Reversing Safely

Design features for a car that will improve safety as it reverses.

Learning Objectives

Students will:

Extend the use of algorithms are capable of carrying out a series of instructions in order
Extend understanding of outputs

Vocabulary

Input, output, algorithm, pseudocode, debug, wait for

Grades

6-8

Subjects

Engineering, STEM, Coding

Duration

45-90 Minutes

Difficulty

Intermediate

Standards

NGSS
MS-ETS1-1. / MS-ETS1-2. / MS-ETS1-3. / MS-ETS1-4

CSTA
2-A-2-1 / 2-A-7-1 / 2-A-7-3 / 2-A-7-4 / 2-A-5-5 / 2-A-3-9 / 2-A-6-10 / 2-C-4-12 / 2-D-5-16 / 2-I-1-20

Materials Needed

LEGO® MINDSTORMS Education EV3 core set.
LEGO MINDSTORMS EV3 Software or Programming app
ROBOTC software (optional)

1

Reversing Safely

CONNECT (5 minutes)

In this lesson, students will use the Brick Status Light and Display Block to program their wheeled robot to simulate visual signals for pedestrians, other car drivers, and passengers when reversing. Students will also use the Touch Sensor to simulate forward and reverse gears.

Ignite a classroom discussion around the following questions:

– What are the dangers of reversing a car or other motorized vehicle?
– What are your area's regional statistics concerning accidents with reversing vehicles?
– How can a vehicle reverse more safely?
– What happens on the outside of vehicles when they are reversing to let pedestrians and other drivers know what they are doing?

Ask your students discuss briefly what happens when cars reverse. Discuss how the reverse lights warn other motorists and pedestrians of what the vehicle is doing. Explain that the indicators on the dashboard communicate to the driver and their passengers which gear the car is in and the direction of travel.

Allow the students to select the tool(s) they find most appropriate for capturing and sharing their ideas. Encourage them to document their thoughts using text, videos, images, sketchnotes, or another creative medium.

CONSTRUCT (15 to 30 minutes)

Build

Students will construct the Robot Educator base model, then they will add the Touch Sensor.
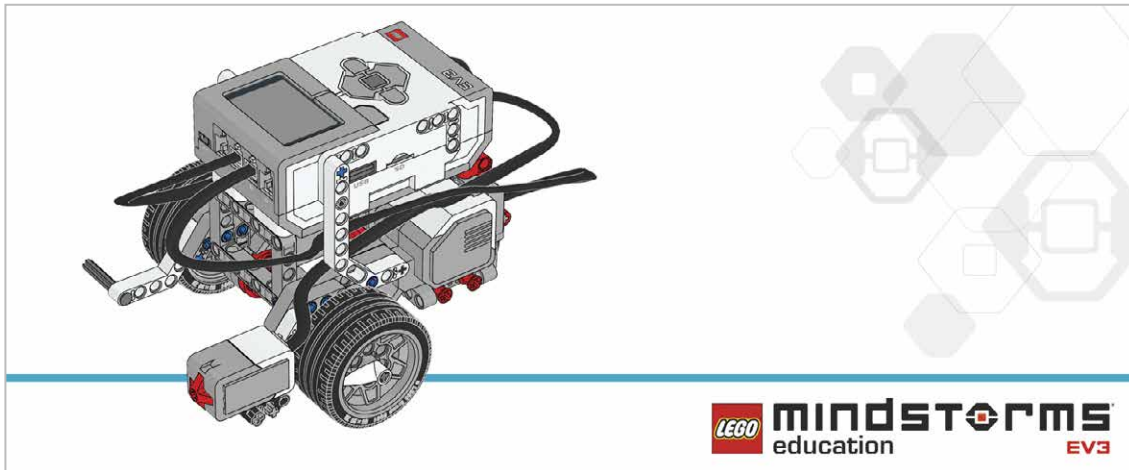
Have the students perform the following building check before they program their robots:
   – Are the wires correctly connected from the motors to ports B and C ?
   – Are the wheels correctly installed?
   – Are the wheels rotating freely?
   – Are the wires correctly connected from the Touch Sensor to port 1 ?
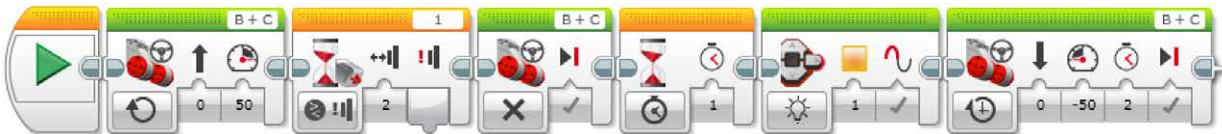
Program

Have the students program their robot to simulate a vehicle reversing, including the use of light indicators.
Have them use the Touch Sensor as a bumper or as a "brake" in order to reverse the robot.
Allow the students to select the tool(s) they find most appropriate for capturing and sharing their pseudocode. Encourage them to use text, videos, images, sketchnotes, or another creative medium.

