

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two Motors to move the car. Use the Joystick to receive input from the user to go straight, turn left or right.
Adjust the speed of the Motors accordingly.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

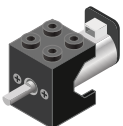
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

1x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the RC car.

2x  Wheel - attach to motor.

1x  Joystick - to the direction of the RC car.

Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

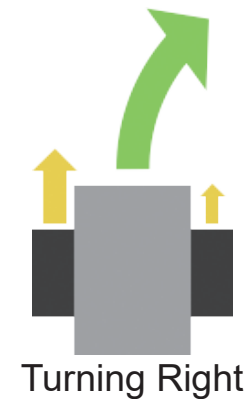
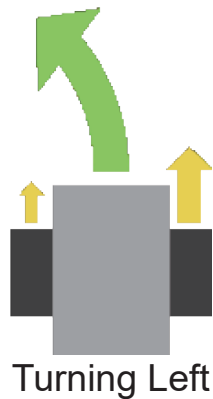
Use two Motors to move the RC car.

Use the Joystick to input the direction to move the car (left, right, up, down).

Depending on the Joystick's direction being held down:

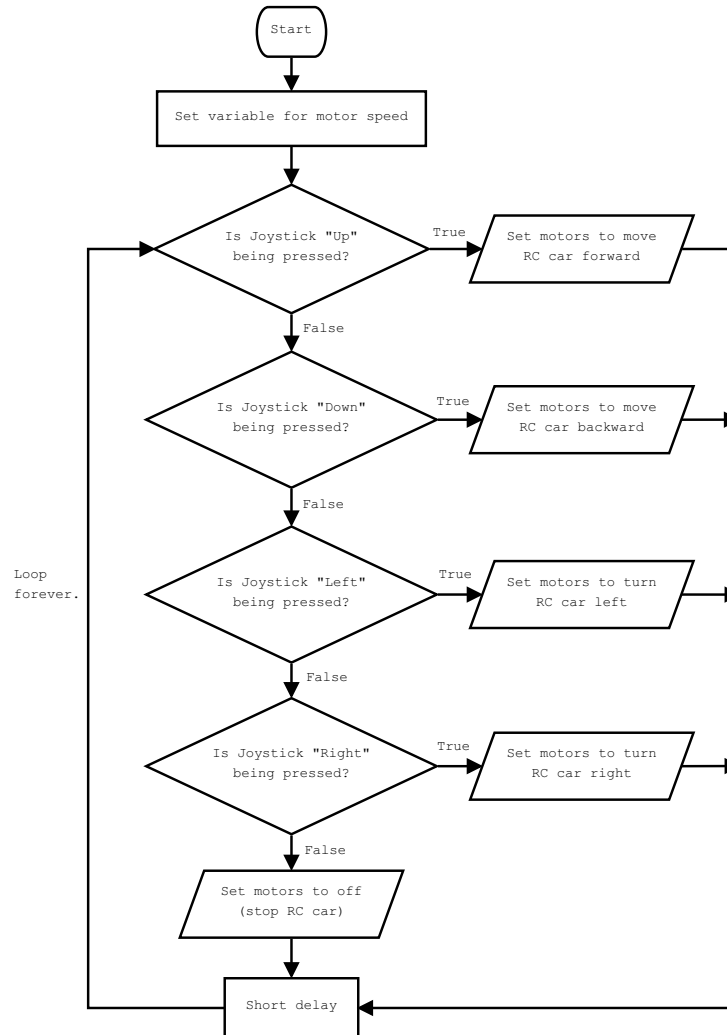
- Up - move car forward (both motors rotating forward).
- Down - move car backward (both motors rotating backwards).
- Left - move car left (motor 1 rotating forward slowly, motor 2 rotating forward fast).
- Right - move car right (motor 1 rotating forward fast, motor 2 rotating forward slowly).
- None - stop car (both motors turned off).

For Left and Right directions, the difference in motor speeds causes the car to turn.



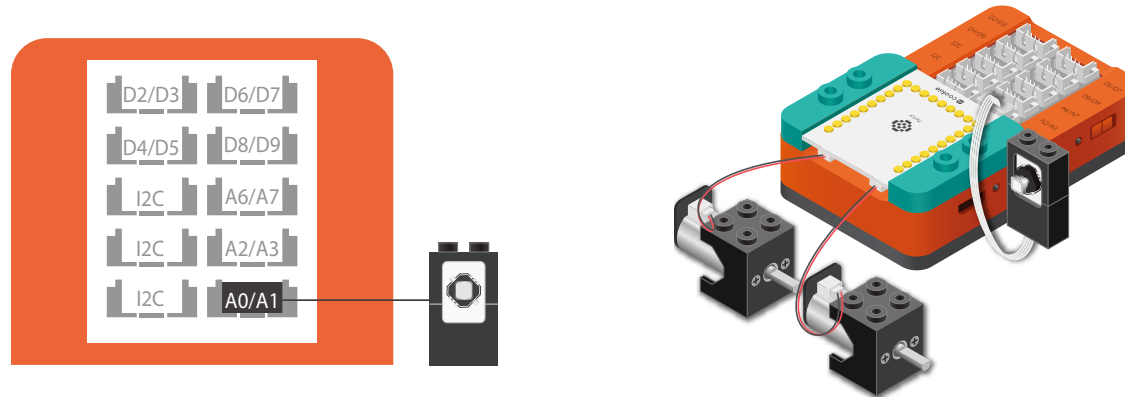
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

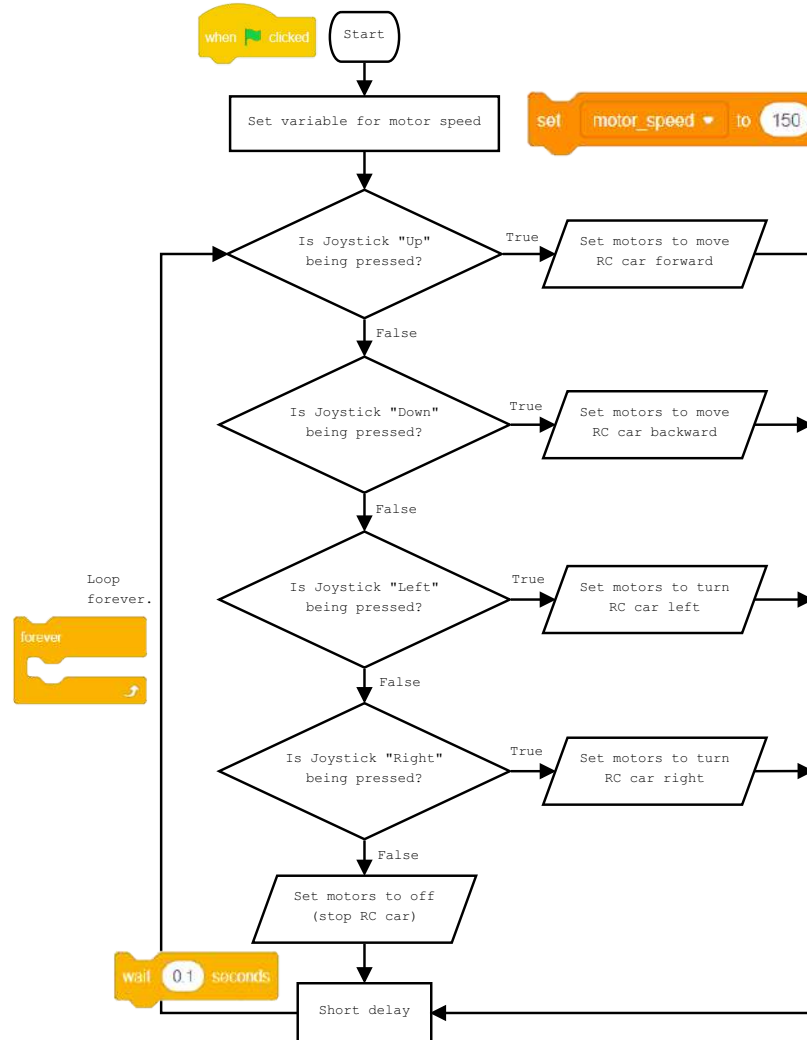
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect the Joystick to pin A0/A1.
5. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Set the variable for the speed of the motors.

Check if the Joystick (pin A0) “Up” key is being pressed.

Set motors to move car forward.

Check if the Joystick (pin A0) “Down” key is being pressed.

Set motors to move car backward.

Check if the Joystick (pin A0) “Left” key is being pressed.

Set motors to turn car left.

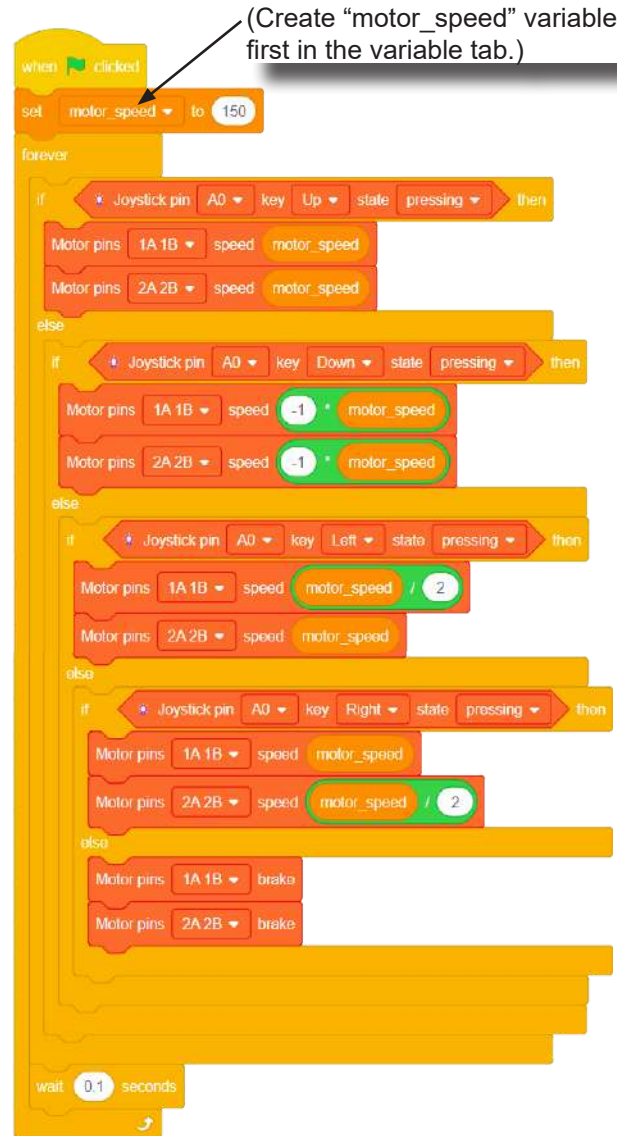
Check if the Joystick (pin A0) “Right” key is being pressed.

Set motors to turn car right.

Otherwise (no key being pressed), turn motors off.

Short delay before looping.

Loop.



Block Locator

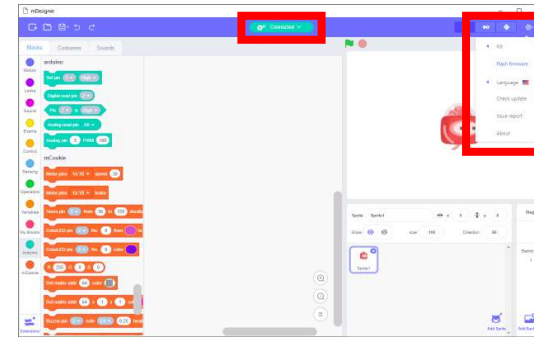


Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

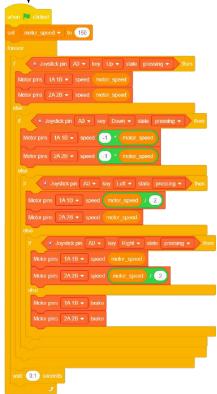
Select Serial Port



Flash firmware

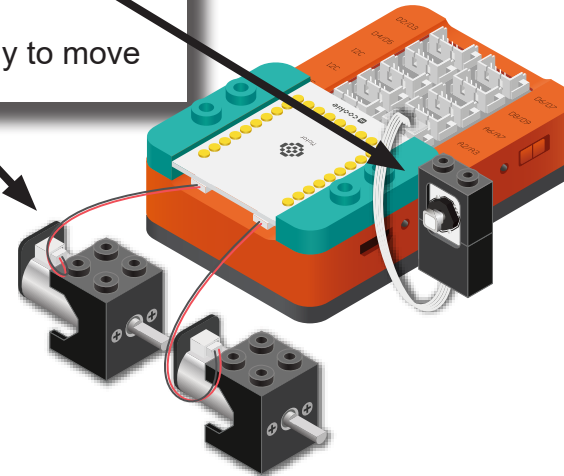
Testing:

Click the green flag to activate the script.



1. Press and hold the Joystick in the Up, Down, Left or Right position.

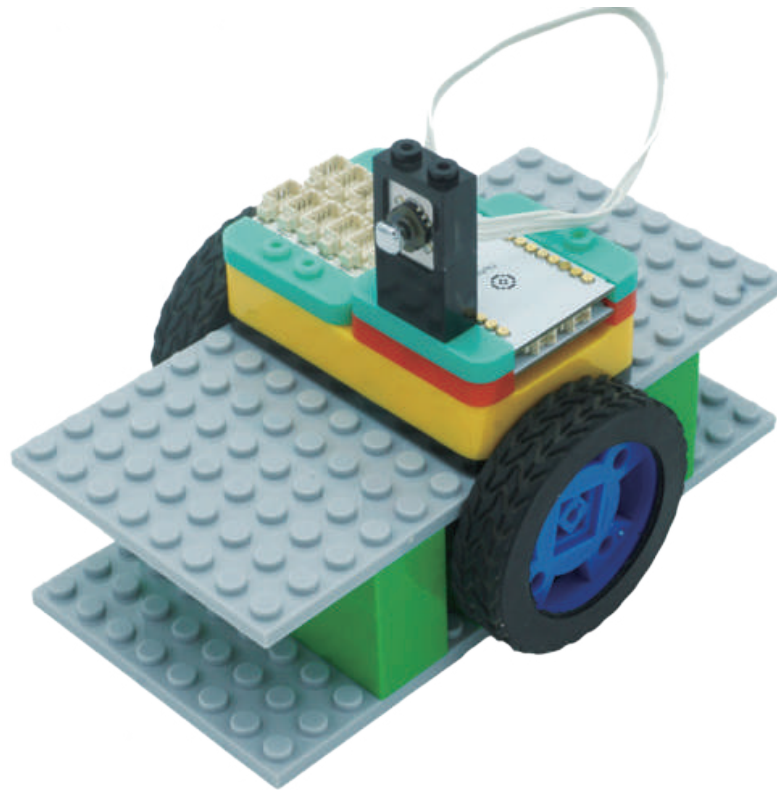
2. The Motors will turn accordingly to move the RC car.



Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

Create a structure for the project using building blocks, paper, or other crafting materials.



Project Challenge

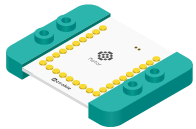
How can you modify the speed in which the car moves?

Adjust the turning radius (quicker turning).

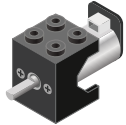
Add another Joystick input ("Center" pressing).

Make this input to cause the car to spin (motor 1 forward, motor 2 backward).

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



Joystick - input module (sensor) which can detect the position of the stick (**up**, **down**, **left**, **right**, **center** (press)).



Joystick State block - returns TRUE if the position of the stick state matches the checking state. Otherwise returns FALSE.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about the history of RC cars. Besides being controllable directly by a wired Joystick, what other methods are used to control RC cars?



WHAT YOU WILL NEED



M-CENTER +



MOTOR
DRIVER



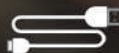
MOTOR



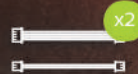
WHEEL



LINE
FINDER



USB CABLE



SENSOR CABLE

MIX4B07

Safety Car

Create a car which moves and stops if there is a nearby object in the front or back.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:



 MICRODUINO

About Collision Avoidance

A collision avoidance system is an automobile safety system designed to reduce the severity of a collision. It is also known as a precrash system, forward collision warning system, or collision mitigating system. It uses radar (all-weather) and sometimes laser (LIDAR) and camera (employing image recognition) to detect an imminent crash. GPS sensors can detect fixed dangers such as approaching stop signs through a location database.

Once the detection is done, these systems either provide a warning to the driver when there is an imminent collision or take action autonomously without any driver input (by braking or steering or both). Collision avoidance by braking is appropriate at low vehicle speeds (e.g. below 50 km/h), while collision avoidance by steering is appropriate at higher vehicle speeds. Cars with collision avoidance may also be equipped with adaptive cruise control, and use the same forward-looking sensors.



Review of Modules

Line Finder



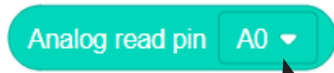
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

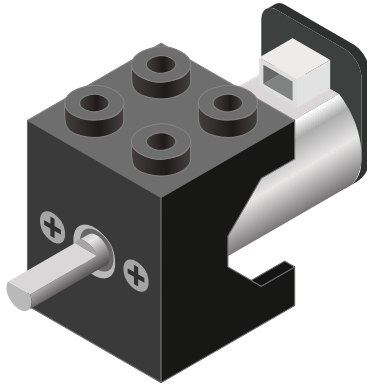


Pin that Line Finder
is connected to.

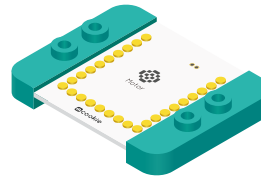
This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to two Motors. A Wheel can be attached to the Motor.



Motor Controller



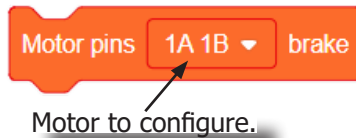
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



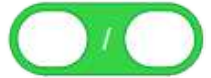
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

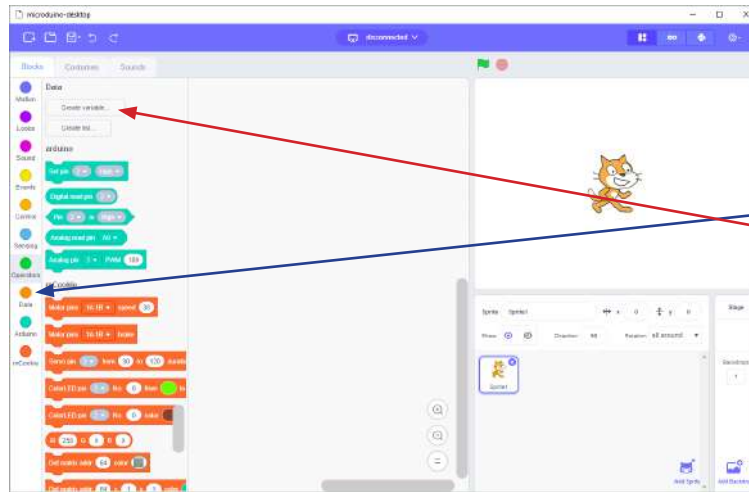
Greater Than Block



This block returns TRUE if the left input is greater than the right input.

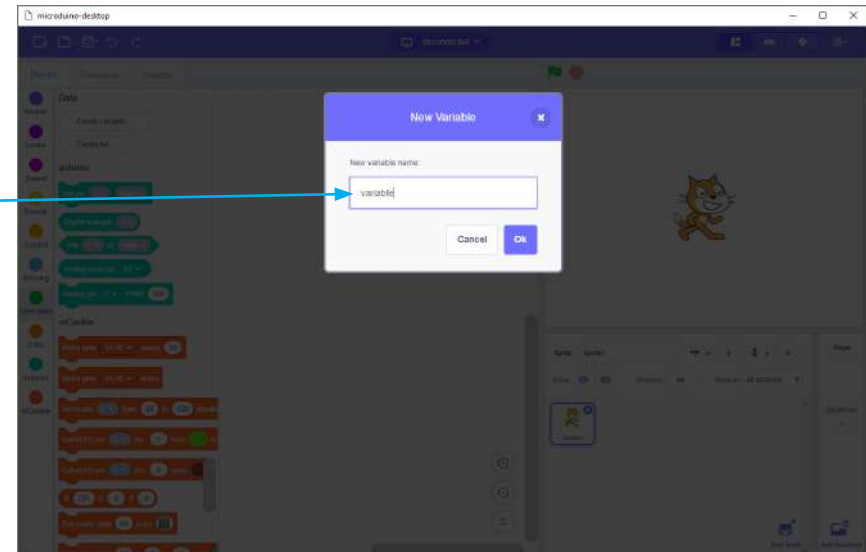
Review of Concepts

Creating a Variable in Scratch



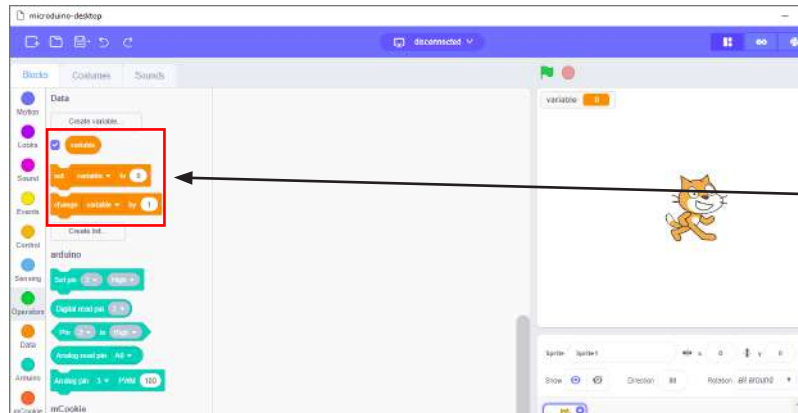
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can we create an autonomous car which moves forward until it reaches an object, then moves backwards until it reaches an object, then repeats?

