ROBOFEST 2021 GAME

Stack Rolls

Scratch with Robot Inventor and SPIKE Prime Workshop

This file can be found under the **eAcademy** → **Workshops** page on the website

www.robofest.net robofest@ltu.edu 248-204-3568

Room J233 Taubman Complex, LTU

21000 West 10 Mile Road, Southfield, MI 48075, USA

Please take Pre Assessment:

https://docs.google.com/forms/d/e/1FAIpQLSc_OIuzY8s4xxfBZlyqztxDleeDA2oHfYe0_rVYZOBlutcsBA/viewform?usp=sf_link

This file can be found under the **eAcademy** on the website **www.robofest.net** along with online training and certifications

2021 Workshops

Presented by

Lawrence Technological
University

Computer Science

Course Overview

- 2021 Robofest competition Stack Rolls
  - Autonomous robot that get points by moving and stacking rolls
- SPbot introduction
- Using SPbot to solve the Stack Rolls challenge

2021 Robofest Competition

- Video overview
- Key tasks

  Task 0: Finding the edge of the table

  Task 1: Following the edge of the table

  Task 2: Stop line following when you reach a corner

  Task 3: Stop line following when you reach a given distance

  Task 4: Turning the robot

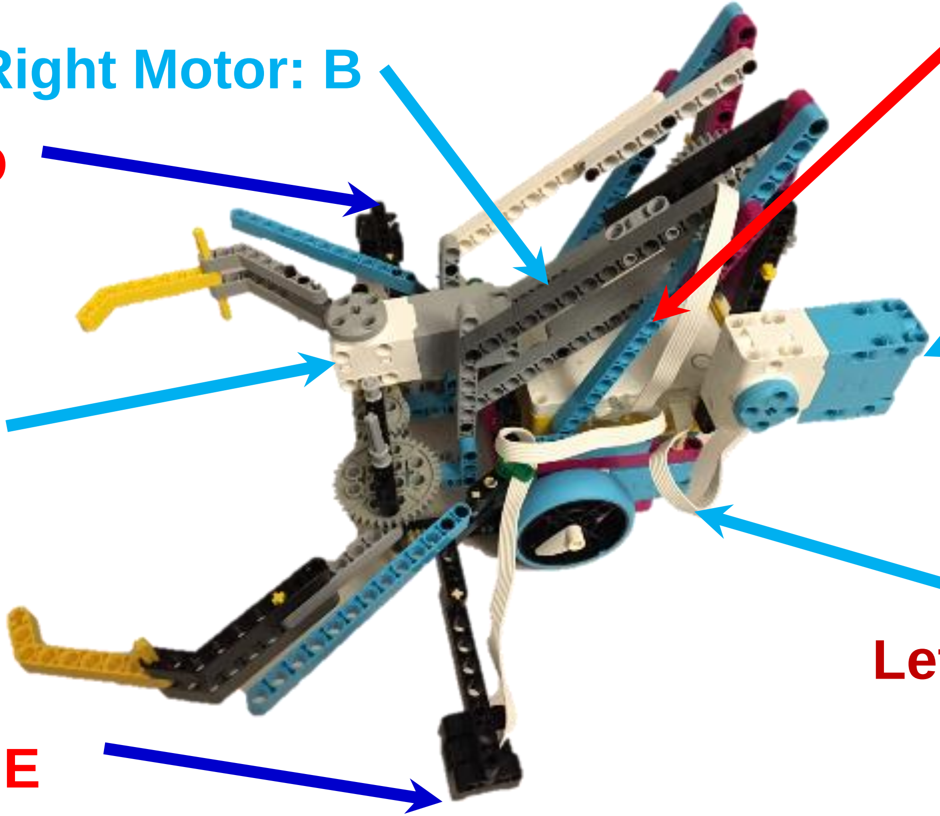
  Task 5: Aligning the robot to an edge

  Task 6: Manipulating Paper Rolls

  Task 7: Building MyBlocks

  Task 8: Gyro Turns

LEGO Spike Prime robot used – SPbot



Remember the connections!

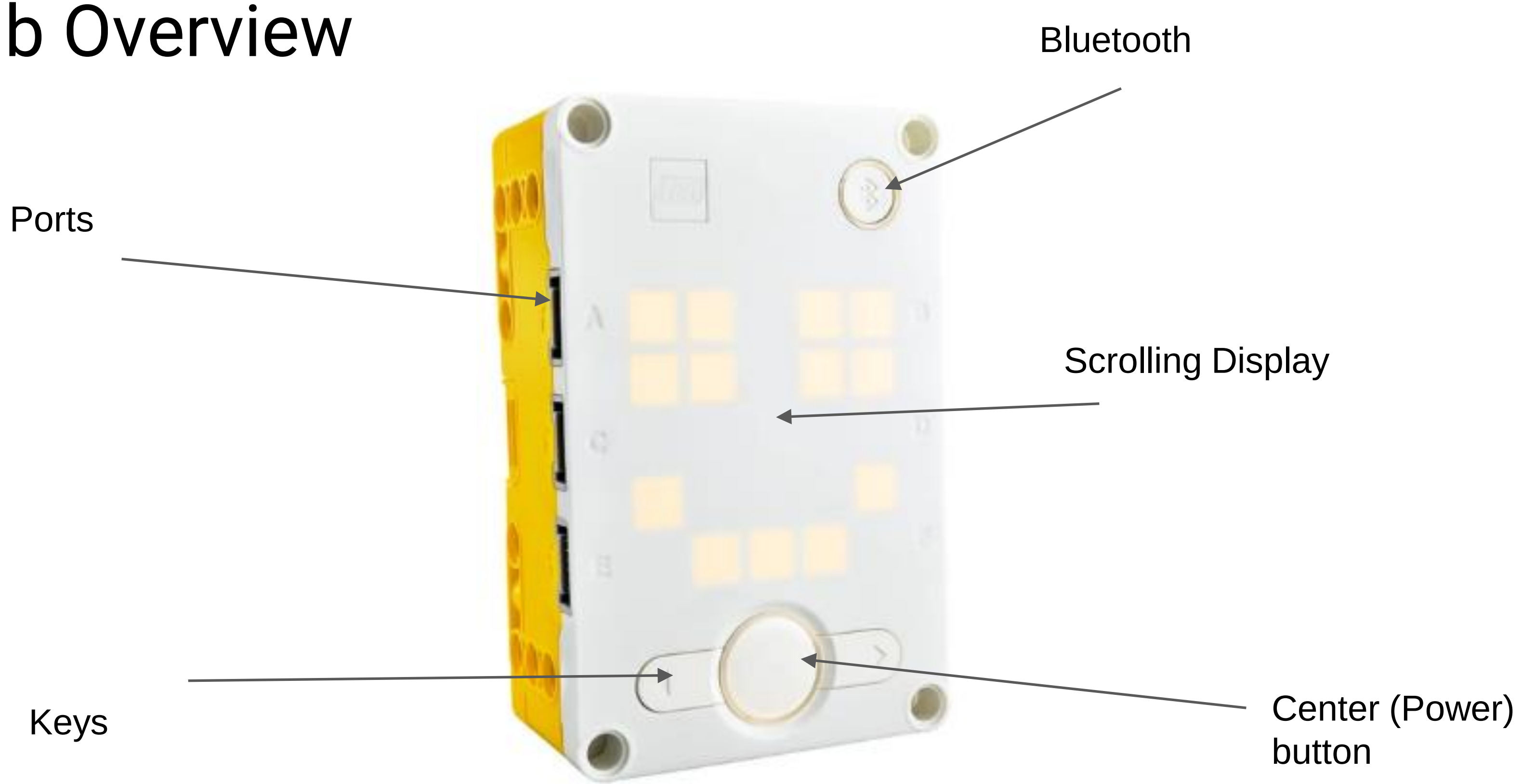
- Left Motor connects to **A**
- Right Motor connects to **B**
- Arm Motor **C**
- RH Color sensor connects to **D**
- LH Color sensor connects to **E**
- Claw Motor **F**