POSSIBLE SOLUTION
FILENAME: CODING-02.EV3 (Tab:1)

BUMP AND REVERSE

1. Start the Program.
2. Drive forward in a straight line at the desired speed.
3. Wait for the Touch Sensor (simulating changing to reverse gear) to be "bumped".
4. Stop the motors.
5. Wait for 1 second.
6. Activate the "reverse lights" (amber lights on EV3 Brick).
7. Reverse direction of travel for 2 seconds.

Program solutions for this lesson are available for download at:
http://www.LEGOeducation.com

Note

You can use the EV3 Software to demonstrate the Brick Status Light, and the three different modes of the Touch Sensor (i.e., pressed, released, bumped).
Refer students to the Robot Educator Tutorials for further assistance.

In the EV3 Software :
    Robot Educator > Hardware > Brick Status Light
    Robot Educator > Hardware > Brick Display
    Robot Educator > Hardware > Touch Sensor

Set up

Before assigning the next task, make sure that there is enough space for them to complete the task. It is a good idea to have students work on a big table or on the floor.

CONTEMPLATE (35 minutes)

Introduce the use of the button on the EV3 Brick to act as the "reverse" gear.
Ask the students how they think the screen could be used as part of an automated car. Discuss ways in which the screen could be used to relay important information to the car's passengers.
Students will build on their previous learning by creating programs that simulate forward and reverse gears in the following ways:
   – Use the Touch Sensor as a bumper to stop the robot.
   – Use the Brick Buttons to reverse the robot.
   – Have the robot indicate its actions.

Allow the students to select the tool(s) they find most appropriate for capturing and sharing their pseudocode. Encourage them to use text, videos, images, sketchnotes, or another creative medium.

POSSIBLE SOLUTION
FILENAME: CODING-02.EV3 (Tab:2)

BUMP, WAIT FOR REVERSE GEAR
1. Start the program.
2. Drive forward in a straight line at the desired speed.
3. Wait for the Touch Sensor (simulating change to reverse gear) to be "bumped".
4. Stop the motors.
5. Wait for the Brick Button (simulating change to reverse gear) to be pressed.
6. Activate the "reverse lights" (amber lights on the EV3 Brick).
7. Reverse the direction of travel for 2 seconds.

Program solutions for this lesson are available for download at:
http://www.LEGOeducation.com

Differentiation Option

Have the students utilize the EV3 Brick screen as a dashboard indicator.
They will further develop their programs by using the Display Block to incorporate the EV3 Brick screen so that the forward and reverse motion of the wheeled robot will trigger the screen to display arrows indicating the direction of travel.
Have the students integrate other signals, such as:
   – Turn signals (a blinking light when turning left or right)
   – Horn