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two Line Finders to detect obstacles (objects) in the front and back of the car. Use two motors to move the car.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

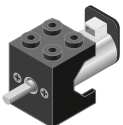
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

2x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the RC car.

2x  Wheel - attach to motor.

2x  Line Finder - detect objects in the front and back of the car.

Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the RC car.

Use two Line Finders (one in front, one in back) to detect objects in the way.

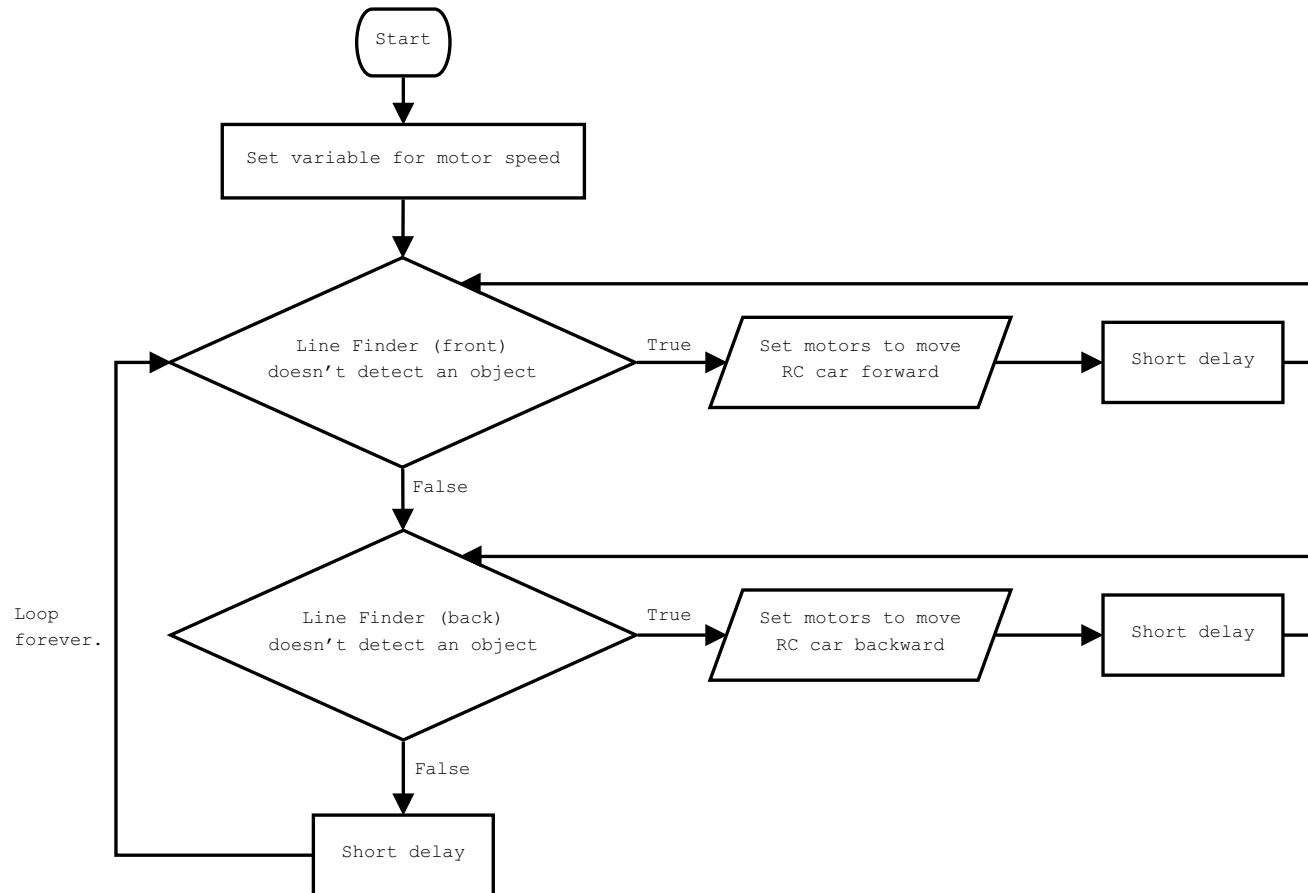
Move forward until an object is detected (Line Finder in front value < 700).

Move backward until an object is detected (Line Finder in back value < 700).

Repeat.

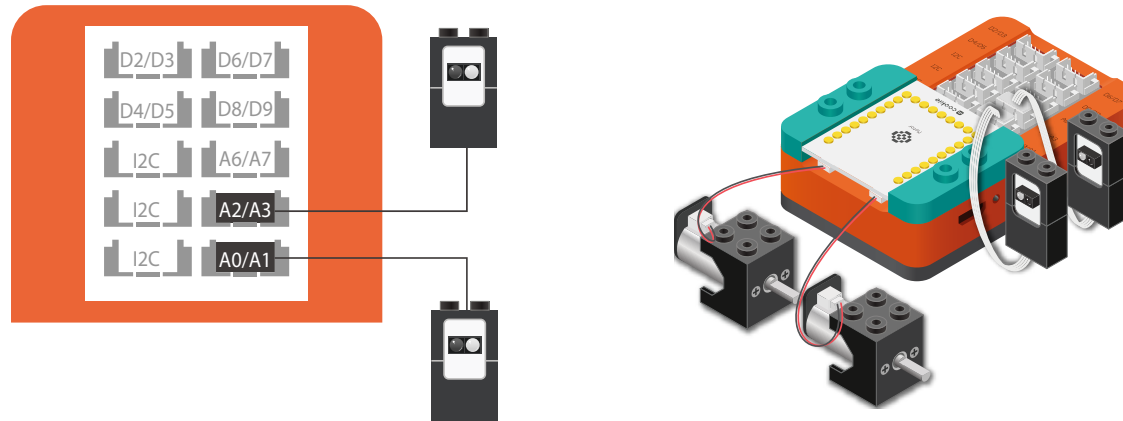
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

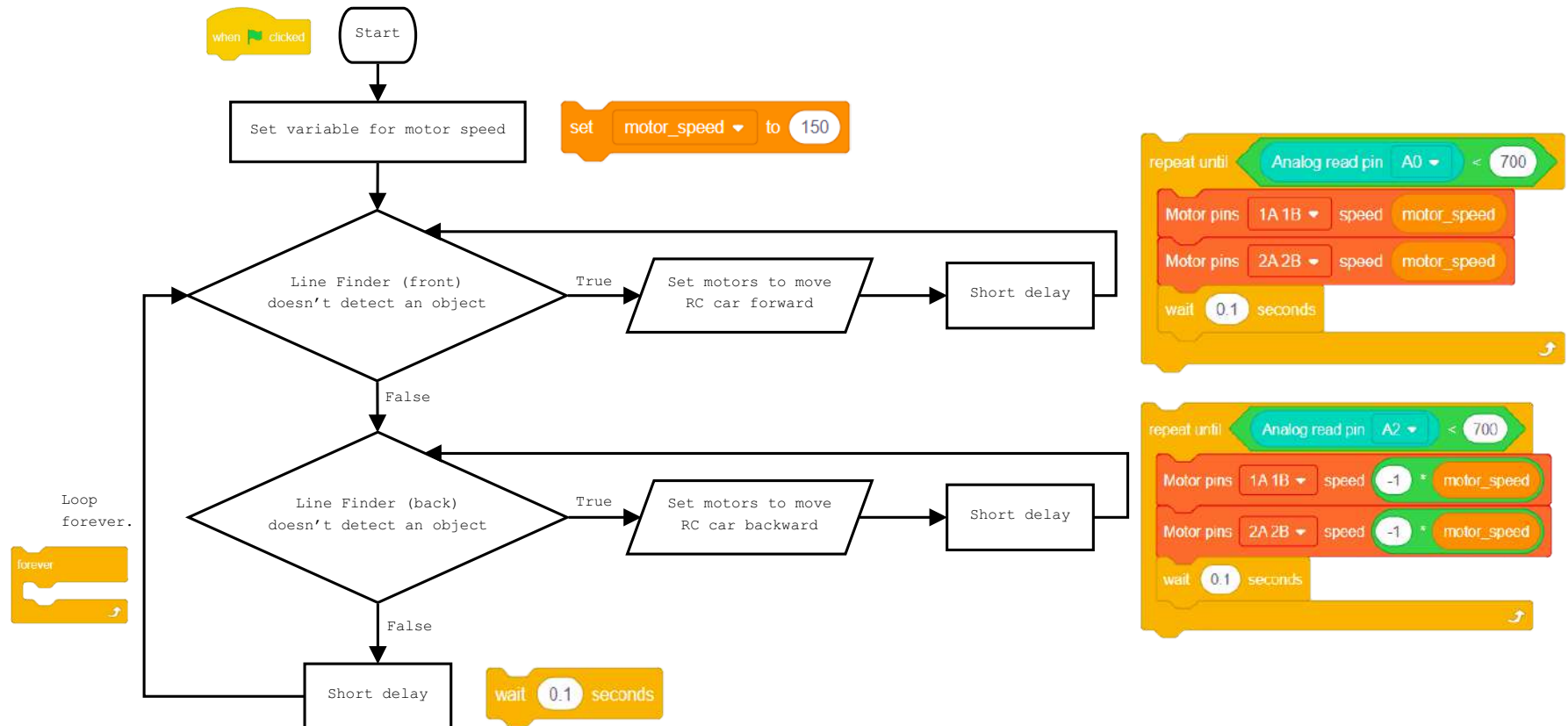
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Line Finder to pin A0/A1.
5. Connect a Line Finder to pin A2/A3.
6. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Set the variable for the speed of the motors.

Repeat until Line Finder (A0) in front detects an object.

Set motors to move car forward.

Short delay before looping.

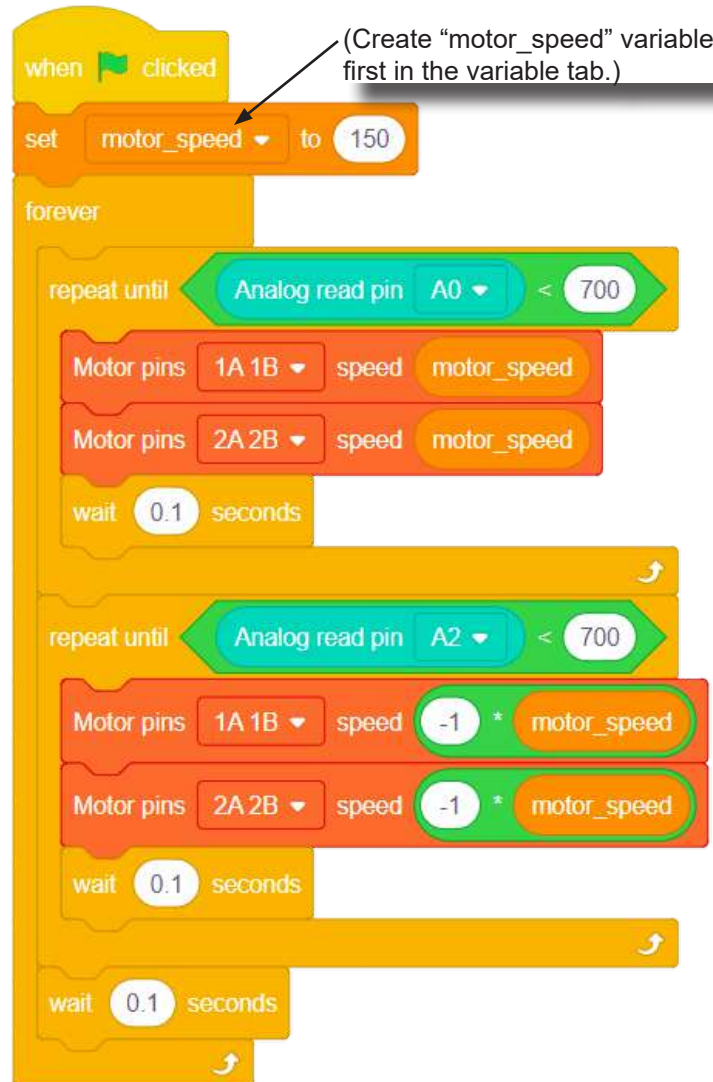
Repeat until Line Finder (A2) in back detects an object.

Set motors to move car backward.

Short delay before looping.

Short delay before looping.

Loop.



Block Locator

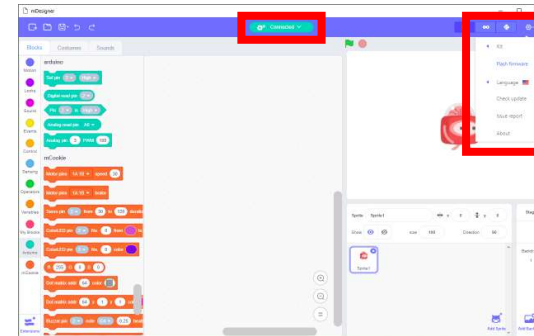


Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

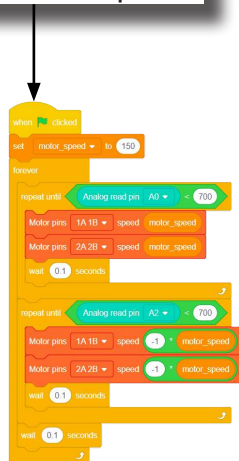
Select Serial Port



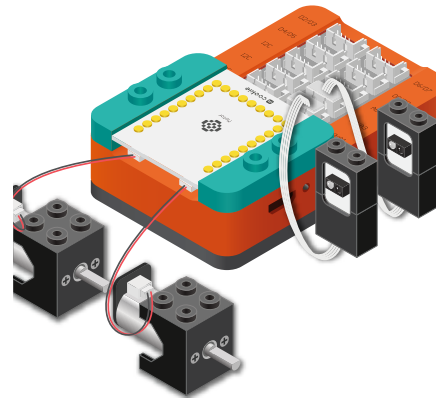
Flash firmware

Testing:

Click the green flag to activate the script.



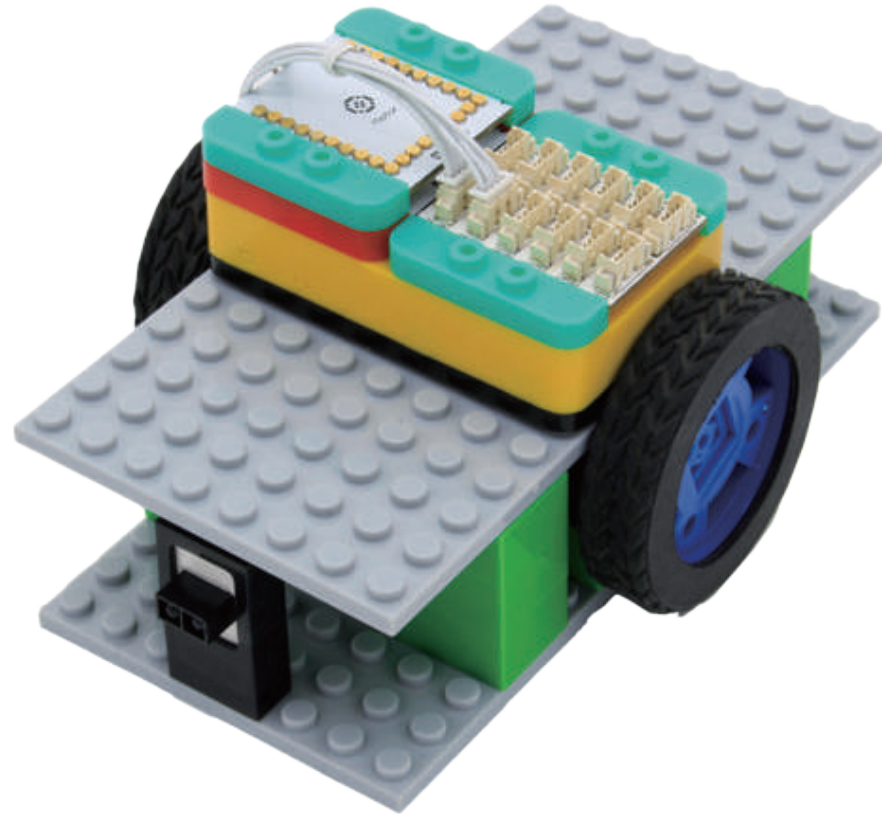
1. The car will move forward until the Line Finder (front) detects an object.
2. The car will then move backward until the Line Finder (back) detects an object.
3. Repeat.



Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

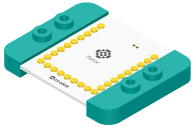
Create a structure for the project using building blocks, paper, or other crafting materials.



Project Challenge

Modify the script to use “If Else” blocks instead of “Repeat Until” blocks.
Add a third condition check to stop the car if both Line Finders detect an object.

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



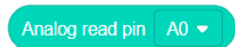
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about obstacle avoidance systems in cars.

What are some interesting techniques that are used in these systems?



WHAT YOU WILL NEED



MCENTER +



IO SPLIT



MOTOR DRIVER



MOTOR



WHEEL



LINE
FINDER



USB CABLE



SENSOR CABLE

MIX4B08

Delivery Truck

Build a toy freight car that will ship packages back and forth.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:

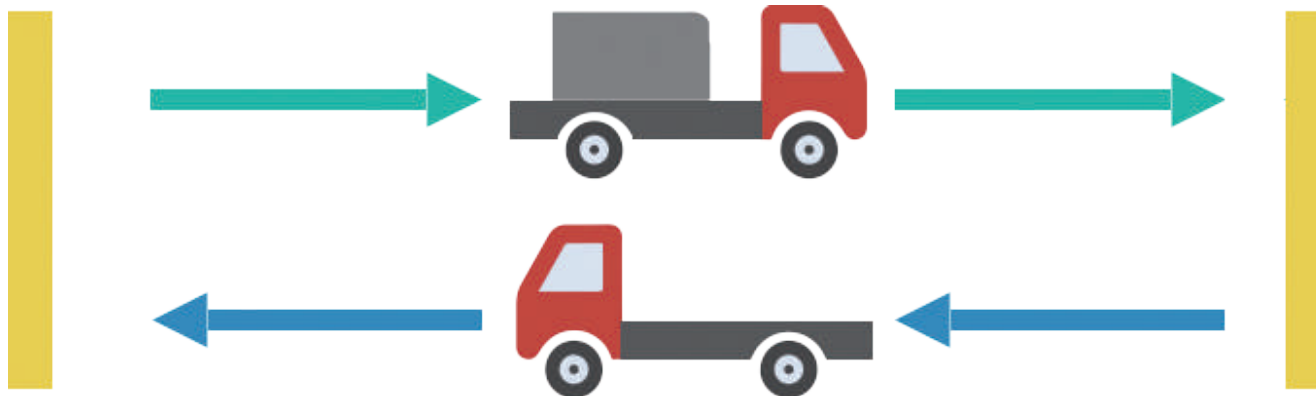


 MICRODUINO

About Deliveries

Delivery is the process of transporting goods from a source location to a predefined destination. There are different delivery types. Cargo (physical goods) are primarily delivered via roads and railroads on land, shipping lanes on the sea and airline networks in the air. Certain specialized goods may be delivered via other networks, such as pipelines for liquid goods, power grids for electrical power and computer networks such as the Internet or broadcast networks for electronic information.

