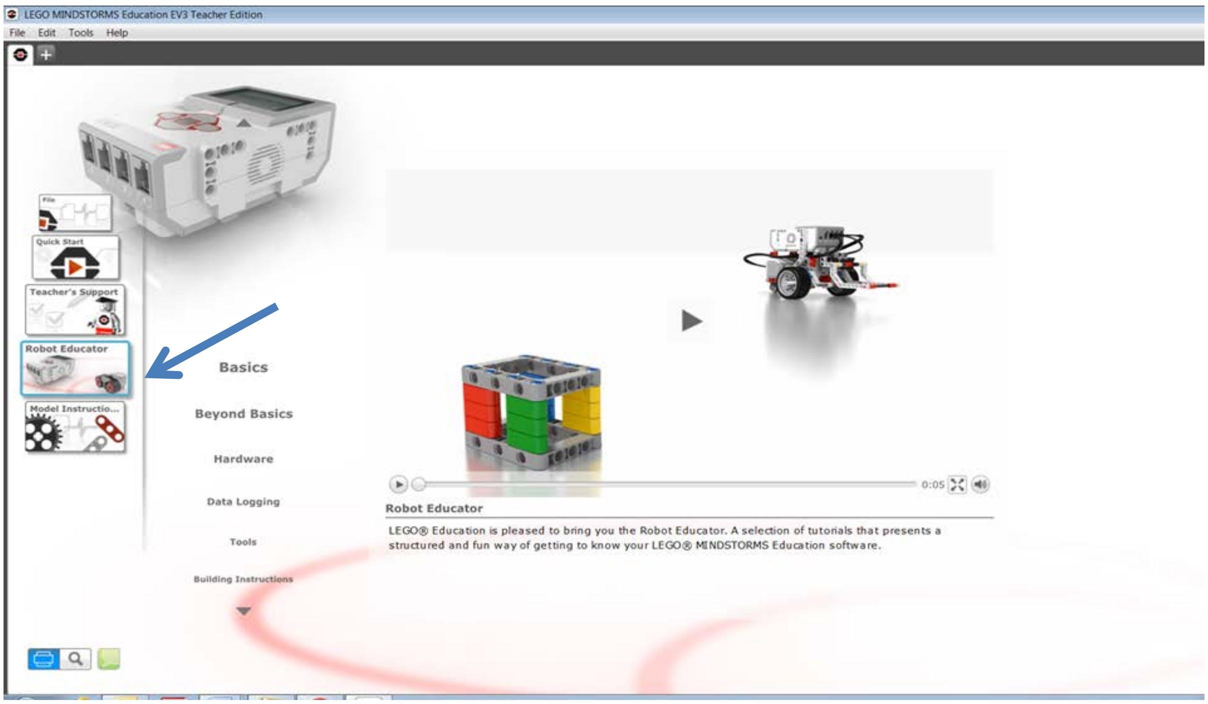


Figure 1: Area to click to select the Robot Educator Tutorial Guide.

Below is a list of teaching resources:

STEMcentric EV3

Texas Tech University Robotics

Dr. Graeme EV3 Lessons

Robotics 1 with EV3 (Ohio State 4-H)

Robotics 2 EV3N More (Ohio State 4-H)

References

Barak, M., & Zadok, Y. (2009). Robotics projects and learning concepts in science, technology and problem solving. *International Journal of Technology and Design Education, 19*(3), 289-307.

Barker, B., & Asnorge, J. (2007). Robotics as Means to Increase Achievement Scores in an Informal Learning Environment. *Journal of Research on Technology in Education, 39*(3), 229-243.

Bayer Corp. (2012). Bayer Facts of Science Education XV: A View from the Gatekeepers-STEM Department Chairs at America's Top 200 Research Universities on Female and Underrepresented Minority Undergraduate STEM Students. *Journal of Science Education and Technology, 21*(3), 317-324.

Beatty, A. S., National Research Council. Committee on Highly Successful Schools or Programs for K-12 STEM Education, National Research Council. Board on Science Education & National Research Council. Center for Education. Board on Testing and Assessment. (2011). *Successful STEM education : a workshop summary*. Washington, D.C.: National Academies Press.

Cejka, E., Rogers, C., & Portsmore, M. (2006). Kindergarten robotics: Using robotics to motivate math, science, and engineering literacy in elementary school. *International Journal of Engineering Education, 22*(4), 711-722.

Chen Yuan, C. (2012). An innovative knowledge management learning cycle by Lego NXT for science education. *International Journal of Innovative Computing, Information & Control, 8*(1B), 791-798.

Committee on Prospering in the Global Economy of the 21st Century & Committee on Science Engineering and Public Policy. (2007). *Rising above the gathering storm : energizing and employing America for a brighter economic future*. Washington, D.C.: National Academies Press.

Fadigan, K. A., & Hammrich, P. L. (2004). A longitudinal study of the educational and career trajectories of female participants of an urban informal science education program. *Journal of Research in Science Teaching, 41*(8), 835-860.

