Speedy Delivery

Learn how to pick up objects and move them around with the VEX Clawbot IQ!

SPARK
SEEK

Discover new hands-on builds and programming opportunities to further your understanding of a subject matter.

The Completed Look of the Build

The completed Clawbot IQ build.

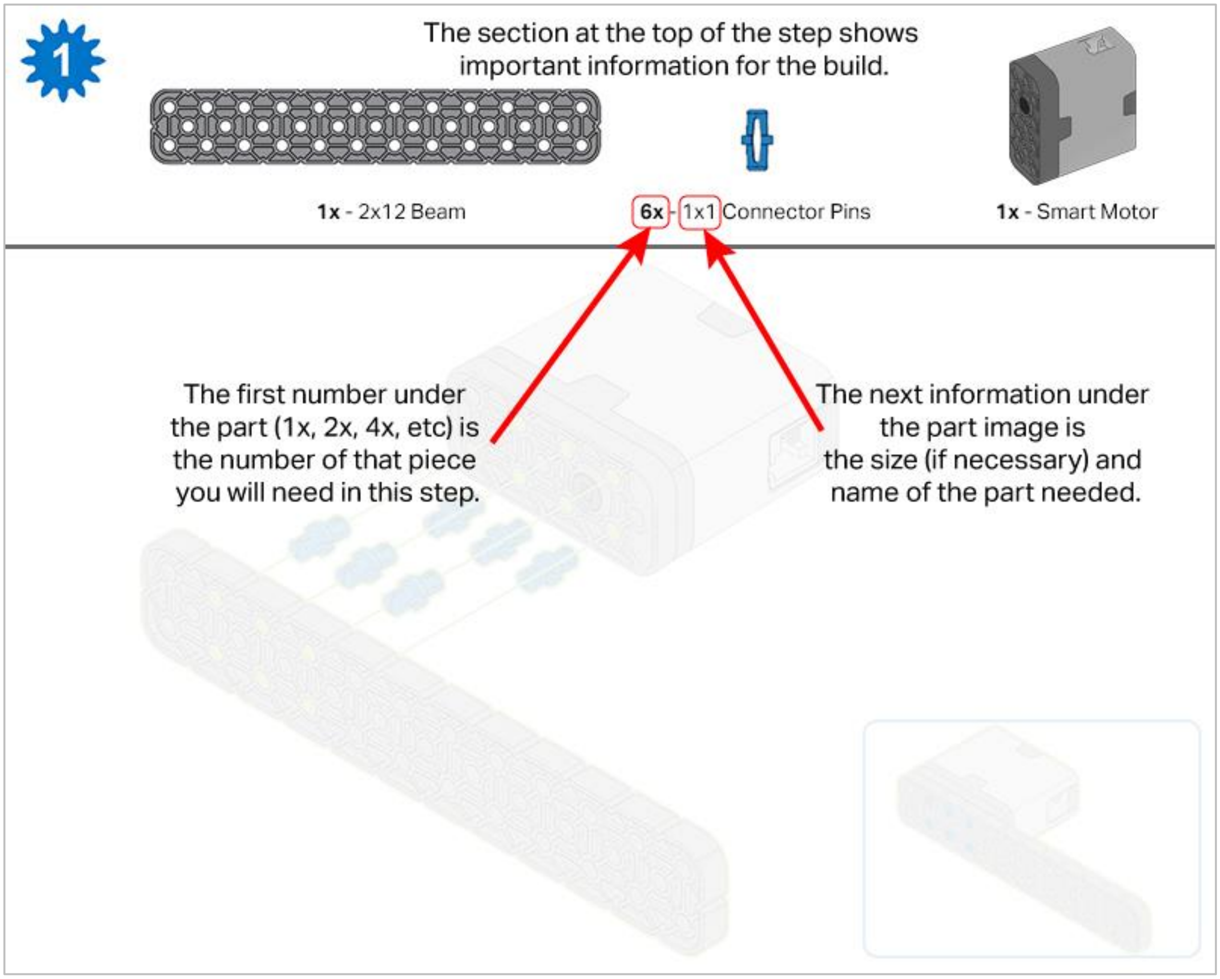
This robot is designed so that it can be built quickly and drive around either autonomously or with the Controller in a short amount of time.

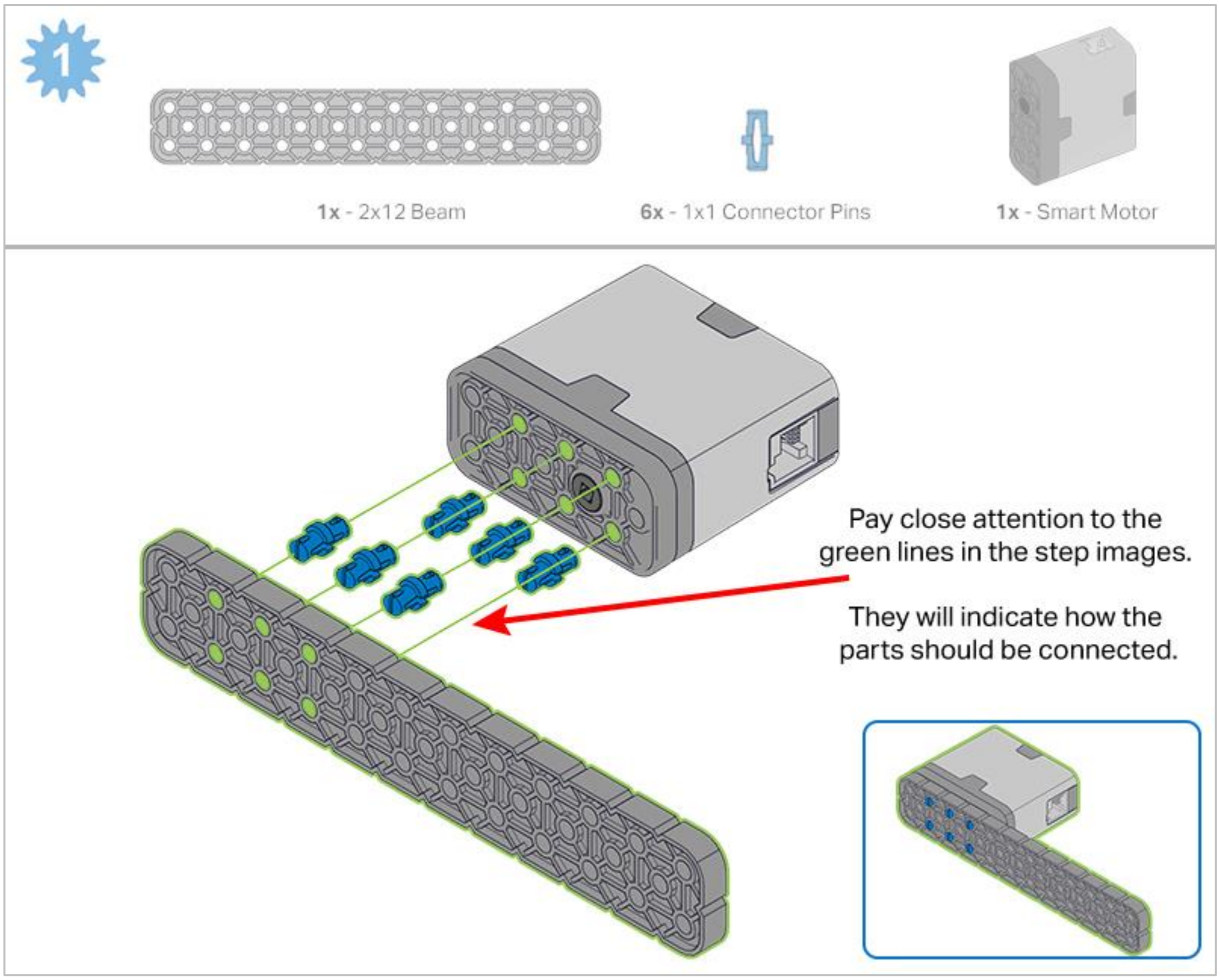
Build Instructions - Drivetrain + Distance

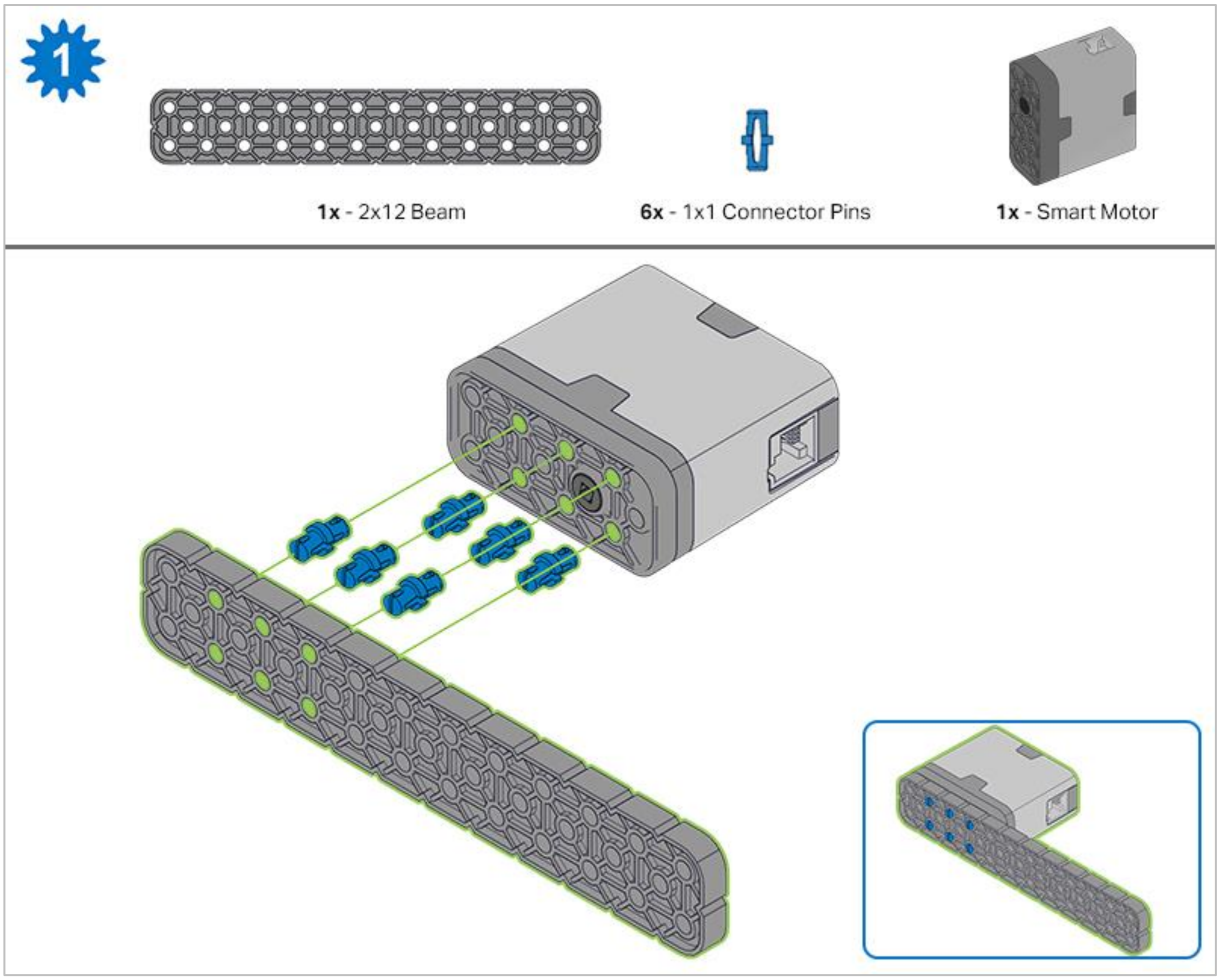
Build Instructions Summary

- **Drivetrain + Distance Building Instructions (19 steps):**
    - Right Wheel: steps 1 to 6
    - Left Wheel: steps 7 to 12
    - Distance Sensor: steps 13 to 19
- **Building Tips for All Steps:**
    - The section at the top of the step shows important information for the build. The first number under the image of the part (1x, 2x, 4x, etc) is the number of that piece you will need in this step. The next information under the part image is the size and description of the part needed.
    - The finished step is illustrated in the box in the lower right corner.

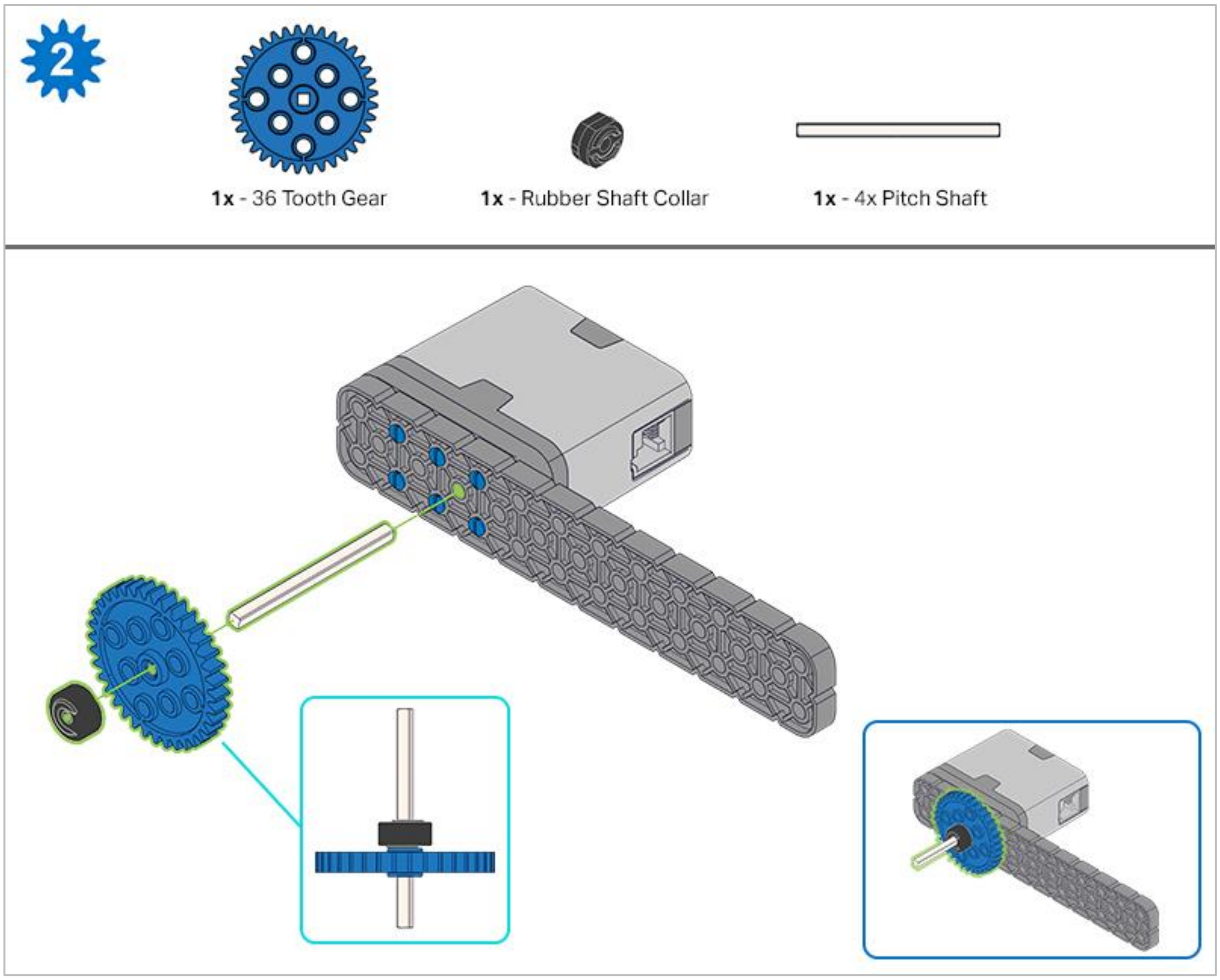
- Play close attention to the green lines in the step images. They will indicate how the parts should be connected.



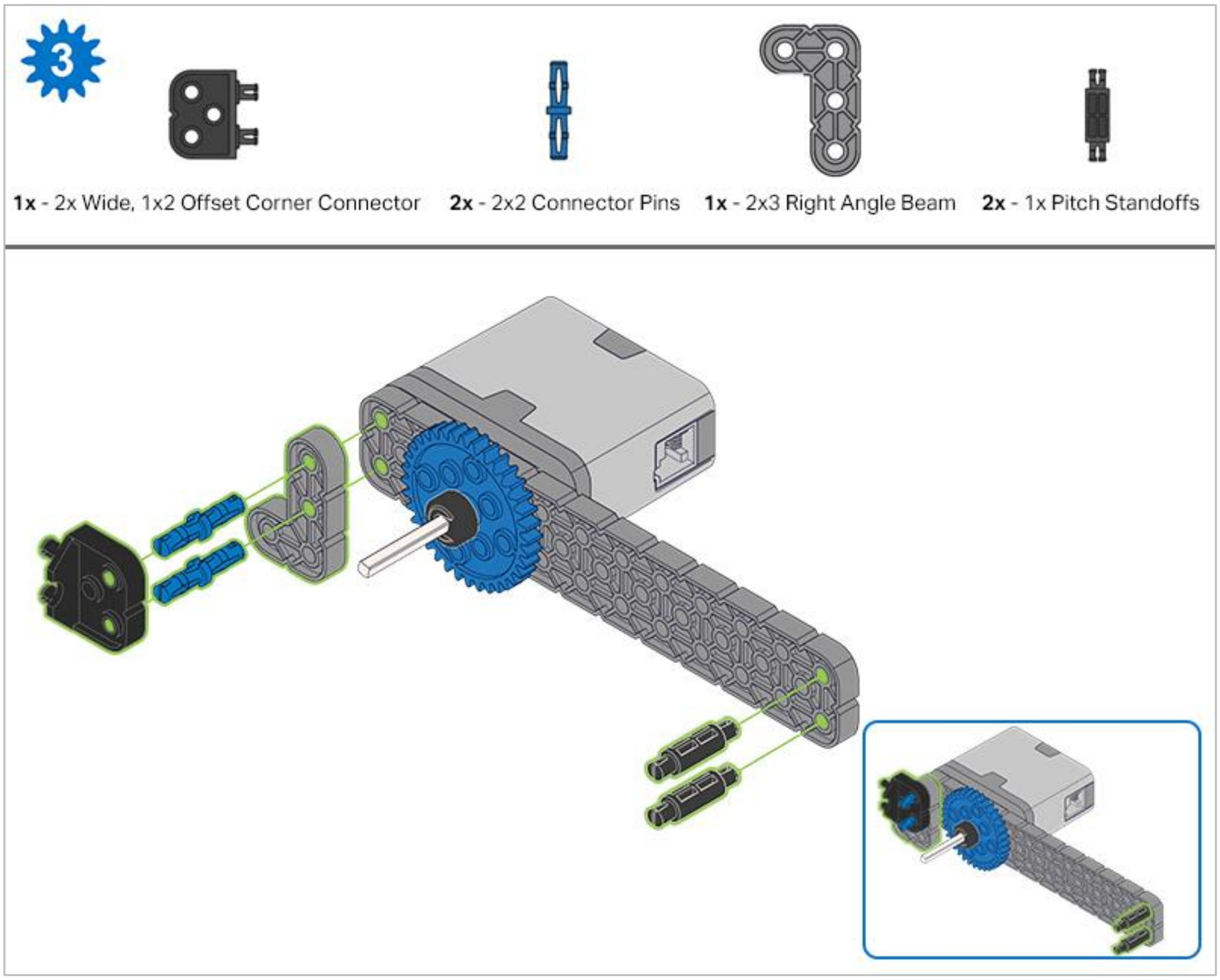

1
1x - 2x12 Beam
6x - 1x1 Connector Pins
1x - Smart Motor
Pay close attention to the green lines in the step images.
They will indicate how the parts should be connected.