The general process of delivering goods is known as distribution. The study of effective processes for delivery and disposition of goods and personnel is called logistics. Firms that specialize in delivering commercial goods from point of production or storage to point of sale are generally known as distributors, while those that specialize in the delivery of goods to the consumer are known as delivery services. Postal, courier, and relocation services also deliver goods for commercial and private interests.



Review of Modules

Line Finder



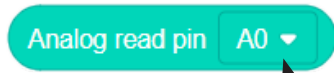
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

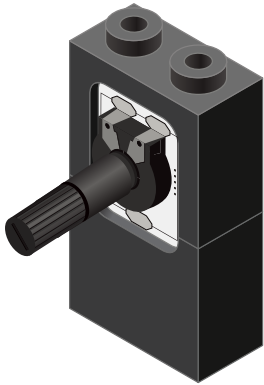


Pin that Line Finder
is connected to.

This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Potentiometer



A Potentiometer is an input module which can be turned or rotated right or left. Based on the position of the knob, it produces a different signal value for the Core module to read. It must be connected to an analog port (ports denoted with "A").

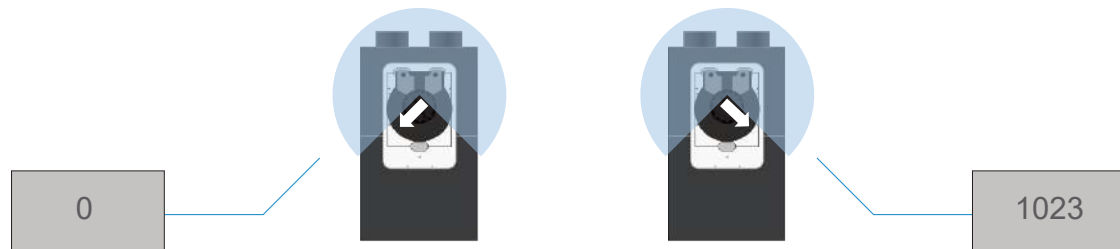
Analog Read block

Analog read pin

A0 ▼

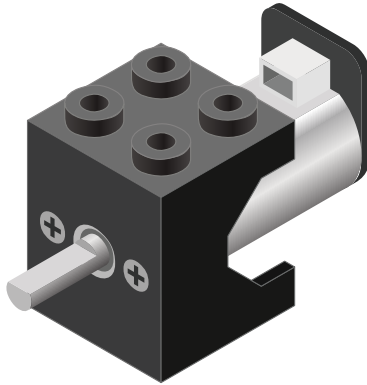
Pin the Potentiometer is connected to.

This block returns the value from the Potentiometer. A value between 0 and 1023 is returned. If the potentiometer is turned all the way to the left, it produces a 0 value. If it is turned all the way to the right, it produces a 1023 value. In the middle it produces a value around 512.

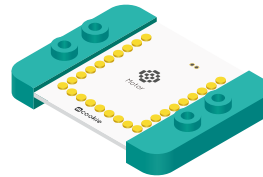


Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to two Motors. A Wheel can be attached to the Motor.



Motor Controller



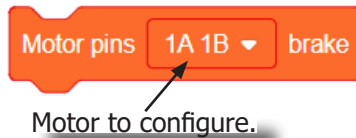
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



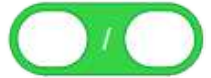
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

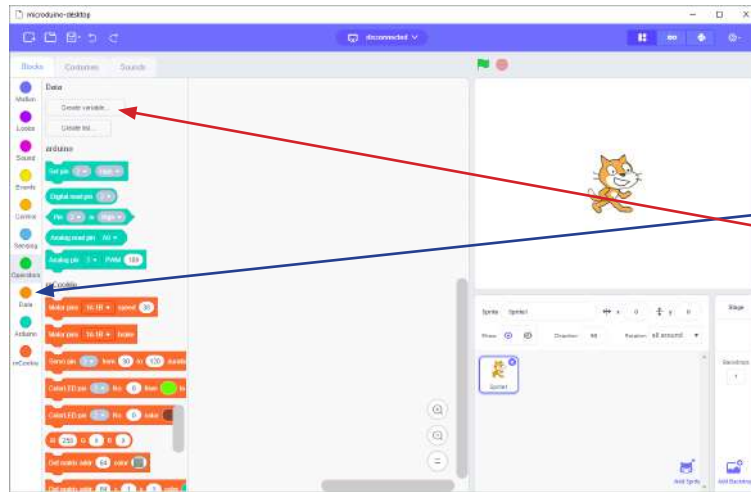
Greater Than Block



This block returns TRUE if the left input is greater than the right input.

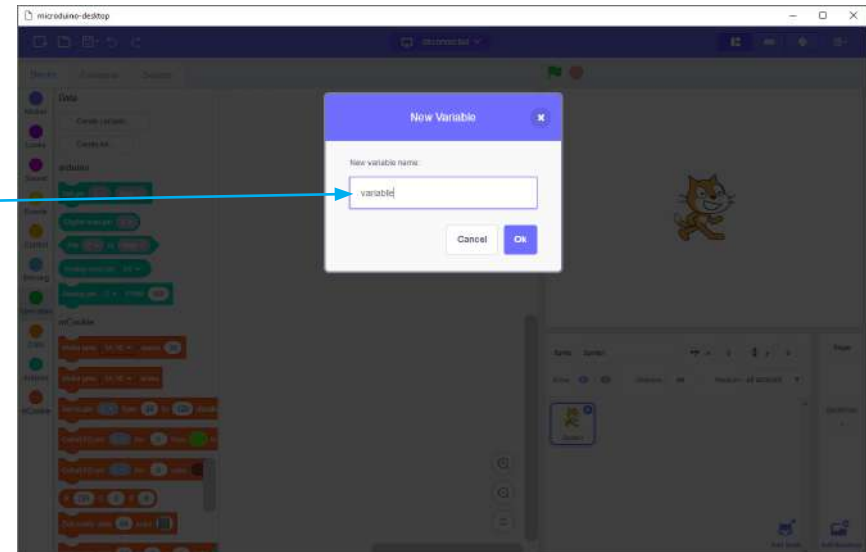
Review of Concepts

Creating a Variable in Scratch



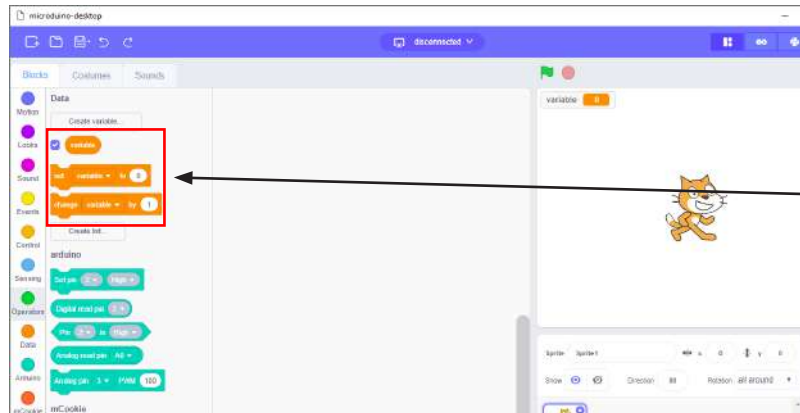
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can we create a delivery truck with adjustable speeds?

Which waits for something to be loaded, then drives it to the destination, waits for it to be unloaded, then drives back.

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two motors to move the delivery truck. Use a Line Finder to detect if cargo is present or not. Use a Line Finder (front) to check if the destination is reached. Use a Line Finder (back) to check if returned back to base.

Use a Potentiometer to adjust the speed of the truck.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

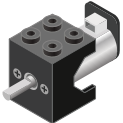
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

5x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the delivery truck.

2x  Wheel - attach to motor.

3x  Line Finder - detect objects in the front and back of the car and if cargo is present.

1x  Potentiometer - to adjust speed of the truck.

1x  IO Splitter - to connect two sensors/trinkets to a single connector.

Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the delivery truck.

Use a Line Finder in the cargo holding location to detect if cargo is loaded or unloaded.

Use a Line Finder in the front to detect if the destination has been reached.

Use a Line Finder in the back to detect if returned to base.

Use a Potentiometer to adjust the speed of the delivery truck.

Wait until cargo is loaded onto truck (Line Finder in cargo holding < 200).

Wait 1 second before driving to destination.

Move forward* until destination is reached (Line Finder in front < 700).

Wait until cargo is unloaded from truck (Line Finder in cargo holding > 400).

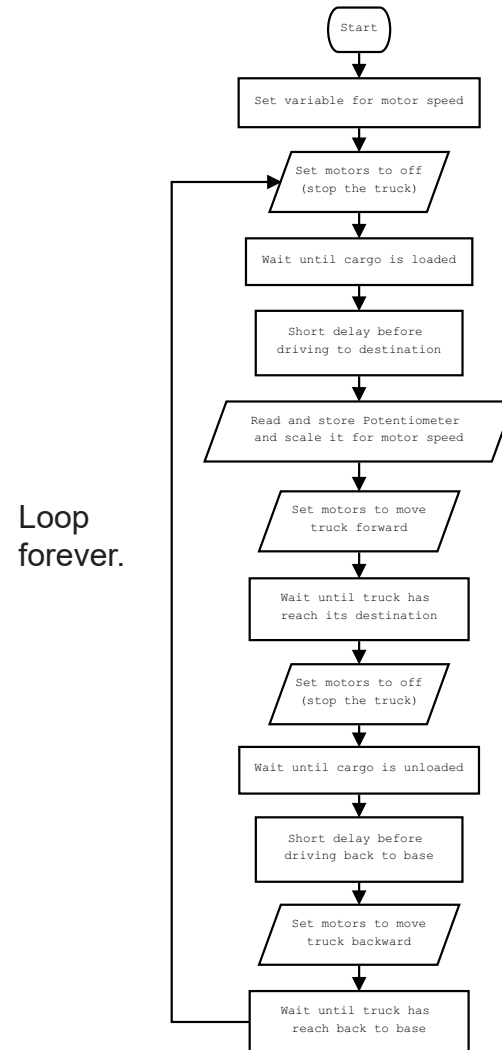
Wait 1 second before driving back to base.

Move backward* until back to base (Line Finder in back < 700).

*When setting the speed to move forward or backward, read the Potentiometer (value between 0 and 1023) first, then scale (divide by 4, $1023/4 \approx 255$) and use that as the speed.

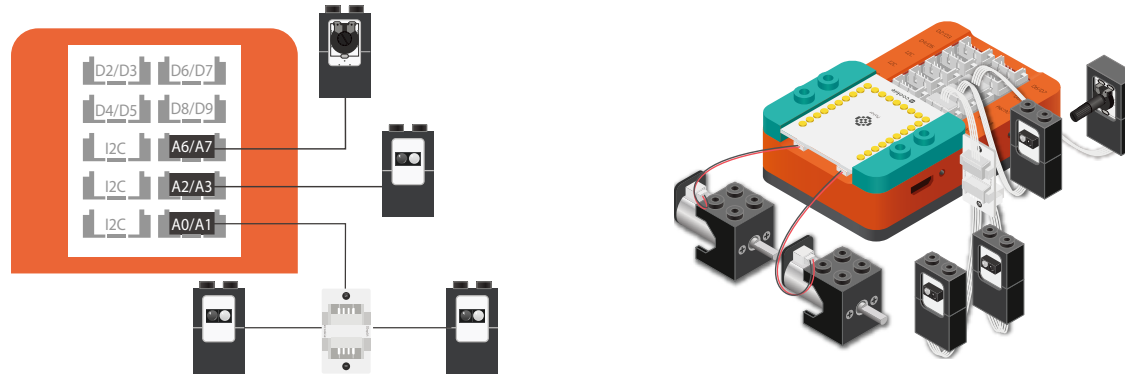
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

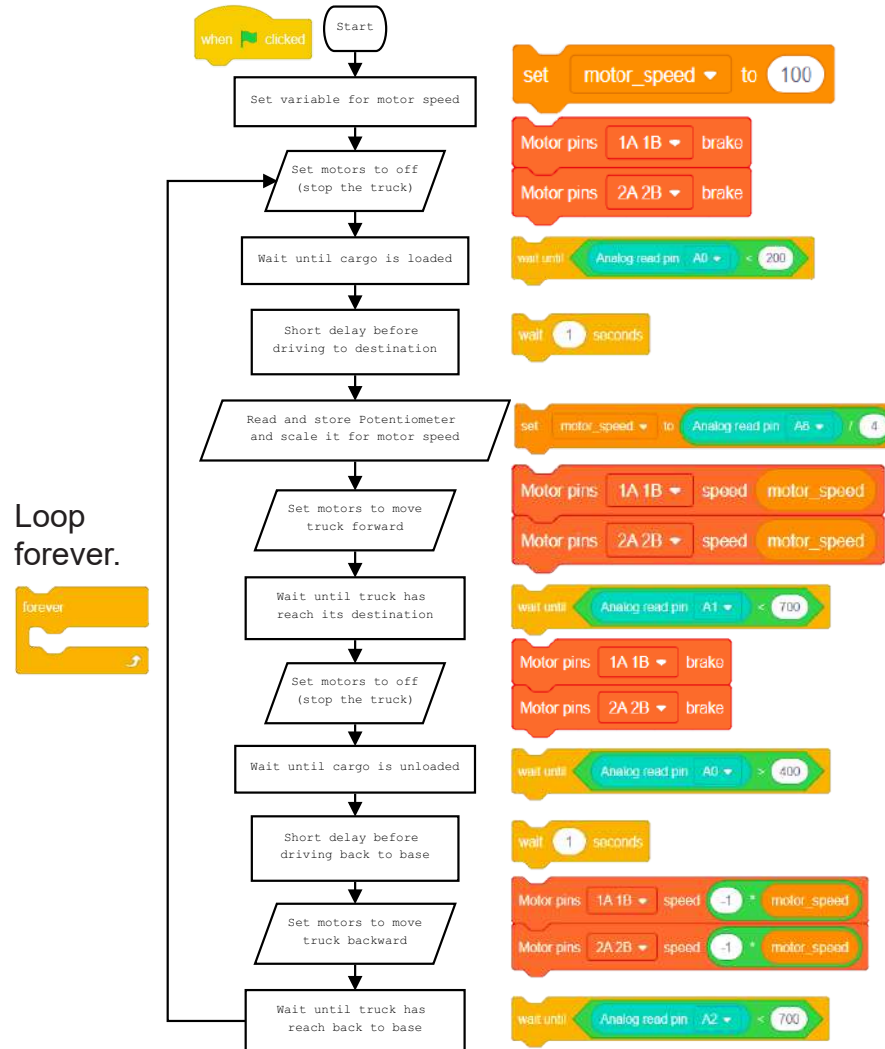
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Line Finder to pin A2/A3.
5. Connect a Potentiometer to pin A6/A7
6. Connect two Line Finders to an IO Splitter. Connect the IO Splitter to pin A0/A1.
7. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Set the variable for the speed of the motors.

Turn motors off (stop the truck).

Wait until Line Finder (A0) detects cargo (loaded).

Wait 1 second before driving to destination.

Scale and store Potentiometer (pin A6) value to motor speed.

Set motors to move truck forward.

Wait until Line Finder (A1) in front detects reaching destination.

Turn motors off (stop the truck).

Wait until Line Finder (A0) detects no cargo (unloaded).

Wait 1 second before driving back to base.

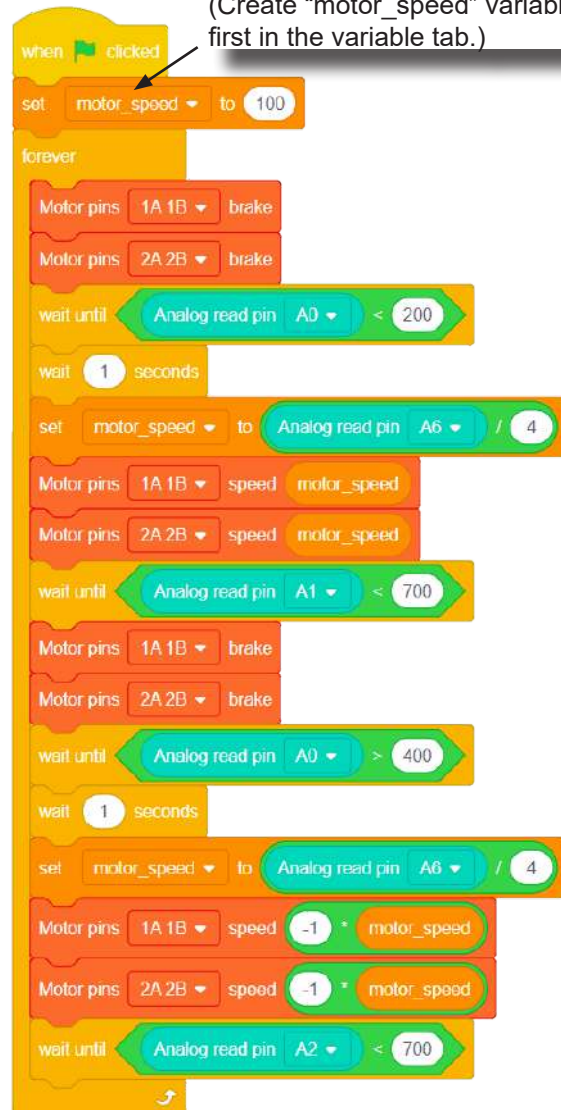
Scale and store Potentiometer (pin A6) value to motor speed.

Set motors to move truck backward.

Wait until Line Finder (A2) in back detects returning to base.

Loop.

(Create "motor_speed" variable first in the variable tab.)



Block Locator

Events



Control



Operators



Arduino



Data



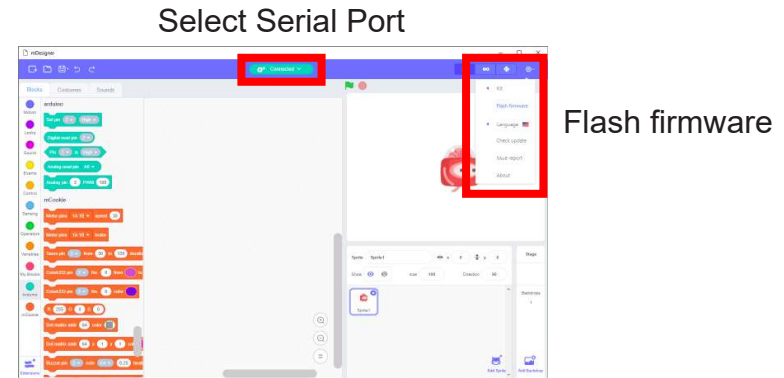
mCookie



Testing the Program

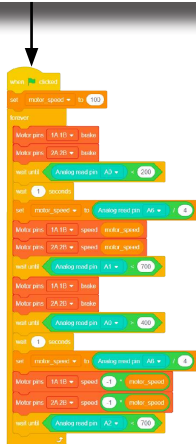
Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

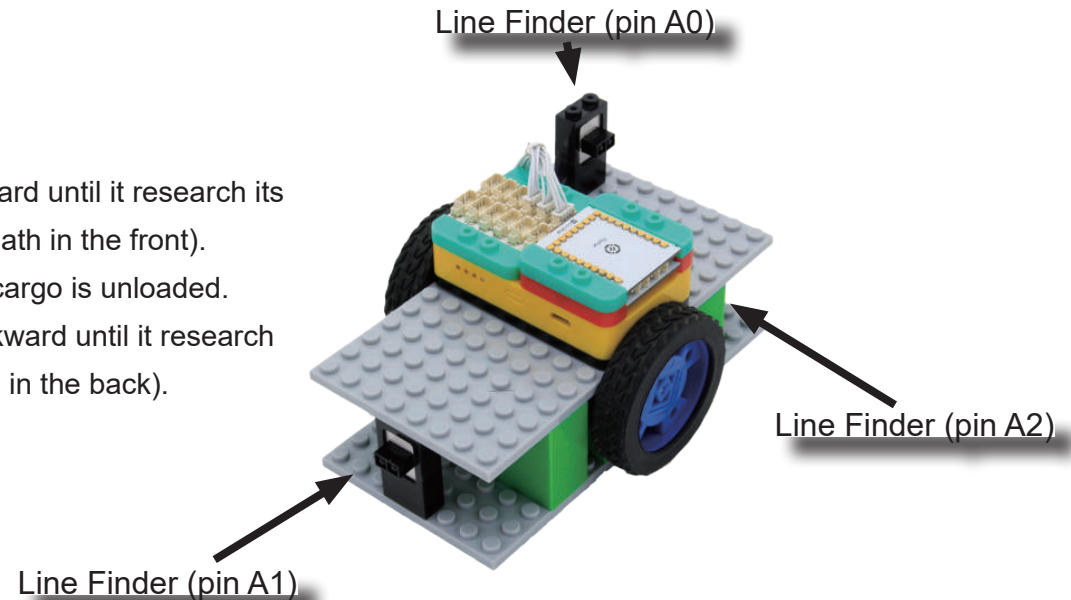


Testing:

Click the green flag to activate the script.