Grasso, D., Callahan, K. M., & Doucett, S. (2004). Defining engineering thought. *International Journal of Engineering Education, 20*(3), 412-415.

Lindh, J., & Holgersson, T. (2007). Does lego training stimulate pupils' ability to solve logical problems? *Computers & Education, 49*(4), 1097-1111.

Mosley, P., & Kline, R. (2006). Engaging students: a framework using LEGO robotics to teach problem solving. *Information Technology, Learning, and Performance Journal, 24*(1), 39-4545.

Nagchaudhuri, A., Singh, G., Kaur, M., George, S. (2002). *LEGO Robotics Products Boost Student Creativity in Pre-College Programs at UMES*. Paper presented at the ASEE/IEEE Frontiers in Education Conference, Boston, MA.

National Research Council. Committee on Highly Successful Schools or Programs for K-12 STEM Education, National Research Council. Board on Science Education, & National Research Council. Center for Education. Board on Testing and Assessment. (2011). *Successful K-12 STEM education: identifying effective approaches in science, technology, engineering, and mathematics*. Washington, D.C.: National Academies Press.

Nielsen, N. & National Research Council. Planning Committee on Evidence on Selected Innovations in Undergraduate STEM Education. (2011). *Promising practices in undergraduate science,* *technology, engineering, and mathematics education : summary of two workshops*. Washington, D.C.: National Academies Press.

Panadero, C. F., Romaacuten, J. V., & Kloos, C. D. (2010). Impact of learning experiences using LEGO Mindstorms reg in engineering courses *2010 IEEE Education Engineering 2010 - The Future of Global Learning Engineering Education* (pp. 503-512)

Slangen, L., van Keulen, H., & Gravemeijer, K. (2011). What pupils can learn from working with robotic direct manipulation environments. *International Journal of Technology and Design Education, 21*(4), 449-469.

Smith, M. H., Heck, K. E., & Worker, S. M. (2012). 4-H boosts youth scientific literacy with ANR water education curriculum. *California Agriculture, 66*(4), 158-163.

Tran, N. A., & Nathan, M. J. (2010). The Effects of Pre-Engineering Studies on Mathematics and Science Achievement for High School Students. *International Journal of Engineering Education, 26*(5), 1049-1060.

Instructor Guide

Title: Distance the robot will travel based on wheel size

Introduction

Calculating the distance the robot will travel for each of the duration variables (rotations, degrees, seconds) can be confusing for participants especially when coupled with a turn or a spin. It is important to remember that rotations and degrees reference the wheel axle, such that these distances can change depending on the size of the tire installed on the robot. This activity outlines how to use the duration variables correctly to make corner turns and to calculate the distance to objects using the circumference of the tire. The answers provided in the instructor guide are based on the basic bot construction from the LEGO® EV3®.

Objectives

Youth will apply basic math functions and geometry.
Youth will gain experience with units of measurement and measurement comparison.
Youth will demonstrate creativity, innovation and critical thinking skills.
Youth will increase their ability to work collaboratively with others.
Youth will better understand the process of programming and evaluating robotic movements.

Preparation and Materials

One-12 inch ruler per robot

Activity: How far will your robot travel?

Take a look at the larger tires connected to the gear motors on the NXT robot. On the side of these tires there are numbers. This is true for *all* tires and tells you the size (diameter and width) of the tire. Automobile tires and bicycle tires also have these numbers to ensure the mechanic places the correct tire size on each axle.

1. What are the numbers on the robots large tire?

   56 X 28

2. Are these numbers are in inches, centimeters or millimeters?

   Millimeters

3. Which number is the diameter and which is the width?

   56 mm = diameter

   28 mm = width

4. Using the diameter of the tire, one can find the *Circumference* of the tire. The circumference of a tire tells the distance a tire travels in one revolution:

   $C = \pi D$

   Where C= circumference, π = 3.14, and D = diameter

What is the circumference of the EV3 tire?

3.14
X 56
1884
15700
175.84 mm

This means one rotation of the tire is 175.84 mm or 17.6 cm.

5. If one rotation is equal to 17.6 cm, how many cm will the robot travel in three tire rotations?

   17.6 X 3 = 52.8 cm

6. Since the wheel of the robot is a circle, one can also speak about degrees of tire rotation. How many degrees does the tire rotate for each tire rotation?

   1 tire rotation – 360 degrees

7. How many cm will the robot travel if it is programmed to travel a duration of 720 degrees?

   720 degrees = 2 tire rotations = 35.2 cm

However, the robot might not travel the distance you calculated. There are several variables that affect the distance travelled by the robot:

- The **power level setting** will affect distance traveled when using the time interval in seconds; however, it will not affect the distance travelled in rotations or degrees. In addition the battery life levels will also impact the amount of power provided to the robot to complete the moves.
- The external environment will impact the amount of **friction** on the tires. A smooth surface will have less friction meaning the robot will travel slightly faster. A carpeted surface will have more friction meaning the robot will travel slightly slower.
- The **mechanics** of the built robot impacts the function of the robot. For example if a tire is not perfectly aligned on the wheel well or on the axle you may experience a wobbly wheel.
- The **temperature** will affect the distance travel. Since the wheels are made out of rubber, they will slightly expand when the temperature is hot and contract with the temperature gets colder.