Software Versions Used

- Examples used Version **1.3.3**
  - Download
    - https://education.lego.com/en-us/downloads/spike-prime/software

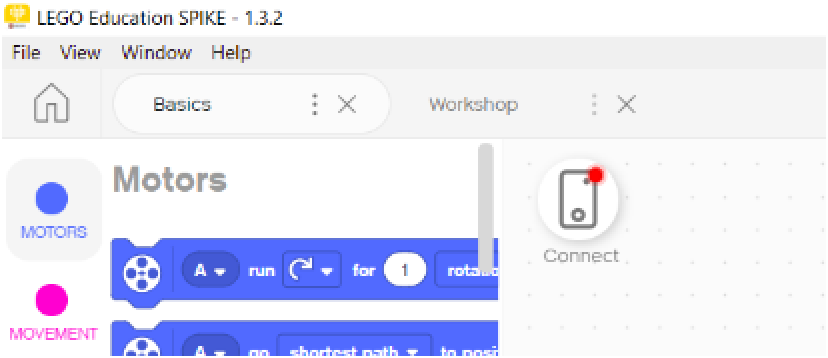
- Presentation and all example programs are available at robofest.net under Tech Resources

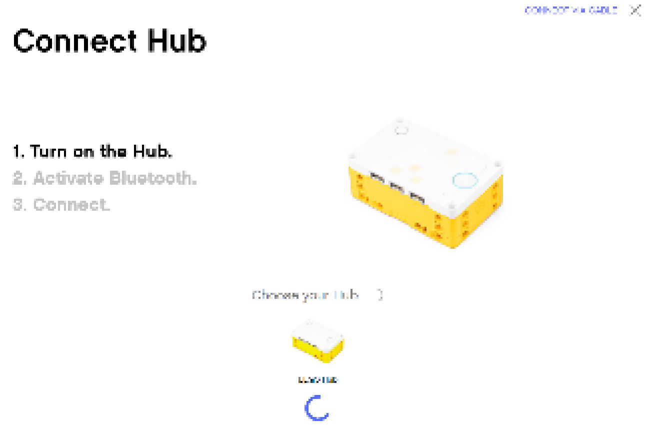
Hub Overview



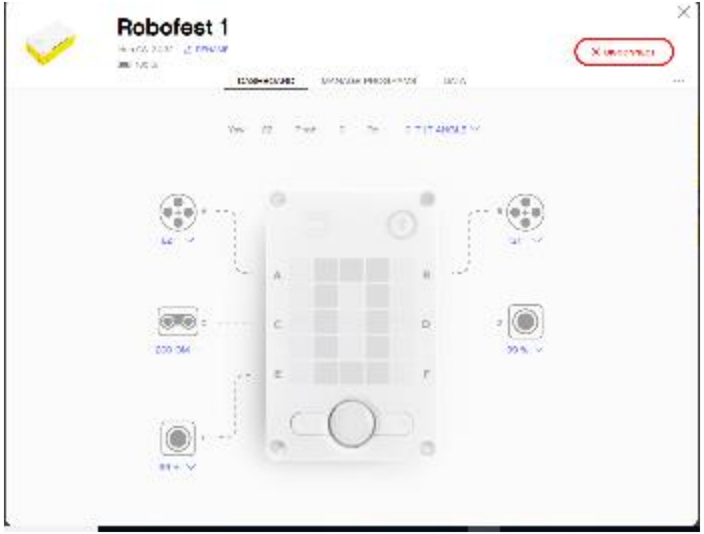
Connect to Hub

- Two options
  - USB Cable
  - Bluetooth
- Follow instructions in software





Hub Settings

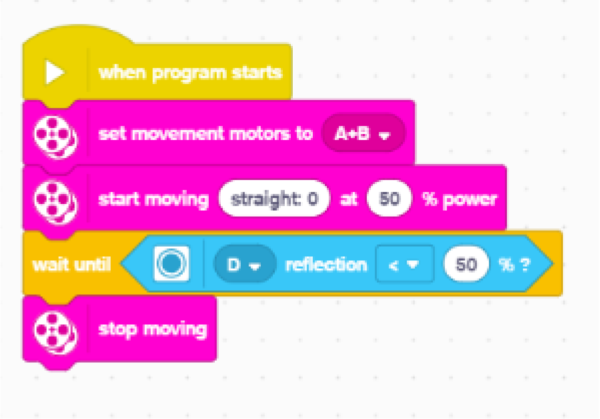


Task 0

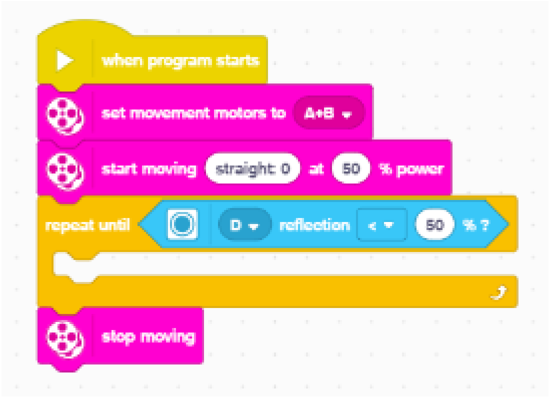
Finding the edge of the table

Task 0: Example Solutions

- Using a wait block



- Using a loop block



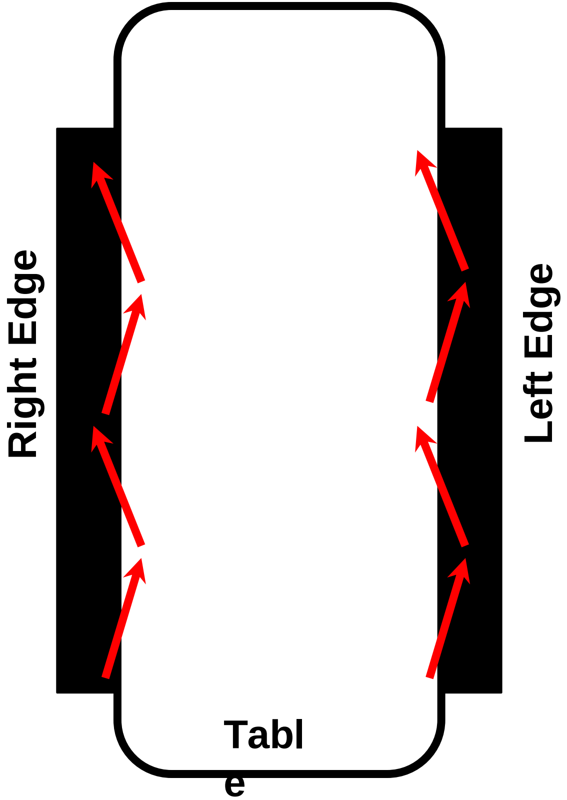
https://youtu.be/goFLg-SG1lo

Task 1

Following the edge of the table

Following The Edge Of The Table

- Use the zig-zag method to follow the edge of the table
- Edge following is also referred to as line following
- The zig-zag method requires the use of a sensor determine when the robot is on or off the table