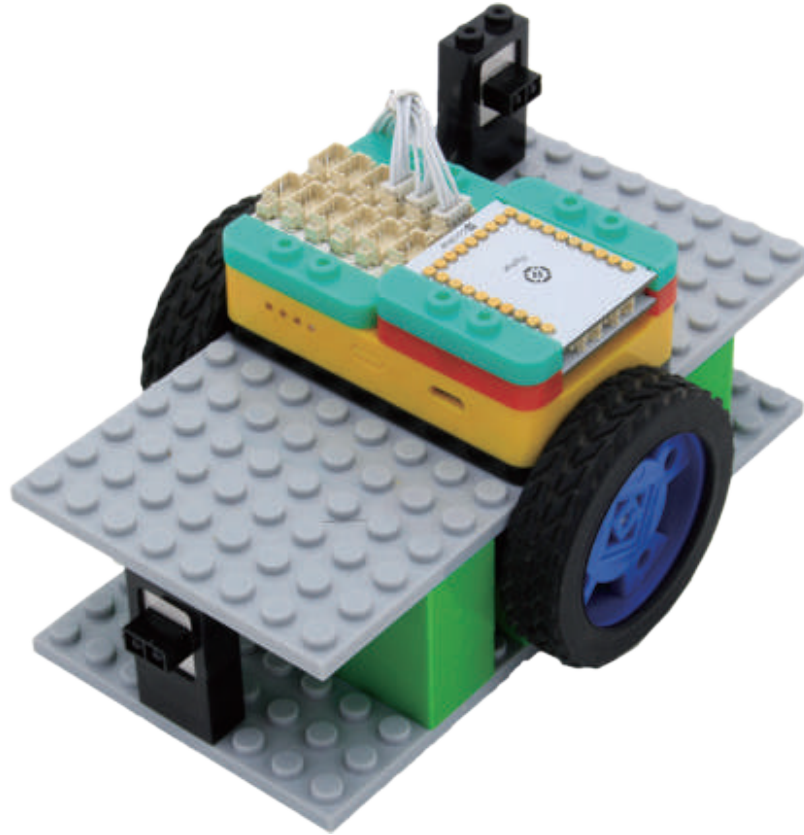
1. Load cargo into the truck.
2. The truck will then move forward until it research its destination (object blocking its path in the front).
3. Truck will stop and wait until cargo is unloaded.
4. The truck will then move backward until it research its base (object blocking its path in the back).



Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

Create a structure for the project using building blocks, paper, or other crafting materials.

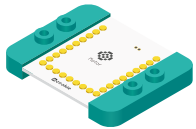


Project Challenge

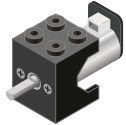
Add a Color LED to your project.

Turn set the Color LED's based on the state the delivery truck.
(Examples: loaded cargo, stopped, moving, unloaded cargo.)

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



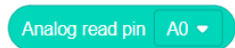
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



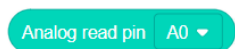
Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.



Potentiometer - input module with an adjustable knob.



Analog Read block - reads the value from the potentiometer. Returns a value between 0 and 1023.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about delivery trucks.

What are some important things delivery trucks must accomplish to be profitable?



WHAT YOU WILL NEED



MCENTER +



MOTOR DRIVER



IO SPLIT



MOTOR



WHEEL



LINE
FINDER



POTENTIOMETER



USB CABLE



SENSOR CABLE

MIX4B09

Obstacle Course Car

Use the four line finders to find and avoid nearby obstacles.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:

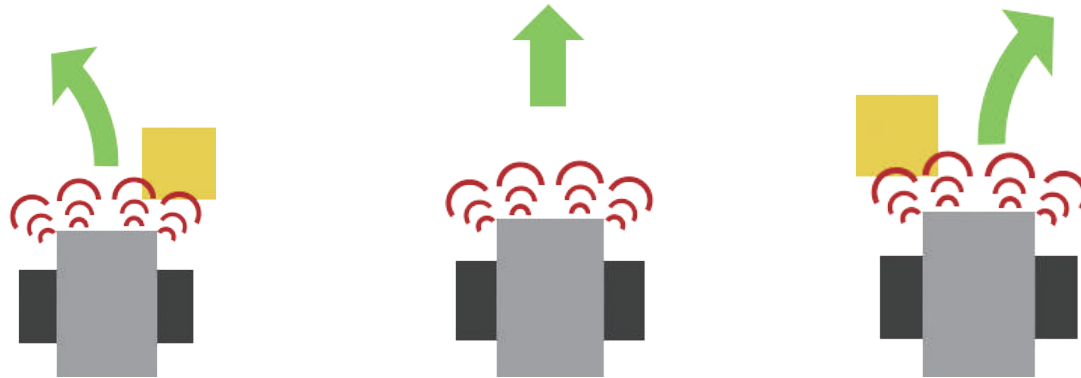


 MICRODUINO

About Obstacle Avoidance

In robotics, obstacle avoidance is the task of satisfying some control objective subject to non-intersection or non-collision position constraints.

What is critical about obstacle avoidance concept in this area is the growing need of usage of unmanned aerial vehicles in urban areas for especially military applications where it can be very useful in city wars. Normally obstacle avoidance is considered to be distinct from path planning in that one is usually implemented as a reactive control law while the other involves the pre-computation of an obstacle-free path which a controller will then guide a robot along.



Review of Modules

Line Finder



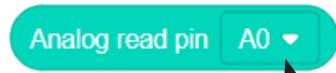
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

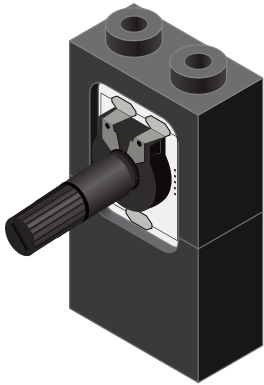


Pin that Line Finder
is connected to.

This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Potentiometer



A Potentiometer is an input module which can be turned or rotated right or left. Based on the position of the knob, it produces a different signal value for the Core module to read. It must be connected to an analog port (ports denoted with "A").

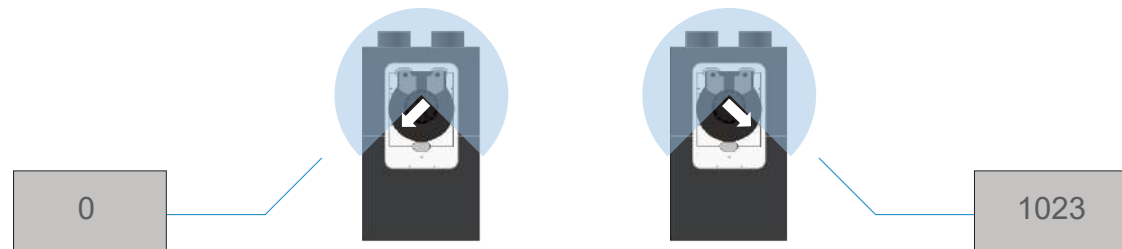
Analog Read block

Analog read pin

A0 ▼

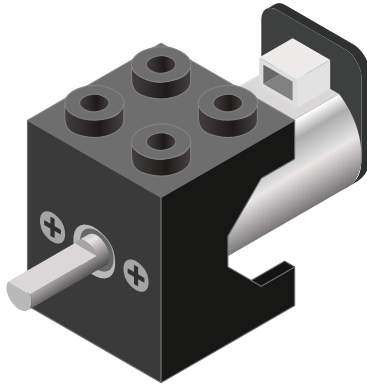
Pin the Potentiometer is connected to.

This block returns the value from the Potentiometer. A value between 0 and 1023 is returned. If the potentiometer is turned all the way to the left, it produces a 0 value. If it is turned all the way to the right, it produces a 1023 value. In the middle it produces a value around 512.

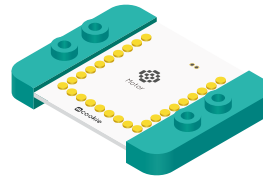


Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to 2 Motors. A Wheel can be attached to the Motor.



Motor Controller



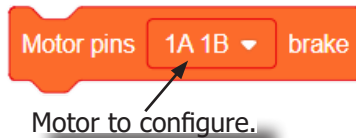
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



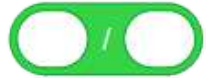
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

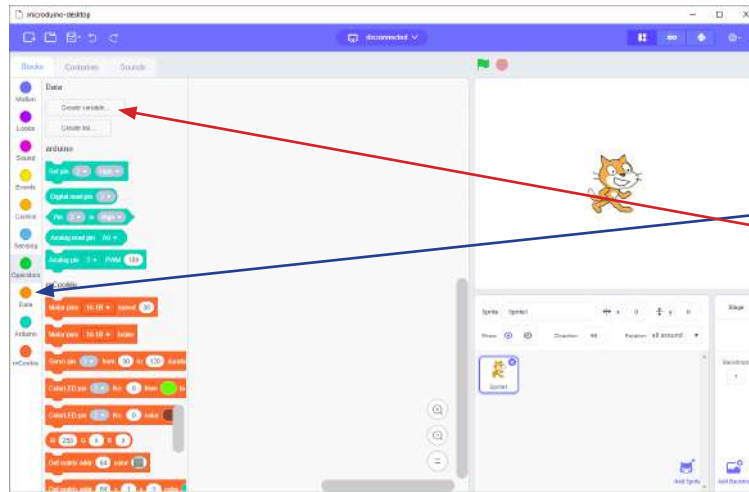
Greater Than Block



This block returns TRUE if the left input is greater than the right input.

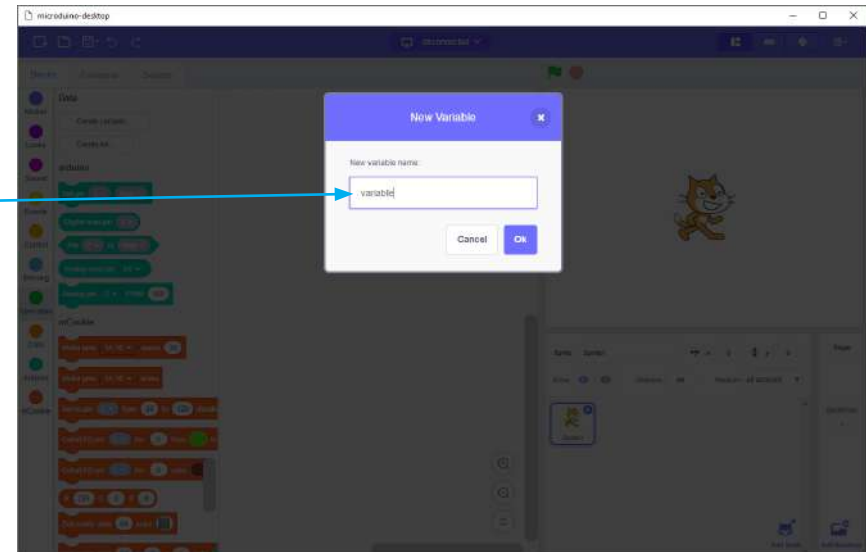
Review of Concepts

Creating a Variable in Scratch



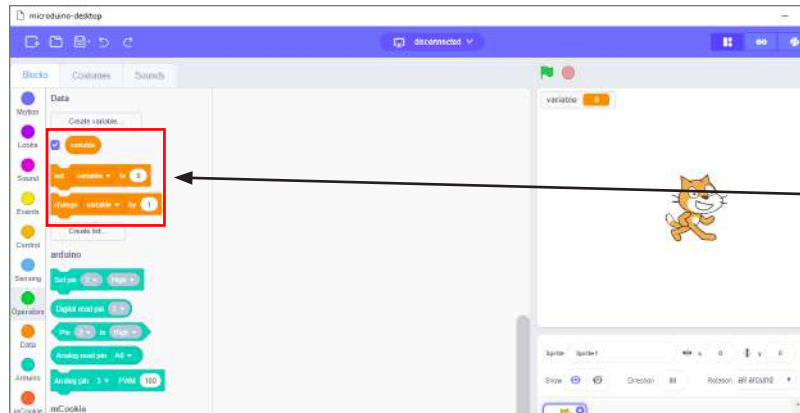
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can we create a car which can detect and avoid objects in the way?

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two motors to move the car. Use 4 Line Finder positioned in front of the car to detect oncoming objects. Use this information to steer the car away to avoid the object. Use a Potentiometer to set the motor speeds.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

7x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the car.

2x  Wheel - attach to motor.

4x  Line Finder - detect objects oncoming objects.

1x  Potentiometer - to adjust speed of the car.

2x  IO Splitter - to connect two sensors/trinkets to a single connector.

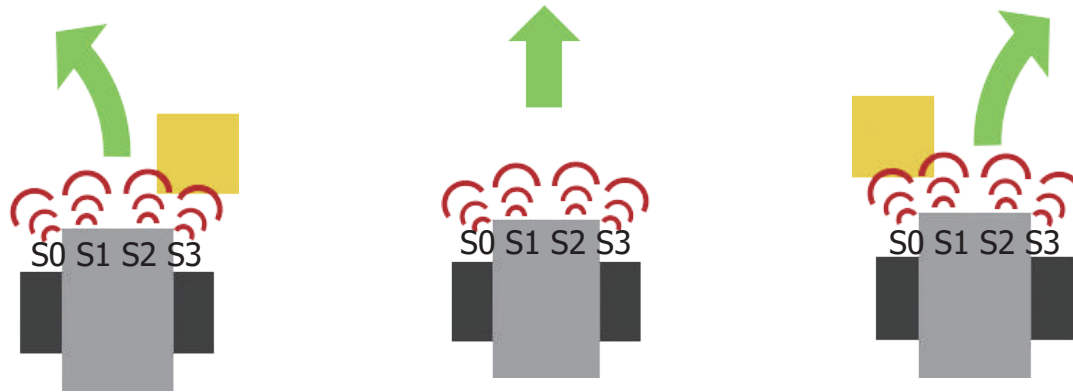
Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the car.

Use 4 Line Finders placed in front of the car to detect oncoming objects.

Use a Potentiometer to set the speed of the car.



S0, S1, S2, S3 are the 4 Line Finders placed in front of the car.

If S0 detects an object, steer to the right.

Else if S1 detects an object, steer to the right.

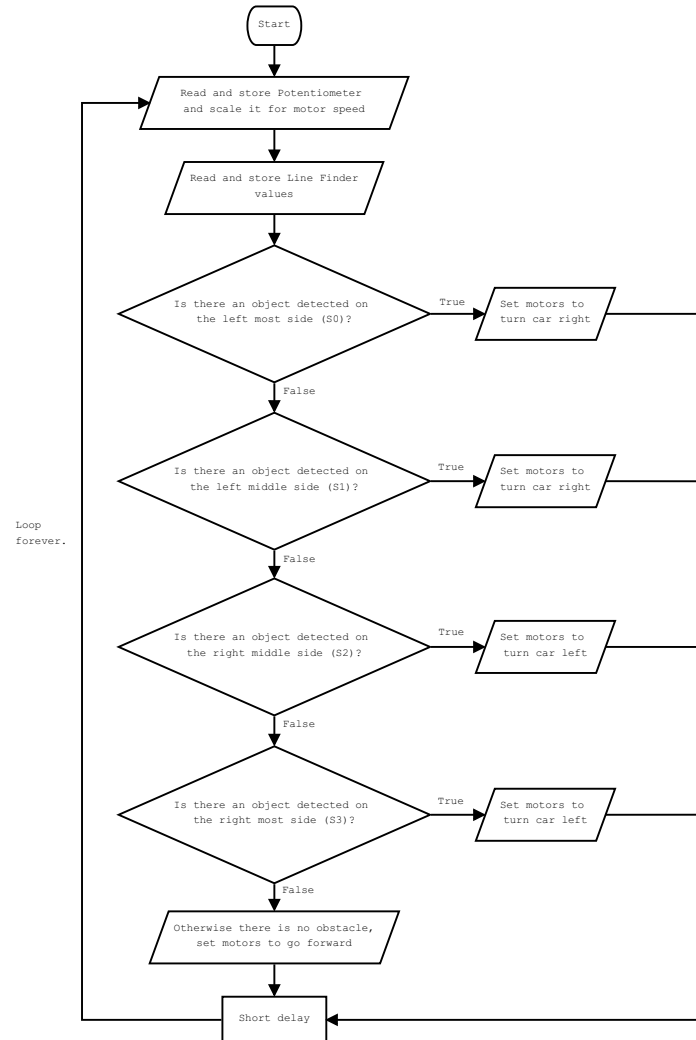
Else if S2 detects an object, steer to the left.

Else if S3 detects an object, steer to the left.

Else no objects are detected, go forward.

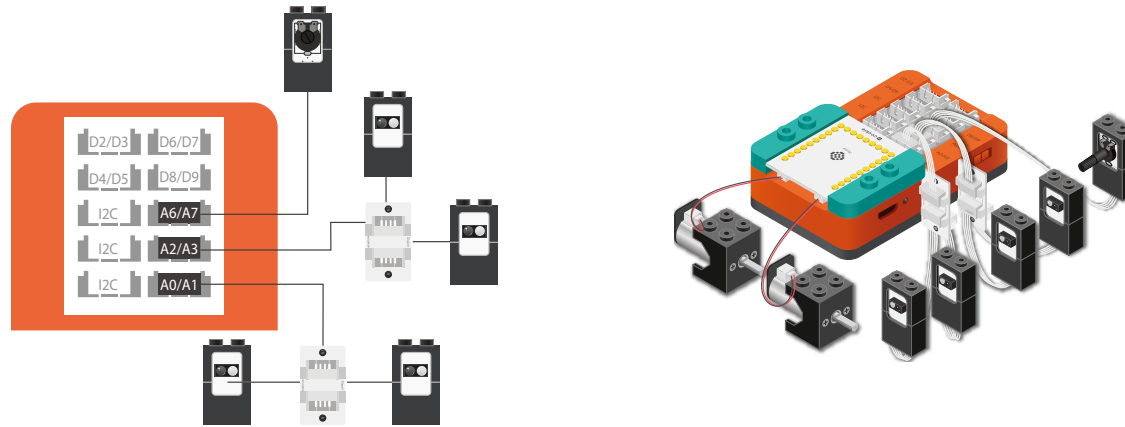
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

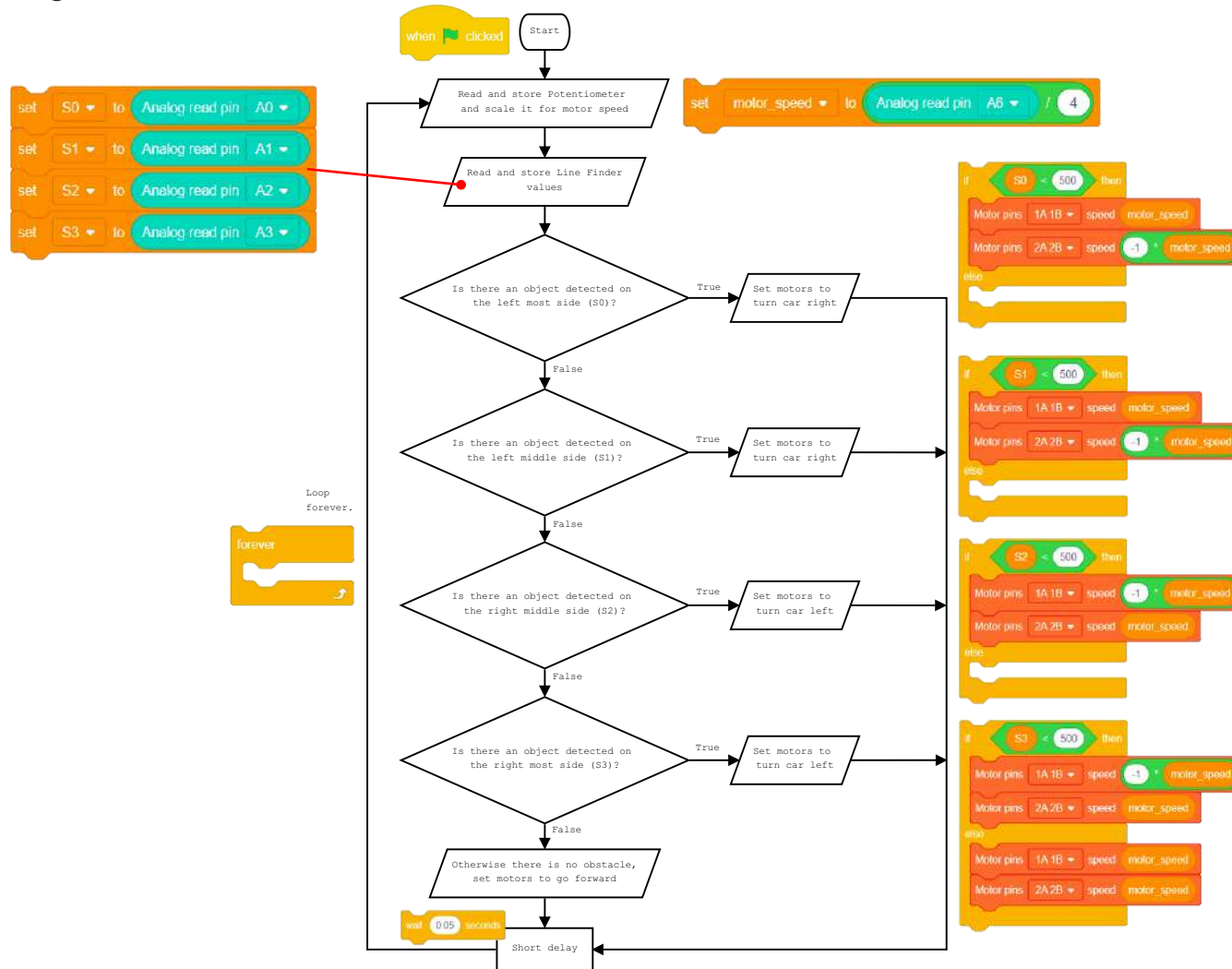
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Potentiometer to pin A6/A7
5. Connect two Line Finders to an IO Splitter. Connect the IO Splitter to pin A0/A1.
6. Connect two Line Finders to an IO Splitter. Connect the IO Splitter to pin A2/A3.
7. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Scale and store Potentiometer (pin A6) value to motor speed.