Instructor Guide

Challenge Title:

EV3 Four Bricks

Introduction:

There is no challenge mat for this activity and no sensors need to be connected to the robot. This challenge introduces participants to the move block and the relationship between the different duration types: rotations, degrees and seconds.

Objectives

Youth will demonstrate creativity, innovation and critical thinking skills. Youth will increase their ability to work collaboratively with others. Youth will be able to improve communication skills. Youth will better understand the process of programming and evaluating robotic movements. Youth will increase their ability to design a solution to a challenge.

Level of Difficulty
Easy

Preparation and Materials

Instructor should determine teaching methodology (text instructions or pictorial programming guide) that best fits the audience.
Print appropriate student materials.

Time Required Programming:

5-10 minutes (if participants are given the text instructions only).

To run the course: less than a minute.

Procedure of Programming Steps

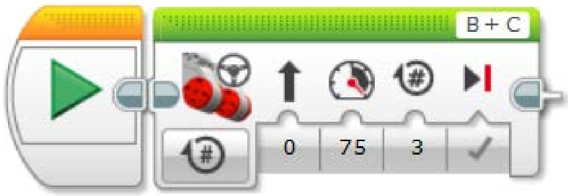
Depending on the expertise level of the participants, the instructor can give the youth the text of the steps involved in the challenge or give them the entire pictorial programming guide.

TEXT:

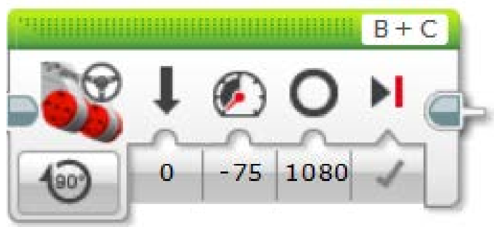
1. Robot moves forward three tire rotations at 75% power.
2. Robot moves backward 1080 degrees at 75% power.
3. Robot moves forward for two seconds at 75% power.
4. Robot moves forward while making a hard turn toward the right for 230 degrees at 75% power.

Program Guide

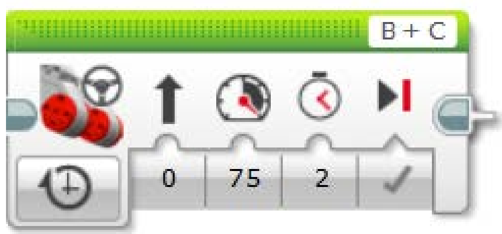
1. Robot moves forward three tire rotations at 75% power.



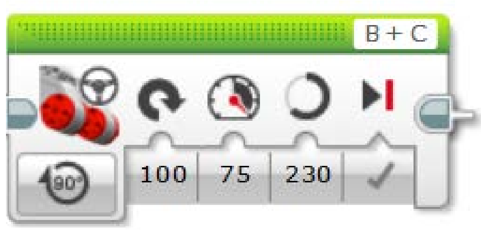
2. Robot moves backward 1080 degrees at 75% power.



3. Robot moves forward for two seconds at 75% power.

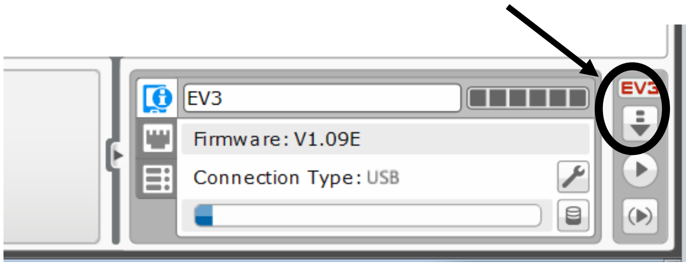


4. Robot moves forward while making a hard turn toward the right for 230 degrees at 75% power



Connect the computer and the robot using the USB cord and make sure the robot is turned on.

A. Download the program onto the robot by pressing the download button at the bottom right corner of the computer screen.



B. Disconnect the robot from the USB cord.

C. Find the program on the robot and use the dark gray button to start the program.

Discussion Questions

What difficulties did you encounter with the programming?

What did you do to overcome these difficulties?

How did the distance traveled by the robot in step 1 compare to the distance the robot traveled in step 2?

The robot moved the same distance. Three tire rotations equals 1080 degrees.

Instructor Guide
Challenge Title:

EV3 Bark Like a Dog

Introduction
This robotic challenge does not need a challenge mat. Youth can use their hand to activate the touch sensor. The robot configuration must include the touch sensor to complete the challenge.

Objectives
Youth will demonstrate creativity, innovation and critical thinking skills.
Youth will increase their ability to work collaboratively with others.
Youth will be able to improve communication skills.
Youth will better understand the process of programming and evaluating robotic movements.
Youth will increase their ability to design a solution to a challenge.