POSSIBLE SOLUTION
FILENAME: CODING-02.EV3 (Tab:3)
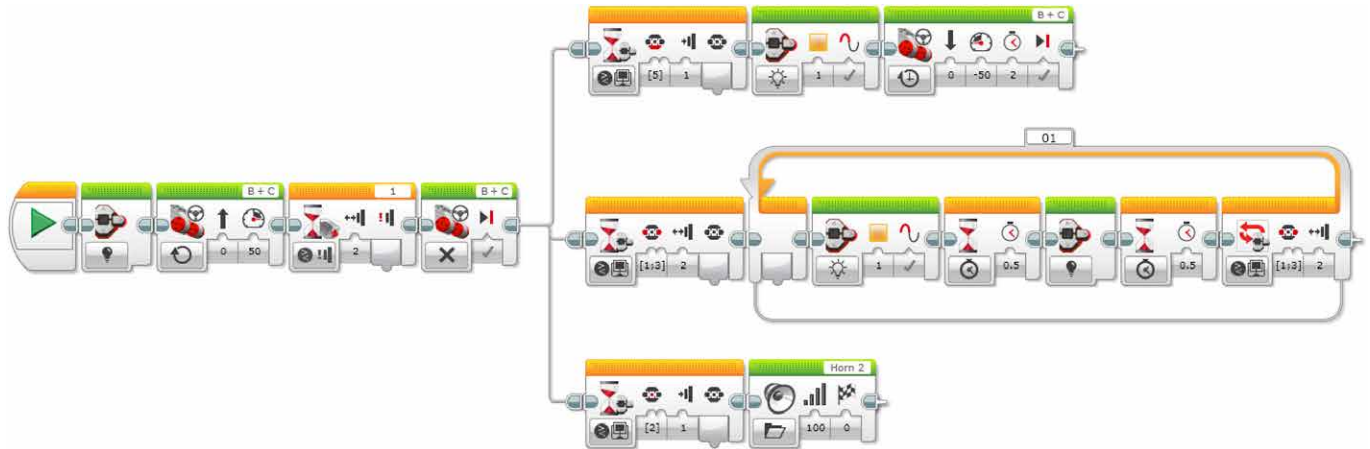
BUMP, WAIT FOR REVERSE GEAR WITH INDICATORS
1. Start the program.
2. Turn off lights on the EV3 Brick.
3. Drive forward in a straight line at the desired speed.
4. Wait for the Touch Sensor (simulating change to reverse gear) to be "bumped".
5. Stop the motors and start 3 simultaneous tasks.
TASK 1
6. Wait for the Down Brick Button (simulating change to reverse gear) to be pressed.
7. Activate the "reverse lights" (amber lights on the EV3 Brick).
8. Reverse the direction of travel for 2 seconds.
TASK 2
9. Wait for the Left or Right Brick Button to be "bumped".
10. Activate the amber lights on the EV3 Brick.
11. Wait for 0.5 seconds.
12. Turn off lights on the EV3 Brick.
13. Wait for 0.5 seconds.
14. Repeat steps x to x until the Left or Right Brick Button is "bumped" again.
TASK 3
15. Wait for Enter Brick Button (simulating horn) to be pressed.
16. Play Horn 2 sound.

Program solutions for this lesson are available for download at:
http://www.LEGOeducation.com

Share

Allow the students to select the tool(s) they find most appropriate for capturing and sharing their creations, unique thinking, and learning process. Encourage them to use text, videos, images, sketchnotes, or another creative medium.
Ask one or two groups to demonstrate their programs and how their wheeled robot moves.
How many variations did the group as a whole come up with? Compare the many possible solutions to the given problem.

Assessment Opportunity
Specific rubrics for assessing computational think can be found under 'Assessment'.

CONTINUE (45 minutes)
Using text-based programming

Have the students explore text-based programming solutions so they can compare different programming languages.

IMPORTANT
The following is a possible solution using the text-based programming language ROBOTC. You may choose to use any other LEGO MINDSTORMS Education EV3 compatible text-based programming languages.

LEGO Education has no ownership of the ROBOTC platform and does not provide any support or guarantee of the quality of the user experience and technology used. All required set up information is provided by ROBOTC at http://www.robotc.net/. We recommend always to reinstall the official LEGO MINDSTORMS EV3 Brick firmware when you finish using other programming languages.

```c
#pragma config(StandardModel, "EV3 _ REMBOT")
/*
Create a program where the robot drives forward until the Touch Sensor
is pressed. The robot then reverses flashing an orange LED.
*/
task main()
{
  //Set the MotorSpeed to 50%.
  setMotorSpeed(motorB, 50);
  setMotorSpeed(motorC, 50);
  //Wait for touch sensor to be pressed.
  while(getTouchValue(touchSensor) ==0)
  {
      sleep(10);
  }
  //Set the MotorSpeed to 0% or off. Wait 1 second.
  setMotorSpeed(motorB, 0);
  setMotorSpeed(motorC, 0);
  sleep(1000);
  //Flash the EV3 LED Orange.
  setLEDColor(ledOrangeFlash);
  //Set the MotorSpeed to -50%(reverse). Wait 2 seconds.
  setMotorSpeed(motorB, -50);
  setMotorSpeed(motorC, -50);
  sleep(2000);
}
```

Program solutions for this lesson are available for download at:
http://www.legoeducation.com

WHAT IS NEXT?
Automatic Headlights