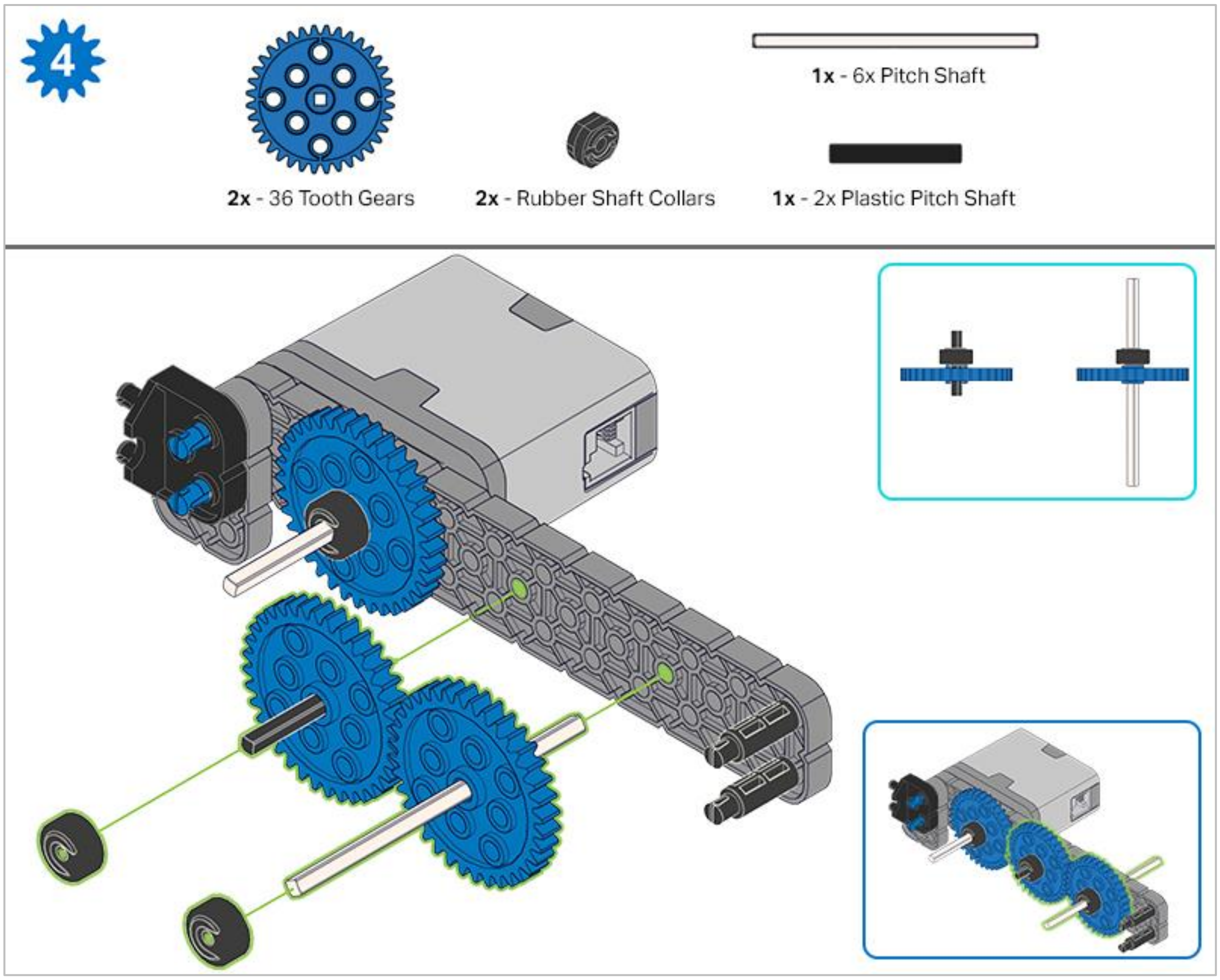


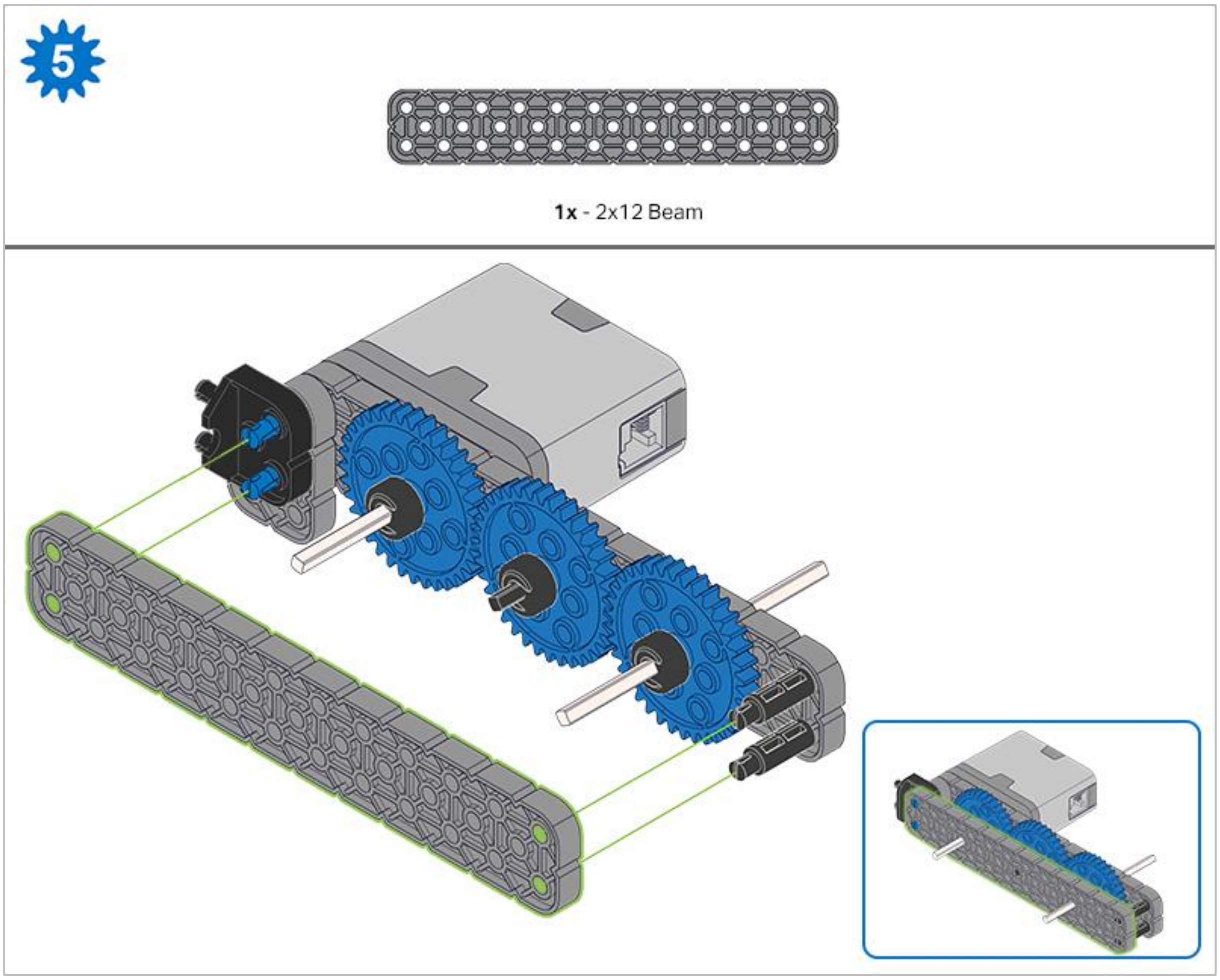
Step 1: Count all pieces before starting your build and have them readily available. Each team member should find the pieces for their section.



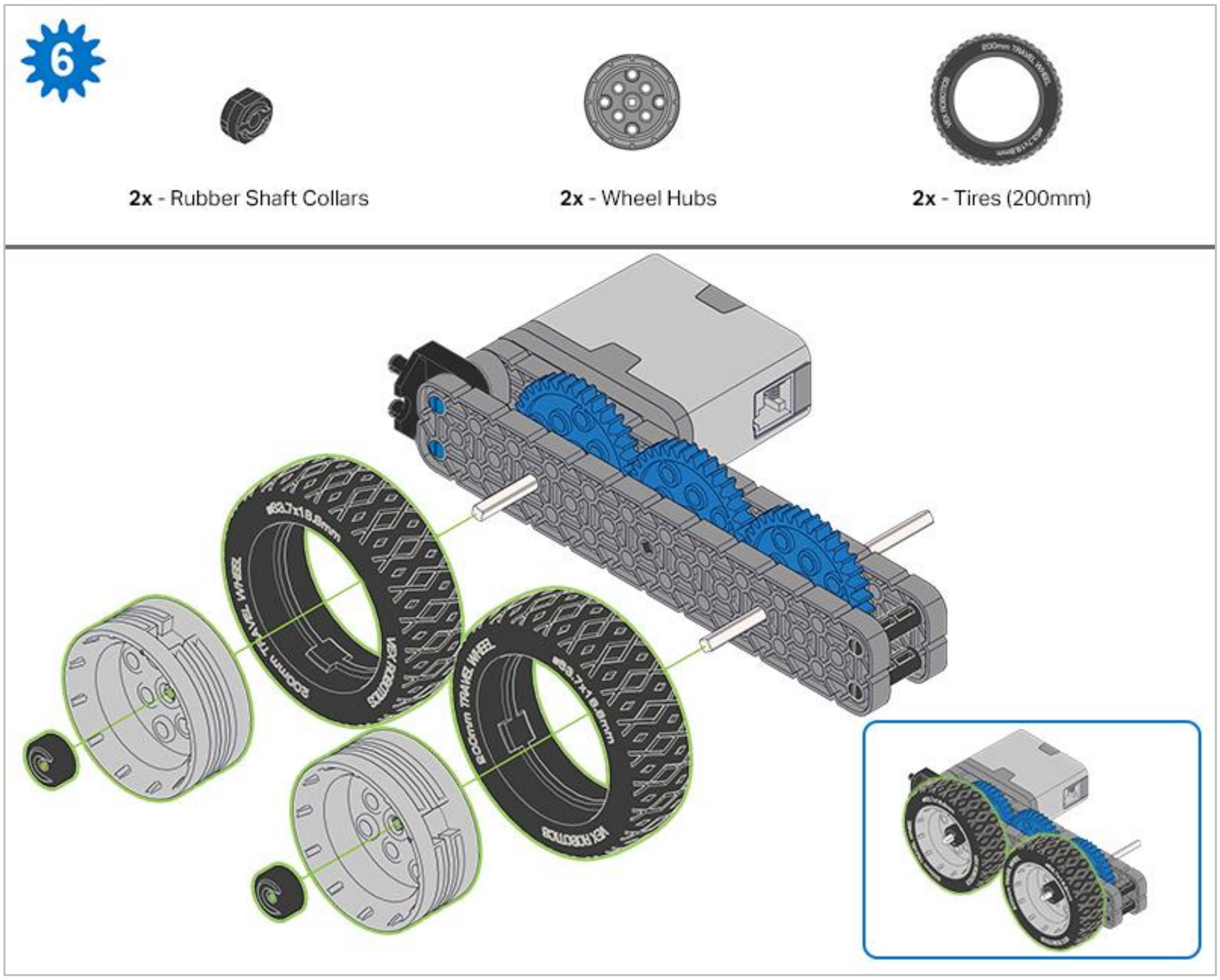
Step 2: When adding the 4x Pitch Shaft, twist the pitch shaft to check for tension while turning. If it spins freely, it is not properly inserted into the motor.


3
1x - 2x Wide, 1x2 Offset Corner Connector
2x - 2x2 Connector Pins
1x - 2x3 Right Angle Beam
2x - 1x Pitch Standoffs

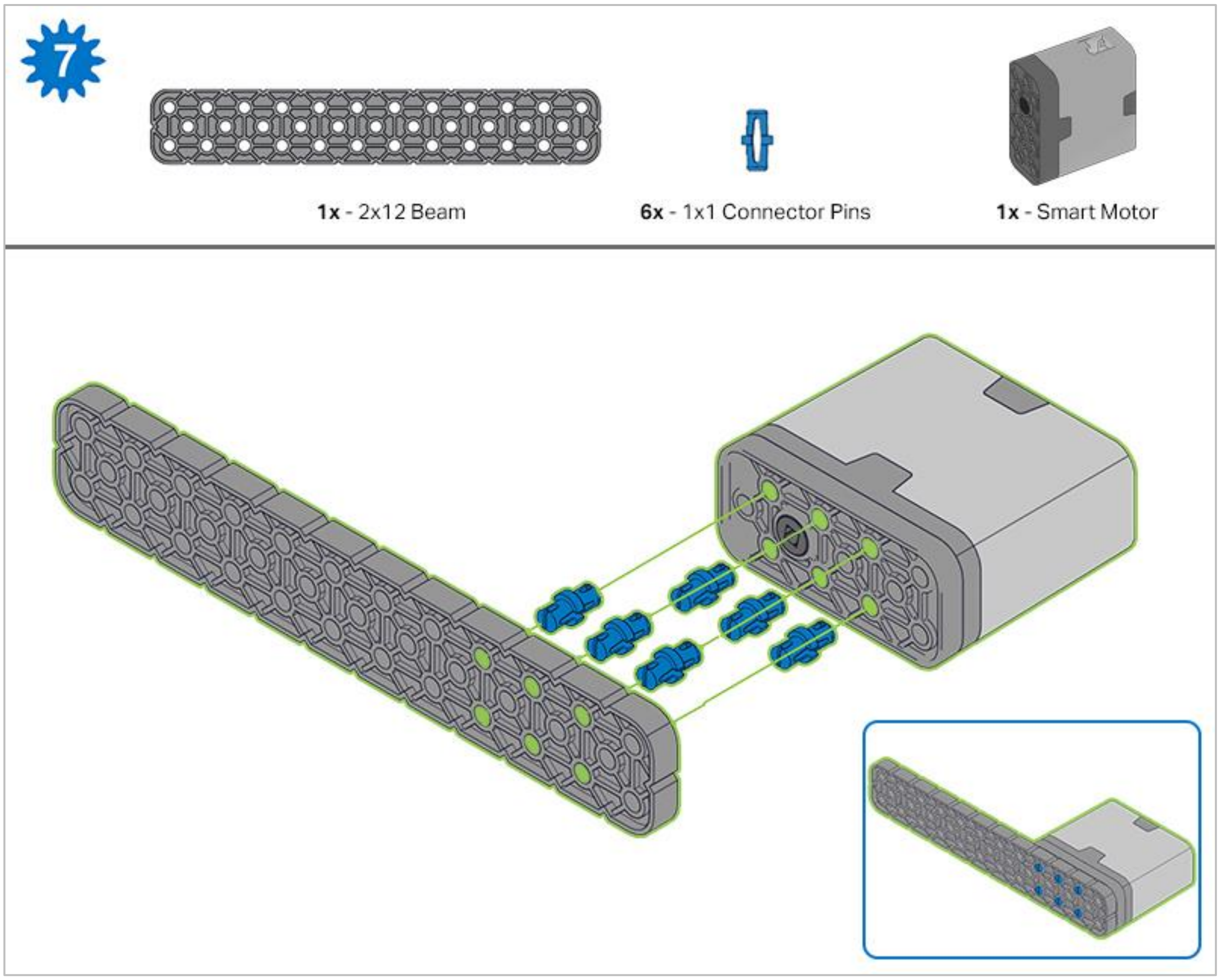

4
2x - 36 Tooth Gears
2x - Rubber Shaft Collars
1x - 6x Pitch Shaft
1x - 2x Plastic Pitch Shaft

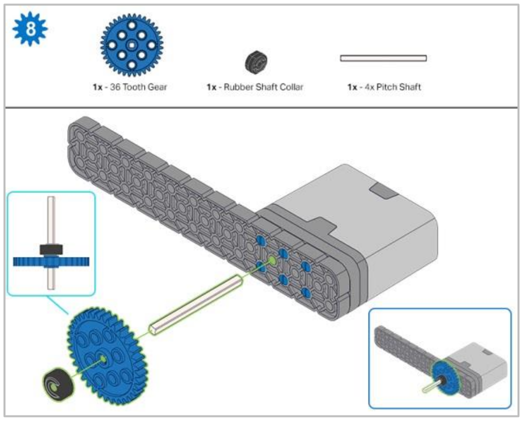


Step 5: Make sure the gears fit together properly before locking the 2x12 Beam in place.

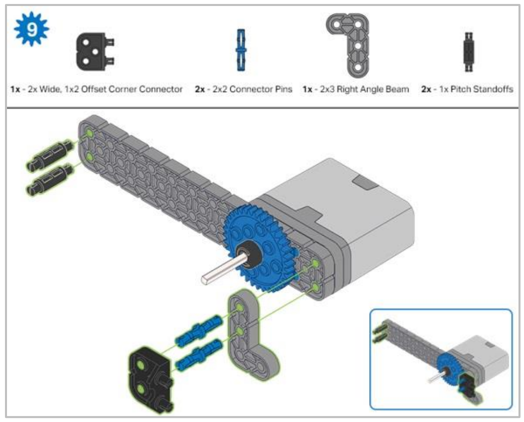


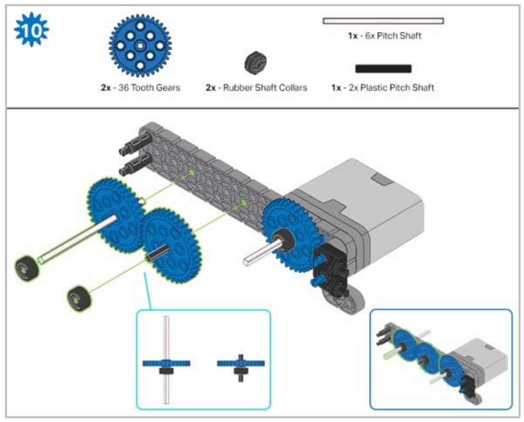
Step 6: After attaching the wheels, twist the wheel that has the shaft going into the motor. If the wheel spins freely and without tension, the 4x Pitch Shaft has slipped out of place.

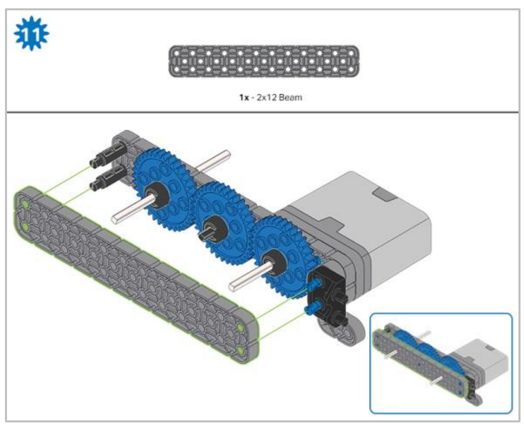




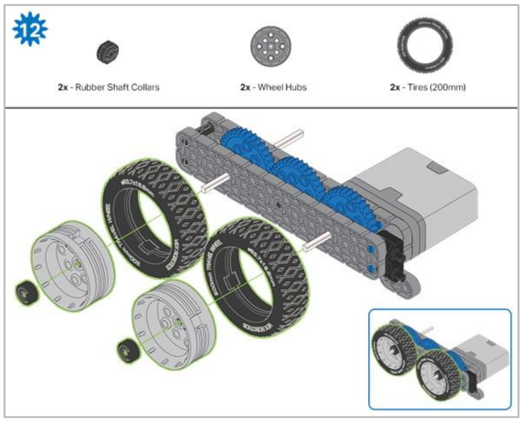
Step 8: When adding the 4x Pitch Shaft, twist the pitch shaft to check for tension while turning. If it spins freely, it is not properly inserted into the motor.



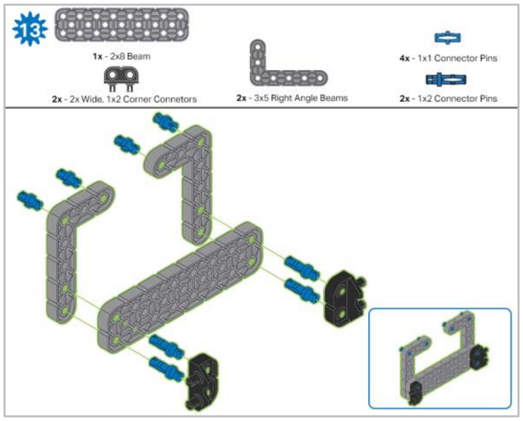


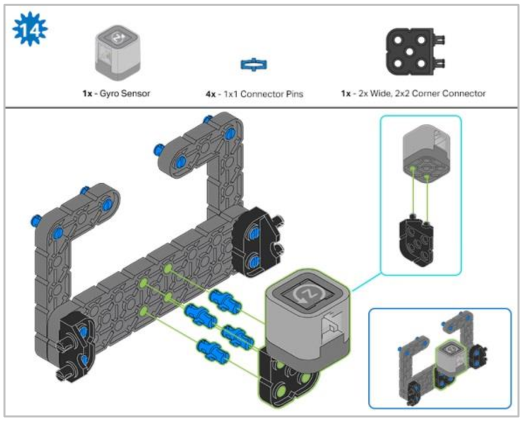


Step 11: Make sure the gears fit together properly before locking the 2x12 Beam in place.

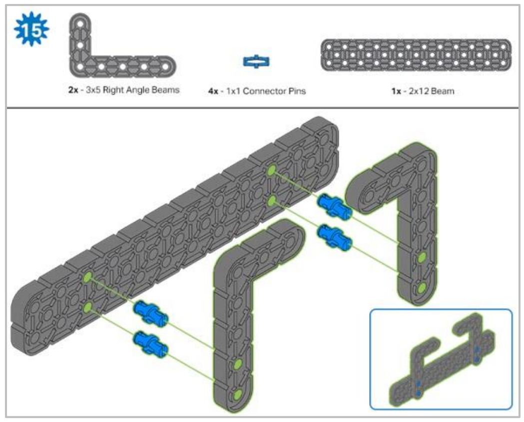


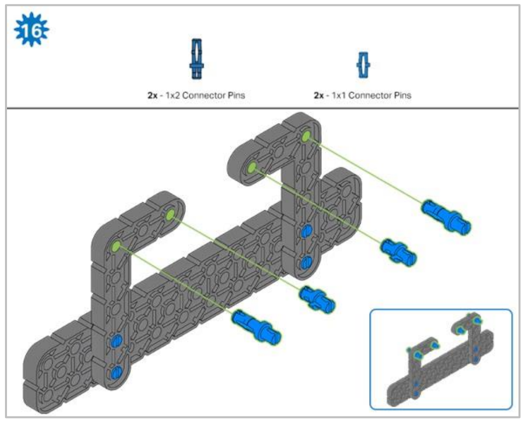
Step 12: After attaching the wheels, twist the wheel that has the shaft going into the motor. If the wheel spins freely and without tension, the 4x Pitch Shaft has slipped out of place.