Read and store Line Finder (pin A0) value.

Read and store Line Finder (pin A1) value.

Read and store Line Finder (pin A2) value.

Read and store Line Finder (pin A3) value.

Check if there is an oncoming object on the left most side.

Set motors to turn car right.

Check if there is an oncoming object on the left center side.

Set motors to turn car right.

Check if there is an oncoming object on the right center side.

Set motors to turn car left.

Check if there is an oncoming object on the right most side.

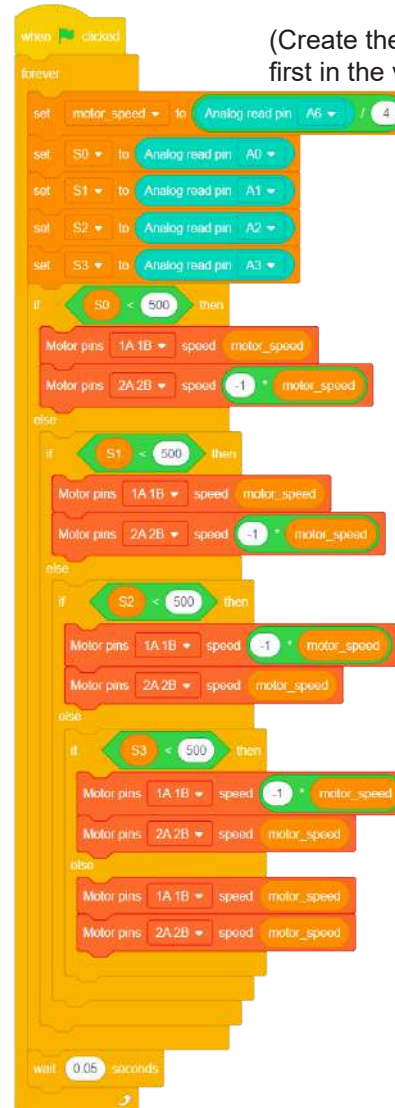
Set motors to turn car left.

Otherwise no coming objects.
Set motors to move car forward.

Short delay before looping.

Loop.

(Create these variables first in the variable tab.)



Block Locator

Events



Control



Operators



Arduino



Data



mCookie

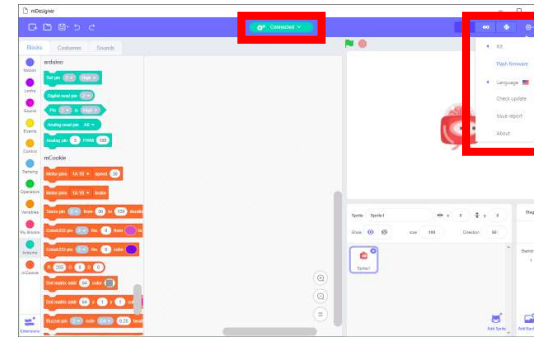


Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

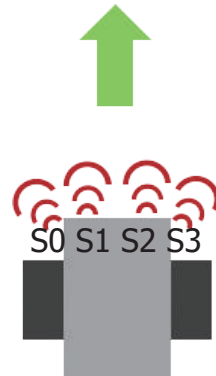
Select Serial Port



Flash firmware

Testing:

Click the green flag to activate the script.



Ensure Line Finders are in the correct position.

Car will automatically avoid oncoming objects by steering left or right.

Speed can be adjusted with the Potentiometer.



Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

Create a structure for the project using building blocks, paper, or other crafting materials.

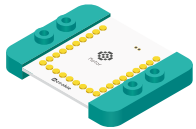


Project Challenge

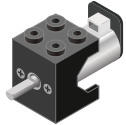
Adjust the script so the car turns slightly right when it detects an object on the left most side (S0).

Adjust the script so the car turns slightly left when it detects an object on the right most side (S3).

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



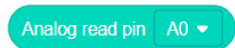
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



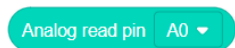
Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.



Potentiometer - input module with an adjustable knob.



Analog Read block - reads the value from the potentiometer. Returns a value between 0 and 1023.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about obstacle avoidance.

Why is this an important feature in safe driving cars?

MIX4B10

Edge Avoidance Car

Use the Line Finder to detect table edges and avoid from falling off them.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:



WHAT YOU WILL NEED



M-CENTER +



MOTOR



MOTOR
DRIVER



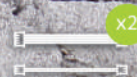
WHEEL



LINE
FINDER

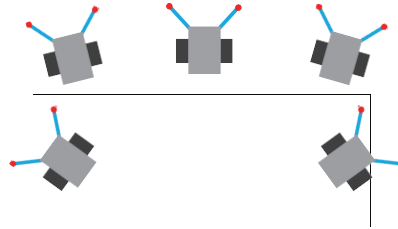


USB CABLE



SENSOR CABLE

About Edge Detection



If you put the a cleaning robot on a desk it won't fall onto the ground, because there are sensors detecting distance from the ground at the bottom of it.

Review of Modules

Line Finder



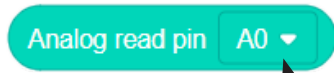
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

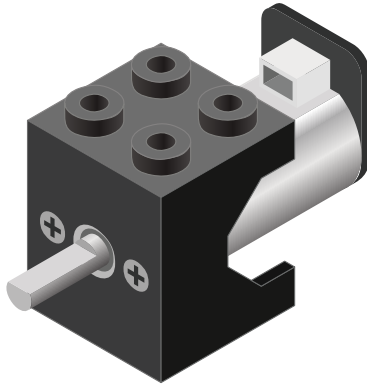


Pin that Line Finder
is connected to.

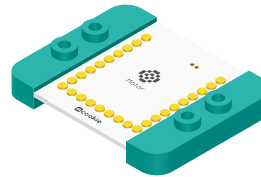
This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to 2 Motors. A Wheel can be attached to the Motor.



Motor Controller



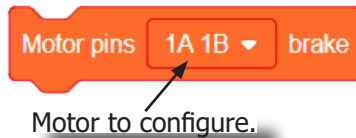
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



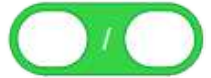
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

Greater Than Block



This block returns TRUE if the left input is greater than the right input.

Review of Concepts

Booleans

Booleans are a type of data, which can only either be TRUE or FALSE. Booleans are used in the condition checks on blocks such as the "If", "If Else", "Wait Until" blocks.

Boolean Operators

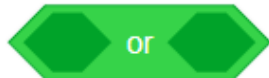
Boolean Operators enables performing operations on boolean types. It is useful for testing multiple conditions.



Left Input	Right Input	Result
True	True	True
True	False	False
False	True	False
False	False	False

And block

This block returns TRUE, when both the inputs are TRUE. Otherwise, it returns FALSE.



Left Input	Right Input	Result
True	True	True
True	False	True
False	True	True
False	False	False

Or block

This block returns TRUE, when one of the input is TRUE. Otherwise, it returns FALSE.



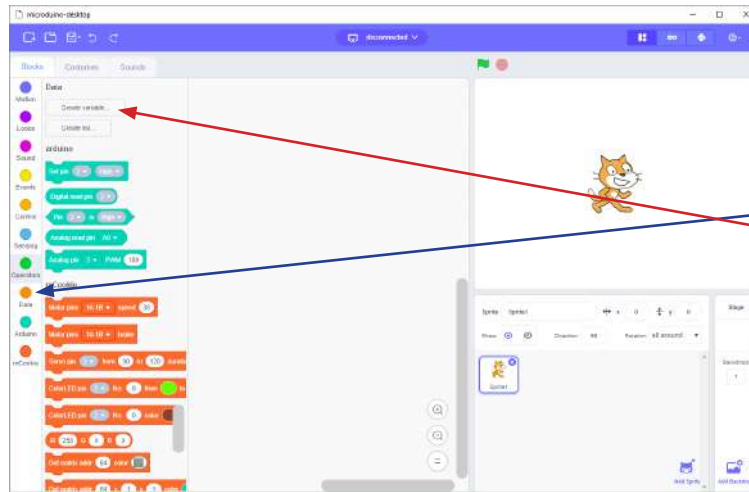
Input	Result
True	False
False	True

Not block

This block returns the opposite of the input. If input is TRUE, this returns FALSE. If input is FALSE, this returns TRUE.

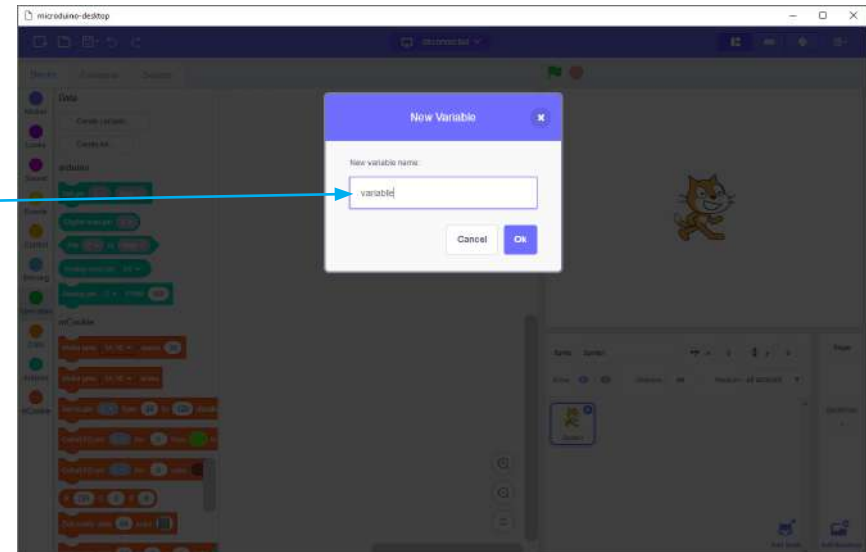
Review of Concepts

Creating a Variable in Scratch



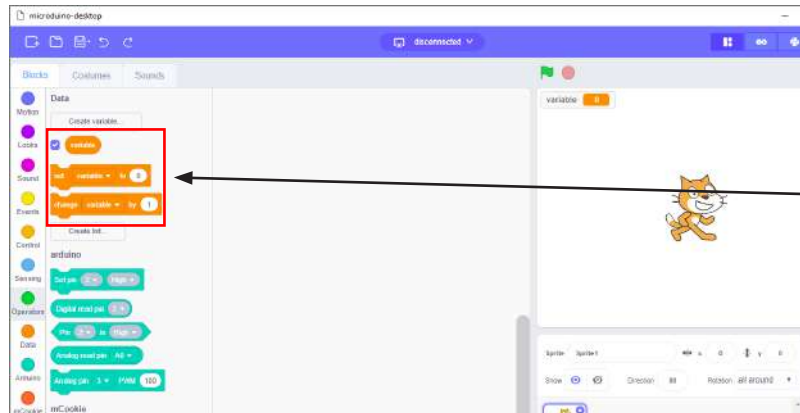
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can we create a car which detects and avoids edges (cliffs)?

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two motors to move the car. Use 2 Line Finders in front of the car pointed downwards to detect if there is an edge (cliff). Steer away if there is a cliff.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

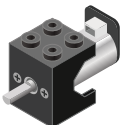
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

2x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the car.

2x  Wheel - attach to motor.

2x  Line Finder - detect if there is an edge / cliff.

Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the car.

Use 2 Line Finders in front of the car pointing downwards to detect edges (cliff).

Read the left Line Finder to see if there is ground ahead (no cliff).

Read the right Line Finder to see if there is a ground ahead (no cliff).

If there is ground in both Left and Right sides, proceed forward.

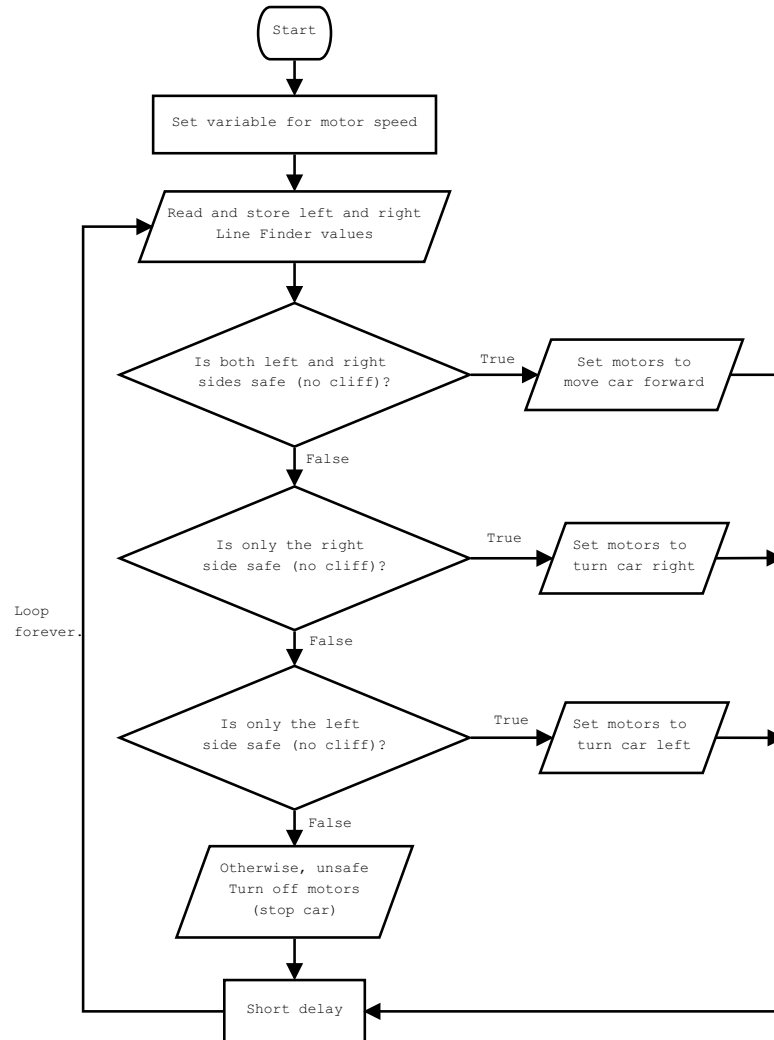
If there is no ground on the Left and ground on the Right, turn right. (Right side is safe to drive on).

If there is ground on the Left and no ground on the Right, turn left. (Left side is safe to drive on).

If there is no ground in both Left and Right sides, stop car.

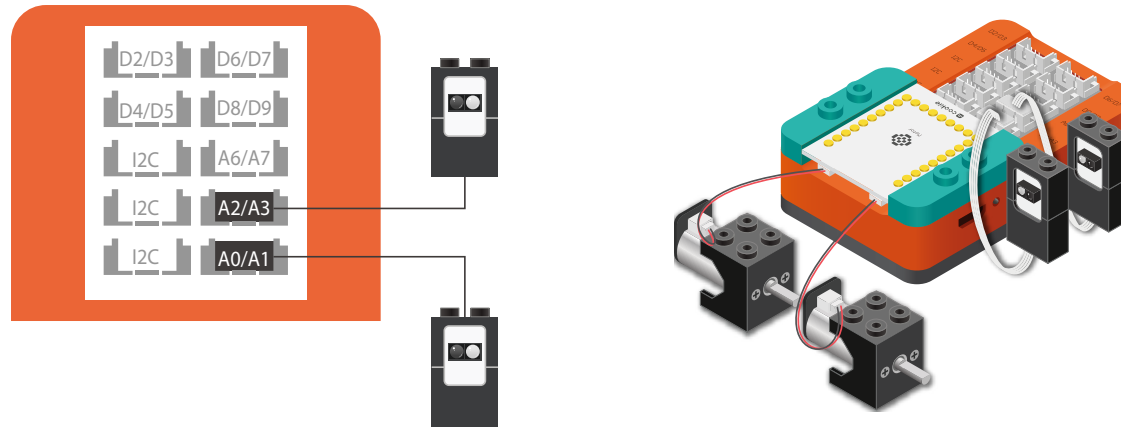
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

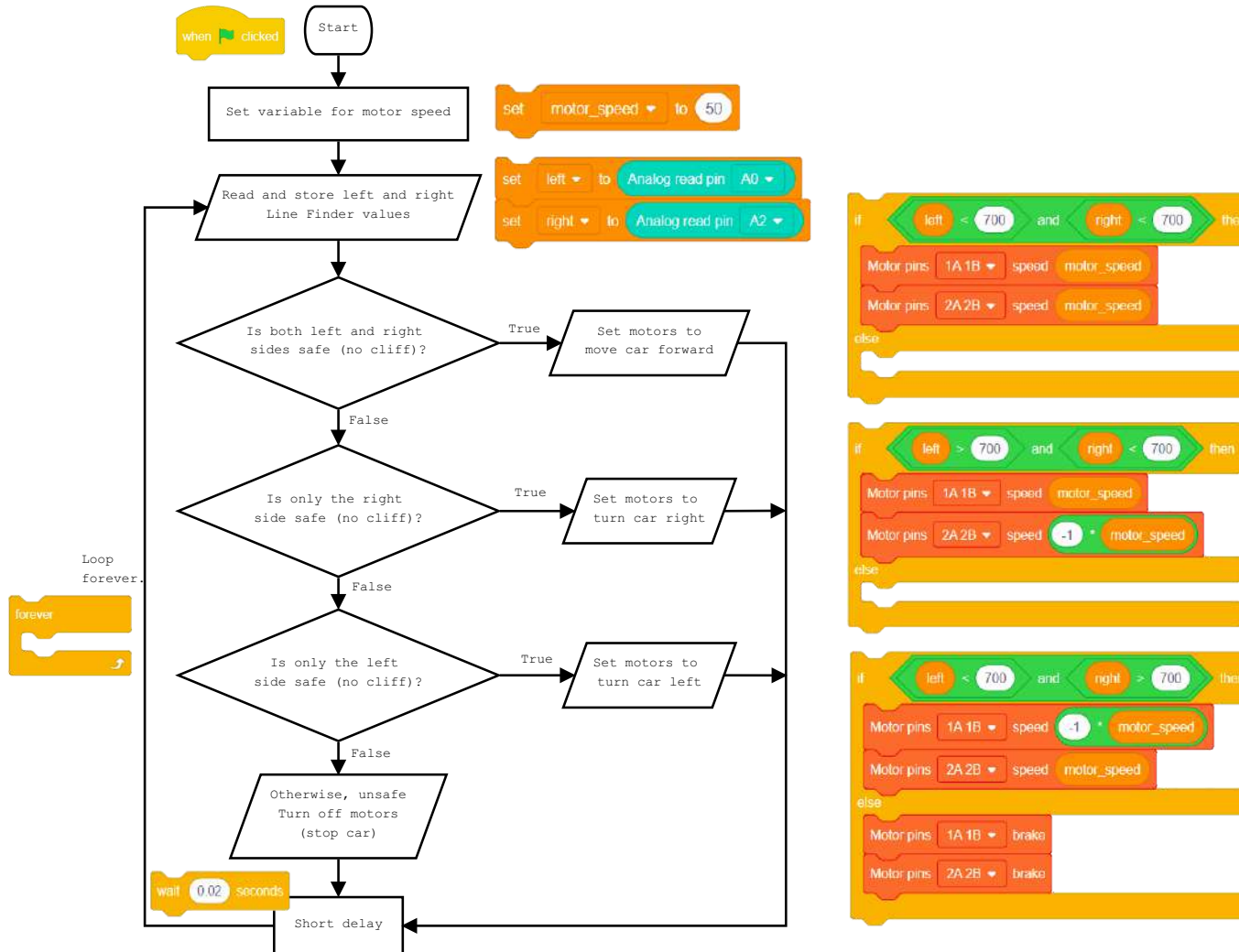
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Line Finder to pin A0/A1.
5. Connect a Line Finder to pin A2/A3.
6. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Set the variable for the speed of the motors.