Level of Difficulty
Easy

Preparation and Materials
Instructor should determine teaching methodology (text instructions or pictorial programming guide) that best fits the audience.
Print appropriate student materials.

Time Required
Programming: 15 minutes (if participants are given text instructions only).
To run the course: less than a minute.

Procedure of Programming Steps
Depending on the expertise level of the participants, the instructor can give the youth the text of the steps involved in the challenge or give them the complete pictorial programming guide.

TEXT:

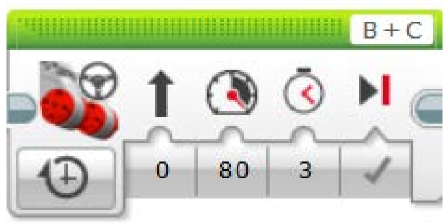
1. The robot waits five seconds before starting the program.
2. The robot barks like a dog.
3. The robot moves forward for 3 seconds at 80% power.
4. The robot stops and waits for you to press the touch sensor.
5. The robot moves backwards four tire rotations.
6. The robot moves forward and uses the touch sensor to hit an obstacle (youth can use their hand to activate the touch sensor).
7. The robot moves backwards one tire rotation and stops.

COMPLETE PICTORIAL PROGRAMMING GUIDE:

1. The robot waits five seconds before starting the program.

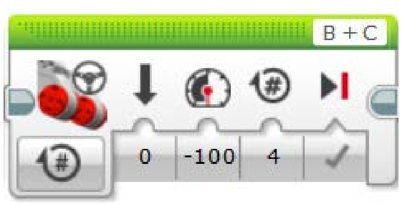
2. The robot barks like a dog.

3. The robot moves forward for 3 seconds at 80% power.

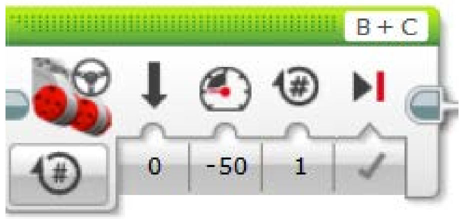


4. The robot stops and waits for you to press the touch sensor.

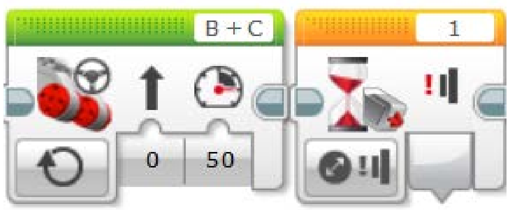
5. The robot moves backwards four tire rotations.



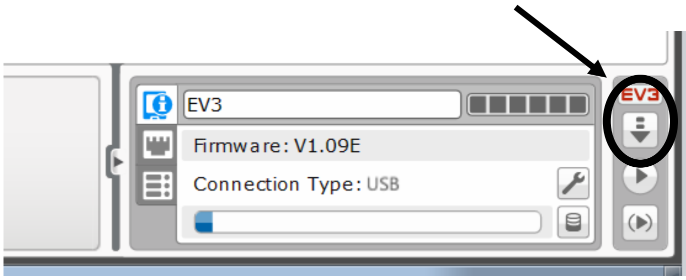
6. The robot moves forward and uses the touch sensor to hit an obstacle (youth can use their hand to activate the touch sensor).



7. The robot moves backwards one tire rotation and stops.



A. Connect the computer and the robot using the USB cord and make sure the robot is turned on.
B. Download the program onto the robot by pressing the download button at the bottom right corner of the computer screen.



C. Disconnect the robot from the USB cord.
D. Find the program on the robot and use the dark gray button to start the program.

Discussion Questions

What difficulties did you encounter with the programming?

What did you do to overcome these difficulties?

What other things might use a touch sensor?

Automatic car wash

Touch screens

What was the difference in the programming for the touch sensor in Step 3 versus Step 6?

The robot was stopped and waited for you to touch the touch sensor in Step 3. In Step 6 the robot moved until the touch sensor was activated. In the programming, you alter the move block before the sensor block.

Why did you need to select "on" for the move block in Step 6?

You needed to select "on" because you didn't know how far away the obstacle was that would activate the touch sensor.

Instructor Guide
Challenge Title:

EV3 X Marks the Spot

Introduction

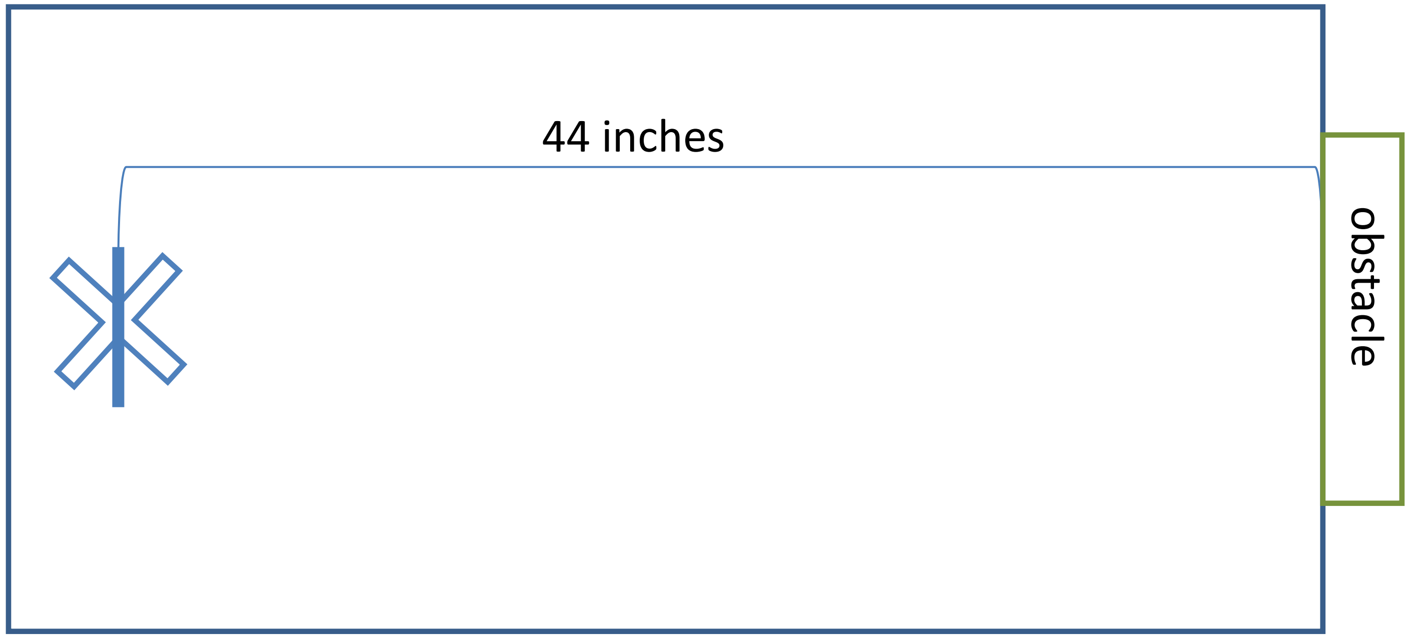
This robotic challenge uses a challenge mat. The robot configuration must include the touch sensor and the ultrasonic (distance) sensor. The obstacle listed on the challenge schematic can be a wall or any other object that will not move when struck by the robot. The participants will not be told the distance from the obstacle to the X. They will need to use trial-and-error to calculate the proper travel duration of the robot. For example, they might have to view how far the robot travels in one tire rotation and then estimate how many rotations it will take to cover the required distance. The youth will also have to use trial-and-error to figure out how to make the robot turn the correct arc distance so the touch sensor is facing the obstacle after the robot spins on the X in step 6. They may also reference the "**Distance the Robot Will Travel Based on Wheel Size"** activity to estimate how many rotations or degrees it will take to successfully complete the challenge.

Objectives
Youth will demonstrate creativity, innovation and critical thinking skills.
Youth will increase their ability to work collaboratively with others.
Youth will be able to improve communication skills.
Youth will better understand the process of programming and evaluating robotic movements.
Youth will increase their ability to design a solution to a challenge.

Level of Difficulty
Intermediate

Preparation and Materials
Instructor should determine teaching methodology (text instructions or pictorial programming guide) that best fits the audience.
Print appropriate student materials.
Create "X Marks the Spot" challenge mat using:

- Masking or painter's tape
- Measuring tape
- Paper or poster board (at least 50 X 12 inches in size)
- Obstacle such as a weighted box, heavy book, or wall.


44 inches
obstacle


44 inches

Time Required
Programming: 45 minutes – 1 hour (if participants are given text instructions only).
To run the course: 2-3 minutes.

Procedure of Programming Steps
Depending on the expertise level of the participants, the instructor can give the youth the text of the steps involved in the challenge or give them the entire pictorial programming guide.

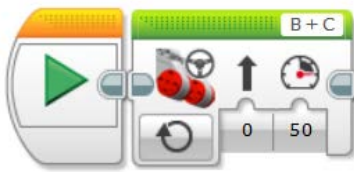
TEXT:
The robot begins the challenge centered on the X facing toward the obstacle (ultrasonic sensor facing obstacle).

1. The robot moves toward the obstacle at 50% power.
2. The robot uses the ultrasonic sensor to detect the obstacle at a distance of 12 inches.
3. Upon sensing the obstacle the robot moves backwards and stops at its original position (centered on the X as in step 1) *without* using any sensors.
4. The robot spins in place one complete rotation so that the touch sensor is facing the obstacle.
5. The robot moves toward the obstacle.
6. The robot touches the obstacle with the touch sensor.
7. Upon sensing the obstacle the robot moves backwards and stops at its original position (centered on the X as in step 1) *without* using any sensors.