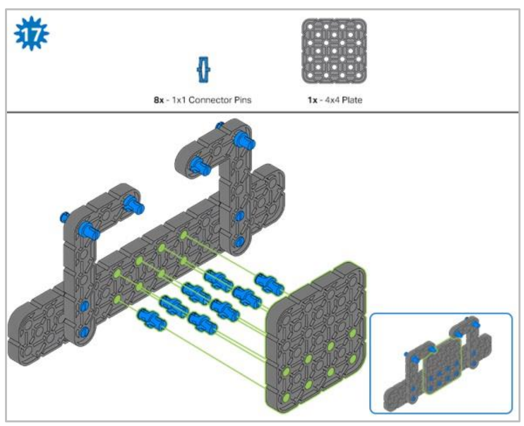


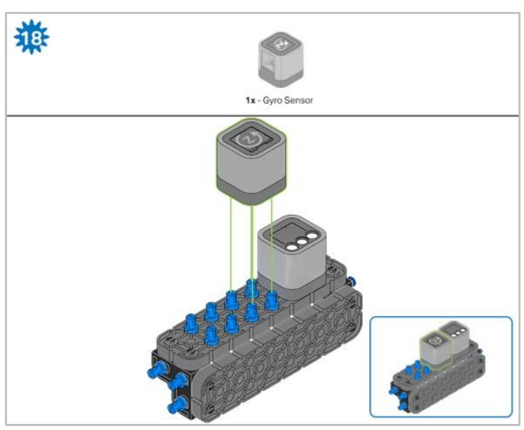


Step 14: Make sure the Gyro is placed the correct way to allow correct cable access.

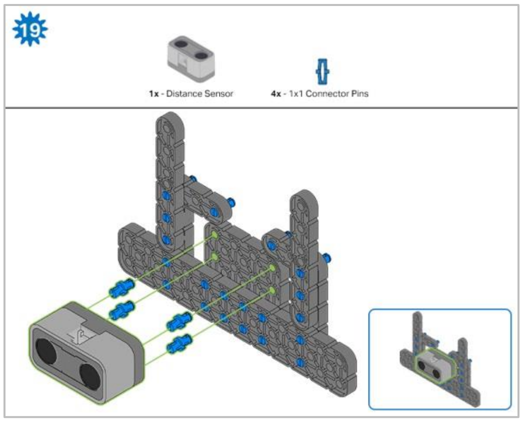








Step 18: The orange arrows mean spin the build around.



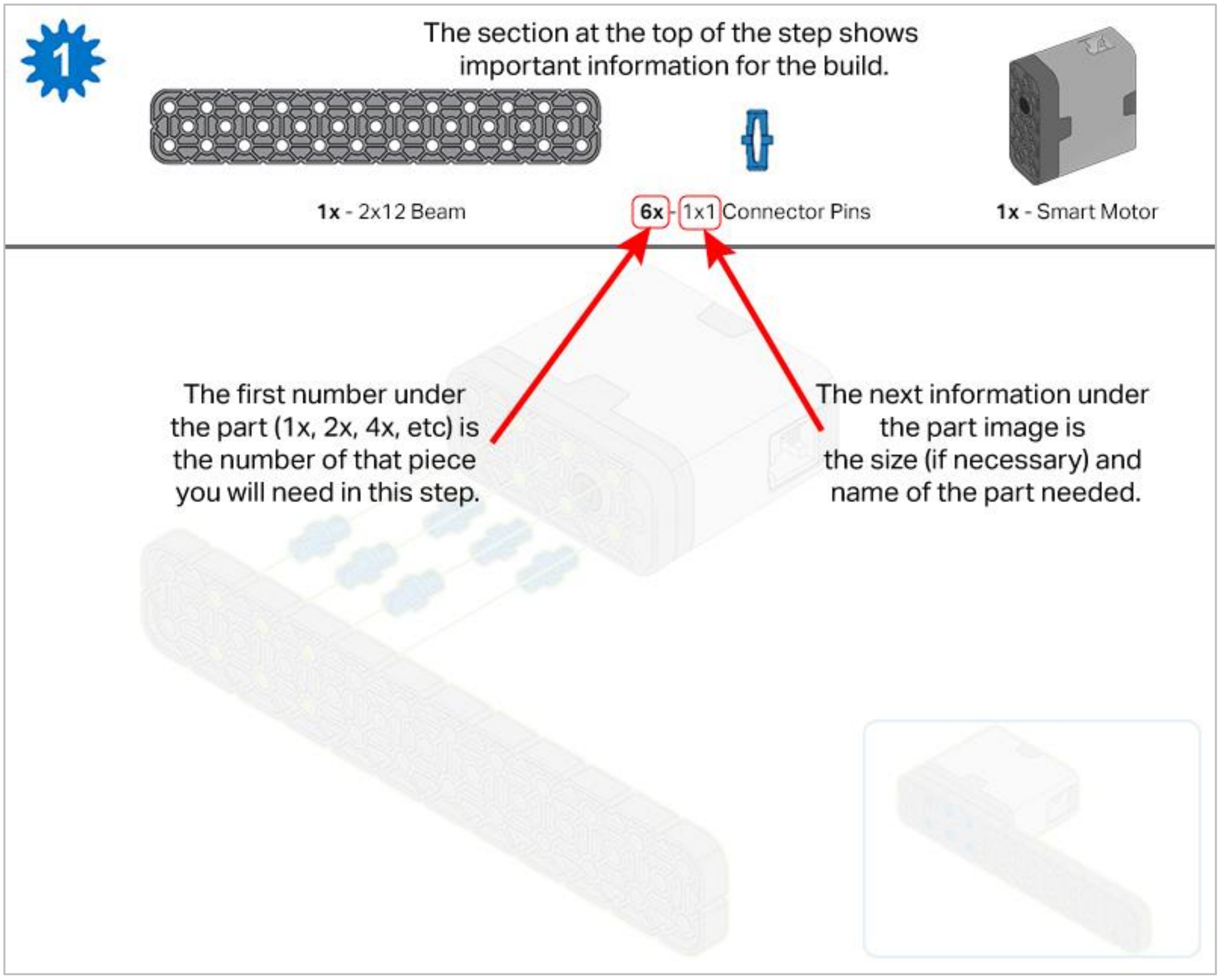
Step 19: When attaching the Distance Sensor, do not push on either of the two mesh covered openings. This will damage the sensor. Ensure the sensor is placed in the correct way to allow cable access.

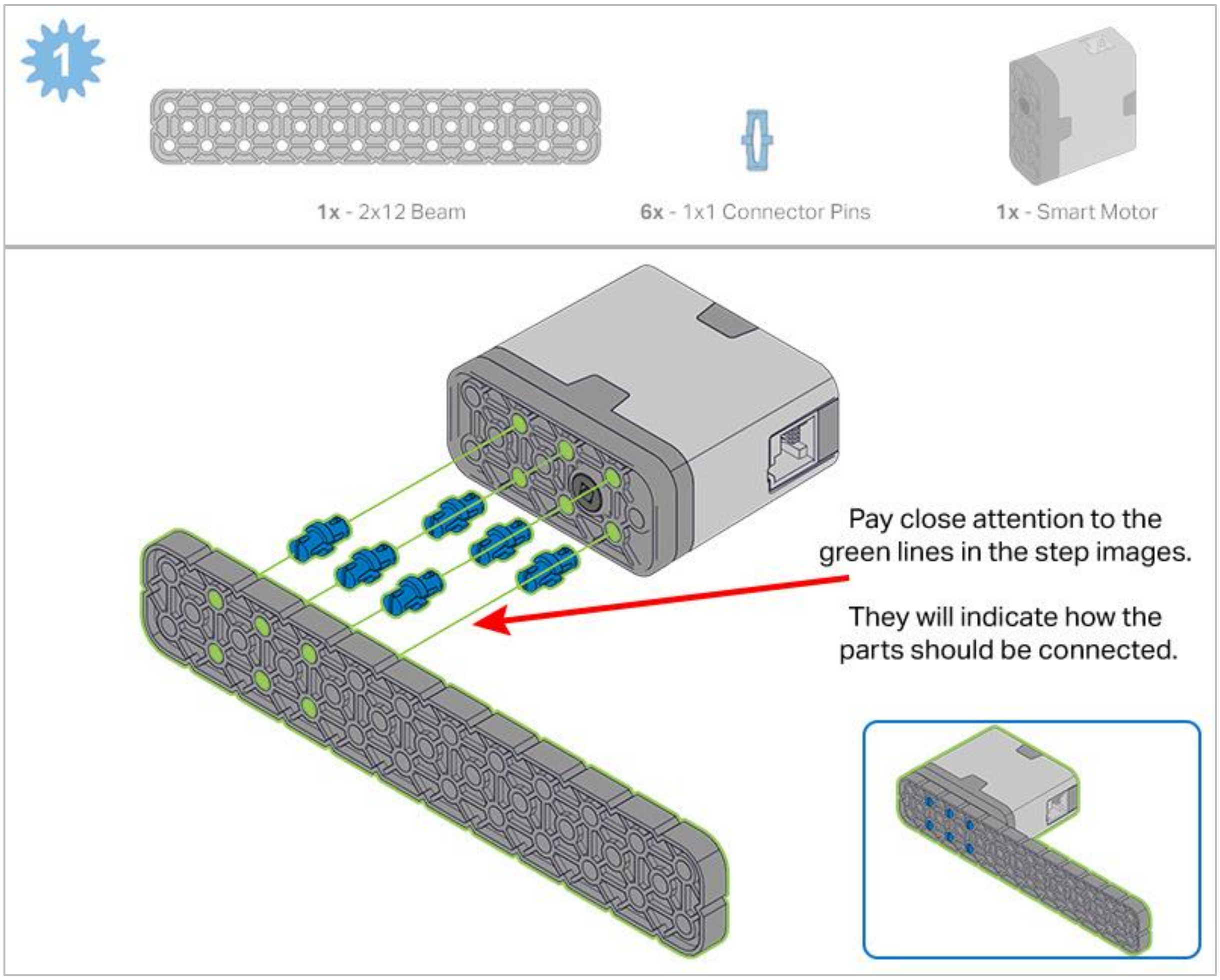
Build Instructions - Robot Frame

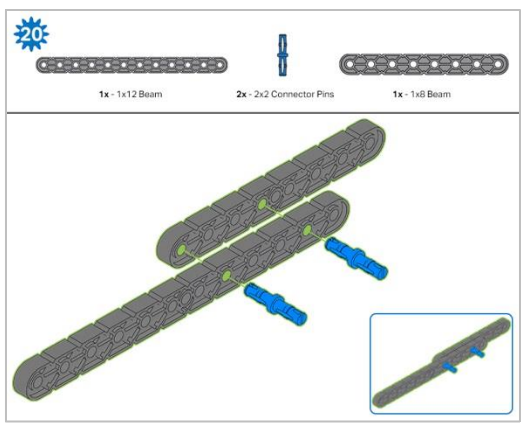
Build Instructions Summary

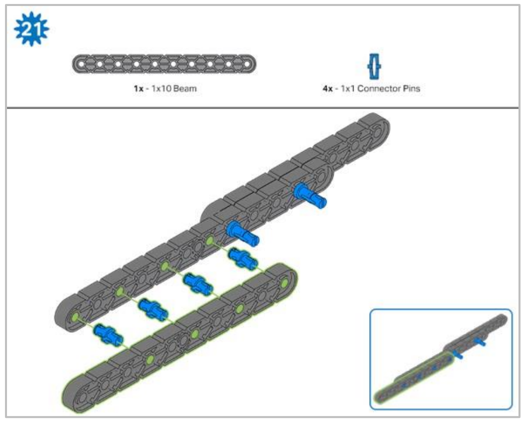
- **Robot Frame Building Instructions (22 steps):**
  - Cargo Holder: steps 20 to 28
  - Arm Base: steps 29 to 41
- **Building Tips for All Steps:**
  - The section at the top of the step shows important information for the build. The first number under the image of the part (1x, 2x, 4x, etc) is the number of that piece you will need in this step. The next information under the part image is the size and description of the part needed.
  - The finished step is illustrated in the box in the lower right corner.

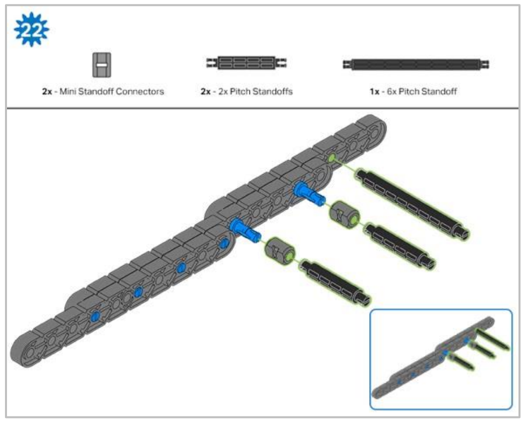
- Play close attention to the green lines in the step images. They will indicate how the parts should be connected.

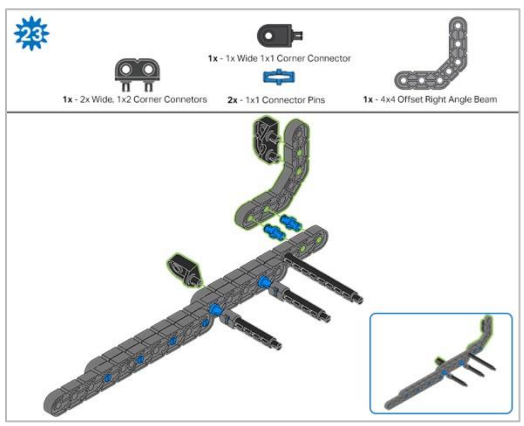


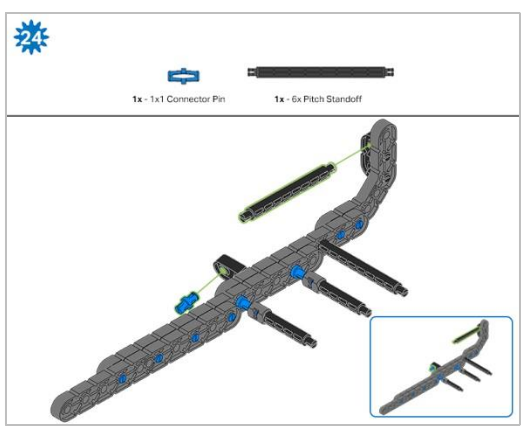

1
1x - 2x12 Beam
6x - 1x1 Connector Pins
1x - Smart Motor
Pay close attention to the green lines in the step images.
They will indicate how the parts should be connected.

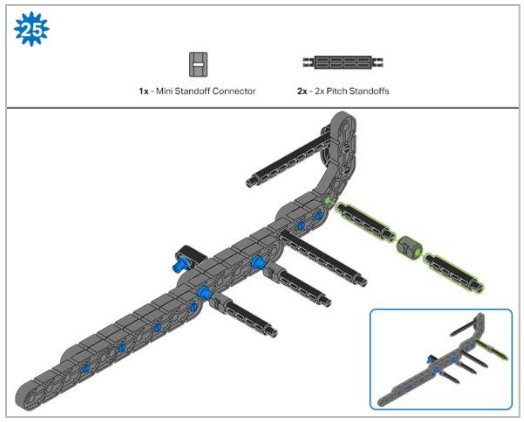

20
1x - 1x12 Beam
2x - 2x2 Connector Pins
1x - 1x8 Beam

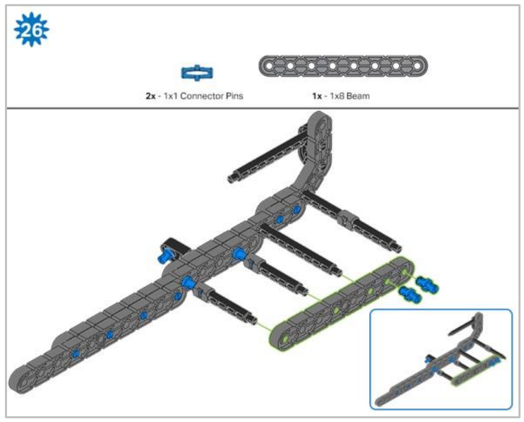

21
1x - 1x10 Beam
4x - 1x1 Connector Pins

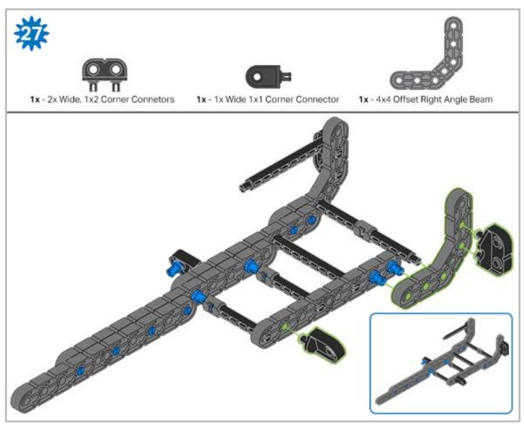

22
2x - Mini Standoff Connectors
2x - 2x Pitch Standoffs
1x - 6x Pitch Standoff

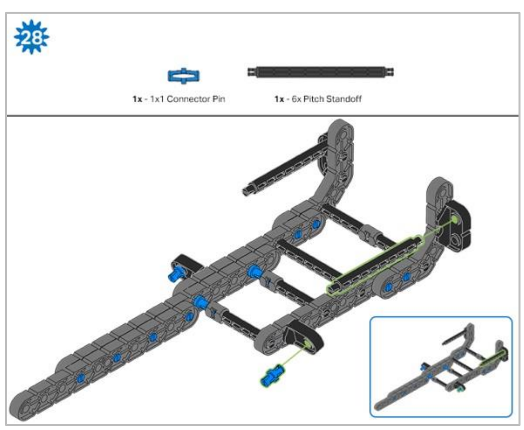

23
1x - 1x Wide 1x1 Corner Connector
1x - 2x Wide, 1x2 Corner Connetors
2x - 1x1 Connector Pins
1x - 4x4 Offset Right Angle Beam

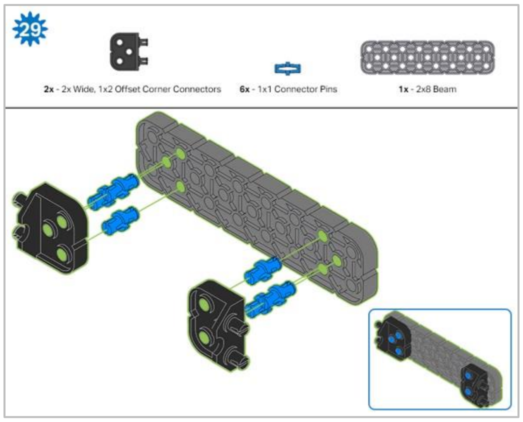

24
1x - 1x1 Connector Pin
1x - 6x Pitch Standoff

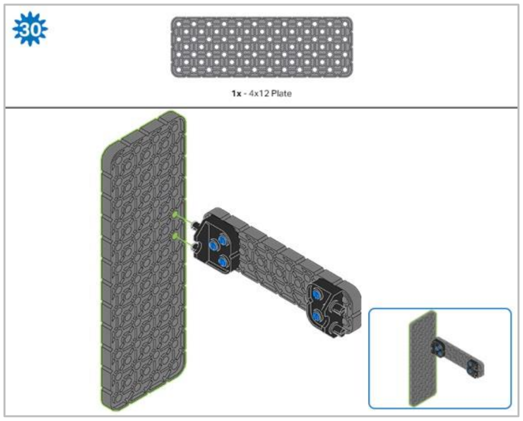

25
1x - Mini Standoff Connector
2x - 2x Pitch Standoffs

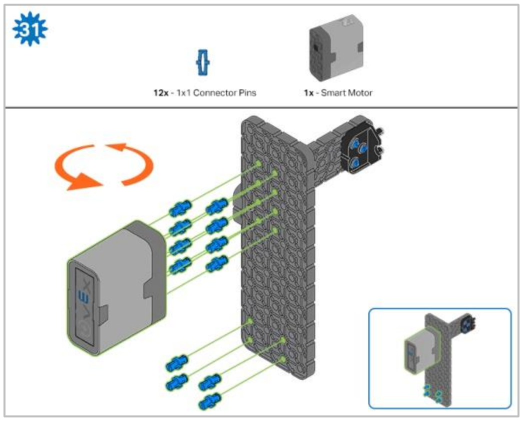

26
2x - 1x1 Connector Pins
1x - 1x8 Beam


27
1x - 2x Wide, 1x2 Corner Connetors
1x - 1x Wide 1x1 Corner Connector
1x - 4x4 Offset Right Angle Beam

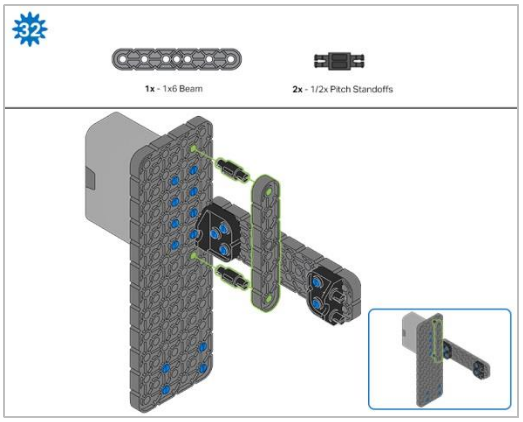

28
1x - 1x1 Connector Pin
1x - 6x Pitch Standoff

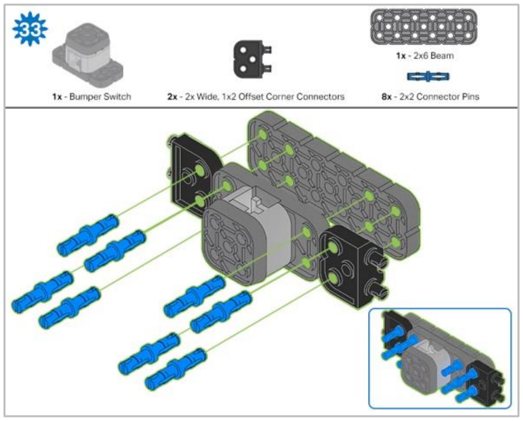




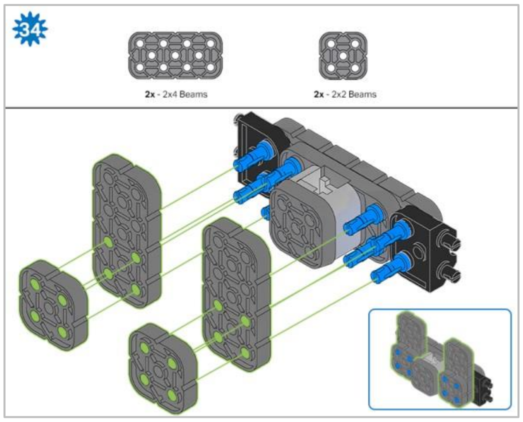


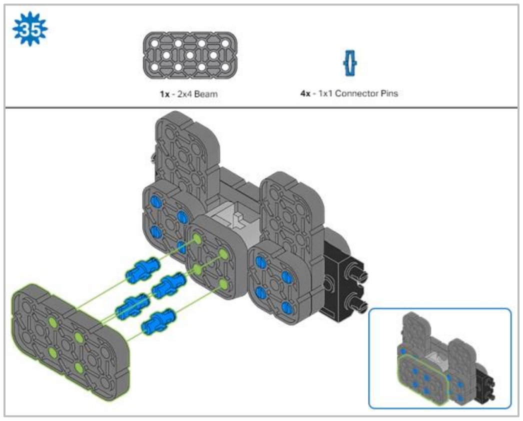
Step 31: The orange arrows mean spin the build around.