Following The Edge Of The Table

- Get color sensor values to determine when the robot is **on** or **off** the table. We will use the color sensor in Reflective Light Intensity mode.

- Color Sensor 1
  - On table = ______ (99)
  - Off table = ______ (5)

Color Sensor 2

On table = ______ (99)

Off table = ______ (5)

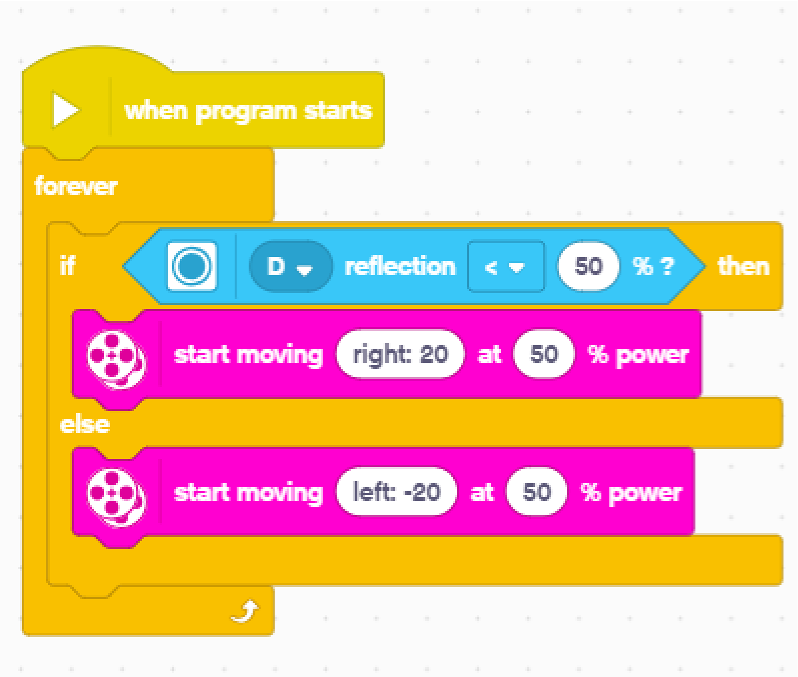
Color Sensor Readings

Following The Edge Of The Table

- For light sensor #2 settings example
  - On table = 40
  - Off table = 0
  - Median threshold = (99+5)/2 = 52

- Two cases
  - Light sensor reading > 52. On table.
  - Light sensor reading < 52. Off table.

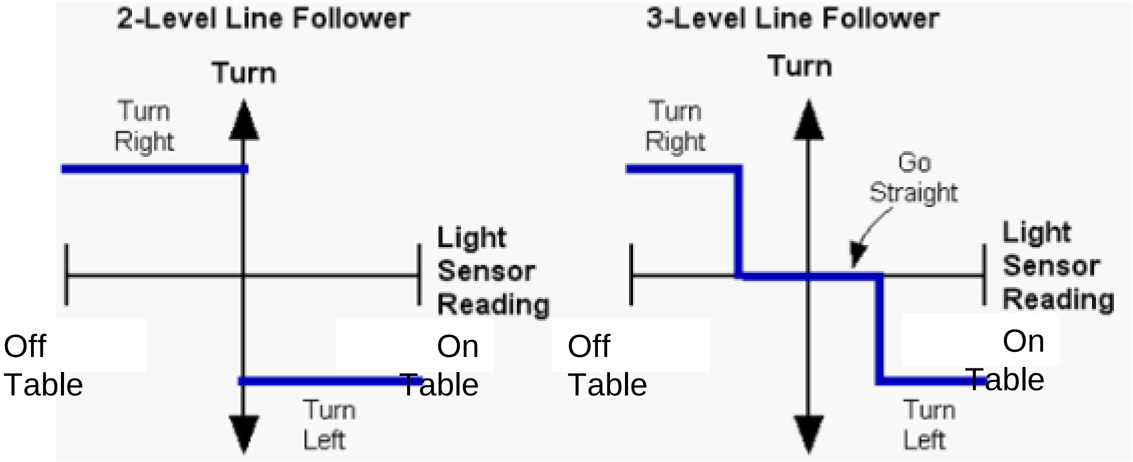
Simple Line Following Algorithm



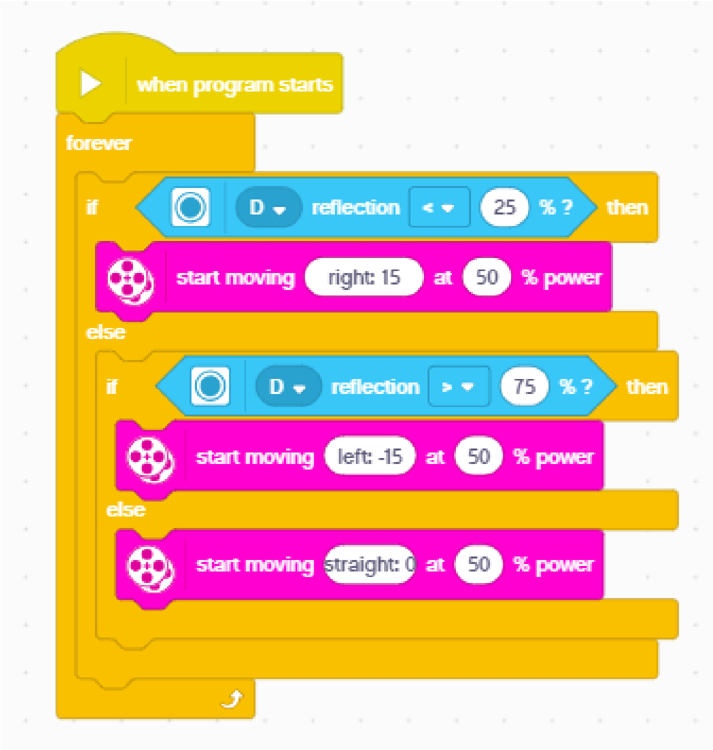
https://youtu.be/RGB8nKO5How

How to improve our line following algorithm

- The zig-zag method can cause a bumpy response
- To improve the response, you can use a 3-level line follower (concept shown below)