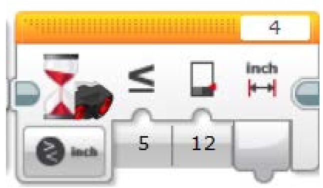
COMPLETE PICTORIAL PROGRAMMING GUIDE:

The robot begins the challenge centered on the X facing toward the obstacle (ultrasonic sensor facing obstacle).

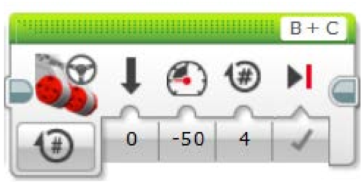
1. The robot moves toward the obstacle at 50% power.



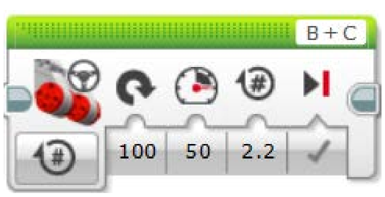
2. The robot uses the ultrasonic sensor to detect the obstacle at a distance of 12 inches.



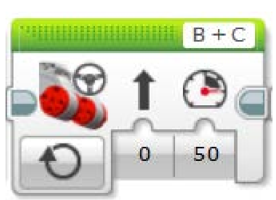
3. Upon sensing the obstacle the robot moves backwards and stops at its original position (centered on the X as in step 1) *without* using any sensors.



4. The robot spins in place one complete rotation so that the touch sensor is facing the obstacle.

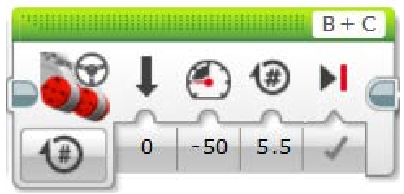


5. The robot moves toward the obstacle.



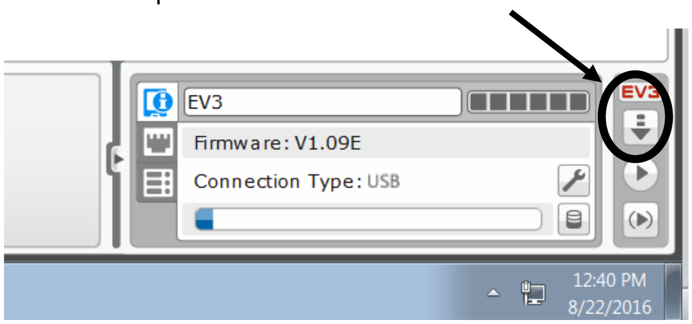
6. The robot touches the obstacle with the touch sensor.

7. Upon sensing the obstacle the robot moves backwards and stops at its original position (centered on the X as in step 1) *without* using any sensors.



Connect the computer and the robot using the USB cord and make sure the robot is turned on.

A. Download the program onto the robot by pressing the download button at the bottom right corner of the computer screen.



B. Disconnect the robot from the USB cord.

C. Find the program on the robot and use the dark gray button to start the program.

Discussion questions

What difficulties did you encounter with the programming?

What did you do to overcome these difficulties?

What strategies did you use to figure out the distance from the obstacle to the X?

What strategies did you use to figure out how to program the robot to turn in step 6?

Why did you need to select "on" for the move/steering block in Step 2?

You needed to select "on" because you didn't know how far away the obstacle was that would activate the ultrasonic sensor.

What other things might use a touch sensor?

Automatic car wash

Touch screens

What other things might use an ultrasonic (distance) sensor?

Bats use ultrasound for navigation.

Many motion detectors use ultrasound (burglar alarms, motion sensing lights).

Instructor Guide
Challenge Title:

EV3 Magic Touch

Introduction

This robotic challenge uses a challenge mat. The robot configuration must include the light sensor and the touch sensor. The obstacle listed on the challenge schematic can be a wall or any other object that will not move when struck by the robot. The participants will not be told the distance from the start box to the corner nor will they be given the distance from the corner to the obstacle. They will need to use trial-and-error to calculate the proper travel duration of the robot. For example, they might have to view how far the robot travels in one tire rotation and then estimate how many rotations it will take to cover the required distance. The youth will also have to use trial-and-error to figure out how to make the robot turn the correct arc distance so the touch sensor is facing the obstacle. They may also reference the "**Distance the Robot Will Travel Based on Wheel Size"** to estimate how many rotations or degrees it will take to successfully complete the challenge.

Objectives

Youth will demonstrate creativity, innovation and critical thinking skills.
Youth will increase their ability to work collaboratively with others.
Youth will be able to improve communication skills.
Youth will better understand the process of programming and evaluating robotic movements.
Youth will increase their ability to design a solution to a challenge.