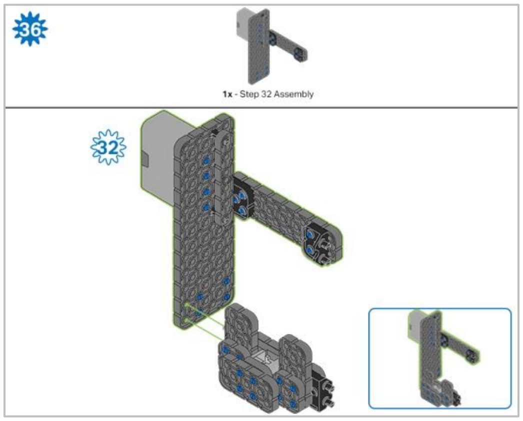




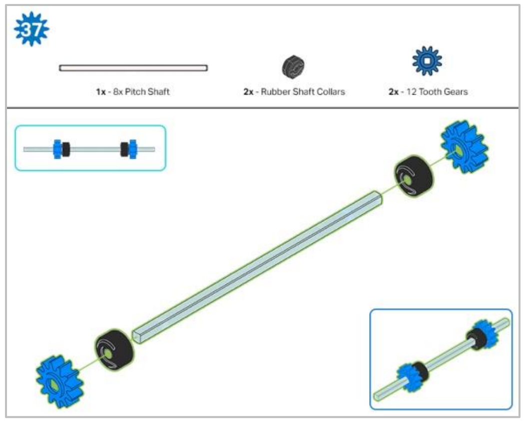
Step 33: Make sure the Bumper Switch is placed in the correct way to allow cable access.

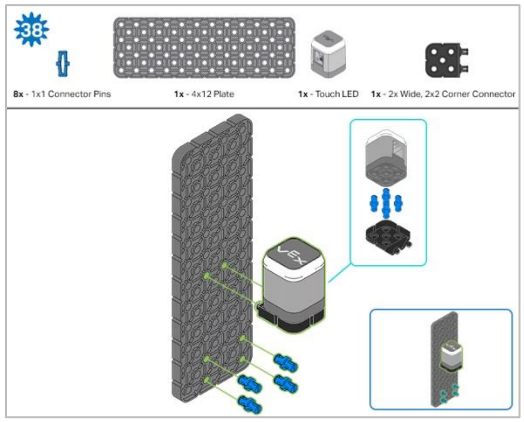




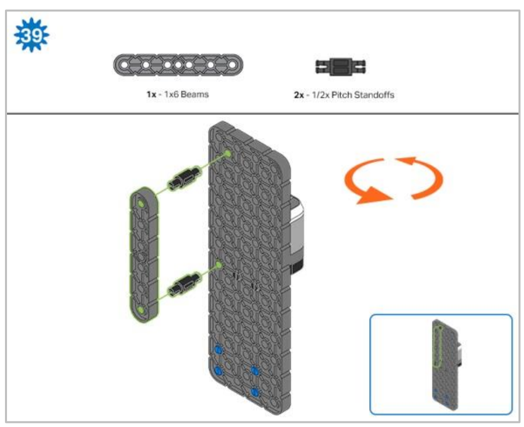


Step 36: Make sure your Smart Motor is are oriented in the correct direction (the hole for the shaft is on the bottom)

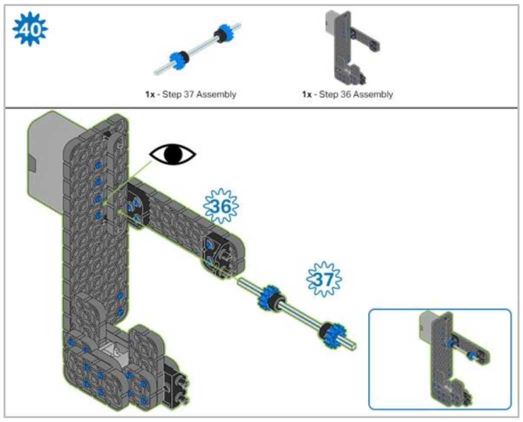




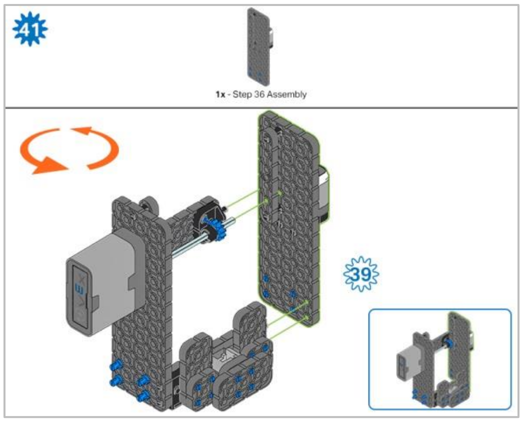
Step 38: Make sure that the Touch LED is placed in the correct way to allow cable access.



Step 39: The orange arrows mean spin the build around.



Step 40: Instead of individual parts, the completed sections of the build needed are shown in the section at the top. When adding the Step 37 Assembly, twist the pitch shaft to check for tension while turning. If it spins freely, it is not properly inserted into the motor.



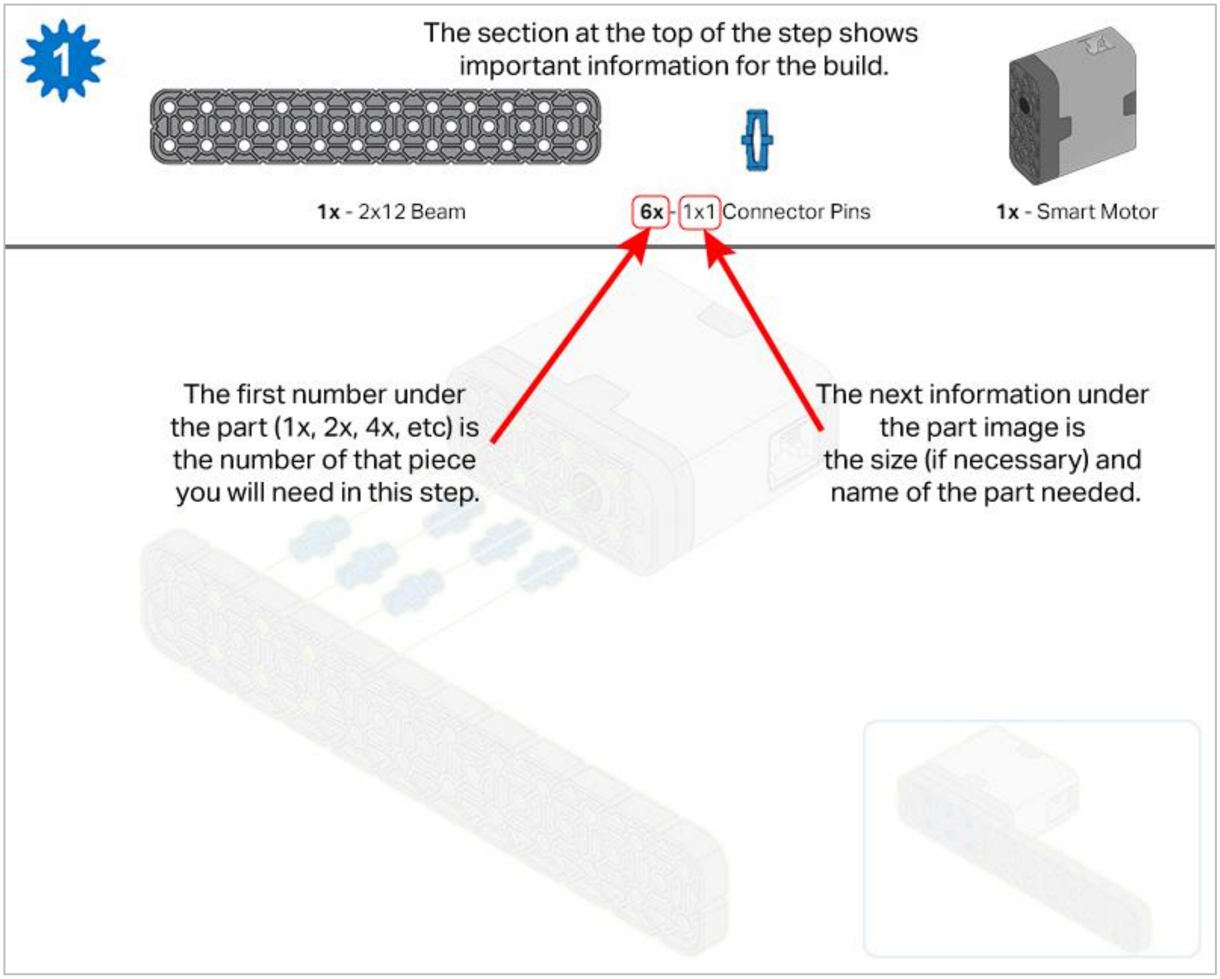
Step 41: Instead of individual parts, the completed sections of the build needed are shown in the section at the top. The orange arrows mean spin the build around.

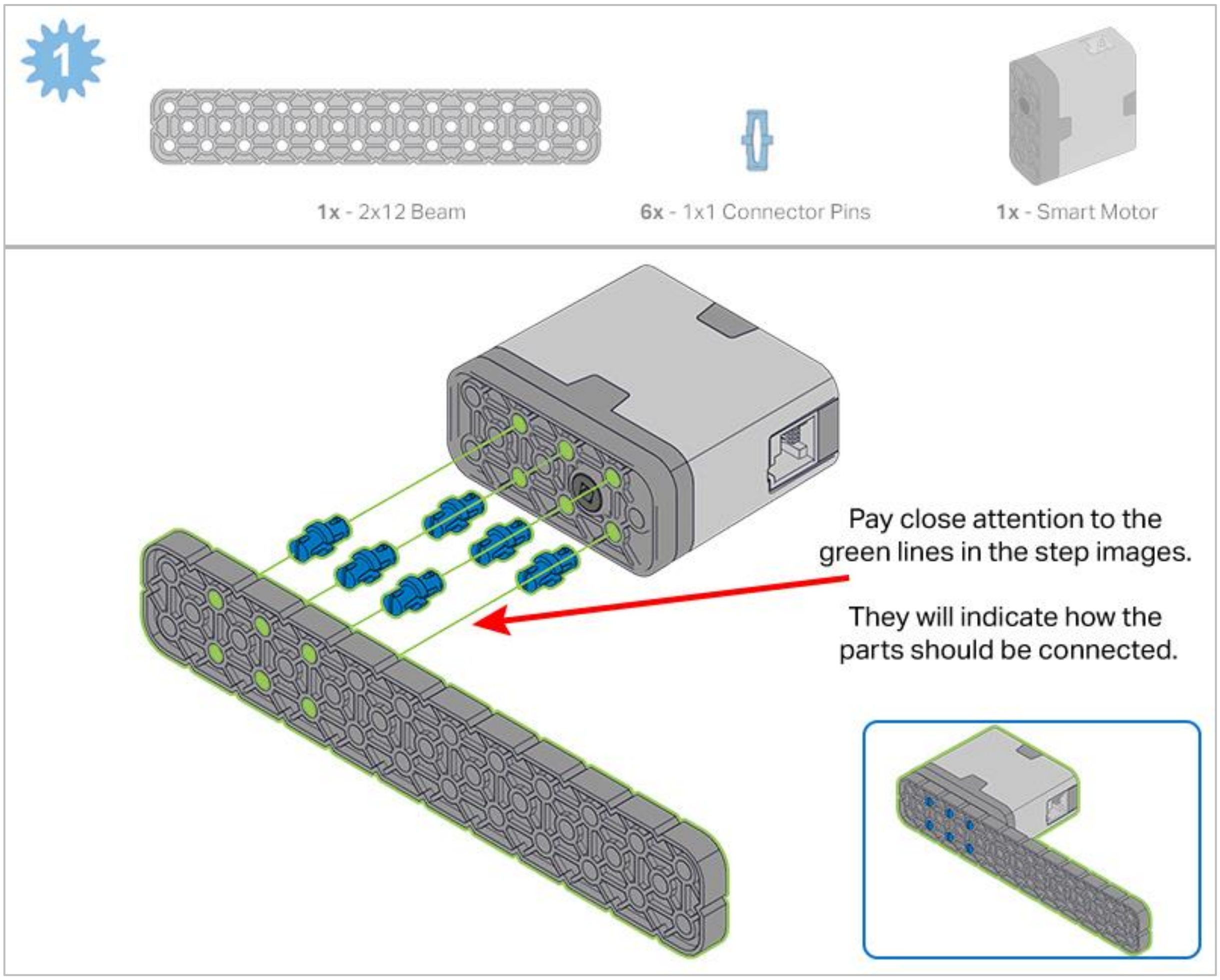
Build Instructions - Arm

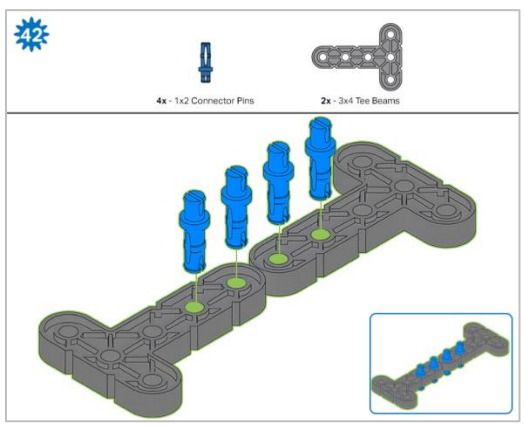
Build Instructions Summary

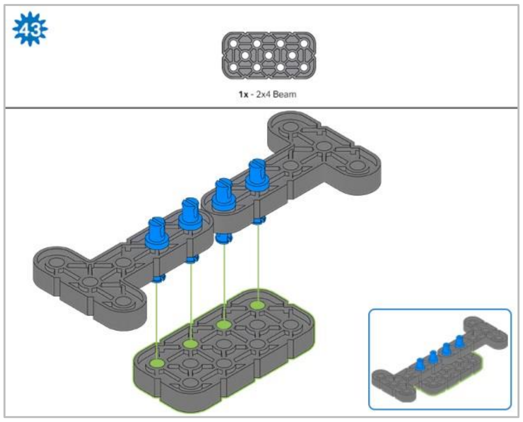
- **Arm Building Instructions (19 steps):**
  - Arm: steps 42 to 60
- **Building Tips for All Steps:**
  - The section at the top of the step shows important information for the build. The first number under the image of the part (1x, 2x, 4x, etc) is the number of that piece you will need in this step. The next information under the part image is the size and description of the part needed.
  - The finished step is illustrated in the box in the lower right corner.

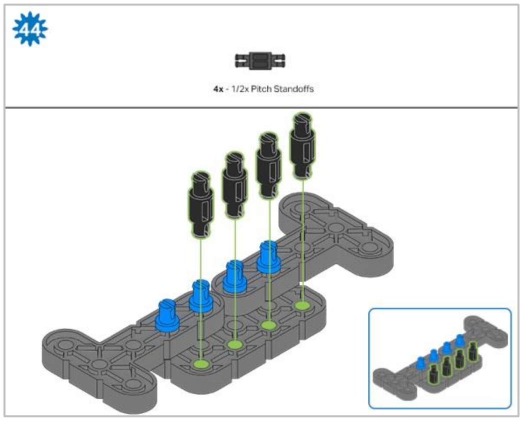
- Play close attention to the green lines in the step images. They will indicate how the parts should be connected.

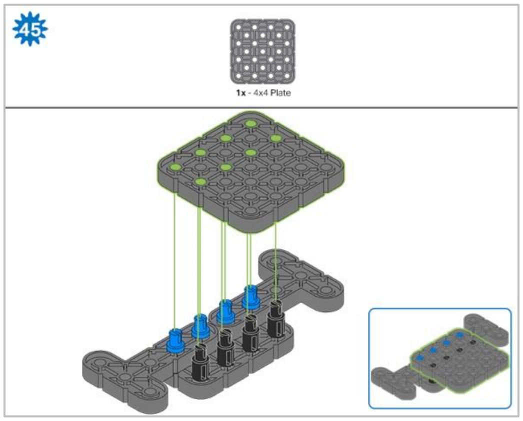


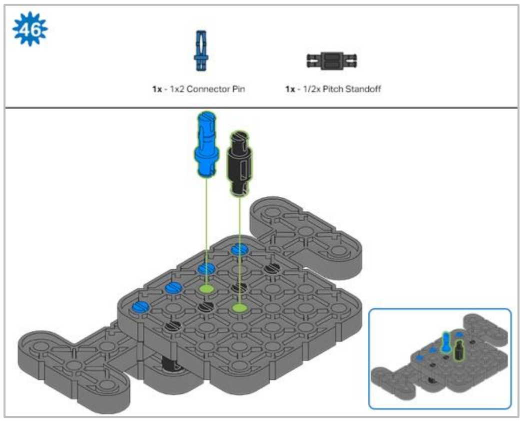

1
1x - 2x12 Beam
6x - 1x1 Connector Pins
1x - Smart Motor
Pay close attention to the green lines in the step images.
They will indicate how the parts should be connected.

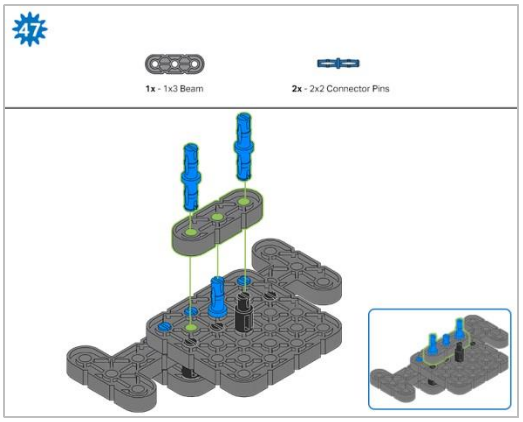

42
4x - 1x2 Connector Pins
2x - 3x4 Tee Beams

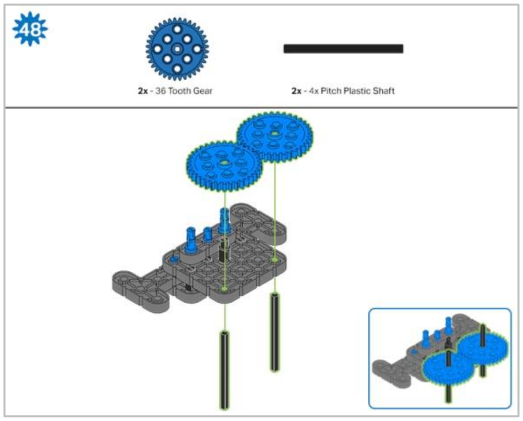

43
1x - 2x4 Beam


44
4x - 1/2x Pitch Standoffs

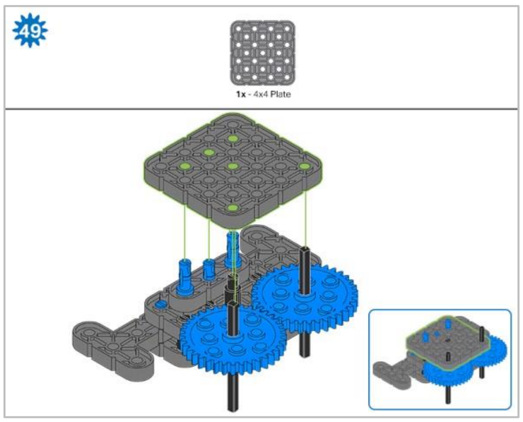

45
1x - 4x4 Plate


46
1x - 1x2 Connector Pin
1x - 1/2x Pitch Standoff

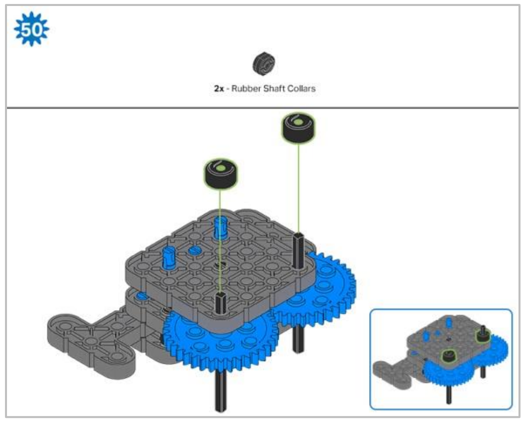


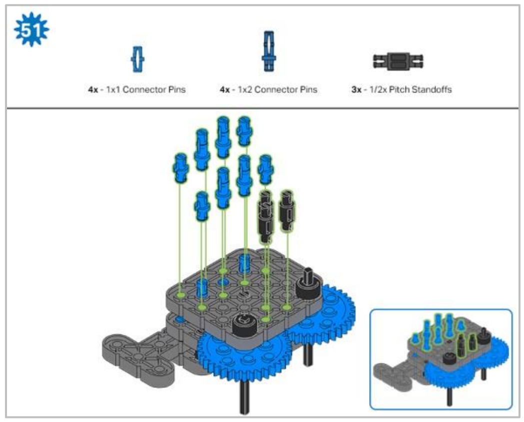


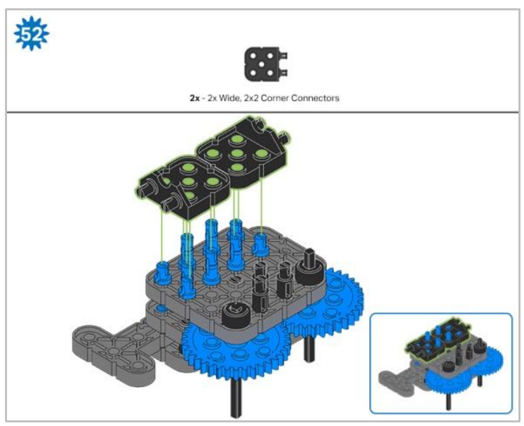
Step 48: Make sure the gears fit together properly before moving on to the next step.