How to improve our line following algorithm

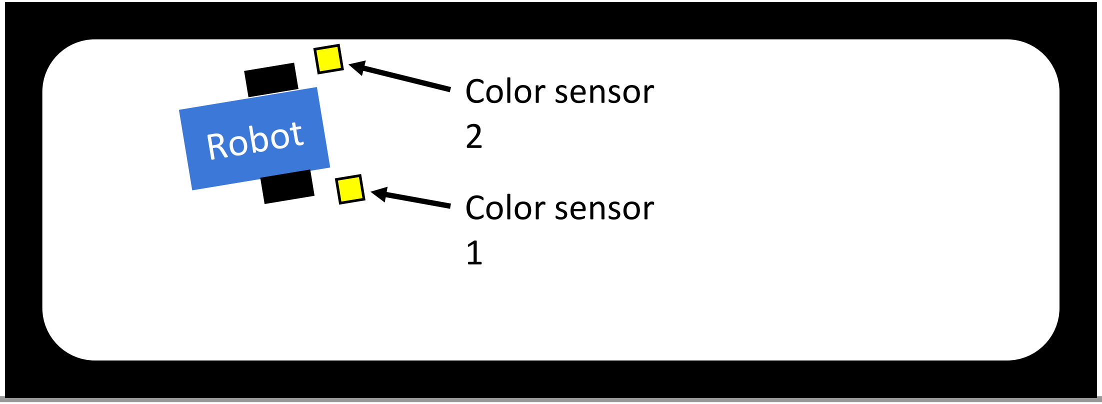


Task 2

Line following to the corner of the table

Line following to the corner

- One method of line following to the corner is to follow the edge of the table with one color sensor and detect the end of the table with a other color sensor
  - Sensor 1 used to locate the end of the table
  - Sensor 2 used to follow the edge of the table

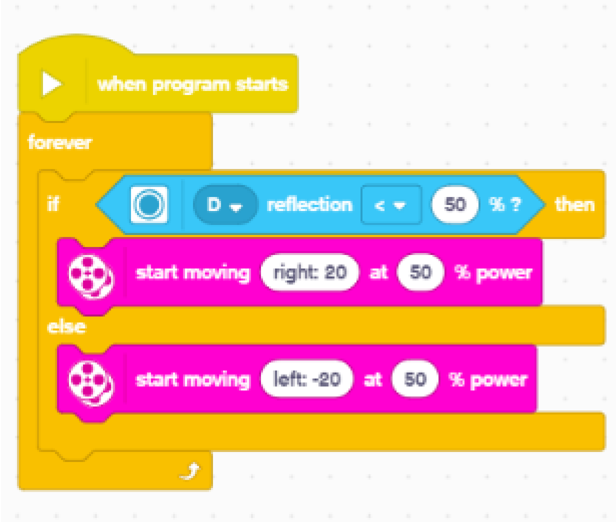


Line following to the corner

- Couple comments regarding moving around the table
  - It is possible to travel around the edge of the table with only one color sensor, but it is more difficult and potentially less reliable than using two colors sensors
  - Black tape is used to denote zones. We can use the black tape to line follow to the end of a **zone**.

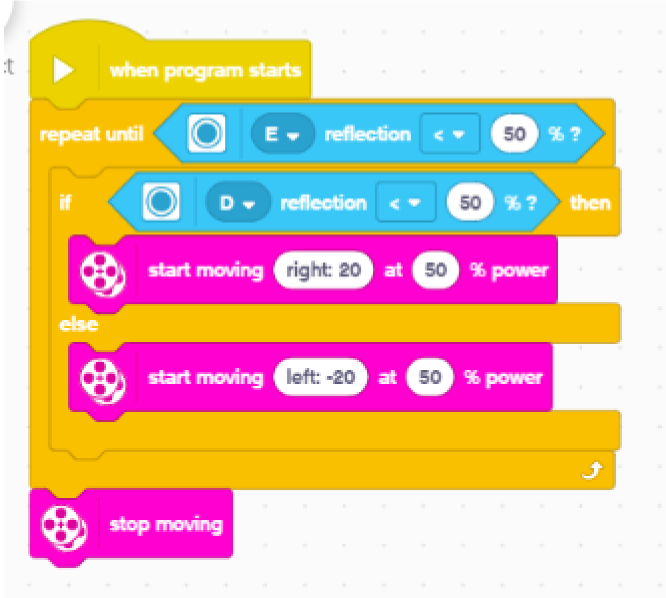
Line following to the corner

- Recall our line following program
  - Let’s modify the program to stop when the robot reaches the end of the table



Using this program, the robot will line follow continuously. How can we make the robot stop when it reaches a corner?

Line following to the corner or to the target zone



https://youtu.be/tC95PcGBs2o

Task 3

Line following a given distance

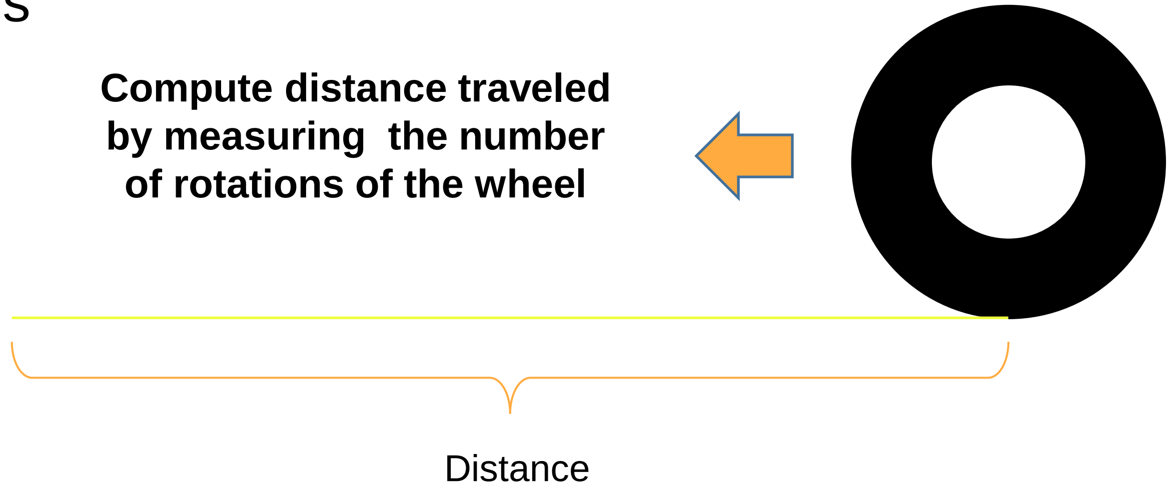
Line following a given distance

- Approach
  - Modify LineFollowStop to stop when the robot travels a given distance