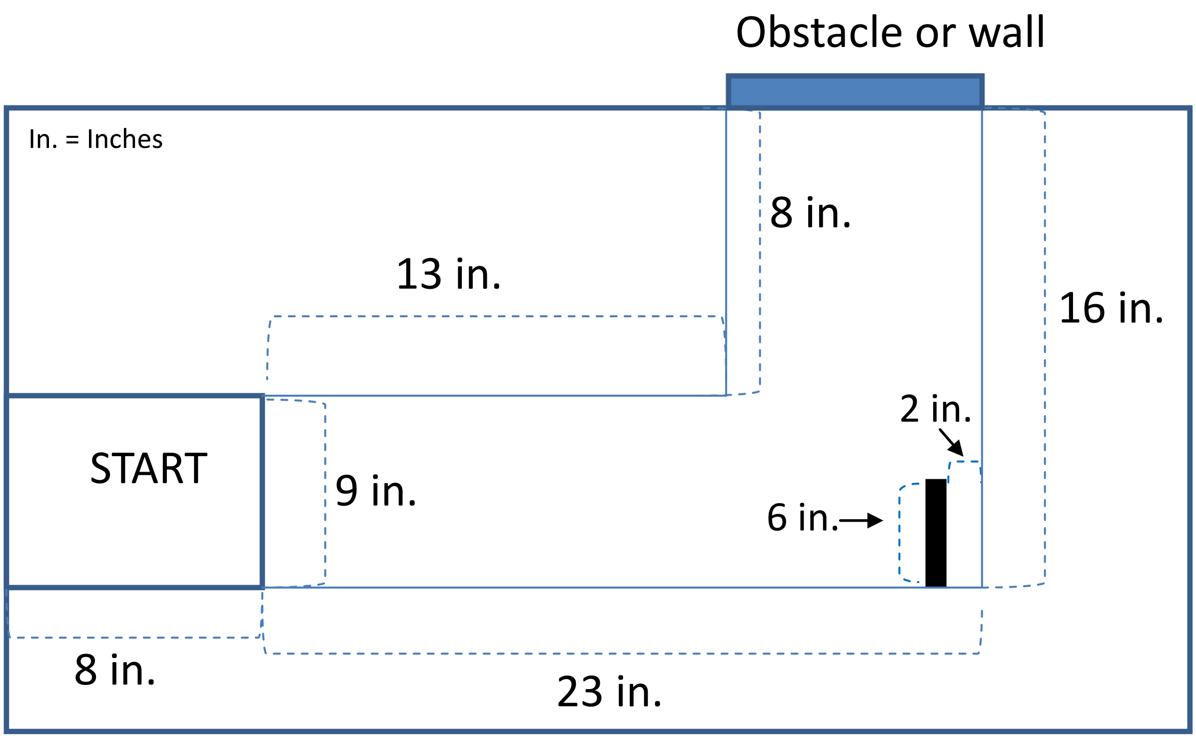
Difficulty

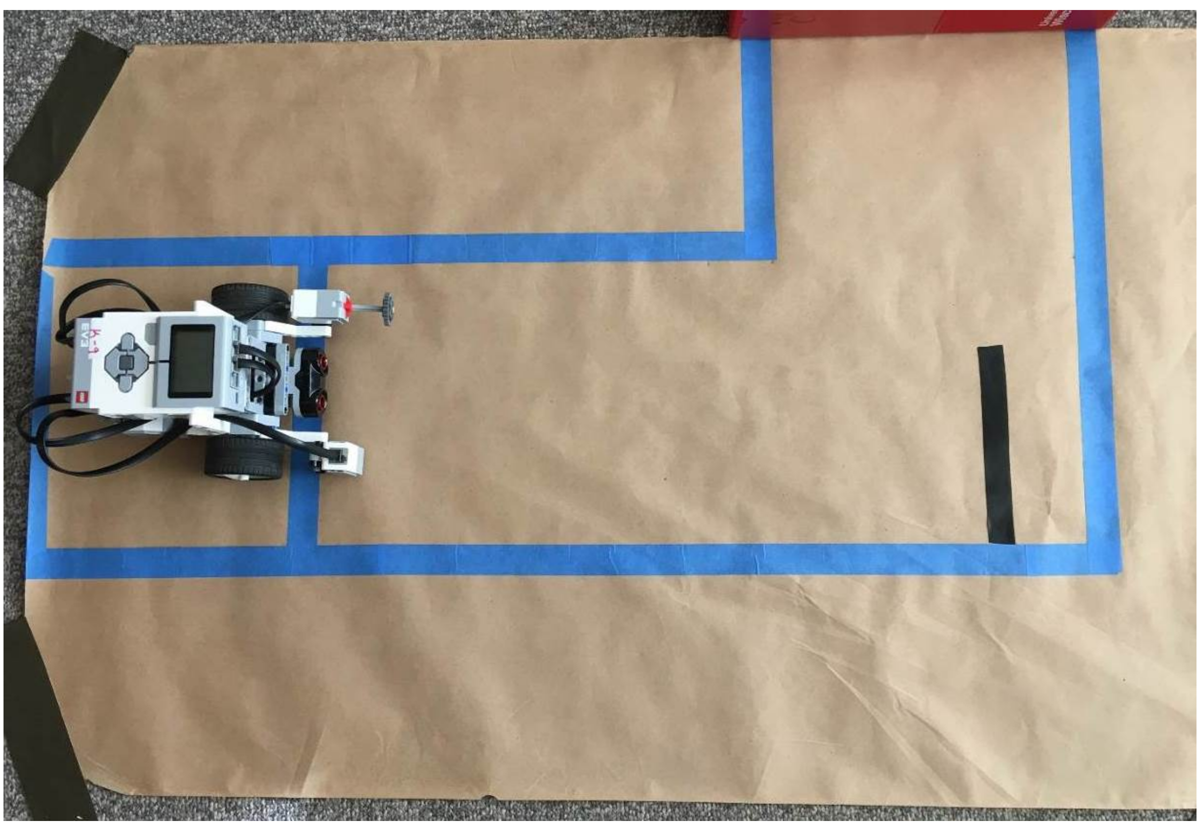
Difficult

Preparation and Materials

Instructor should determine teaching methodology (text instructions or pictorial programming guide) that best fits the audience.
Print appropriate student materials.
Create "Magic Touch" challenge mat using:

- Masking or painter's tape
- Black tape
- Measuring tape
- Paper or poster board (at least 40 X 20 inches in size)
- Obstacle such as a weighted box, heavy book or wall





This picture demonstrates the position of the robot in step 1 and at the end of step 6

Time Required
Programming: 30-45 minutes (if participants are given text instructions only)
To run the course: 1-2 minutes

Procedure of Programming Steps
Depending on the expertise level of the participants, the instructor can give the youth the text of the steps involved in the challenge or give them the entire pictorial programming guide.

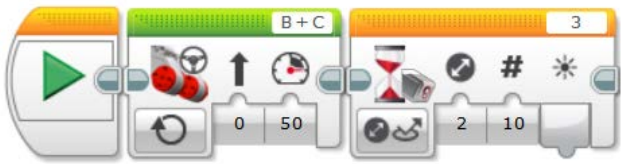
TEXT:
The front wheels of the robot must be completely behind the start line and the robot must stay inside the lines at all times.

1. The robot uses the light sensor to move from the start area to the dark tape line.
2. The robot turns such that the touch sensor is facing the obstacle.
3. The robot touches the obstacle with the touch sensor.
4. Once the robot touches the obstacle it must return to its initial starting position (facing forward and ready to complete the course again) *without* using any sensors.

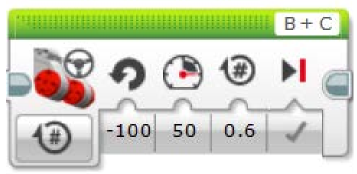