Read and store the left Line Finder (pin A0) value.

Read and store the right Line Finder (pin A2) value.

Check if it is safe on both Left and Right sides.

Set motors to proceed forward.

Check if there is a cliff on Left side and safe on Right side.

Set motors turn right.

Check if it is safe on Left side and a cliff on Right side.

Set motors turn left.

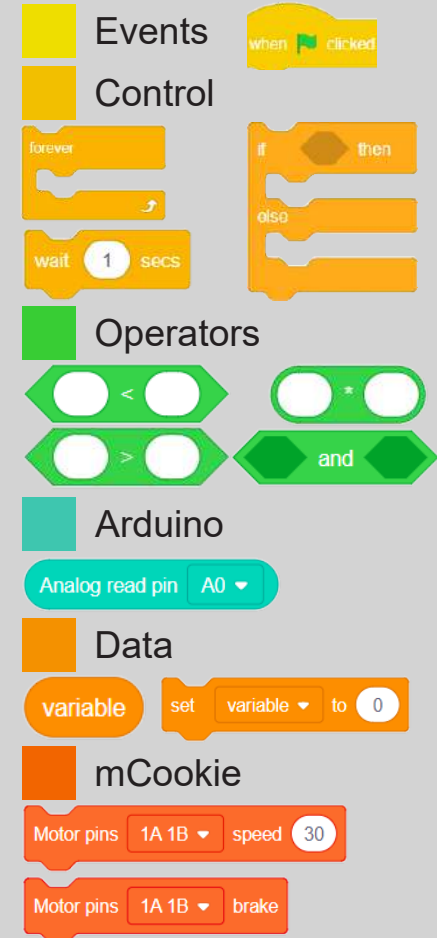
Otherwise both sides are unsure,
set motors to stop car.

Short delay before looping.

Loop.



Block Locator

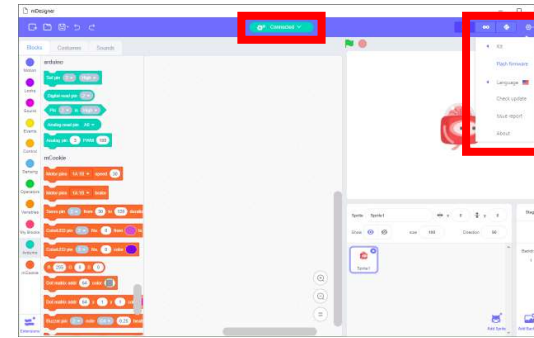


Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

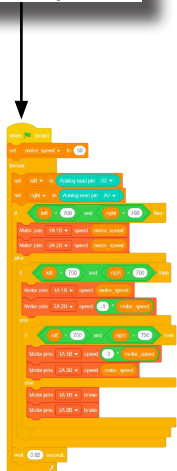
Select Serial Port



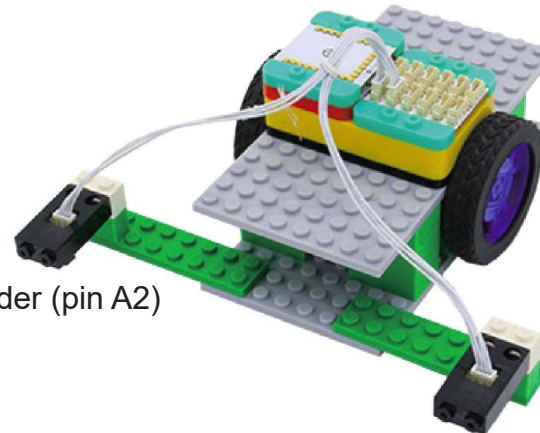
Flash firmware

Testing:

Click the green flag to activate the script.



1. The car will automatically move forward if left and right sides are safe (no cliff).
2. The car will turn right if only the right side is safe.
3. The car will turn left if only the left side is safe.
4. The car will stop if both left and right sides are unsafe (cliff).



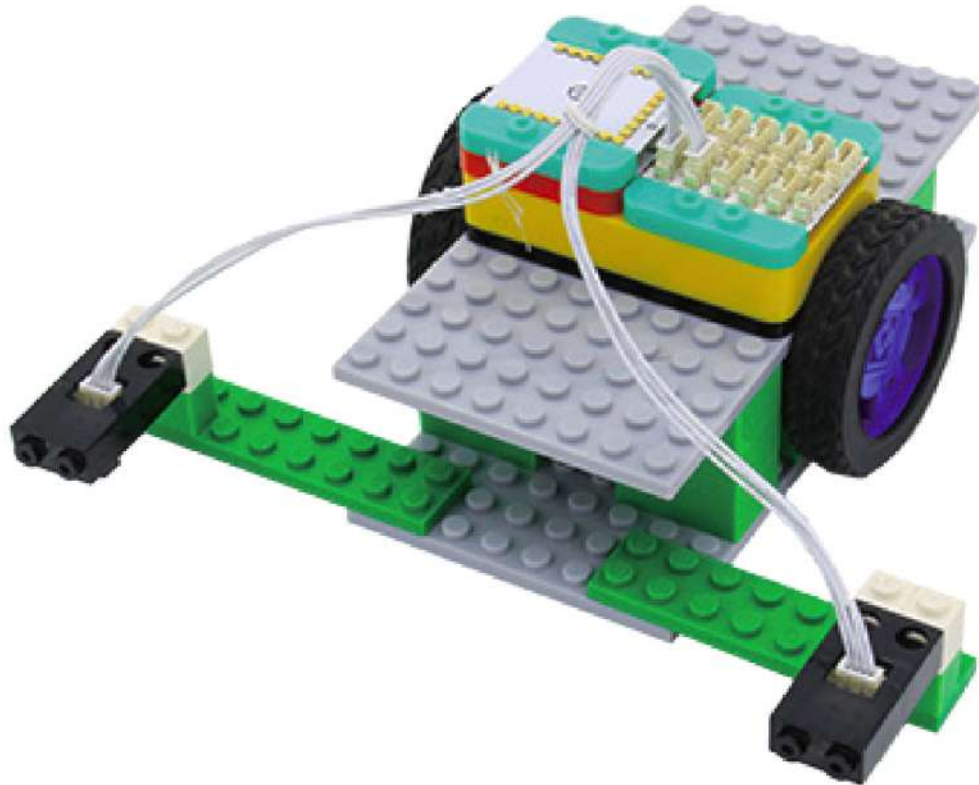
Right Line Finder (pin A2)

Left Line Finder (pin A0)

Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

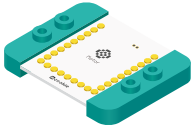
Create a structure for the project using building blocks, paper, or other crafting materials.



Project Challenge

The car stops when the left and right sides are unsafe.
What other options can the car make in this situation?
Implement it on your own.

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



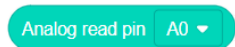
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about edge/cliff detection. What things implement this feature?
Why is it important for each of those things to have this feature?

WHAT YOU WILL NEED



Mcenter +



MOTOR



MOTOR
DRIVER



WHEEL



LINE
FINDER



USB CABLE



SENSOR CABLE

MIX4B011

Line Finding Car 1

Create a car that will use the
Line Finder to move along
the line.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:



 MICRODUINO

About Line Following Cars

Some of the earliest Automated Guided Vehicles (AGVs) were line following mobile robots. They might follow a visual line painted or embedded in the floor or ceiling or an electrical wire in the floor.

Most of these robots operated a simple “keep the line in the center sensor” algorithm. They could not circumnavigate obstacles; they just stopped and waited when something blocked their path.



Review of Modules

Line Finder



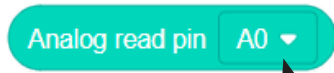
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

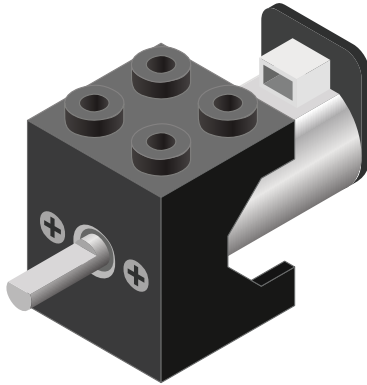


Pin that Line Finder is connected to.

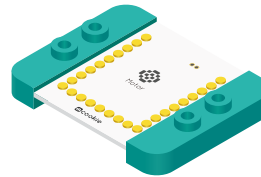
This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to 2 Motors. A Wheel can be attached to the Motor.



Motor Controller



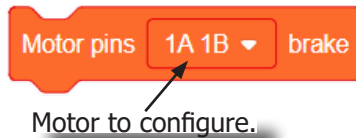
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



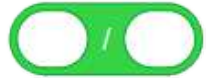
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

Greater Than Block



This block returns TRUE if the left input is greater than the right input.

Review of Concepts

Booleans

Booleans are a type of data, which can only either be TRUE or FALSE. Booleans are used in the condition checks on blocks such as the "If", "If Else", "Wait Until" blocks.

Boolean Operators

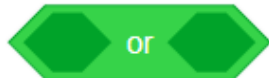
Boolean Operators enables performing operations on boolean types. It is useful for testing multiple conditions.



Left Input	Right Input	Result
True	True	True
True	False	False
False	True	False
False	False	False

And block

This block returns TRUE, when both the inputs are TRUE. Otherwise, it returns FALSE.



Left Input	Right Input	Result
True	True	True
True	False	True
False	True	True
False	False	False

Or block

This block returns TRUE, when one of the input is TRUE. Otherwise, it returns FALSE.



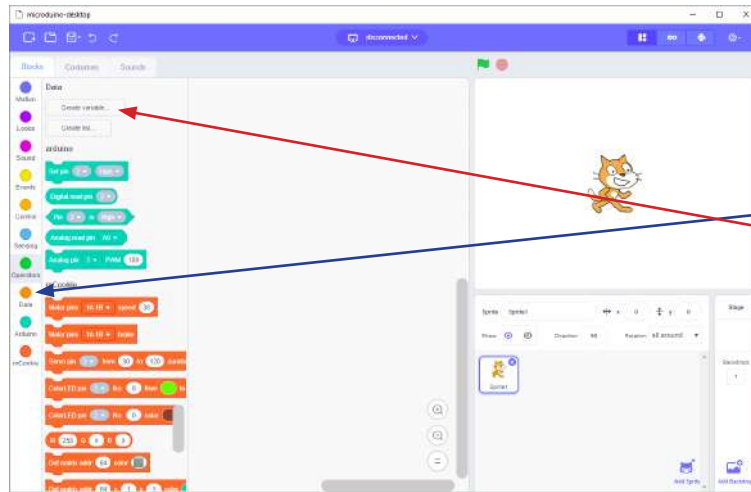
Input	Result
True	False
False	True

Not block

This block returns the opposite of the input. If input is TRUE, this returns FALSE. If input is FALSE, this returns TRUE.

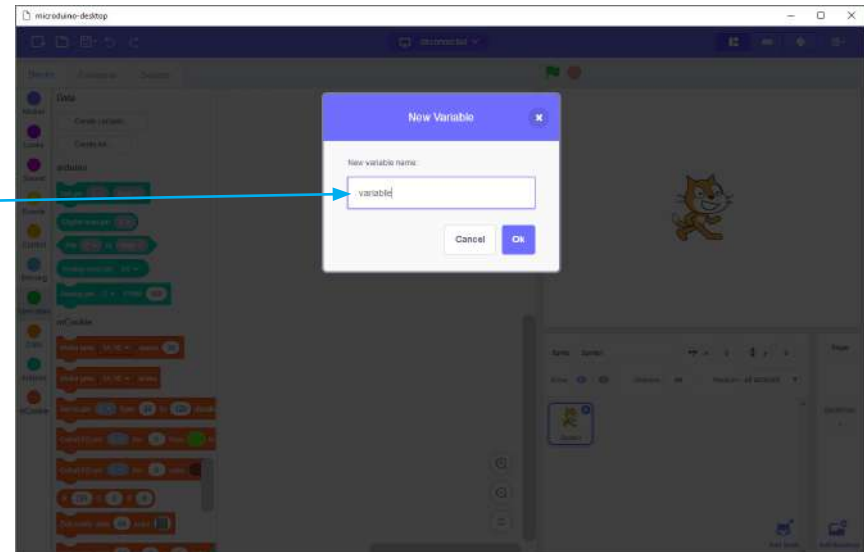
Review of Concepts

Creating a Variable in Scratch



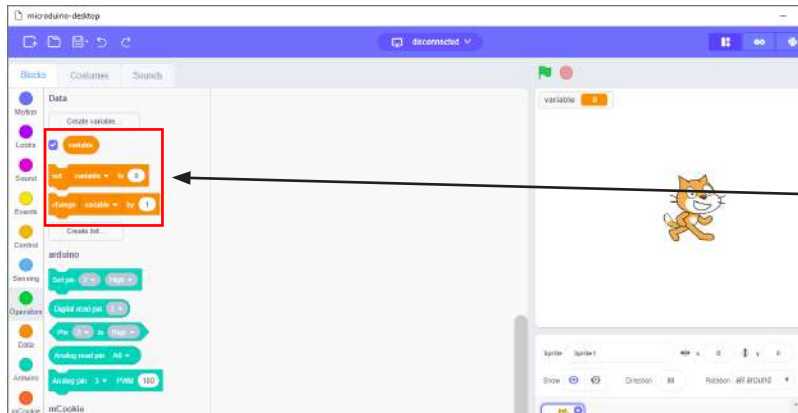
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can we create a car which follows a black line (track)?

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Use two motors to move the car. Use Two Line Finders pointed downward to detect the black line. Adjust accordingly to stay on the black line.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

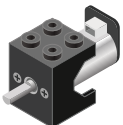
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

2x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the car.

2x  Wheel - attach to motor.

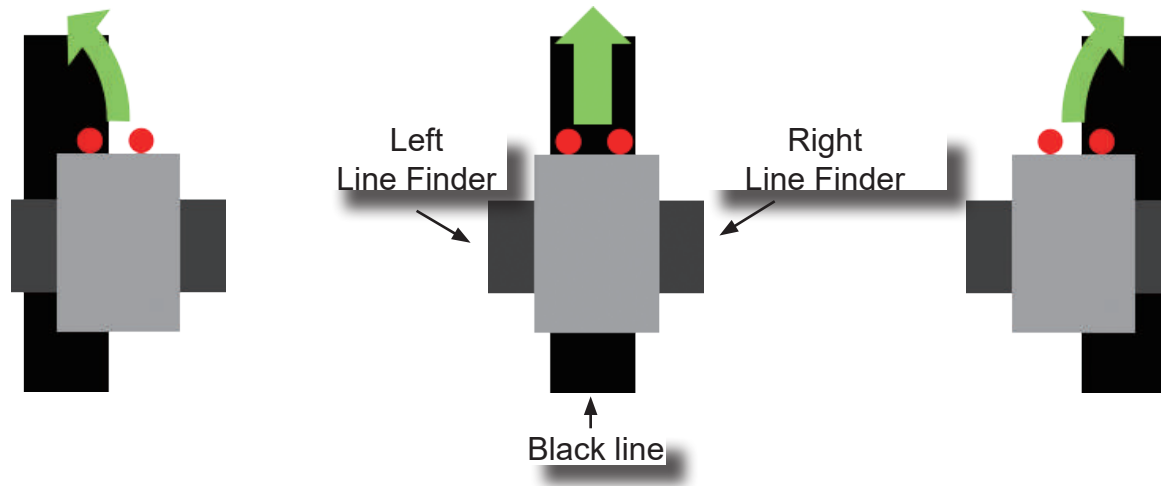
2x  Line Finder - to detect the black line.

Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the car.

Use 2 Line Finders pointed downward to detect the black line.



When the both Line Finders do not detect the black line, proceed forward.

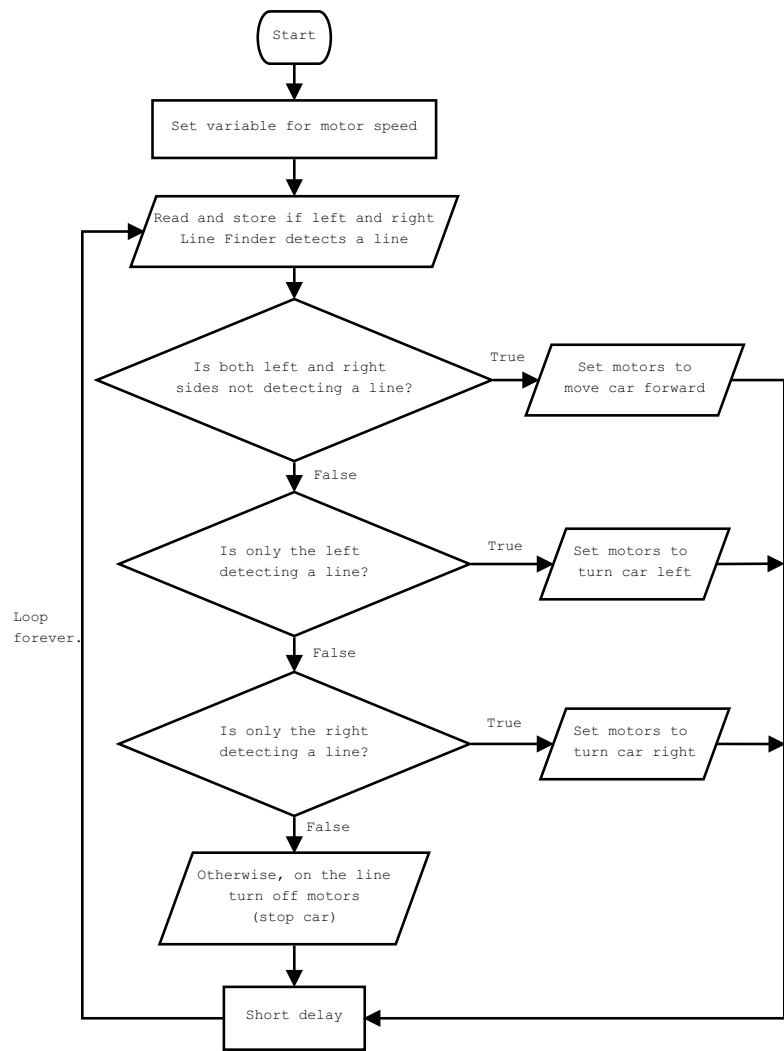
When the left Line Finder detects the black line, steer left.

When the right Line Finder detects the black line, steer right.

*Note: The Line Finders must be spaced apart slightly more than the width of the black line (as illustrated).