- Tools needed
  - Line following
  - Measure distance traveled

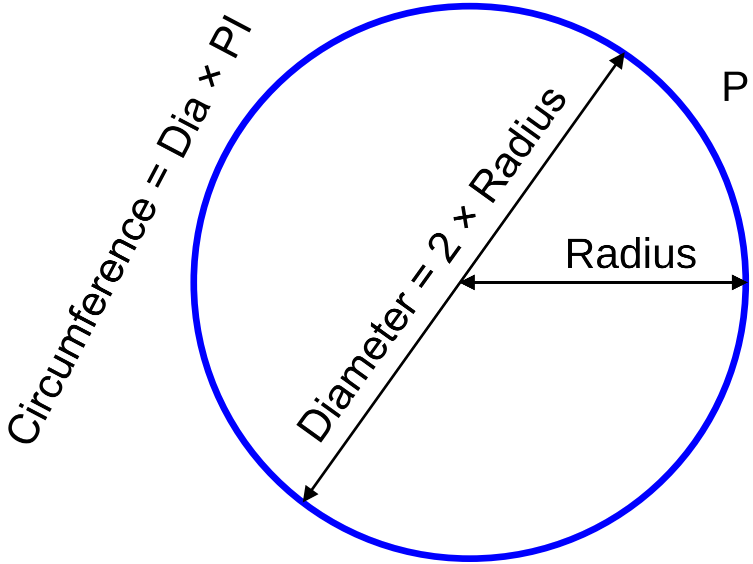
Measuring Distances

- How do we measure distance traveled?
- Let’s determine how far the robot travels moving forward for 2 seconds



Measuring Distances

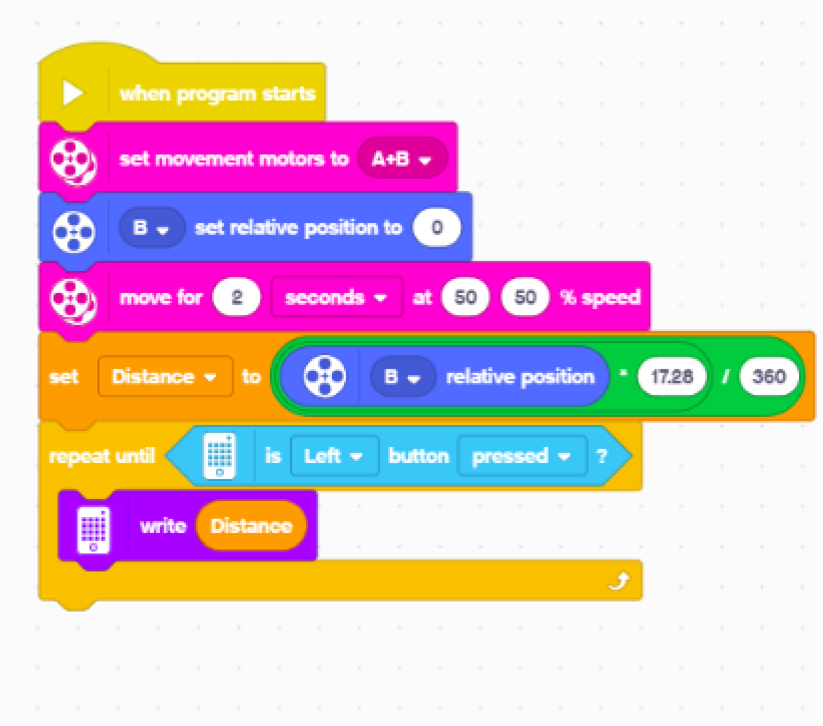
- Use the wheel geometry



How can we use this information?

Measuring Distances

- For each rotation of the wheel, the robot travels (Wheel Diameter) x (PI)
  - Distance = (Wheel Diameter) x (PI) x (# Rotations)
  - Distance = (5.5 cm) x (PI) x (# Rotations)
  - Distance = (17.28 cm) x (# degrees/360)
- Note: Right side motor (B) moves in positive direction, Left side motor (C) side moves in negative direction



https://youtu.be/IsRGvC7vaMI

Measuring Distances

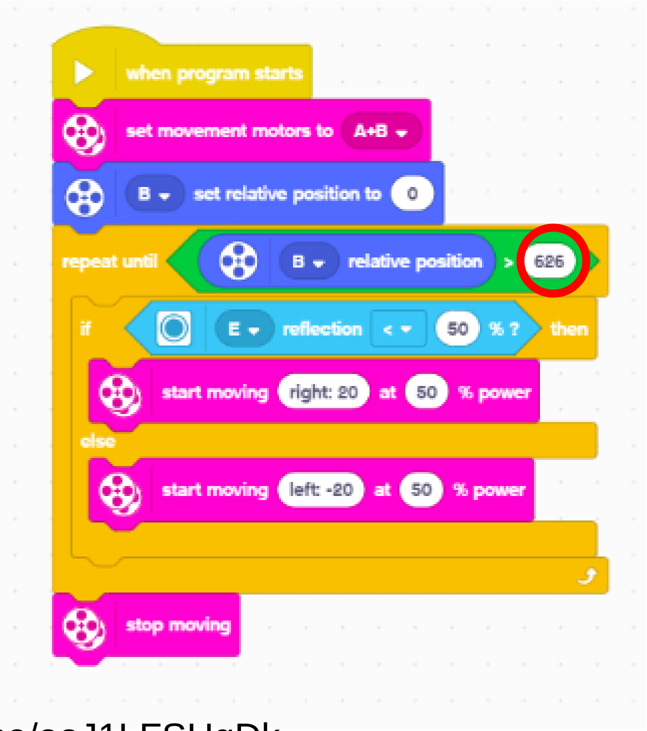
- Example
  - Let's program the robot to line follow for 30 cm
    - Distance = 30 cm
  - Number of rotations
    - Distance = (Wheel Diameter) x (PI) x (# Rotations)
    - Distance = (Wheel Diameter) x (PI) x (# degrees/360)
    - Solve for (# degrees)

$$(\#\ degrees) = \frac{Distance \times 360}{(Wheel\ Diameter) \times (PI)}$$

$$(\#\ degrees) = \frac{30\ cm \times 360}{(5.5\ cm) \times (PI)} = 626\ degrees$$

Line following a given distance

- Line follow a desired distance



https://youtu.be/ooJ1LFSHgDk

Task 4

Turning the robot

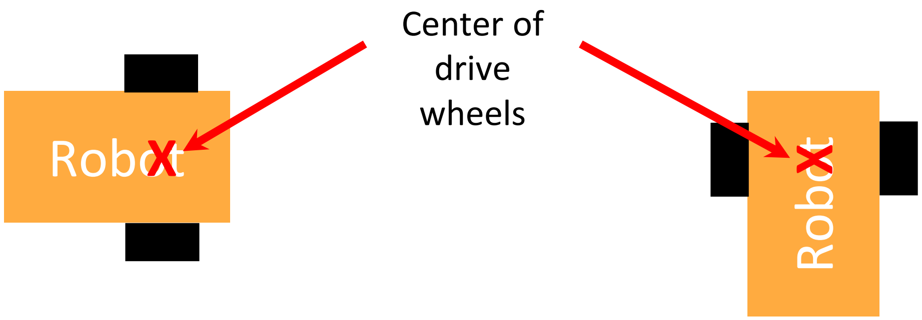
Turning The Robot

- For our example here, we wish to turn the robot 90 degrees
- There are several methods for turning a tripod robot. We will focus on two methods
  - “Spin” turn
  - “Swing” turn