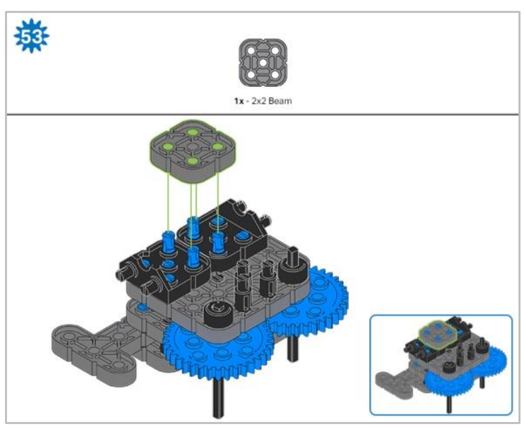


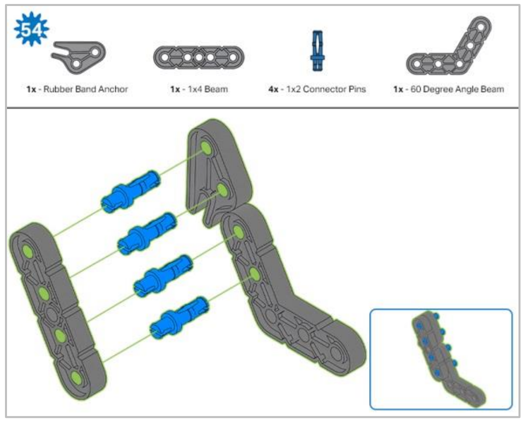
Step 49: Turn one of the black shafts in the center of the gear to make sure they are together and both turn at the same time before adding the 4x4 Plate.

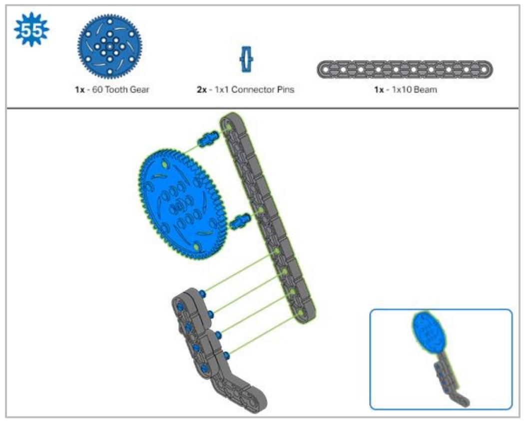

50
2x - Rubber Shaft Collars

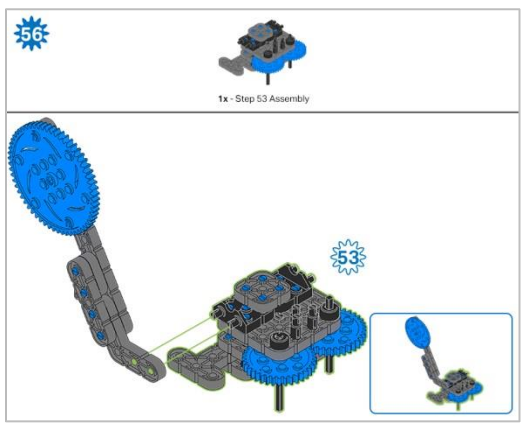

51
4x - 1x1 Connector Pins
4x - 1x2 Connector Pins
3x - 1/2x Pitch Standoffs


52
2x - 2x Wide, 2x2 Corner Connectors

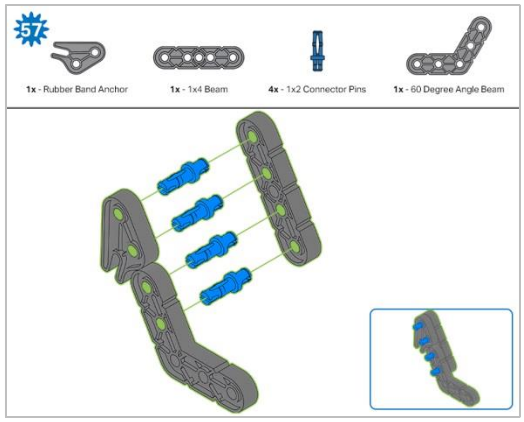

53
1x - 2x2 Beam

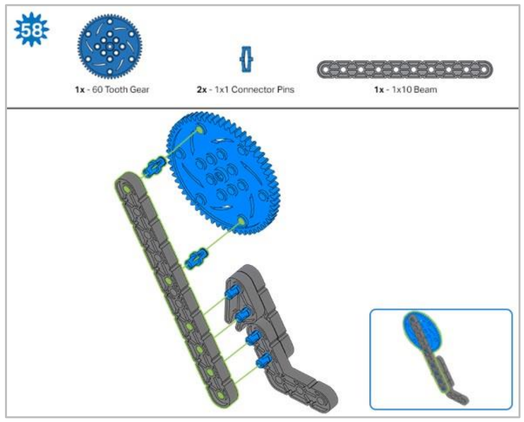


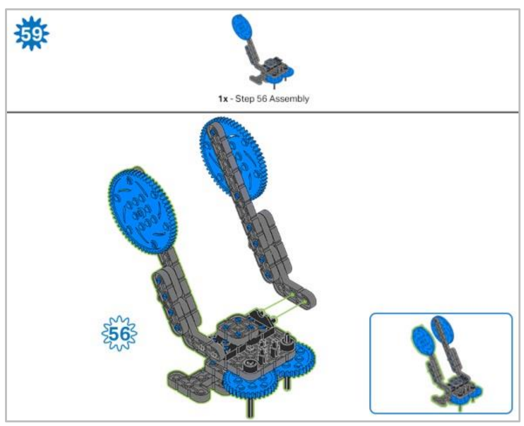




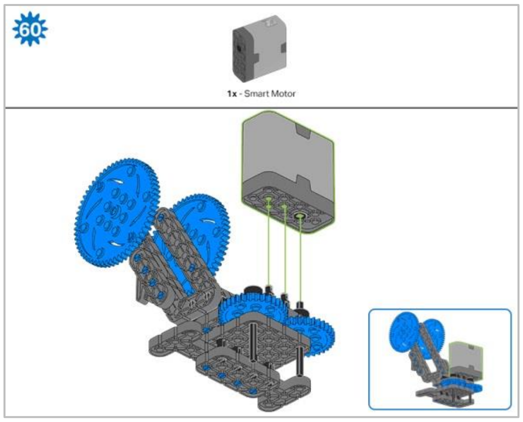
Step 56: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.







Step 59: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.



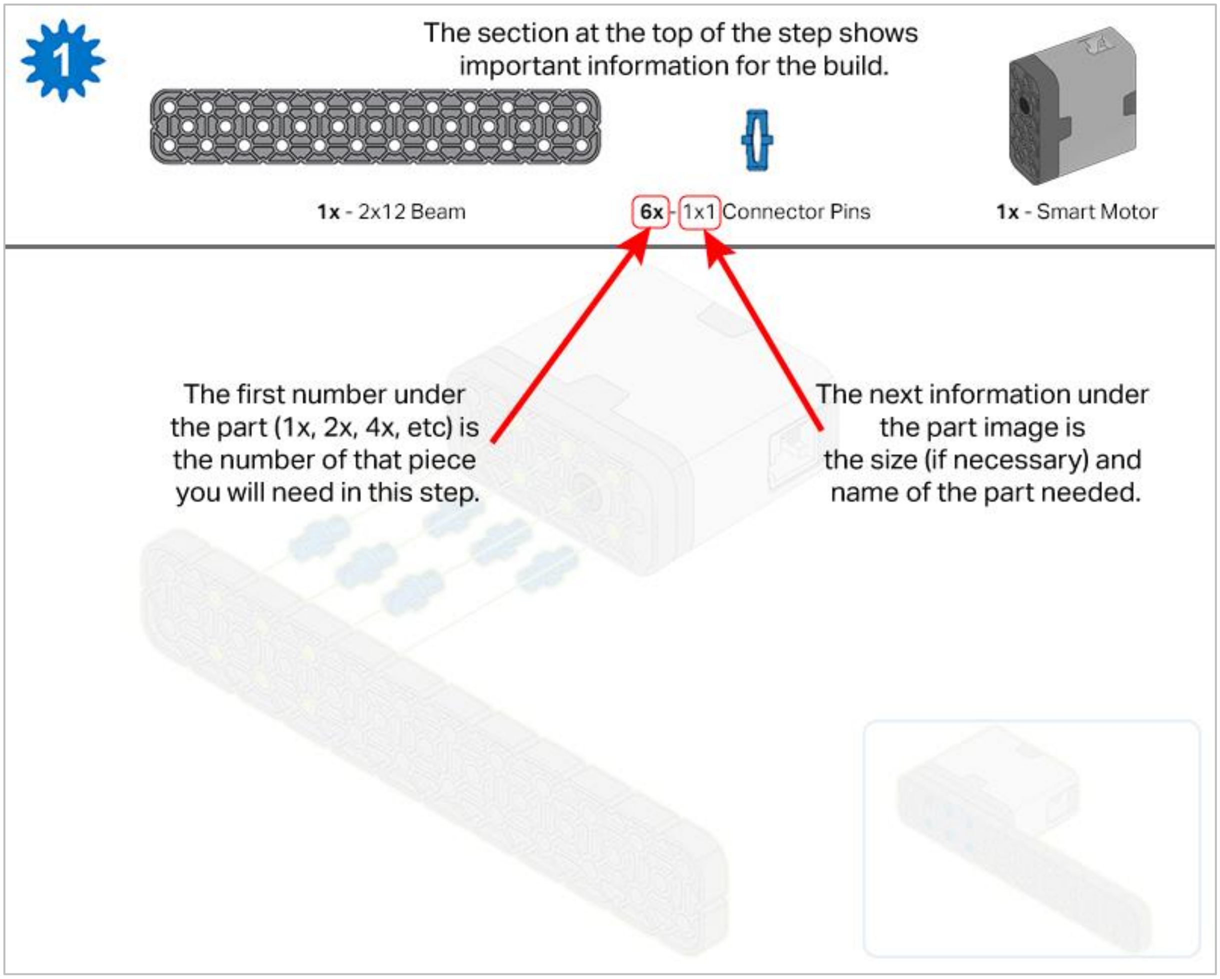
Step 60: Make sure your Smart Motor is are oriented in the correct direction (the hole for the shaft is on the right). After adding motor, turn one of the gears to check for tension while turning. If it spins freely, it is not properly inserted into the motor.

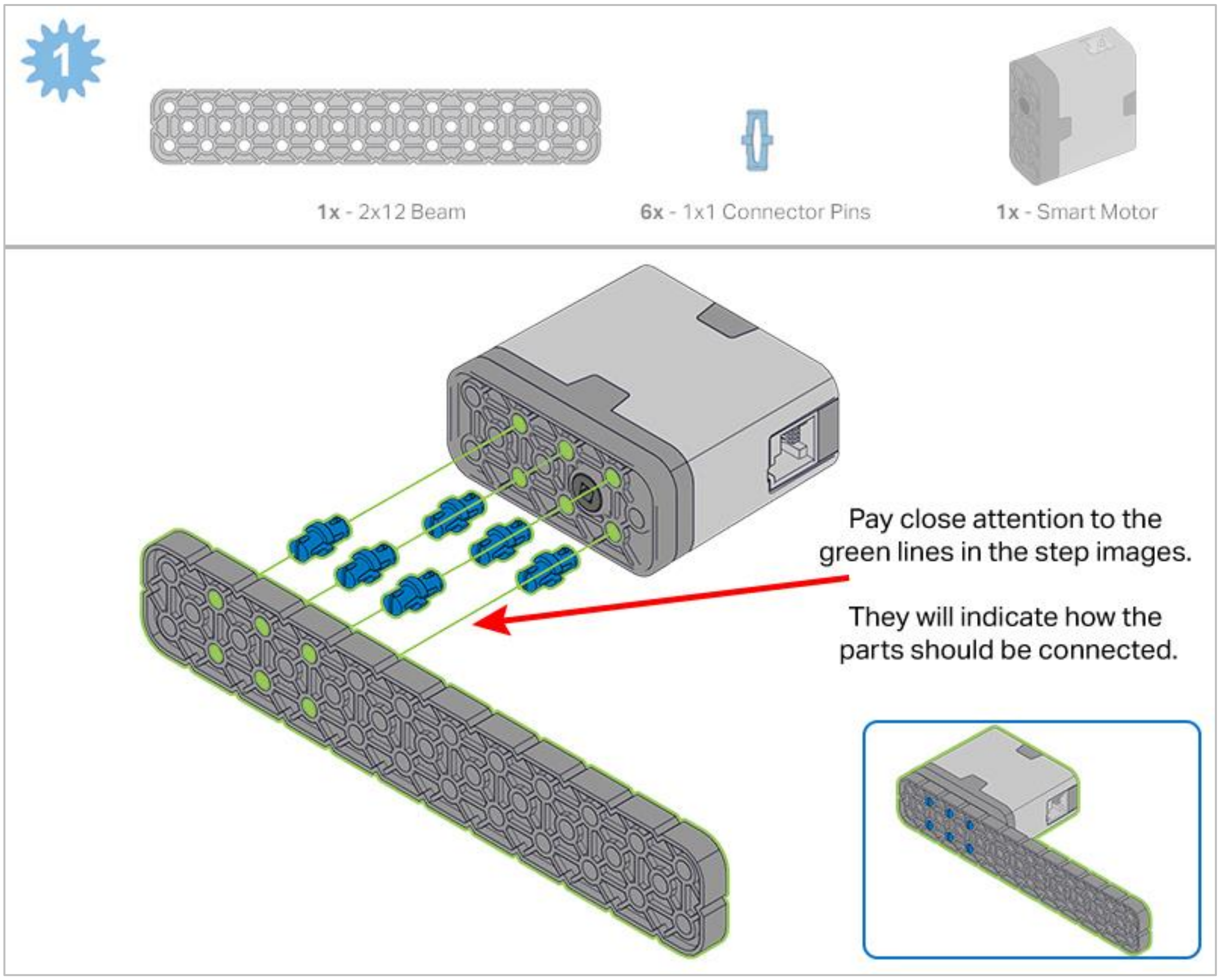
Build Instructions - Claw

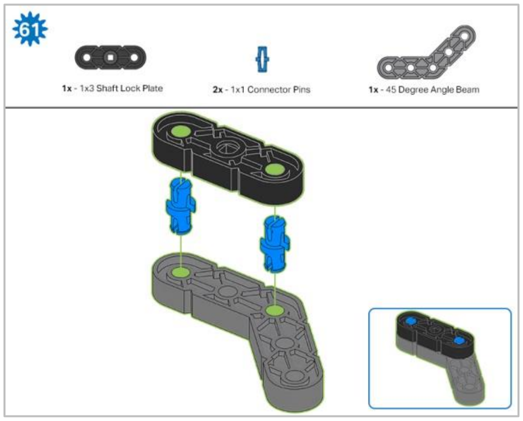
Build Instructions Summary

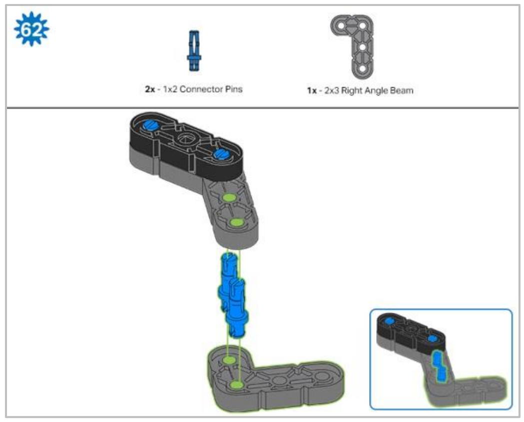
- **Claw Building Instructions (22 steps):**
  - Claw: steps 61 to 82
- **Building Tips for All Steps:**
  - The section at the top of the step shows important information for the build. The first number under the image of the part (1x, 2x, 4x, etc) is the number of that piece you will need in this step. The next information under the part image is the size and description of the part needed.
  - The finished step is illustrated in the box in the lower right corner.

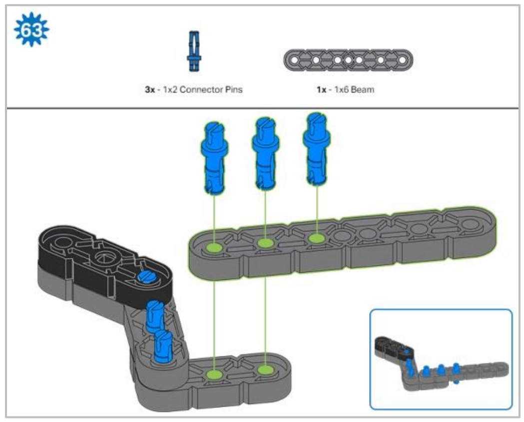
- Play close attention to the green lines in the step images. They will indicate how the parts should be connected.

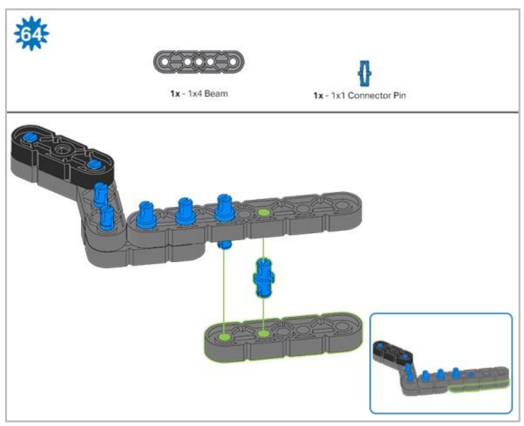


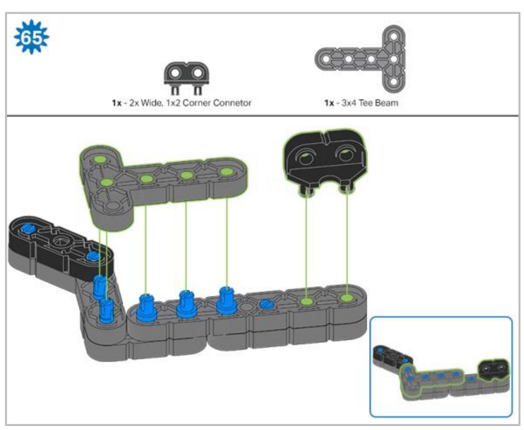

1
1x - 2x12 Beam
6x - 1x1 Connector Pins
1x - Smart Motor
Pay close attention to the green lines in the step images.
They will indicate how the parts should be connected.