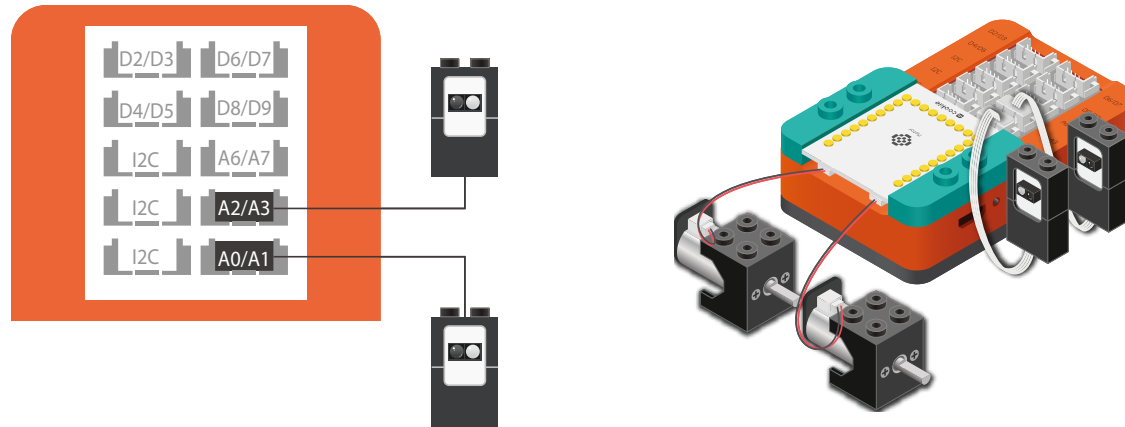
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

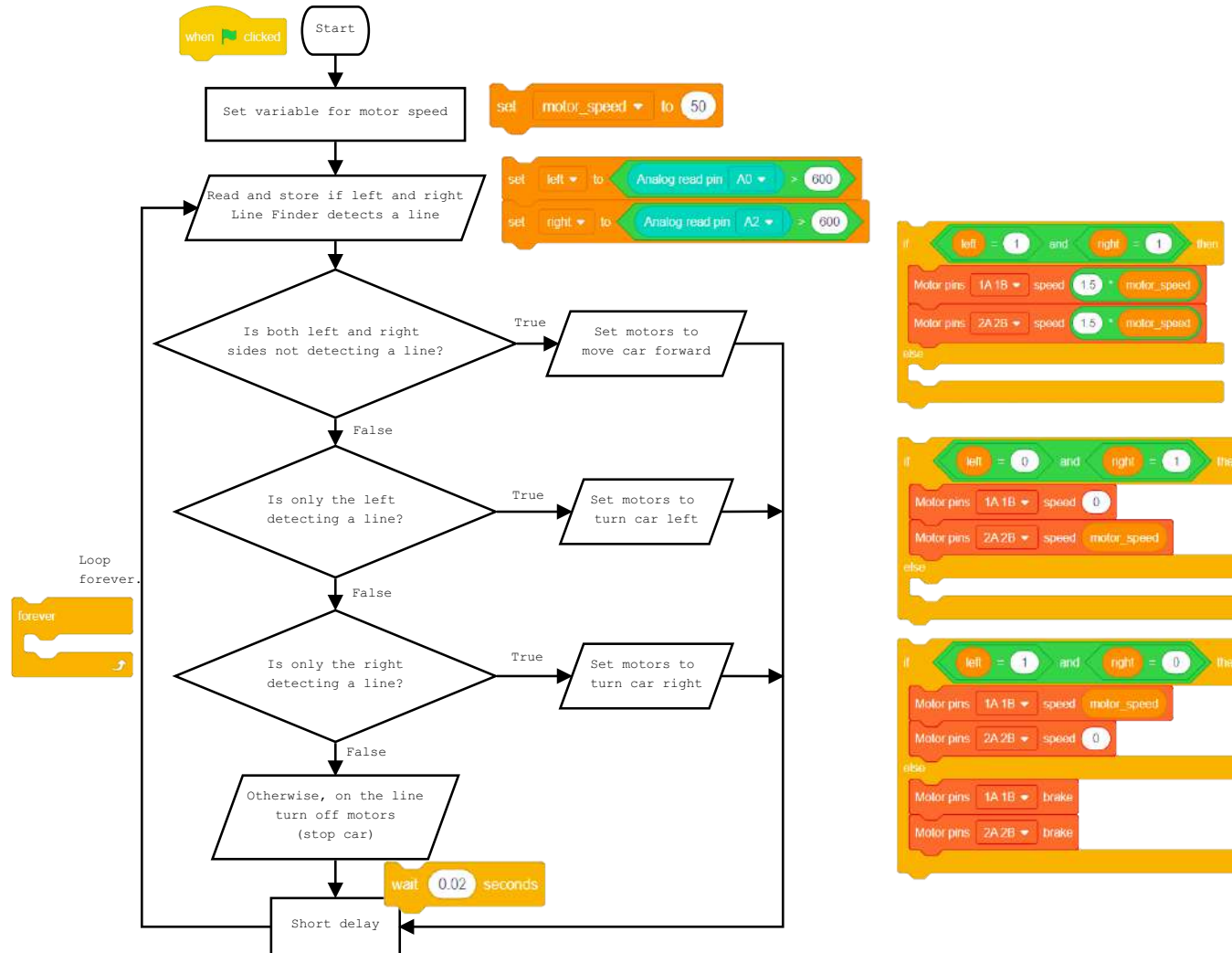
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Line Finder to pin A0/A1.
5. Connect a Line Finder to pin A2/A3.
6. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.

Set the variable for the speed of the motors.

Read and store Left side if detects a black line.

0 (false) = detecting line. 1 (true) = not detecting line.

Read and store Right side if detects a black line.

0 (false) = detecting line. 1 (true) = not detecting line.

Check if not detecting line on both sides.

Set motors to move car forward.

Check if detecting line on left side.

Set motors to turn car right.

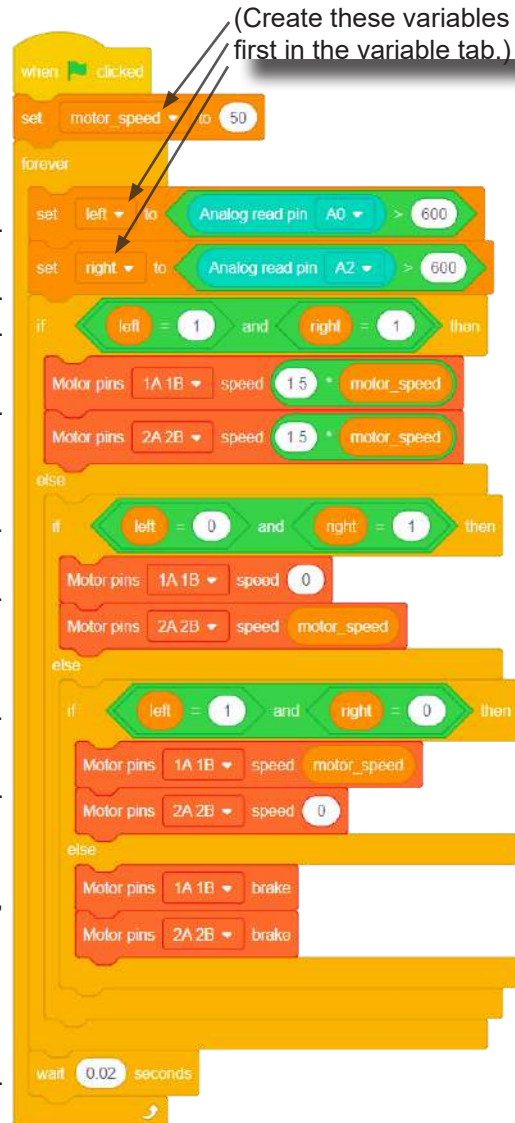
Check if detecting line on right side.

Set motors to turn car left.

Otherwise car is on the black line,
Set motors to off (stop the car).

Short delay before looping.

Loop.



Block Locator

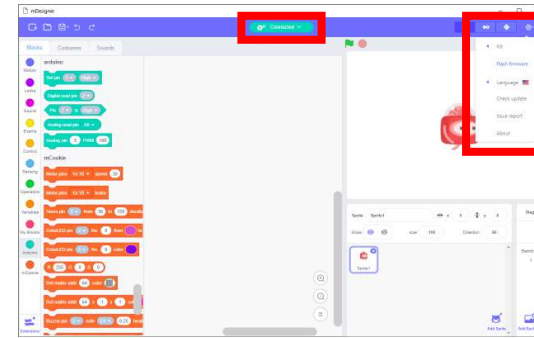


Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

Select Serial Port



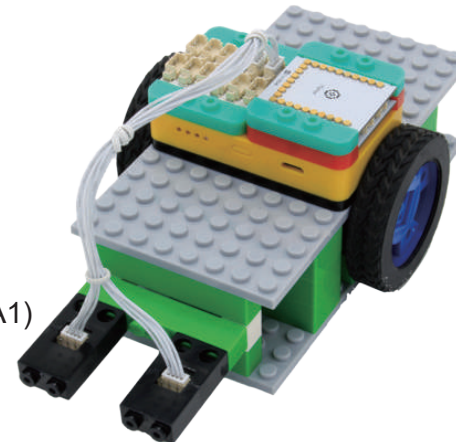
Flash firmware

Testing:

Click the green flag to activate the script.



1. Create a straight line with black tape.
2. Make sure your Line Finders are spaced apart SLIGHTLY larger than the width of the black tape.
3. Place the car on the black tape. The car will trace the black tape.



Right Line Finder (pin A1)

Left Line Finder (pin A0)

Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

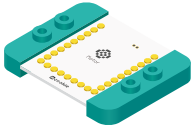
Create a structure for the project using building blocks, paper, or other crafting materials.



Project Challenge

The car stops if it detects the line on both Line Finders (left and right).
What is another solution for this scenario?
Implement it on your own.

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



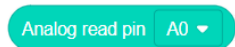
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research more about Line Finding cars.
What are some other features and advancements have been added to this type of car?

WHAT YOU WILL NEED



MBATTERY



MOTOR



MOTOR
DRIVER



POTENTIOMETER



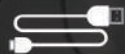
IO SPLIT



WHEEL



LINE
FINDER



USB CABLE



SENSOR CABLE

MIX4B12

Line Finding Car 2

Enhance the line detection
abilities of Line Finding Car 1.

PROJECT TIME: 90 minutes

PROJECT DIFFICULTY:



ASSEMBLY DIFFICULTY:



 MICRODUINO

About Line Following Cars

Some of the earliest Automated Guided Vehicles (AGVs) were line following mobile robots. They might follow a visual line painted or embedded in the floor or ceiling or an electrical wire in the floor.

Most of these robots operated a simple “keep the line in the center sensor” algorithm. They could not circumnavigate obstacles; they just stopped and waited when something blocked their path.



Review of Modules

Line Finder



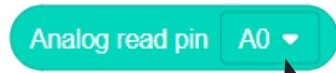
Usage

The Line Finder module is able to detect reflectivity or gray level of a surface. In general, darker or less reflective surfaces produces a higher value. While a lighter or more reflective surface produces a lower value. **It must be connected to a pin with an "A" prefix (A0/A1, A2/A3, A6/A7).**

Surface Reflectivity



Analog Read block

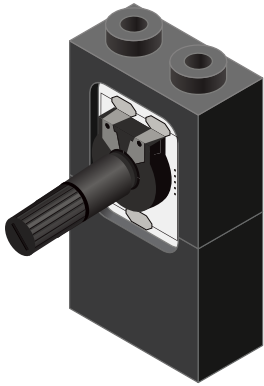


Pin that Line Finder
is connected to.

This block returns the value that the Line Finder detects. A value between 0 and 1023 is returned.

Review of Modules

Potentiometer



A Potentiometer is an input module which can be turned or rotated right or left. Based on the position of the knob, it produces a different signal value for the Core module to read. It must be connected to an analog port (ports denoted with "A").

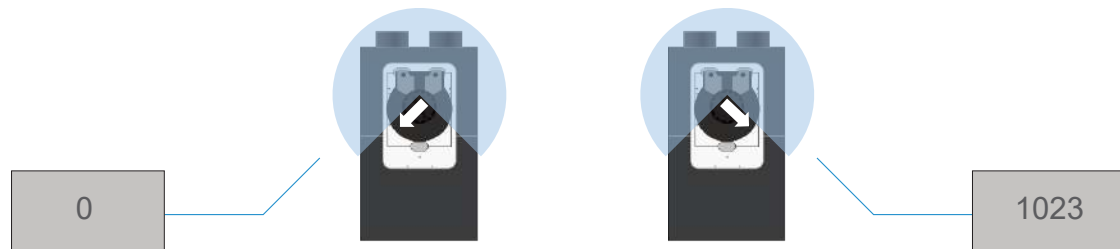
Analog Read block

Analog read pin

A0 ▼

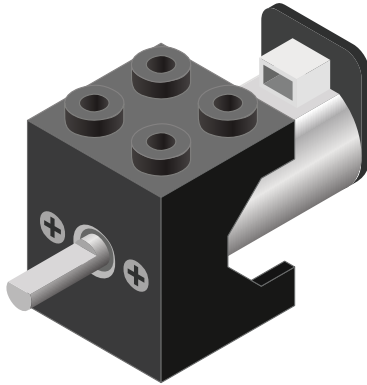
Pin the Potentiometer is connected to.

This block returns the value from the Potentiometer. A value between 0 and 1023 is returned. If the potentiometer is turned all the way to the left, it produces a 0 value. If it is turned all the way to the right, it produces a 1023 value. In the middle it produces a value around 512.

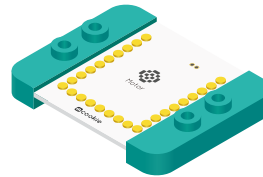


Review of Modules

Motor



A Motor is a electrical machine which converts electrical energy to mechanical energy. It is capable of rotating clockwise and counter clockwise. It is controlled by a Motor Controller. A Motor Controller can control up to 2 Motors. A Wheel can be attached to the Motor.



Motor Controller



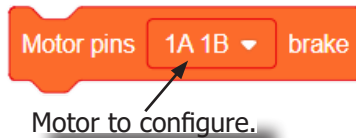
Wheel

Motor - Set Speed block



This block sets the speed of the motor. Positive values turns the Motor in one direction. While negative values turns the Motor in the opposite direction. A value of 0 stops the Motor. Values between -255 and 255 are accepted. 255 is maximum clockwise speed. -255 is maximum counter-clockwise speed.

Motor - Brake block



This block sets the speed to 0 to stop the Motor.

Review of Blocks

Addition Block



This block returns the sum of the left input and right input.

Subtraction Block



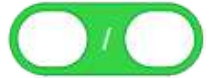
This block returns the value of subtracting the left input by the right input.

Multiplication Block



This block returns the product of the left input and right input.

Division Block



This block returns the value of dividing the left input by the right input.

Less Than Block



This block returns TRUE if the left input is less than the right input.

Equal Block



This block returns TRUE if the left input is equal to the right input.

Greater Than Block



This block returns TRUE if the left input is greater than the right input.

Review of Concepts

Booleans

Booleans are a type of data, which can only either be TRUE or FALSE. Booleans are used in the condition checks on blocks such as the "If", "If Else", "Wait Until" blocks.

Boolean Operators

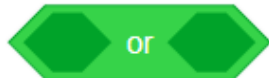
Boolean Operators enables performing operations on boolean types. It is useful for testing multiple conditions.



Left Input	Right Input	Result
True	True	True
True	False	False
False	True	False
False	False	False

And block

This block returns TRUE, when both the inputs are TRUE. Otherwise, it returns FALSE.



Left Input	Right Input	Result
True	True	True
True	False	True
False	True	True
False	False	False

Or block

This block returns TRUE, when one of the input is TRUE. Otherwise, it returns FALSE.



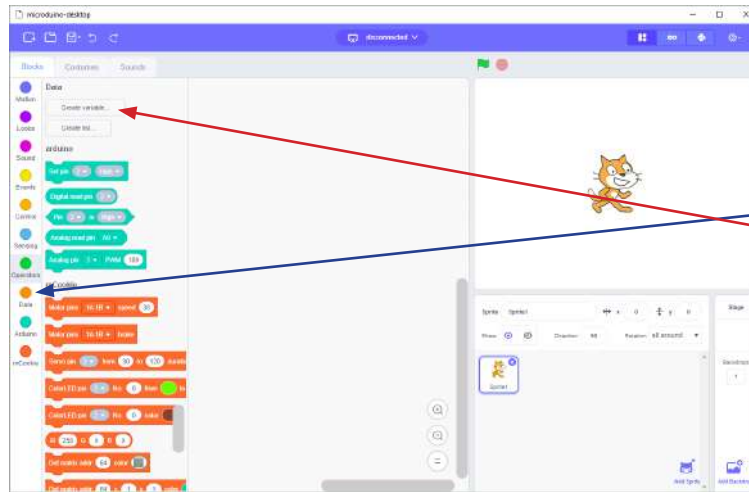
Input	Result
True	False
False	True

Not block

This block returns the opposite of the input. If input is TRUE, this returns FALSE. If input is FALSE, this returns TRUE.

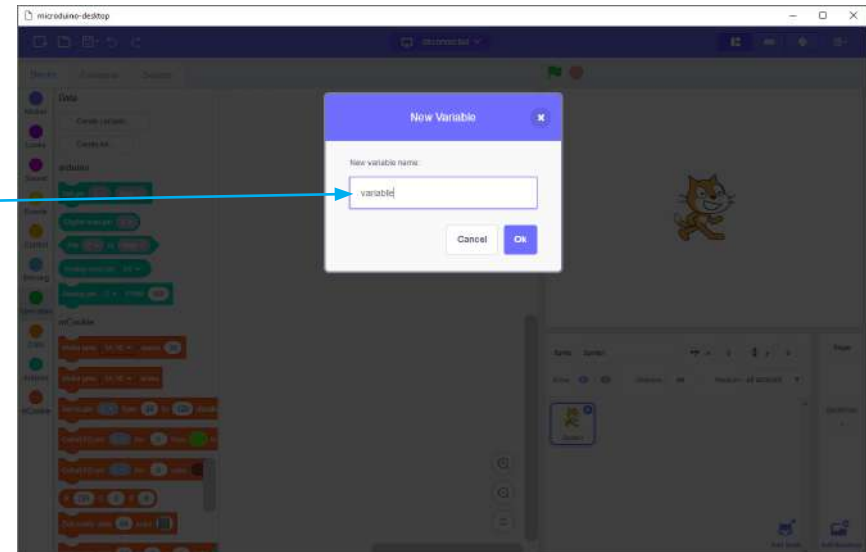
Review of Concepts

Creating a Variable in Scratch



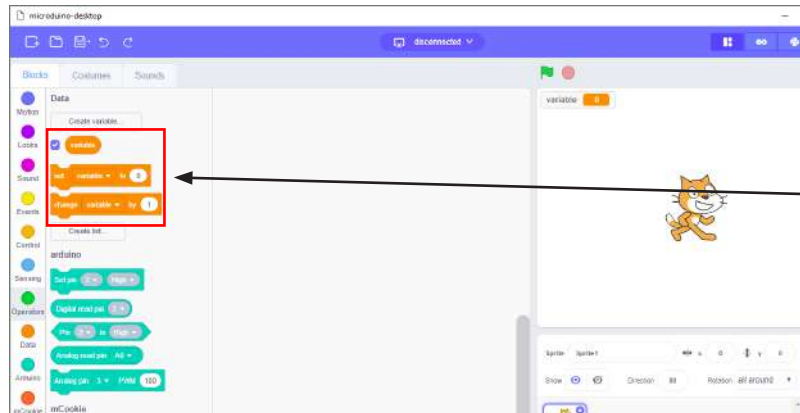
Click on the "Variable" category.
Then click on the "Create variable..." button.

Name your variable.



Review of Concepts

Creating a Variable in Scratch (continued)



Several new blocks will appear which allows access and modification of the new variable.

Variable Value block

variable This block returns the current value of the variable.

Variable Set block

set variable to 0 This block sets the value of the variable. It will overwrite the current value.

Variable Change block

change variable by 1 This block changes the variable value by adding the input value (negative values or blocks are accepted).

input value

The Problem

How can enhance the line following robot to increase its accuracy?

Project Worksheet

Complete the worksheet below to your best abilities.

1. A solution.

Add two additional Line Finders to increase the line detection accuracy.

2. List all the parts needed for the solution. What is the purpose of each part?

3. What is a simple explanation of the logic for the solution?

4. Create a flow chart of the solution.

5. How do you assemble and connect the modules for the solution?

6. Configure blocks to relate to the flow chart.

Project Worksheet - Answers

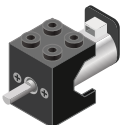
2. List all the parts needed for the solution. What is the purpose of each part?

1x  mCenter+ - control the project, connect sensors and trinkets to project, and supply power.

1x  USB Cable - connect mCenter+ to a computer for charging and programming the project.

6x  Hub Connector Cable - connect sensors and trinkets to the mCenter+.

1x  Motor Controller - controls and drives the attached motors.

2x  Motor - to move the car.

2x  Wheel - attach to motor.

4x  Line Finder - to detect the black line.

1x  Potentiometer - to adjust speed of the truck.

Project Worksheet - Answers

2. List all the parts needed for the solution. What is the purpose of each part? (Continued)

1x



IO Splitter - to connect two sensors/trinkets to a single connector.