90 Degree Spin

- Let's have the robot spin 90 degrees CCW
- The robot will rotate about center of the drive wheels

Starting Position **Final Position**



90 Degree Spin

- To spin 90 degrees CCW, we use the Move block as shown here

- Set the steering to -100. This causes:
    - Right wheel to rotate forward
    - Left wheel to rotate reward
    - Equal and opposite rotations

- Now, we need to determine the correct number of rotations

90 Degree Spin

- You can determine the proper number of rotations mathematically; however, the result typically needs some adjustment due to lash in the motors

- For today's class, we will use trial and error to find the number of rotations that cause the robot to turn 90 degrees

90 Degree Spin

- We can use one block to spin the robot

- For our sample robot, it takes 0.55 rotations to spin the robot 90 degrees

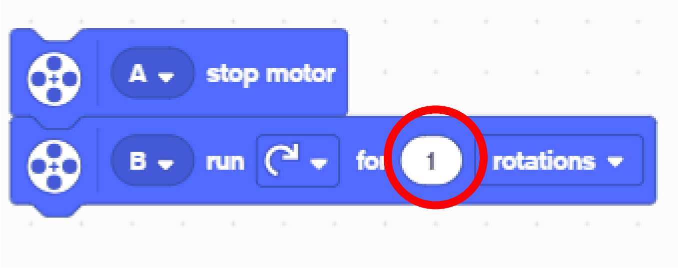
https://youtu.be/k5cqhPFShFc

90 Degree Swing

- Let's have the robot swing 90 degrees CCW
- The robot will rotate about a locked wheel (denoted by red X)

90 Degree Swing

- To swing, we lock the left motor and power the right motor to turn the robot



- For our sample robot, it takes 1.0 rotations to swing the robot 90 degrees

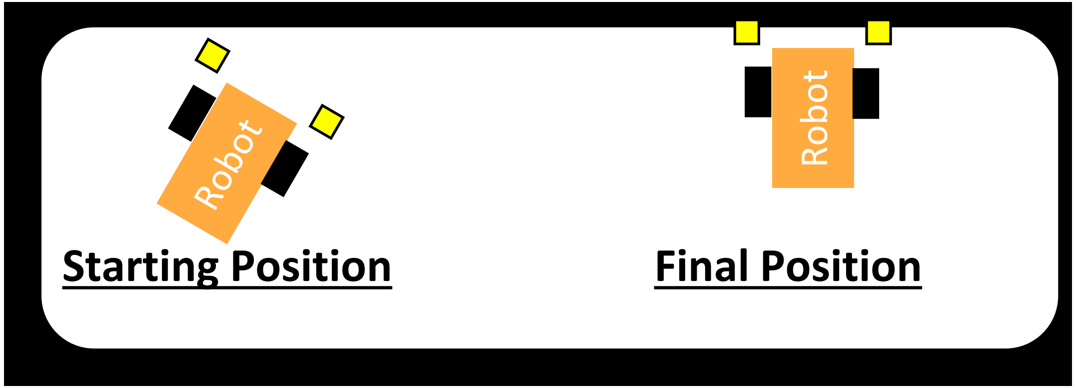
https://youtu.be/OlCKeLEf9Us

Task 5

Aligning the robot to an edge

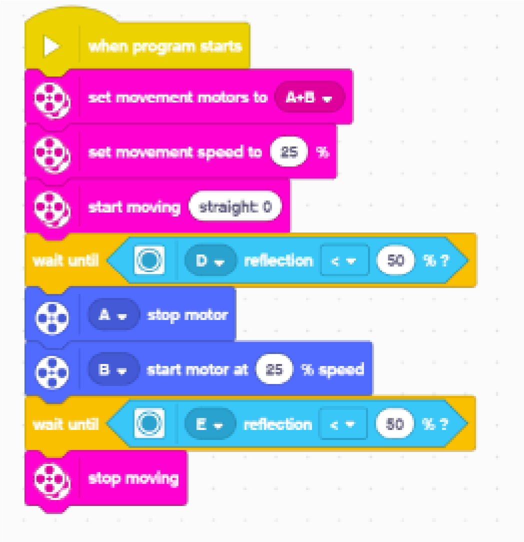
Aligning the robot to an edge

- In some situations we desire align with robot to an edge of the table as shown below
- Assuming the starting position below, how can we program the robot to reach the final position that is aligned with the edge of the table?

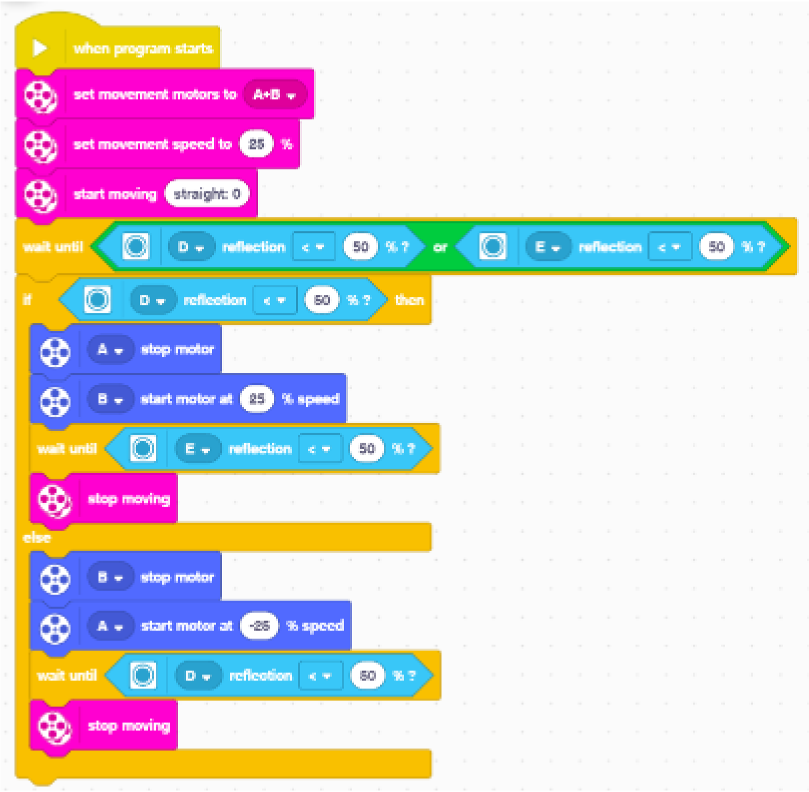


Aligning the robot to an edge

- Travel until color sensor D reaches the edge, swing robot until it is aligned with the edge