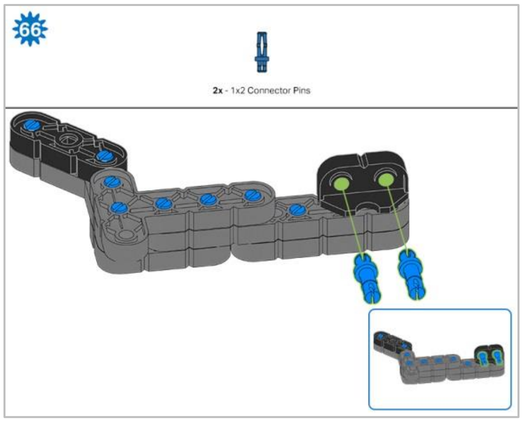

61
1x - 1x3 Shaft Lock Plate
2x - 1x1 Connector Pins
1x - 45 Degree Angle Beam

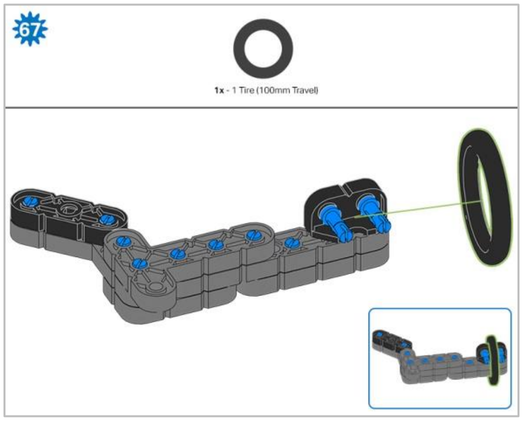

62
2x - 1x2 Connector Pins
1x - 2x3 Right Angle Beam


63
3x - 1x2 Connector Pins
1x - 1x6 Beam

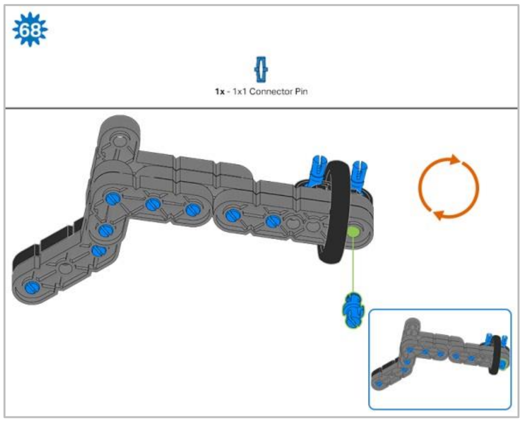

64
1x - 1x4 Beam
1x - 1x1 Connector Pin


65
1x - 2x Wide, 1x2 Corner Connetor
1x - 3x4 Tee Beam

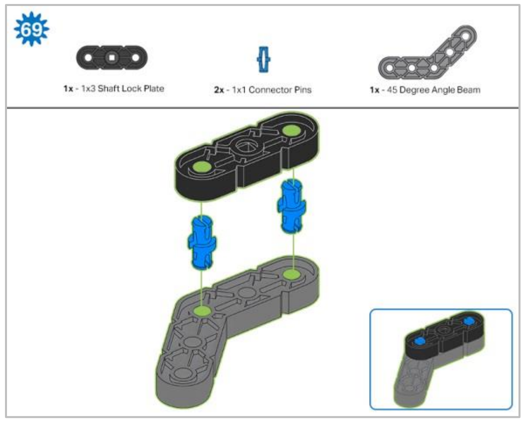


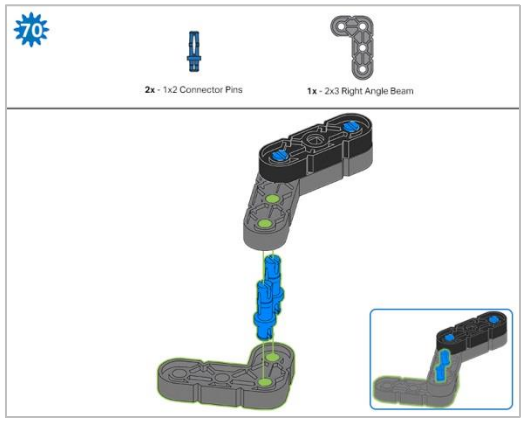


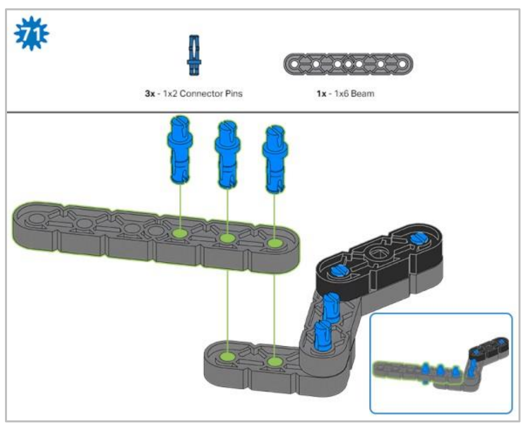
Step 67: Make sure that the 100mm Travel Tire fits snugly in the grove of the 2x Wide, ½ Corner Connector.

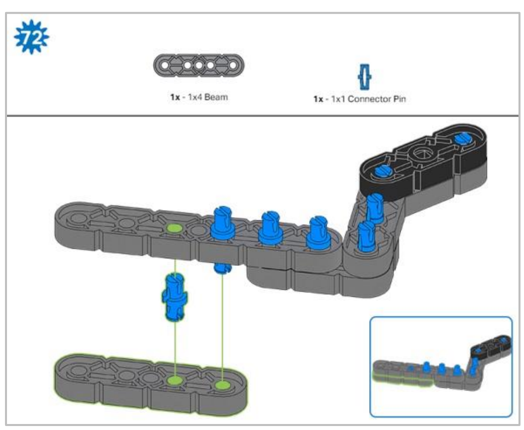


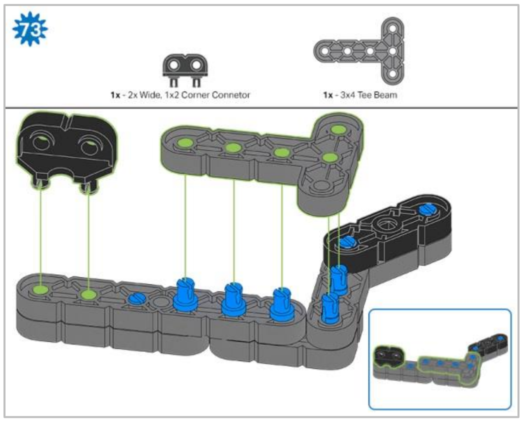
Step 68: The orange arrows mean spin the build around.

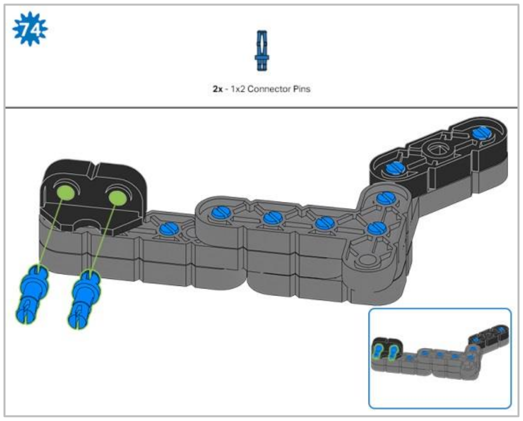

69
1x - 1x3 Shaft Lock Plate
2x - 1x1 Connector Pins
1x - 45 Degree Angle Beam

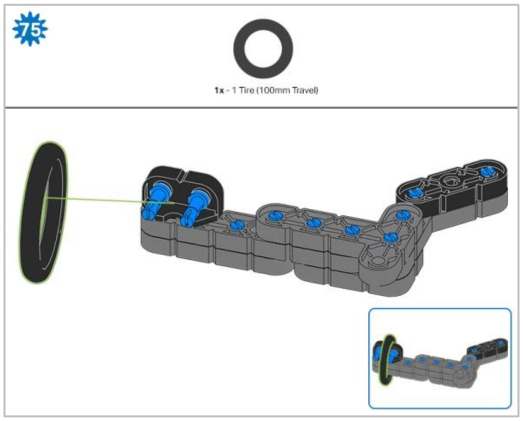

70
2x - 1x2 Connector Pins
1x - 2x3 Right Angle Beam


71
3x - 1x2 Connector Pins
1x - 1x6 Beam

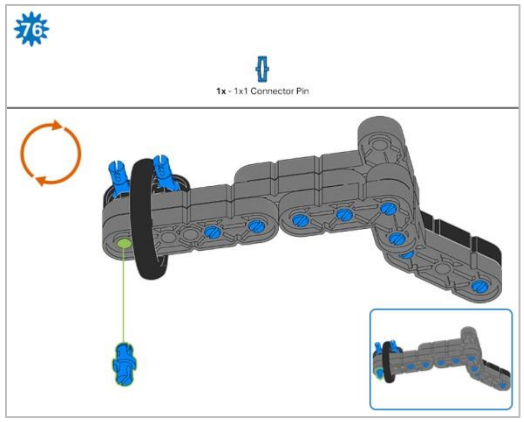

72
1x - 1x4 Beam
1x - 1x1 Connector Pin



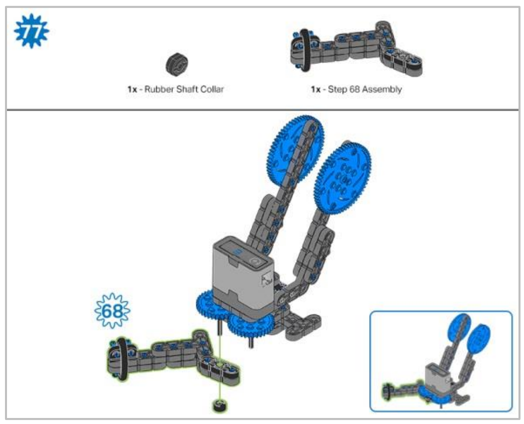




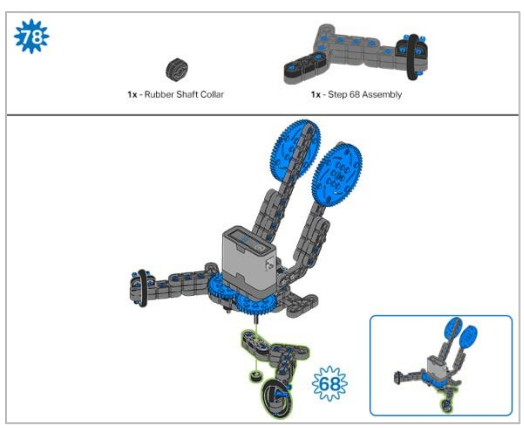
Step 75: Make sure that the 100mm Travel Tire fits snugly in the grove of the 2x Wide, ½ Corner Connector.



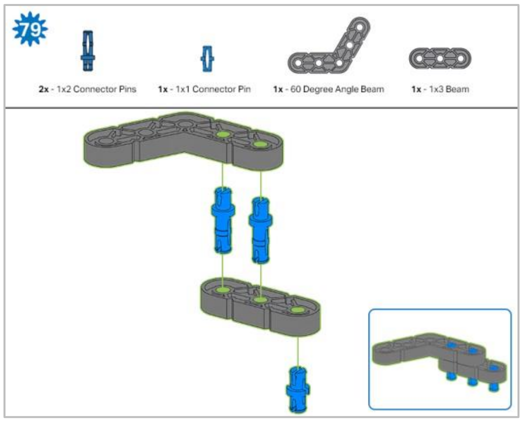
Step 76: The orange arrows mean spin the build around.

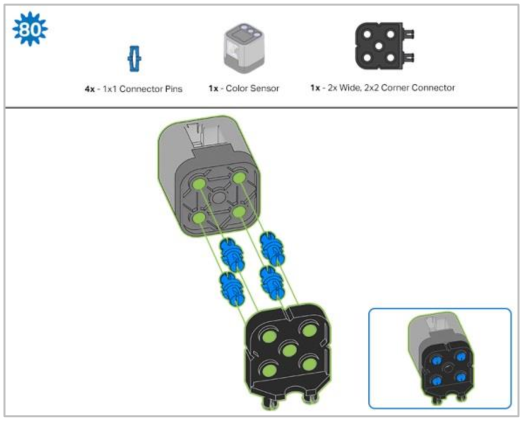


Step 77: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.

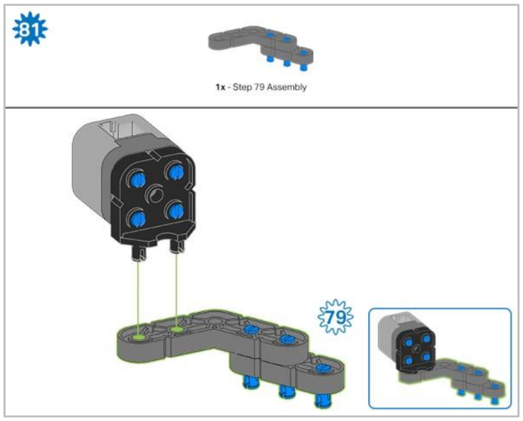


Step 78: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.

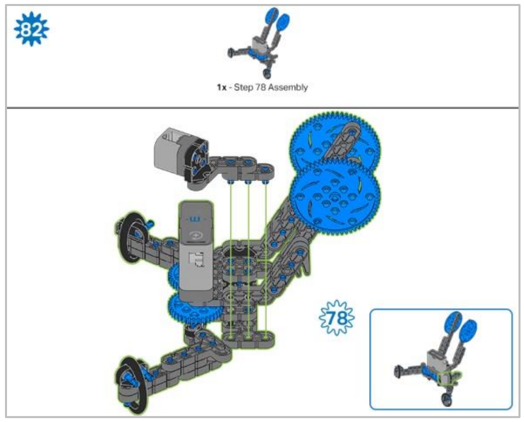




Step 80: The sensor being attached is the Color Sensor. Ensure the sensor is placed in the correct way to allow cable access.



Step 81: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.

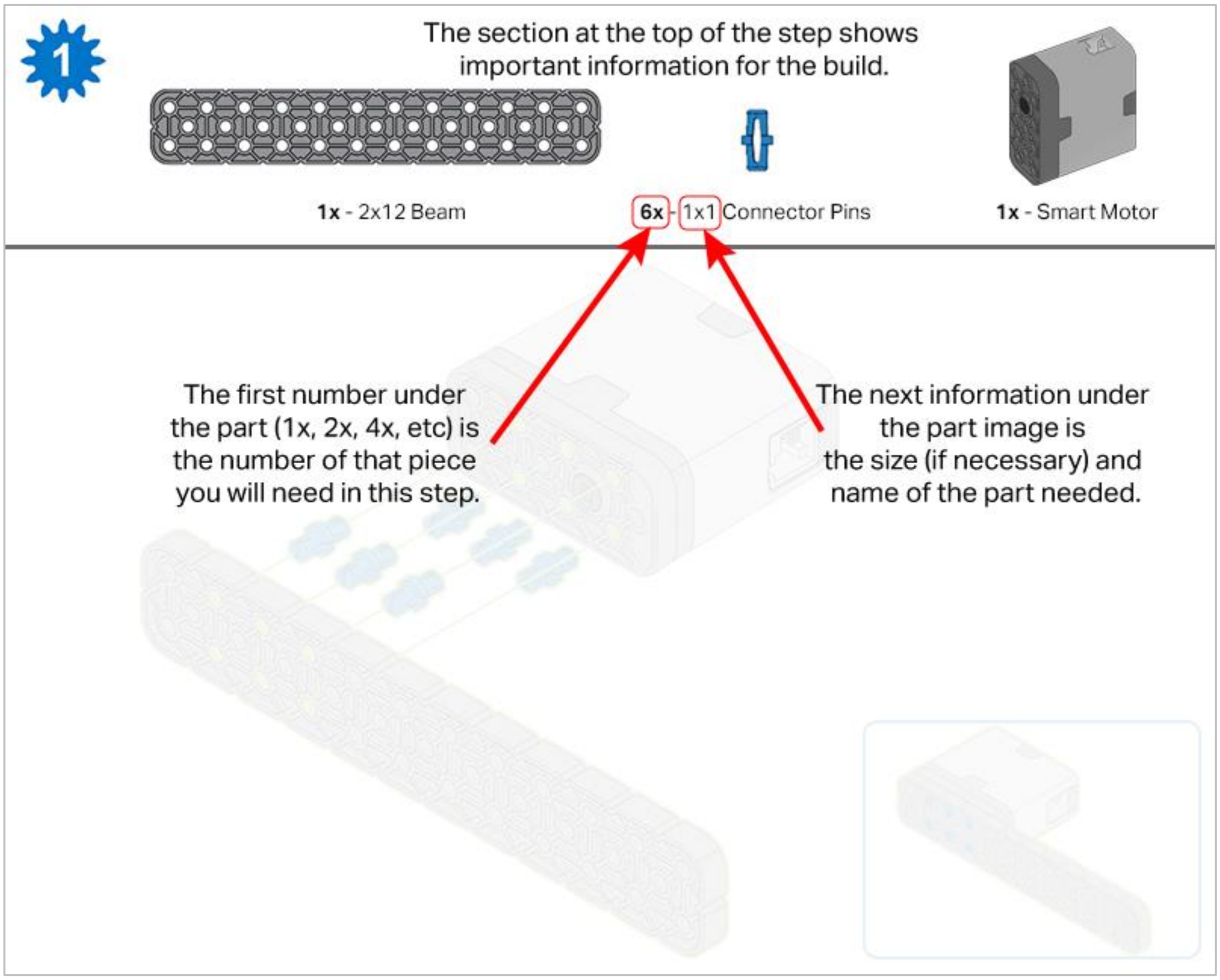

82
1x - Step 78 Assembly
78

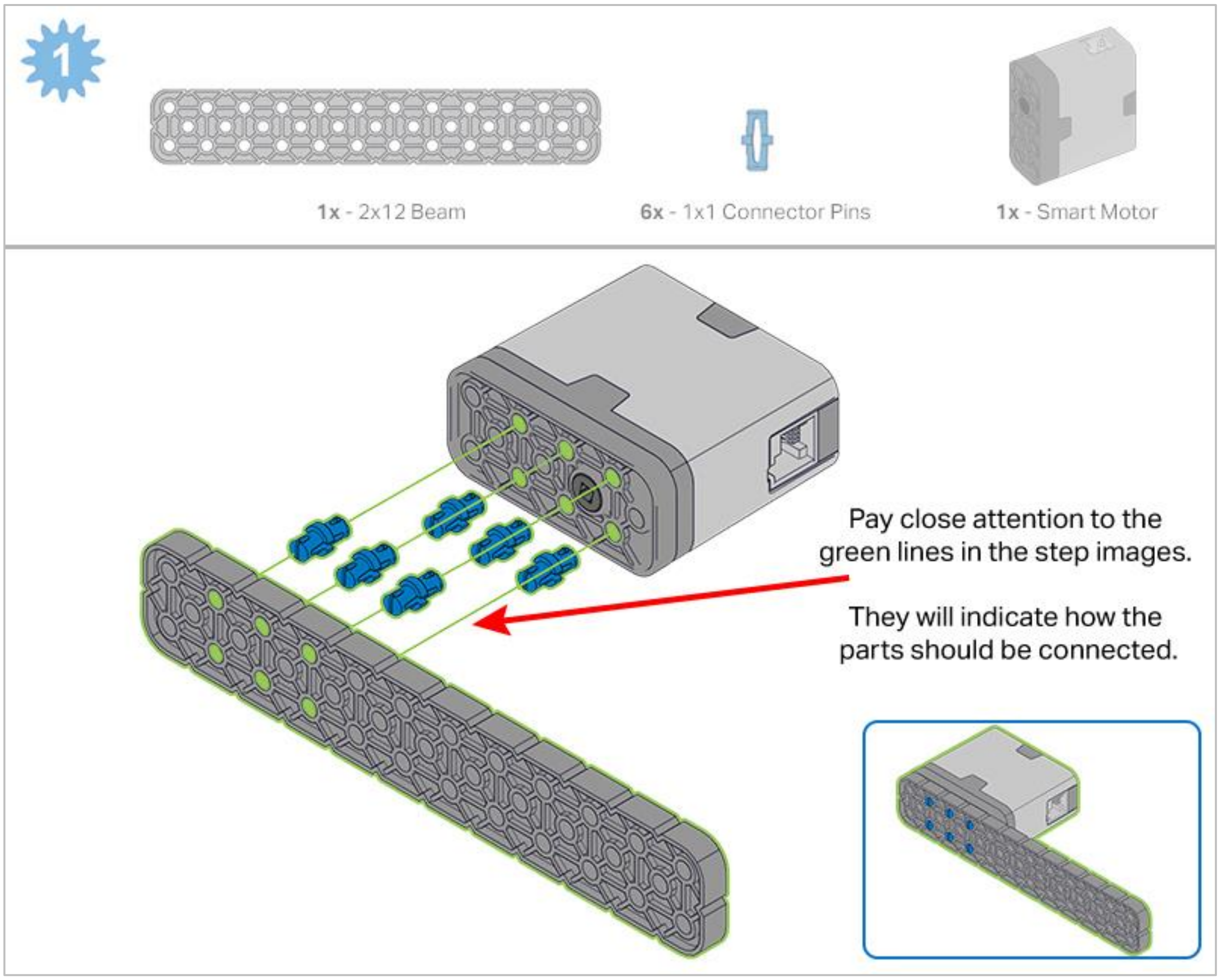
Build Instructions - Assembly and Wiring

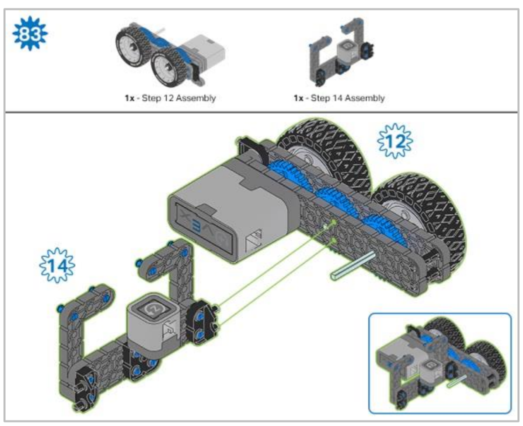
Build Instructions Summary

- **Assembly and Wiring (11 steps):**
  - Final Assembly: steps 83 to 93
  - The group is also responsible for making sure the sensors and motors are attached to the correct ports using the designated Smart Cables.
    - Port 1: Left Wheel
    - Port 2: Touch LED
    - Port 3: Color Sensor
    - Port 4: Gyro Sensor
    - Port 6: Right Wheel
    - Port 7: Distance Sensor
    - Port 8: Bumper Switch
    - Port 10: Arm Motor
    - Port 11: Claw Motor
- **Building Tips for All Steps:**
  - The section at the top of the step shows important information for the build. The first number under the image of the part (1x, 2x, 4x, etc) is the number of that piece you will need in this step. The next information under the part image is the size and description of the part needed.
  - The finished step is illustrated in the box in the lower right corner.
  - Play close attention to the green lines in the step images. They will indicate how the parts should be connected.
- **Building Tips for Steps 87-89:**

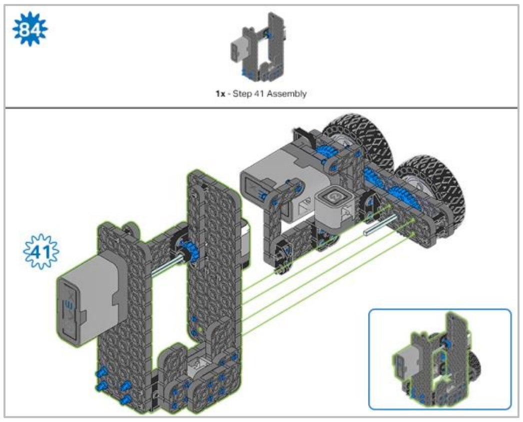
- The solid green numbers represent the numbered port the cable will be connected into. The outlined green number indicates the sensor that cable will connect into. Use the indicated Smart Cable for each sensor or motor. When attaching the Smart Cables, make sure they are tucked away so as to not block the Smart Sensors or interfere with the Clawbot's movement.