COMPLETE PICTORIAL PROGRAMMING GUIDE:

The front wheels of the robot must be completely behind the start line and the robot must stay inside the lines at all times.

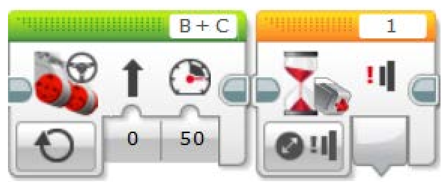
1. The robot uses the light sensor to move from the start area to the dark tape line.



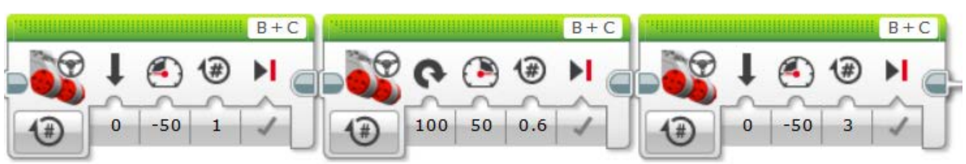
2. The robot turns such that the touch sensor is facing the obstacle.



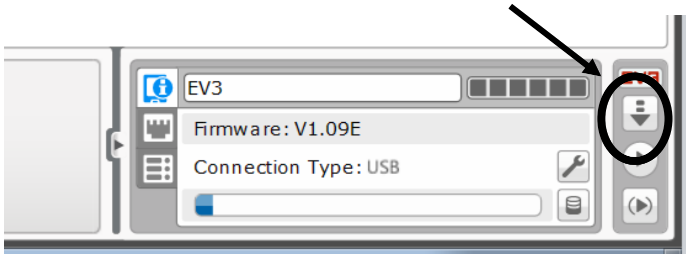
3. The robot touches the obstacle with the touch sensor.



4. Once the robot touches the obstacle it must return to its initial starting position (facing forward and ready to complete the course again) *without* using any sensors.



A. Connect the computer and the robot using the USB cord and make sure the robot is turned on.
B. Download the program onto the robot by pressing the download button at the bottom right corner of the computer screen.



C. Disconnect the robot from the USB cord.
D. Find the program on the robot and use the dark gray button to start the program.

Discussion Questions

What difficulties did you encounter with the programming?

What did you do to overcome these difficulties?

What strategies did you use to figure out how to program the robot to turn in step 2?

Why did you need to select “on” for the move block in Step 3?

You needed to select “on” because you didn’t know how far away the obstacle was that would activate the touch sensor.

What other things might use a touch sensor?

Automatic car wash

Touch screens