Align if either side detects edge first



https://youtu.be/998DFfMUXw0

ROBOFEST
LAWRENCE TECHNOLOGICAL UNIVERSITY

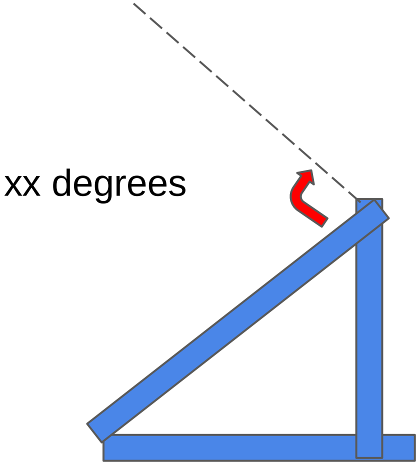
Task 6

Manipulating Paper Rolls
Moving the Arm and Claw

How to Control the Arm and Claw

- Time
- Rotation(encoder degrees)

Note: To use rotation, you need to determine what direction to run the motor to raise the arm. For this workshop, the arm is run counter clockwise (negative direction)

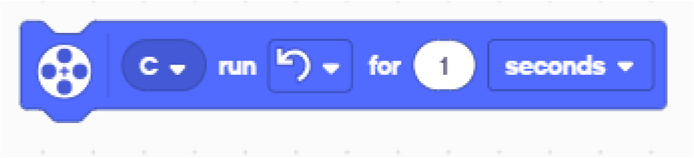


Arm all the way down
0 degrees

Moving the Arm- Using time

- Advantages
  - Simple
  - Easy to program
  - Will not get stuck
- Disadvantages
  - Can be imprecise
  - Repeatability
- How to do it
  - Set a motor block to “seconds”
  - Select the motor port
  - Select the direction
  - Select the duration

Raise Arm



Lower Arm

Moving the Arm- Using encoder

- Advantages
  - More precise
  - More repeatable
- Disadvantages
  - More difficult to program
  - Can get stuck
- How to do it
  - Establish a “zero” point
  - Determine direction of motor
  - Set limits

Raise arm,

Wait 2sec

Lower arm back to 0

Moving the Claw

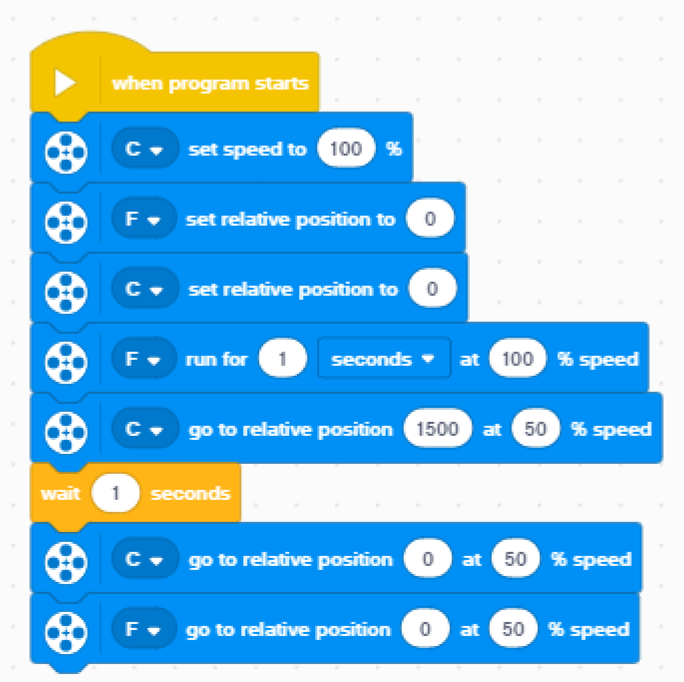
- May need to overdrive motor to get enough grip
- Use encoder to open back to zero position

Set Power and zero position at beginning of program

Use time to close

Use rotation to open

Combine Arm and Claw Movement



Relative Motor Position

This block returns the number of degrees that the specified motor has turned since the program started or was reset by the *Set Relative Motor Position to 0* Block.

Set Relative Motor Position to 0

This block sets the relative position of one or more motors to a specified value. Use a value of '0' to reset the relative position.

Go to Relative Motor Position at Speed

This block runs one or more motors to a relative position at a specified speed. Unlike the absolute position that's used in the Go to Position Block, the relative position has no range limit and can be preset with the *Set Relative Motor Position to 0 Block*.

https://youtu.be/5QyelRBtxcc

Task 7

Building MyBlocks

My Blocks

- Solving the Robofest Game challenge will typically require a fairly large program (around 100 blocks is not unreasonable)
- Very large programs can be difficult to understand, navigate and use
- To alleviate this issue, the Scratch software has a My Block Builder to create custom blocks that can replace sections of your program

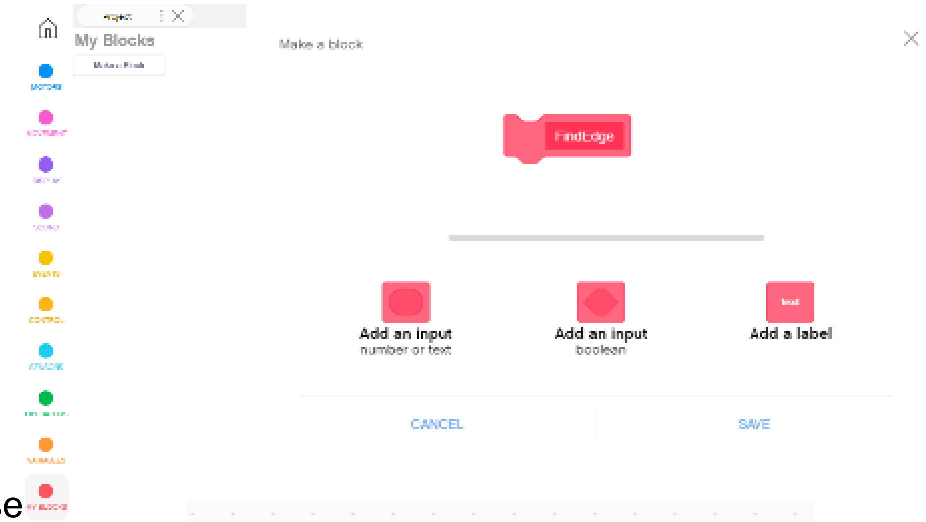
My Blocks

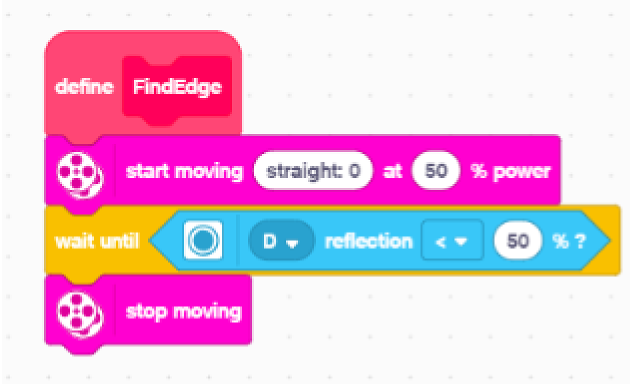
- For example, let's assume you have a section of code that completes the following:
    - Move forward until the edge of the table is found with color sensor D, then stop
- The code may look like this