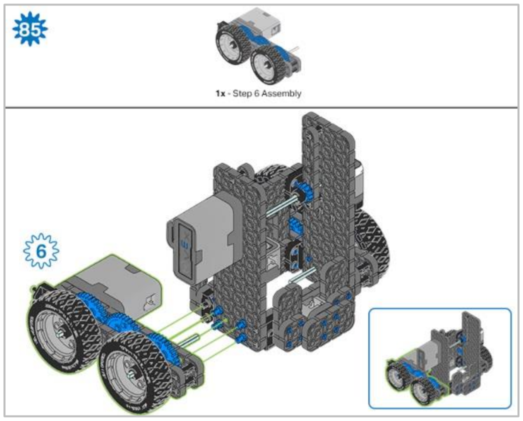




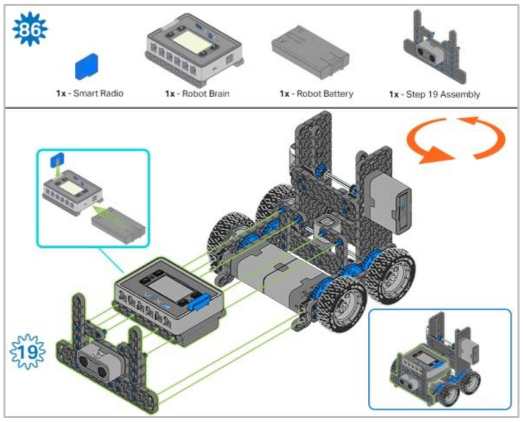
Step 83: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.



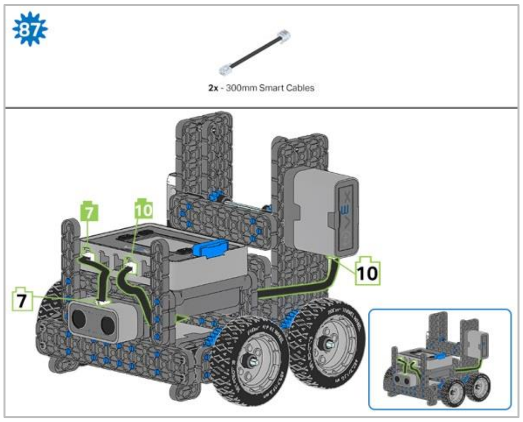
Step 84: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.



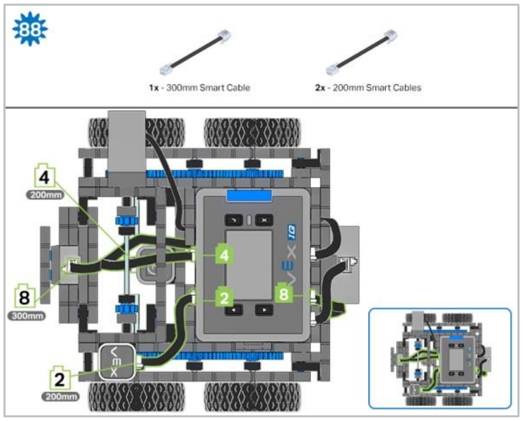
Step 85: Instead of individual parts, the completed sections of the build needed are shown in the section at the top.

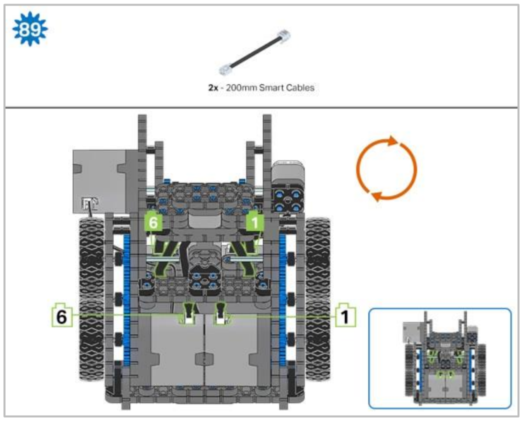


Step 86: Make sure the Smart Radio is pushed in securely. Make sure the Robot Battery is oriented the correct way before inserting. The orange arrows mean spin the build around.

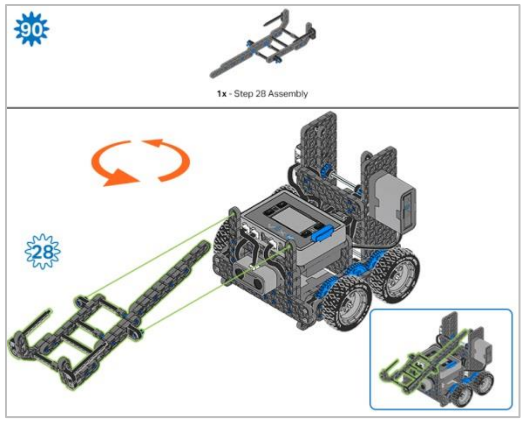


Step 87: The Smart cable for the Arm Motor can be tucked under the Brain and plugged into the correct port (port 10).

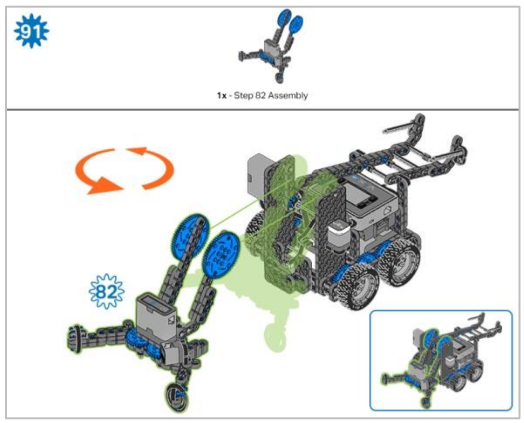




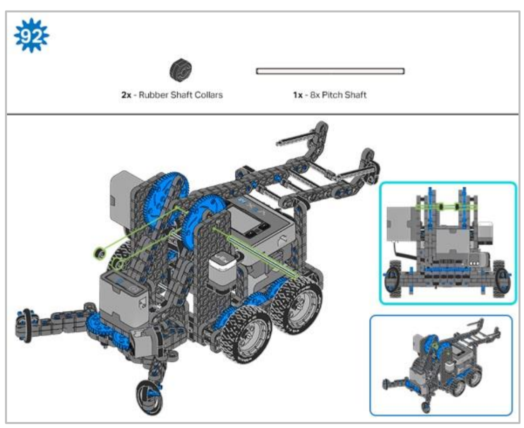
Step 89: The orange arrows mean spin the build around.



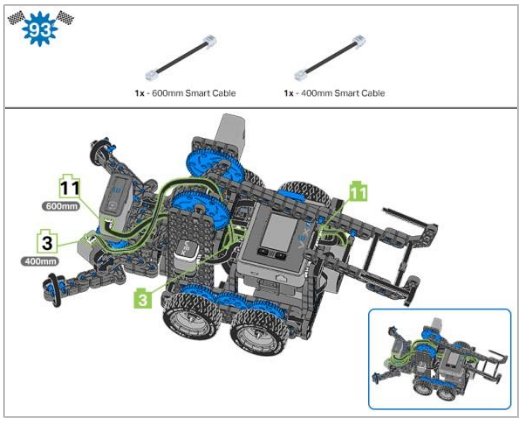
Step 90: Instead of individual parts, the completed sections of the build needed are shown in the section at the top. The orange arrows mean spin the build around.



Step 91: Instead of individual parts, the completed sections of the build needed are shown in the section at the top. The orange arrows mean spin the build around.



Step 92: When adding the 8x Pitch Shaft, twist the pitch shaft to check for tension while turning. If it spins freely, it is not properly inserted into the gears.

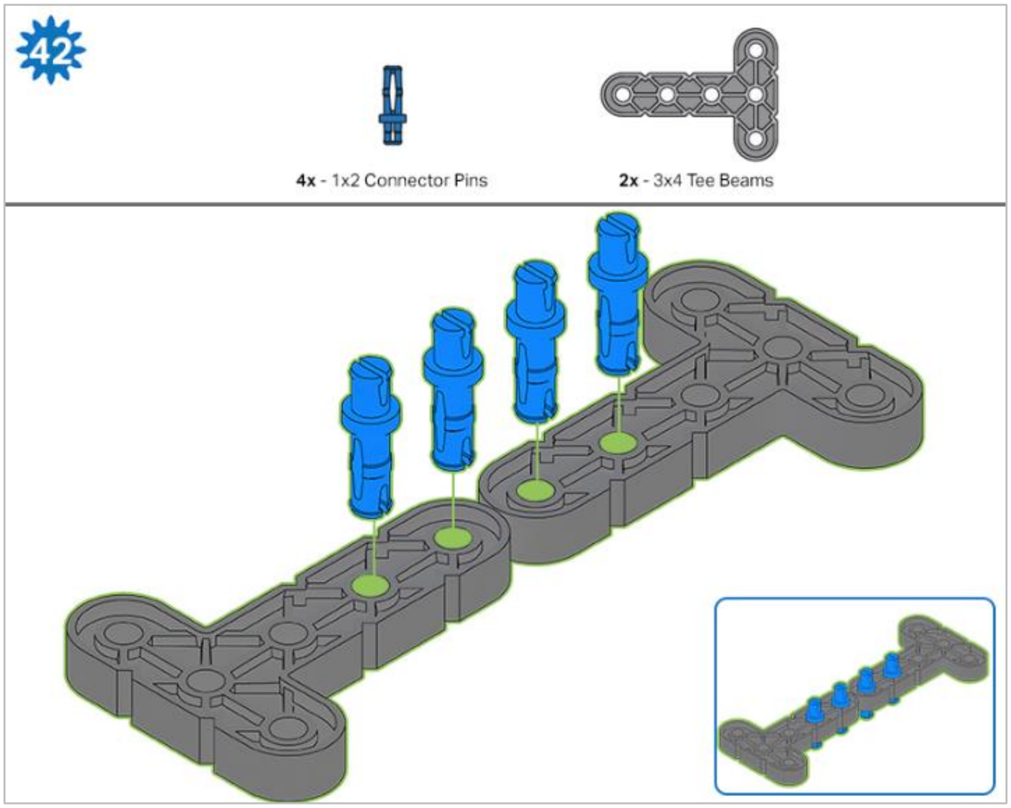


Step 93: The solid green numbers represent the numbered port the cable will be connected into. The outlined green number indicates the sensor that cable will connect into. Use the indicated Smart Cable for each sensor. When attaching the Smart Cables, make sure they are tucked away so as to not block the Smart Sensors or interfere with the Clawbot's movement.

Exploration

Now that you've finished the build, test what it does. Explore your build and then answer these questions in your engineering notebook.

How would the range of the claw change if the two 3x4 Tee Beams used in Step 42 of building the arm were replaced with one 1x4 Beam?



- For help with this question, observe how far the claw can open with the two Tee Beams.
    - Do you think the claw will open more or less with the 1x4 Beam?
    - Be sure to justify your answer with data from your observation.

What You'll Need to Know

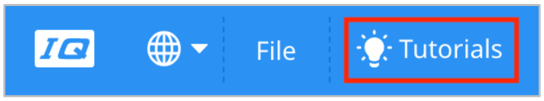
Learning Resources

In order to successfully complete this STEM Lab, there are some things you'll need to know before getting started. You can use the tutorial videos, the example projects in VEXcode IQ Blocks, or the links to the other STEM Labs to learn how to drive and turn the robot before proceeding.

If you have not programmed your robot to drive or turn before, make sure to complete each with your robot before moving on!

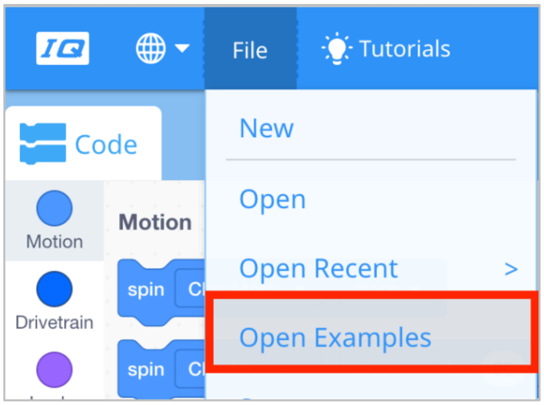
Tutorial Videos

Tutorial Videos can be found in VEXcode IQ Blocks or below:

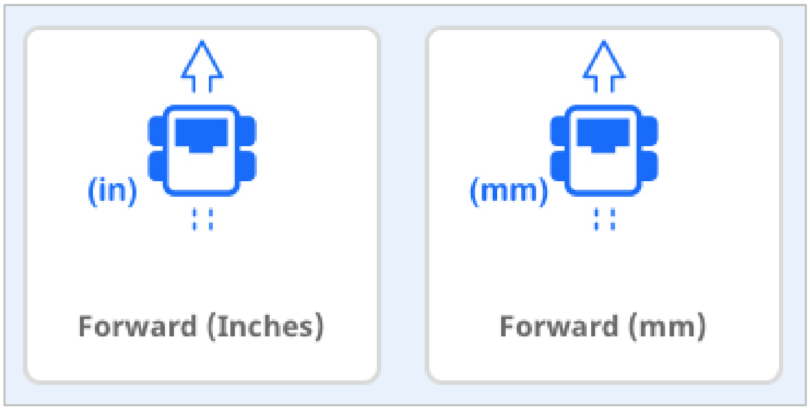


Example Projects

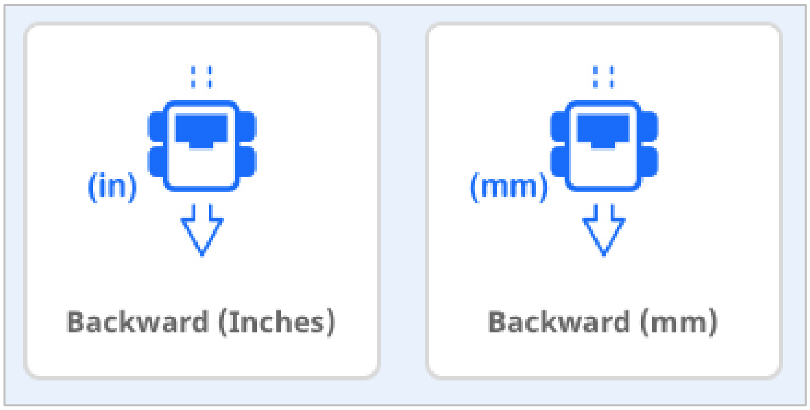
Example Projects found in VEXcode IQ Blocks:



- Forward (Inches or mm)



- Backward (Inches or mm)



STEM Labs

Basic Movement Programming Concepts in STEM Labs

- Drive Forward and Reverse STEM Lab
    - Drive Forward and Reverse Exploration - Part 1
    - Drive Forward and Reverse Exploration - Part 2
- Turning STEM Lab
    - Turning Left and Right
    - Turning Right and Left Exploration - Part 1
    - Turning Right and Left Exploration - Part 2

SPARK PLAY

Test your build, observe how it functions,
and fuel your logic and reasoning skills
through imaginative, creative play.

Range of Motion

Let's explore range of motion!

This exploration will allow you to see the minimum and maximum degrees that the arm and claw can extend.

- Make sure you have the hardware required and your engineering notebook.

Hardware/Software Required:

Quantity	**Hardware/Other Items**
1	VEX IQ Super Kit
1	VEXcode IQ Blocks (latest version, Windows, MacOS, Chromebook, iPad)
1	Engineering Notebook

1. Preparing for the Exploration

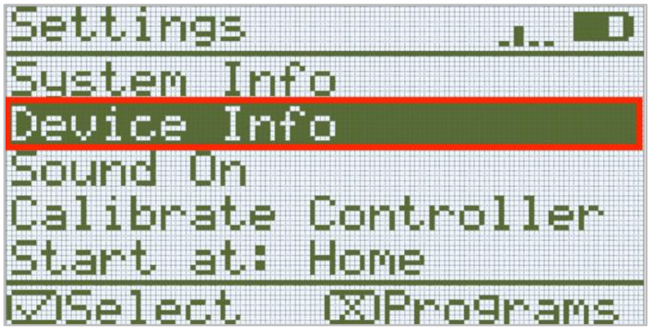
Before you begin the activity, do you have each of these items ready? The Builder should check each of the following:

- Are all of the motors and sensors plugged into the correct ports?
- Are the smart cables fully inserted into all of the motors and sensors?
- Is the Brain turned on?
- Is the battery charged?
- Is the Radio inserted into the Robot Brain?

2. The Device Menu

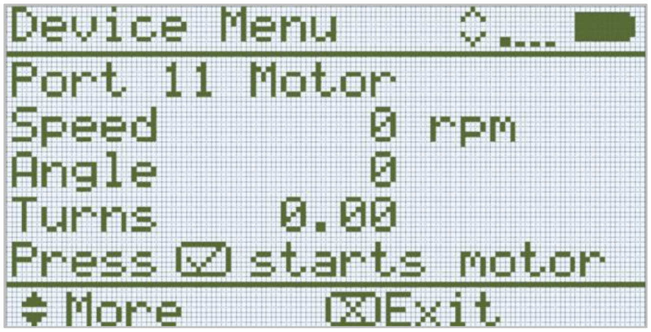
Begin by turning on the Robot Brain and selecting the X Button to navigate to the Settings menu.

Gently open the Clawbot's Claw fully by using your fingers.
Once the Settings menu is open, use the Up and Down buttons on the Brain to select Device Info to open the Device Menu.



The Device Menu screen displays information about the device that is connected to that port. There are 12 ports on the IQ Brain.

Use the arrows to go to the Port 11 Motor, which is the Claw Motor.



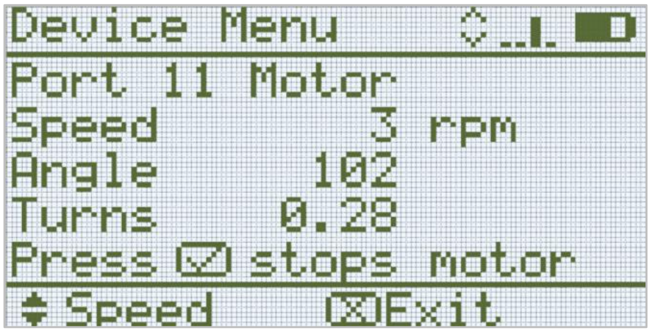
- Port 11 Motor: The Claw Motor.
- Speed: Displays how fast (in revolutions per minute) the motor is spinning.
- Angle: Displays the current position of the motor in degrees.
- Turns: Displays how many turns the motor has rotated.
- Press the Check Button to start and stop the motor. The Claw can also be open and closed manually .

3. Exploring the Claw and Arm's Motion

- If you opened your claw fully before opening the Device Menu, then the Claw Motor considered its fully opened position to be 0 degrees - as it displayed in the Device Menu.

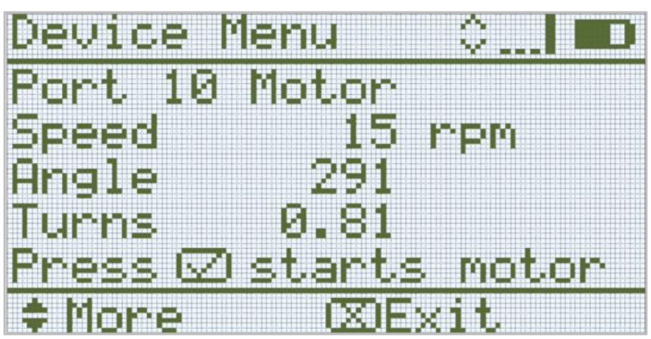
  In your engineering notebook, predict what the values of the Port 11 Motor (Claw Motor) will be when you close the claw by gently pushing the sides together. What will the Angle value in degrees be when the claw is closed?

  - *Hint:* The value reported will not be the same one shown in the image below.
- Test your prediction by gently pushing the claw closed. What angle is now shown in the Device Menu for the Claw Motor?
- Continue using your hands to gently open and close the claw so that you can see the angle changing.



- What do you notice about the range of the angle in degrees for the Claw Motor? Do the Angle values continue to increase, or do they have limits?
- Write down the range of the Angle value for the Claw Motor: the Angle value when fully opened to the Angle value when fully closed.
- Are the Angle values for when the claw is open always the same? Are the Angle values for when the claw is closed always the same? Why do you think that is?