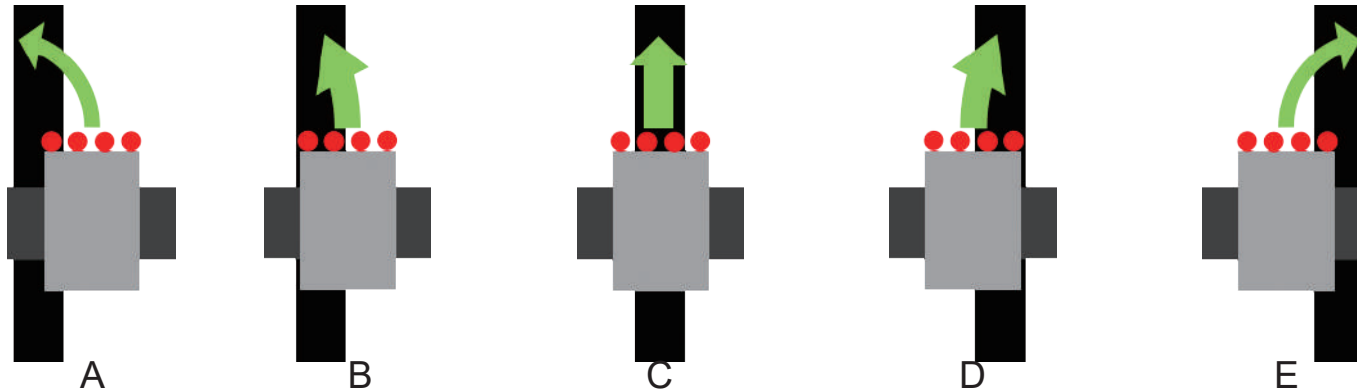
Project Worksheet - Answers

3. What is a simple explanation of the logic for the solution?

Use two Motors to move the car.

Use 4 Line Finders pointed downward to detect the black line.

Use a Potentiometer to configure the Line Finder threshold for registering a black line.

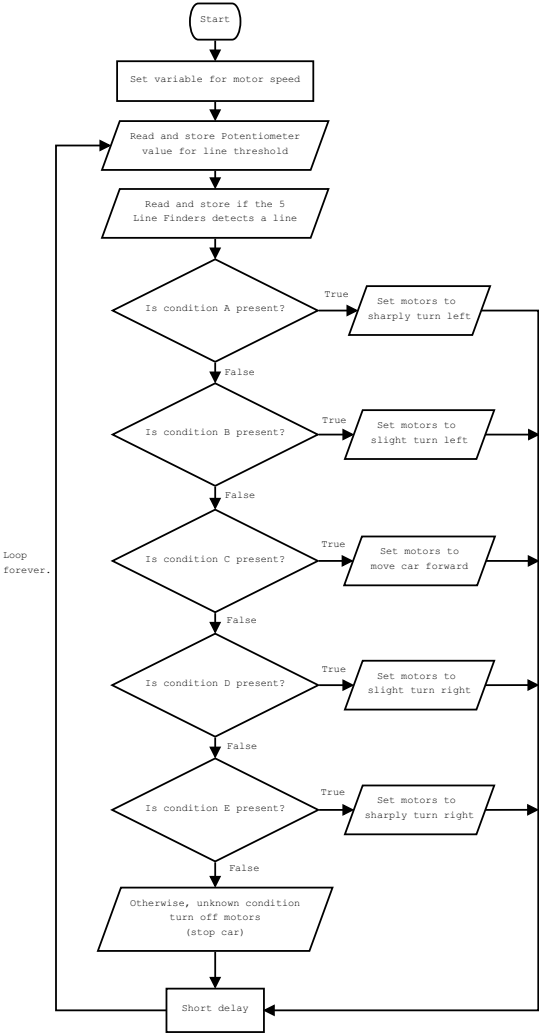


Leftmost Line Finder is labeled S0, Left middle is S1, Right middle is S2, Rightmost is S3.

- A. Only Leftmost detects line - turn sharply to the left to get back on line.
- B. Only Leftmost and Left middle detects line - turn slightly to left to get back on line.
- C. Only Left middle and Right middle detects line - already on the line, go straight.
- D. Only Right middle and Rightmost detects line - turn slightly to the right to get back on line.
- E. Only Rightmost detects line - turn sharply to the right to get back on line.

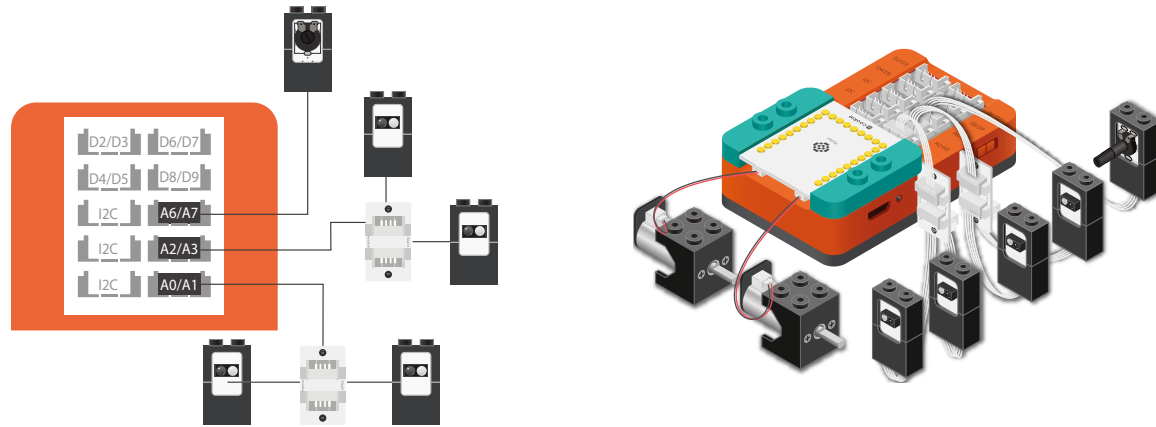
Project Worksheet - Answers

4. Create a flow chart of the solution.



Project Worksheet - Answers

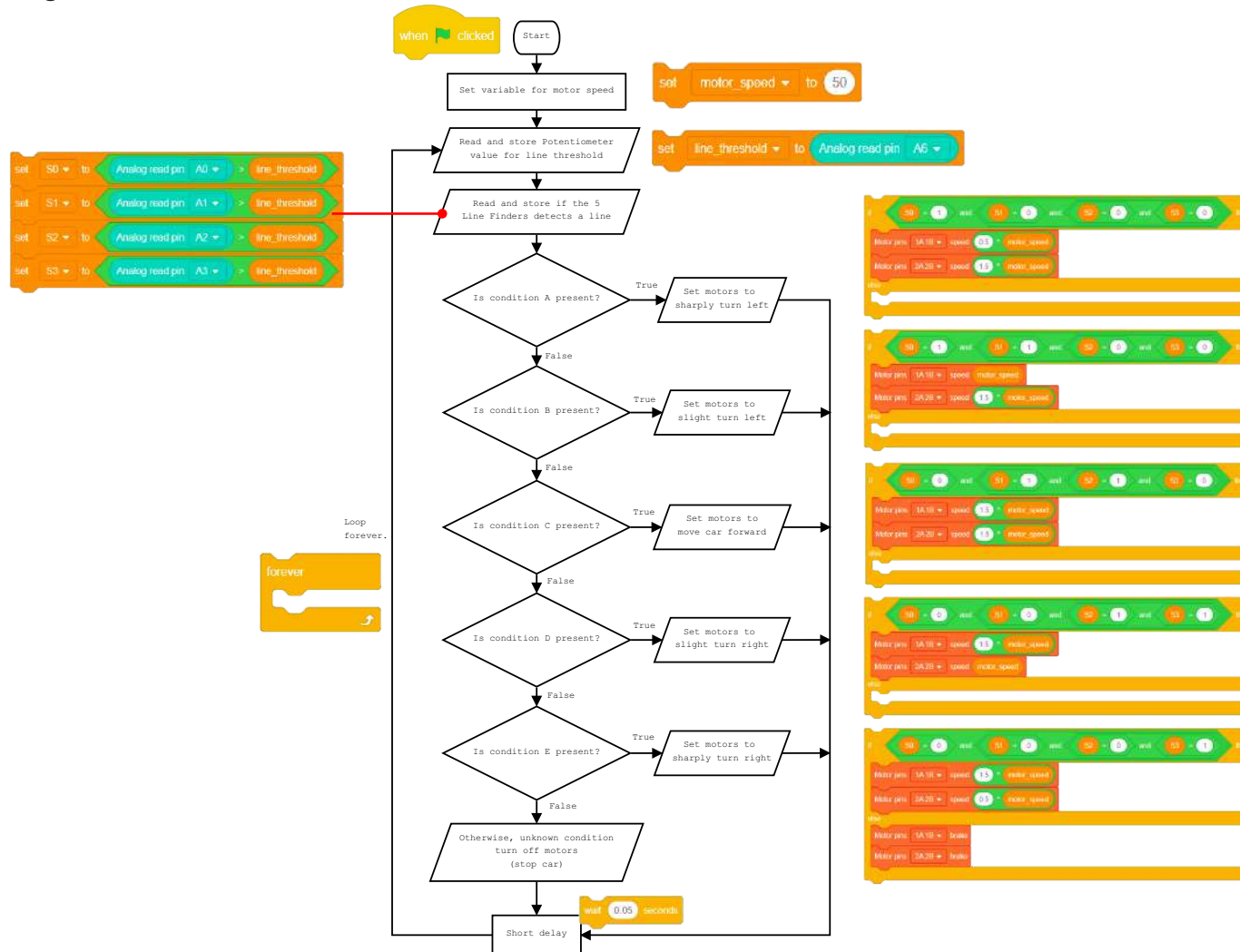
5. How do you assemble and connect the modules for the solution?



1. Stack the Motor Controller onto mCenter+.
2. Connect a Motor to connector "1A1B" on the Motor Controller.
3. Connect a Motor to connector "2A2B" on the Motor Controller.
4. Connect a Potentiometer to pin A6/A7
5. Connect two Line Finders to an IO Splitter. Connect the IO Splitter to pin A0/A1.
6. Connect two Line Finders to an IO Splitter. Connect the IO Splitter to pin A2/A3.
7. Insert one end of the USB cable into mCenter+ and connect the other to a computer.

Project Worksheet - Answers

6. Configure blocks to relate to the flow chart.



The Script

Click Green Flag to activate script.
Set the variable for the speed of the motors.

Read and store Potentiometer (pin A6) for line threshold.

Read and store Line Finder values.

"1" represents detecting line.
"0" represents" no line detected.

Check for condition A.

Set motors to turn sharply to the left.

Check for condition B.

Set motors to turn slightly to the left.

Check for condition C.

Set motors to move quickly forward.

Check for condition D.

Set motors to turn sharply to the right.

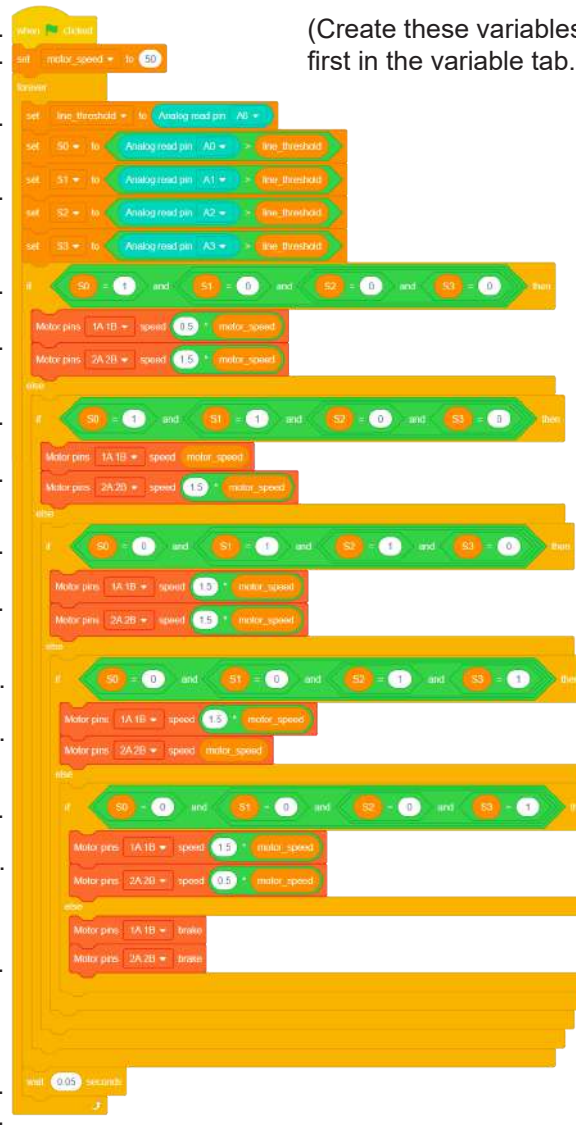
Check for condition E.

Set motors to turn slightly to the right.

Unknown condition,
turn motors off (stop car).

Short delay before looping.
Loop.

(Create these variables
first in the variable tab.)



Block Locator

Events

- when green flag clicked

Control

- forever
- if then
- also
- wait 1 secs

Operators

- >
- =
- and

Arduino

- Analog read pin A0

Data

- variable set variable to 0

mCookie

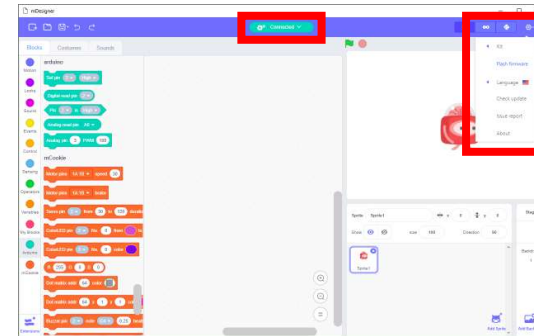
- Motor pins 1A 1B speed 30
- Motor pins 1A 1B brake

Testing the Program

Important:

- Ensure that mCenter+ is on. Flip the switch on mCenter+ to turn it on. An LED on mCenter+ will light up to indicate power is being supplied.
- Ensure you have selected the Serial Port and flashed the special firmware (mCookie) first before testing (this only needs to be done once unless the firmware is overwritten).
- Ensure you have selected the Serial Port and connected before testing.

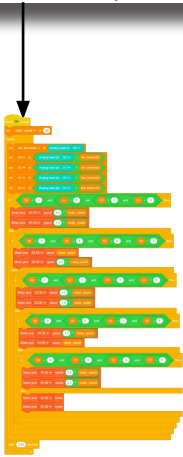
Select Serial Port



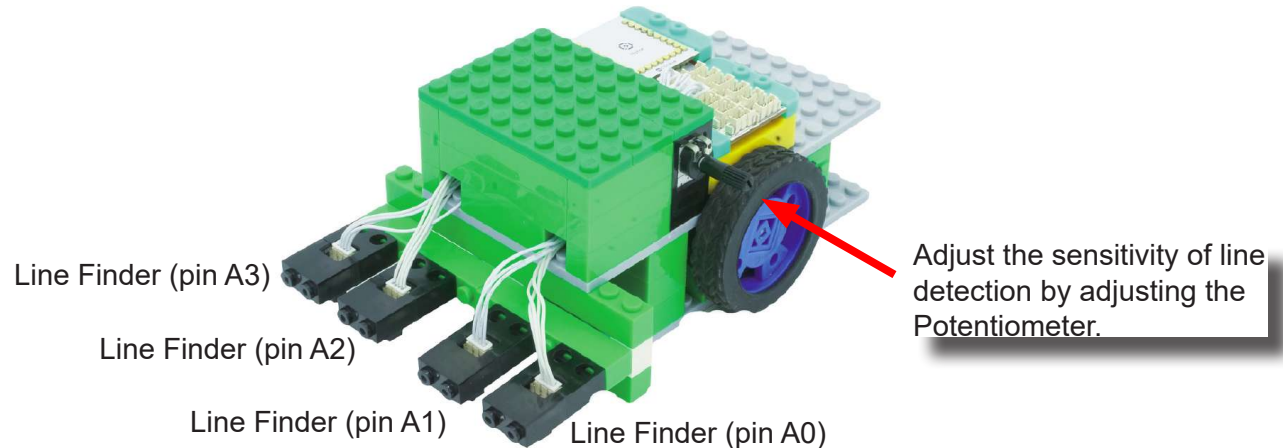
Flash firmware

Testing:

Click the green flag to activate the script.



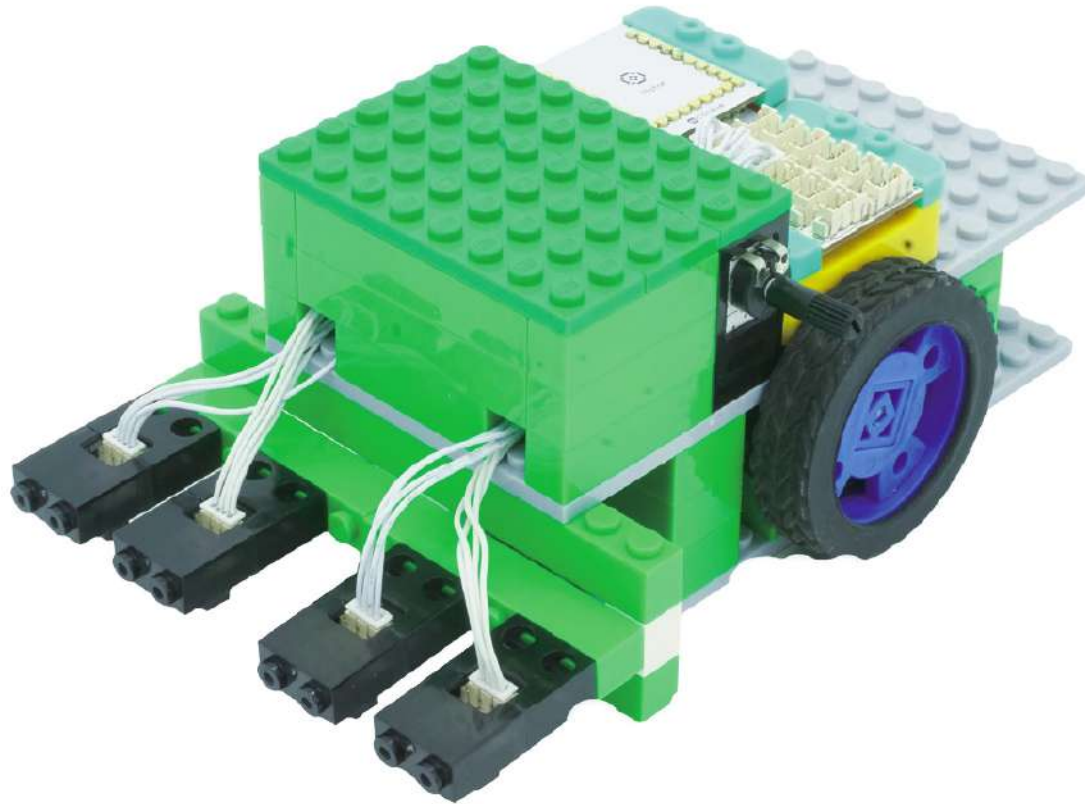
1. Create a straight line with black tape.
2. Make sure the Line Finders are side by side with the correct positions. The two middle Line Finders should be detecting the black tape.
3. Place the car on the black tape. The car will trace the black tape.



Note: If the program is working correctly and you wish to use the program without being connected to your computer. Switch to Arduino mode (top right), then "Flash Program" (this overrides the special firmware, you will need to reflash to use Live mode).

Create a Structure

Create a structure for the project using building blocks, paper, or other crafting materials.

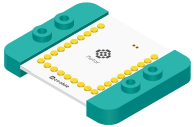


Project Challenge

Add a Color LED to the project.
Have it indicate the current condition detected (A~E).

Modify the behavior to your liking when it reaches the unknown condition.

Review



Motor Controller - controls and drives attached motors. Can control up to 2 motors.



Motor - converts electrical energy into mechanical energy in the form of torque.



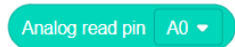
Wheel - attaches to the Motor.



Motor - Set Speed block - set the speed of a motor.



Line Finder - detects the reflective surface or distance of an object.



Analog Read block - returns the value of the reflective surface from the Line Finder sensor.

Expanding the Project

Write down a way of expanding this project. Create it on your own.

Research

Research new technologies for self driving vehicles.
What kind of methods are used in these self-driving cars?

FCC Warning

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and

(2) this device must accept any interference received, including interference that may cause undesired operation.

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment. This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.

- Increase the separation between the equipment and receiver.

- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

- Consult the dealer or an experienced radio/TV technician for help.

The device has been evaluated to meet general RF exposure requirement.

The device can be used in portable exposure condition without restriction.