- My blocks will allow us to convert this to a single block

My Blocks

- Creating a My Block
  1. Click on “MyBlocks”
  2. Click on “Make a Block”
  3. Create a name “FindEdge” for the block
  4. Click “save
  5. Add blocks to be run when My Block is use





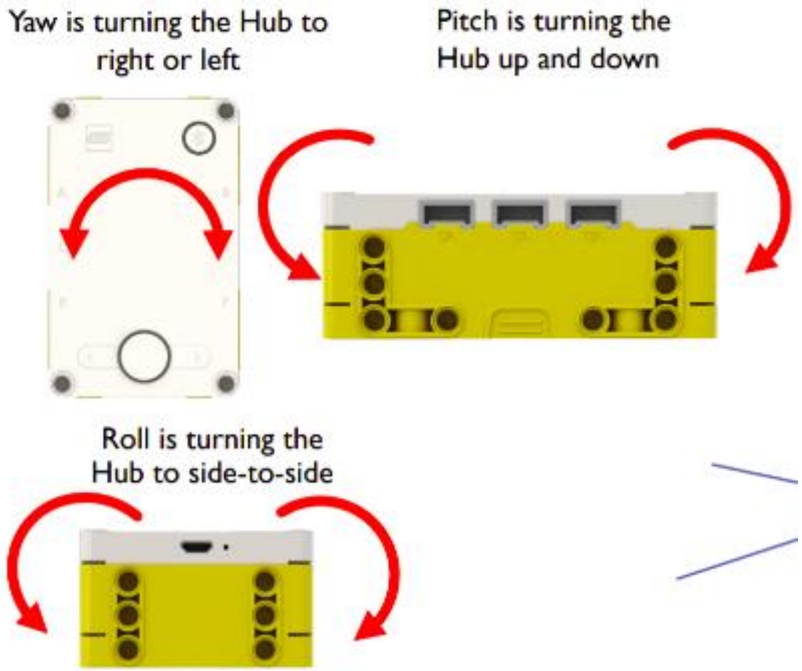
This creates a My Block called FindEdge that will be located in the My Blocks Pallet

Task 8

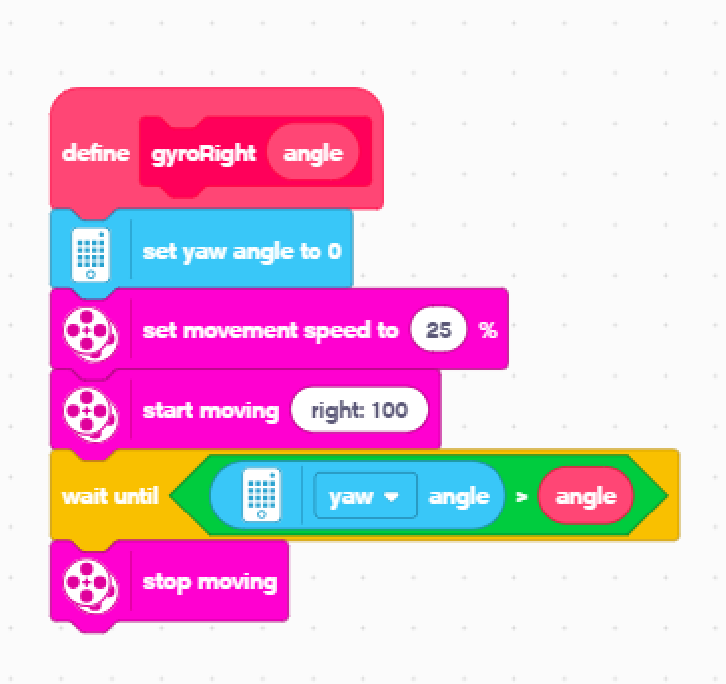
Gyro Turns

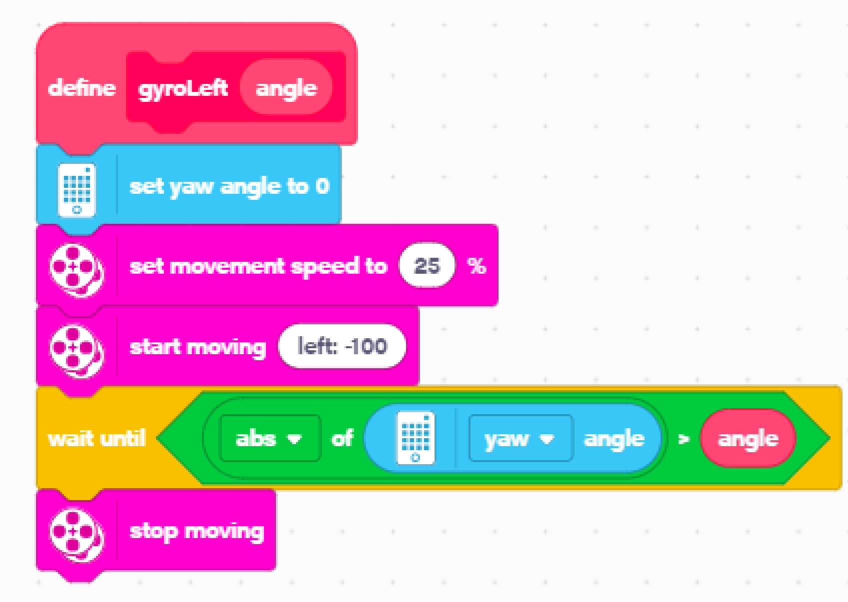
More Precise Turns with a Gyro

- Spike Prime has a built in Gyro sensor
- Gyro sensor can be used for turns



Turning with Gyro





Putting It All Together

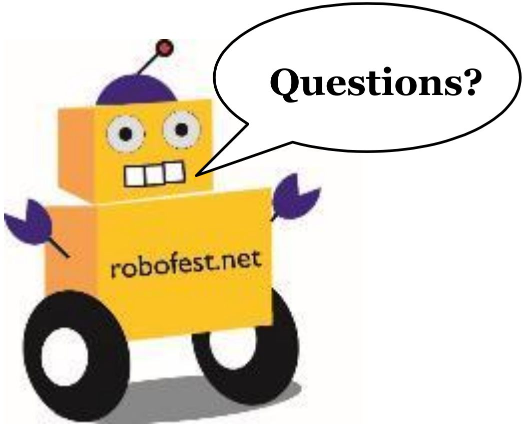
- In this course we learned about
  - Finding the edge of the table
  - Following the edge of the table
  - Stop line following
    - When you reach a corner
    - When you reach a given distance
  - Turning the robot
  - Aligning the robot to an edge
  - Manipulating Paper Rolls
  - Building MyBlocks
  - Gyro turns

Test for Knowledge

Post Assessment Link:

https://docs.google.com/forms/d/e/1FAIpQLSfnXgkErHErtzUakWI9Da1twiDdcBfyC24D3RxKF5FitXH8qQ/viewform?usp=sf_link

Little Robots, Big Missions



robofest@LTU.edu
LTU Computer Science

2021 Workshops

Sponsored by

Lawrence Technological University

Computer Science