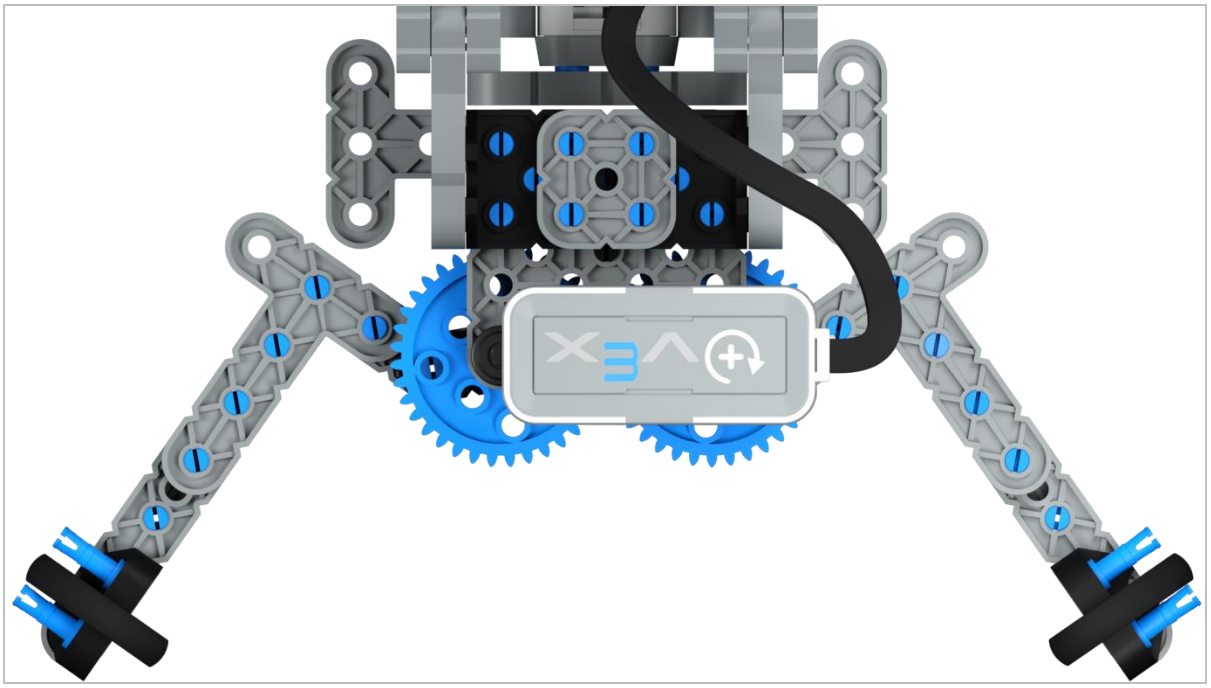
Now, let's explore how the arm moves. The Arm Motor is connected to Port 10.



- Use the Down button to navigate to Port 10 in the Device Menu.
- Once Port 10 is selected, manually move the arm up until it is just above the motor and then lower it down. What do you notice about the range of the angle in degrees? Is it the same as the claw?

- At what angle is the Arm Motor when completely down? At what angle is the Arm Motor when the arm is all of the way at the back of the Clawbot? Write the values in your engineering notebook.

4. Programming with a Range of Motion



IQ Claw opened

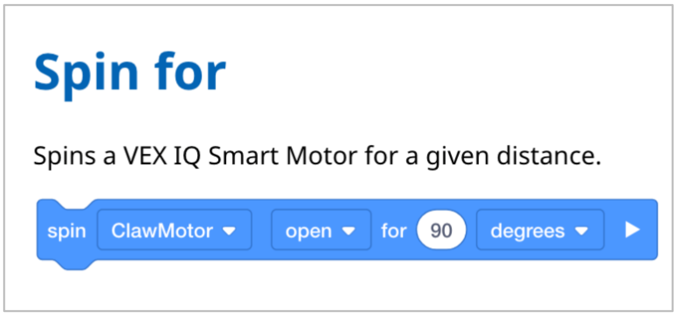
Subsystems such as claws or arms usually have a limited range of motion, which prevent them from spinning continuously. Claws can only open or close so much before reaching a mechanical limit. Likewise, the range of motion of an arm is often limited by the ground or the body of the robot itself. When working with subsystems with a limited range of motion, it is very important to stay within that range, regardless of whether you are remote controlling the robot or programming it to move autonomously. Continuing to provide power to the motors once a subsystem has reached a limit will cause unnecessary stress on the motor and any connected components.

Before learning how to adjust for the claw's and arm's limited ranges of motion, let's look at the blocks used to program the claw and arm.

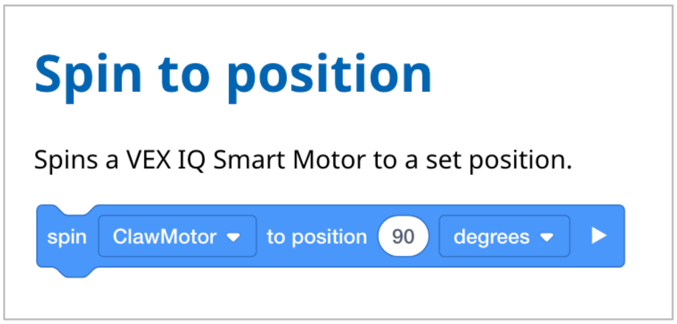
There are two blocks in VEXcode IQ Blocks that can be used to raise and lower the arm and open and close the claw to a specific position.

The *spin for* block and the *spin to position* block.

- The *spin for* block spins a motor in a selected direction for a selected distance from where it is currently located.



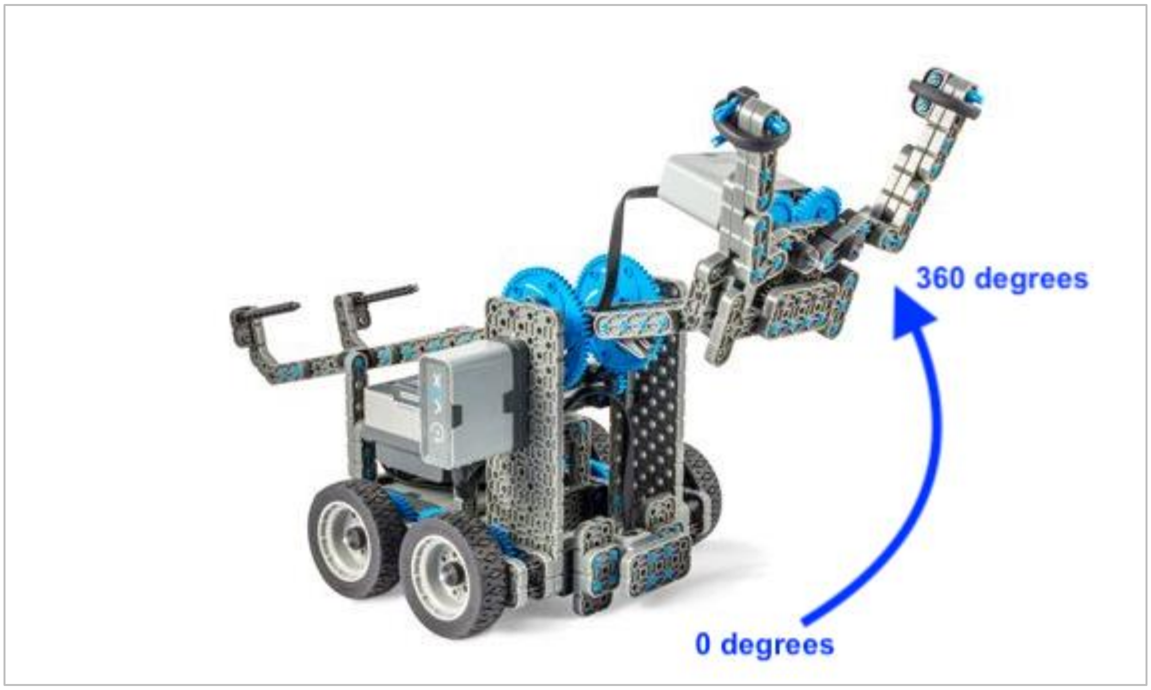
- The *spin to position* block spins a motor to a selected position based on the current position of the motor. The *spin to position* block determines the best direction to rotate in order to get to the position.



When would these blocks be used? Imagine that you program your arm to raise and lower, but when it lowers, it doesn't fully lower back down to its starting position of zero degrees. Instead, it lowers back to 15 degrees. If you then use the *spin for* block to raise it 90 degrees - the arm will raise 90 degrees from where it currently is and really be raised up to 105 degrees.

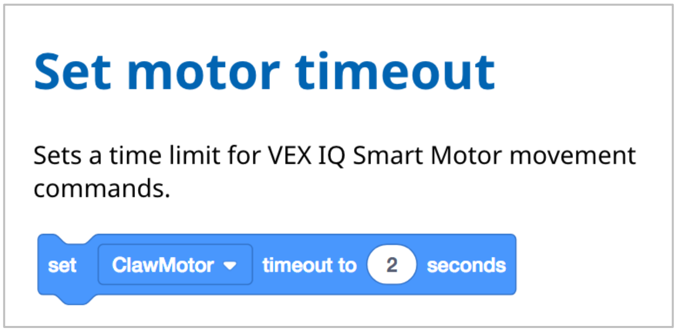
However, in the same situation, if the arm is at 15 degrees and the *spin to position* block is used to raise it to 90 degrees, the arm will raise 75 degrees to reach the desired position of 90 degrees.

This is important to understand, because if the s*pin for* block is used and the arm is not fully lowered or the claw was not fully closed, the arm or claw could approach its limit for how far it can move.



Let's look at blocks to use with the *spin for* and *spin to position* blocks that help them to program your robot more precisely.

- The *set motor timeout* block is used to prevent motion blocks that do not reach their position from preventing other blocks in the stack from running. An example of a motor not reaching its position is an arm or claw that reaches its mechanical limit and cannot complete its movement.

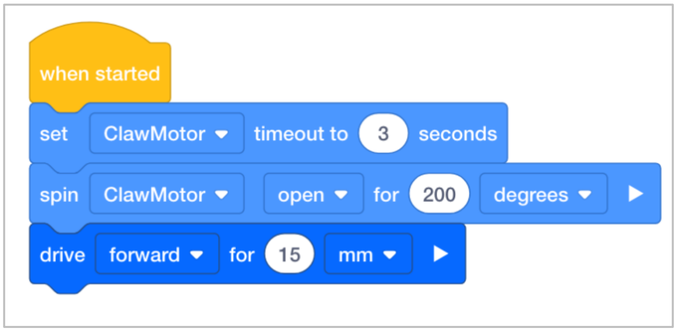


- What happens if a *spin for* block is used and the claw or arm reaches its limit for its range of motion? Will the project stop because the arm or claw cannot move any further?

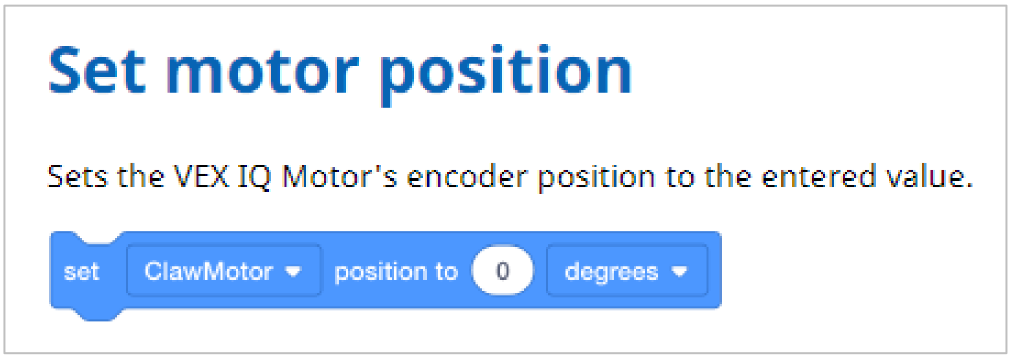
  The project will not stop until the block has completed its task. If the claw is trying to open 100 degrees but starts from 50 degrees and is trying to spin beyond its range of motion, the claw will continue to try to open even though it cannot. This is not a good situation because this can strain the parts and drain the battery.

  In this case, the s*et motor timeout block* can be used. This block acts as a fail-safe so that if a motor reaches its mechanical limit, it can continue on with the rest of the project after a certain amount of time.

In the following example, the robot will drive forward after the claw has opened the full 200 degrees or reached the timeout of three seconds.



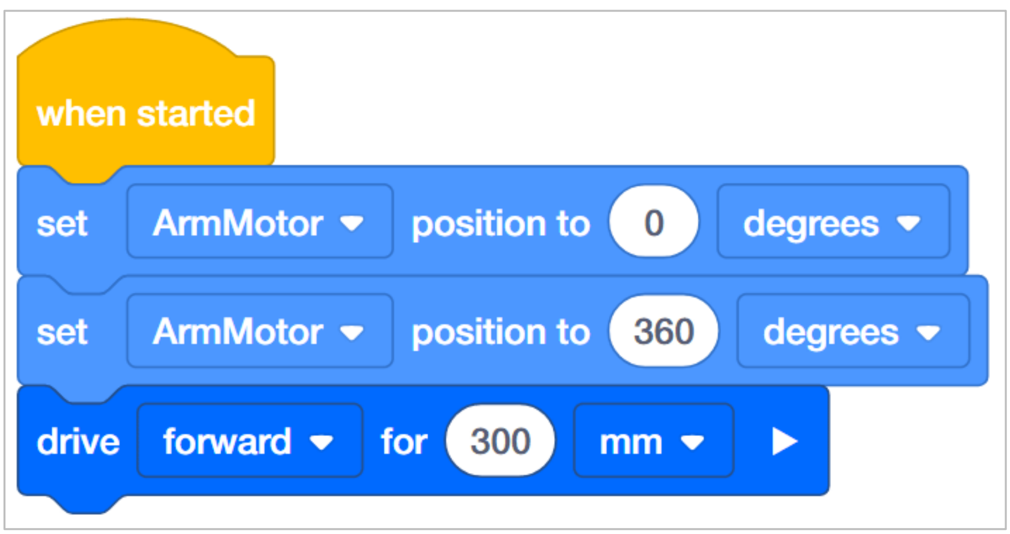
- The *set motor position* block is used to set the motor's Angle value (its position) to a selected value. It can also be set to 0 degrees to reset the motor's position.



- A *spin to position* block is easier to program with when you know what the motor's angle currently is. But sometimes, the arm might look like it's fully down when it is actually raised a few degrees.

  The *set motor position block* lets you set the degrees that you want the motor's Angle to be at. This is very useful for resetting the motor's position to 0 degrees.

  In the following example, the robot's Arm Motor is reset to 0 degrees no matter where it currently is before it spins to the 360 degrees position and drives forward.



Programming a Sequence - VEXcode IQ Blocks

Let's program a sequence!

In this activity, you will outline a plan for the sequence of movements that your robot will need to make in order to grab, lift, and move an object.

You will first review two tutorial videos about programming the claw and arm. Then you will identify the correct sequence of steps to approach, grab, and lift an object, and plan that project using pseudocode.
You will then create, download, and run the project using the pseudocode that helped you plan!

- Make sure you have the hardware required, your engineering notebook, and VEXcode IQ Blocks downloaded and ready.

Hardware/Software Required:

Quantity	**Hardware/Other Items**
1	VEX IQ Super Kit
1	VEXcode IQ Blocks (latest version, Windows, macOS, Chromebook, iPad)
1	Engineering Notebook
1	Meter stick or ruler
1	Clawbot (Drivetrain 2-motor) template example project
1	Aluminum can or empty water bottle

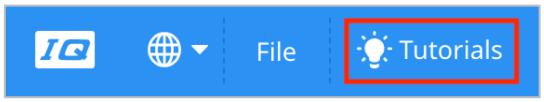
1. Preparing for the Exploration

Before you begin the activity, do you have each of these items ready? The Builder should check each of the following:

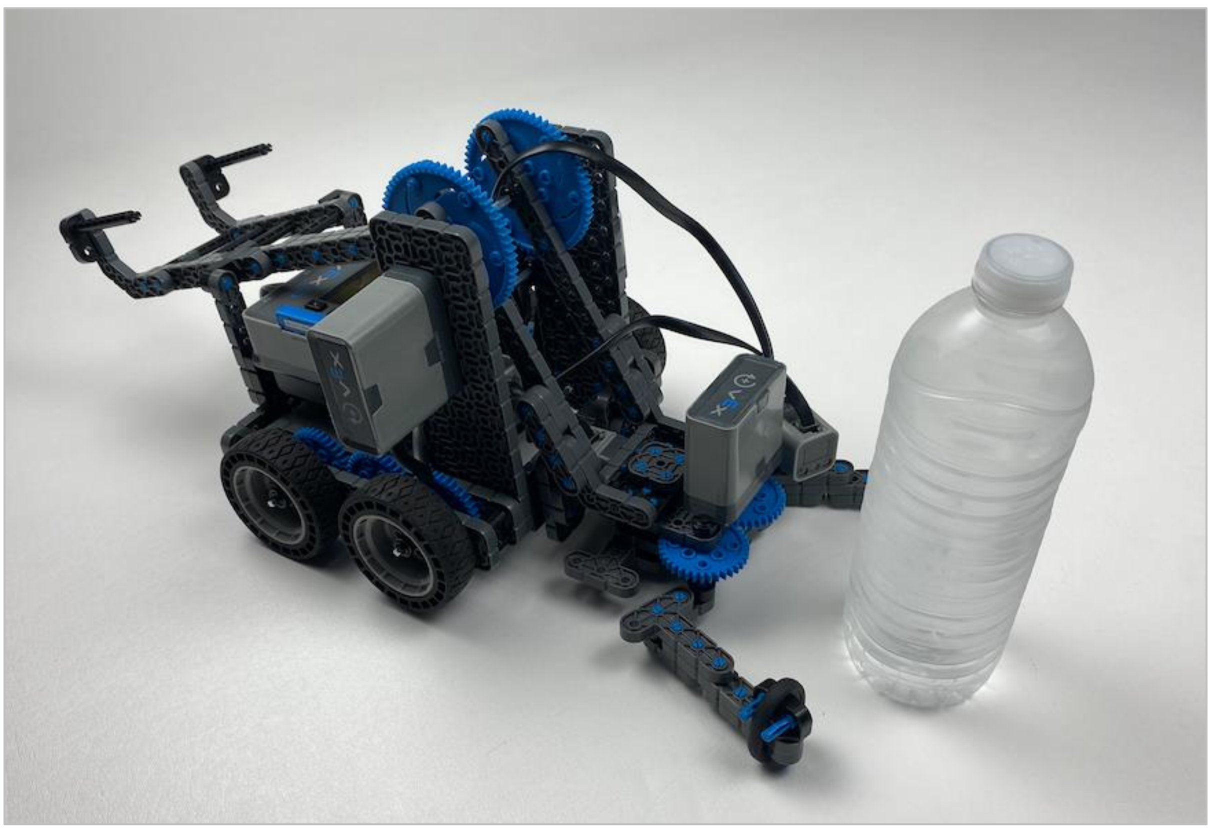
- Are all of the motors and sensors plugged into the correct ports?
- Are the smart cables fully inserted into all of the motors and sensors?
- Is the Brain turned on?
- Is the battery charged?
- Is the Radio inserted into the Robot Brain?

2. Start Planning a Path

Before you begin planning the path that your robot will take, first review the Moving the Arm and Open the Claw tutorial videos in VEXcode IQ Blocks.



Now, in your engineering notebook, use pseudocode to write the correct sequence of steps to approach, grab, and lift an object like an empty water bottle or aluminum can.

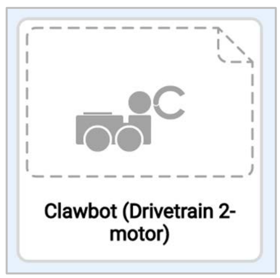


- Parts of this problem to keep in mind:

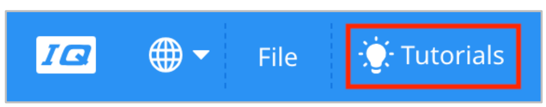
    - First, you will need to measure in millimeters how far away the object is from the robot. You will need this to determine how far forward and in reverse the robot should move.
    - Identify how many degrees the claw will have to open and close depending on its range of motion and how large the object is. For more help, refer to the previous page in the Play section.
  - *Hint*: Open the Device Menu and see how many degrees the claw will close with the object inside.
    - Identify how far up in degrees the arm will raise to carry the object.
    - When you approach an object, the claw should already be opened. If you approach an object with the claw closed, the closed claw could knock the object over.
    - The robot will also have to grasp the object in its claw, raise the object up, move in reverse to relocate the object, and then place the object back down and release it.

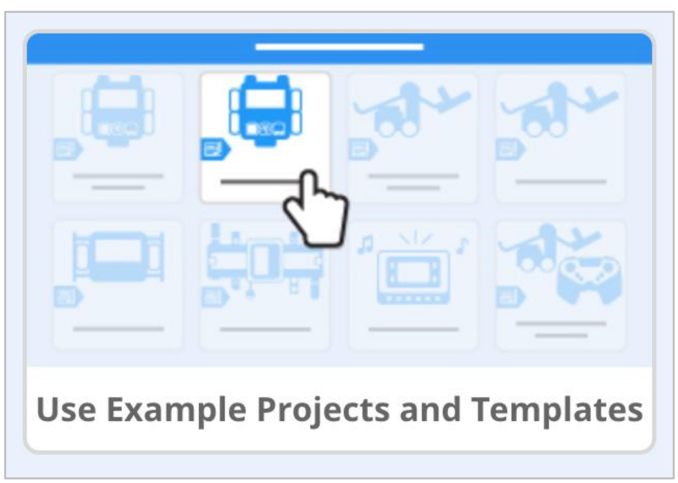
3. Programming a Sequence

- Open the Clawbot (Drivetrain 2-motor) template example project.

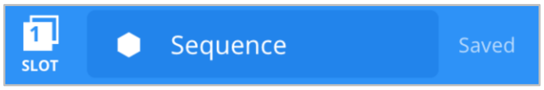


- For help opening an example project, view the Use Example Projects and Templates tutorial video in VEXcode IQ Blocks.



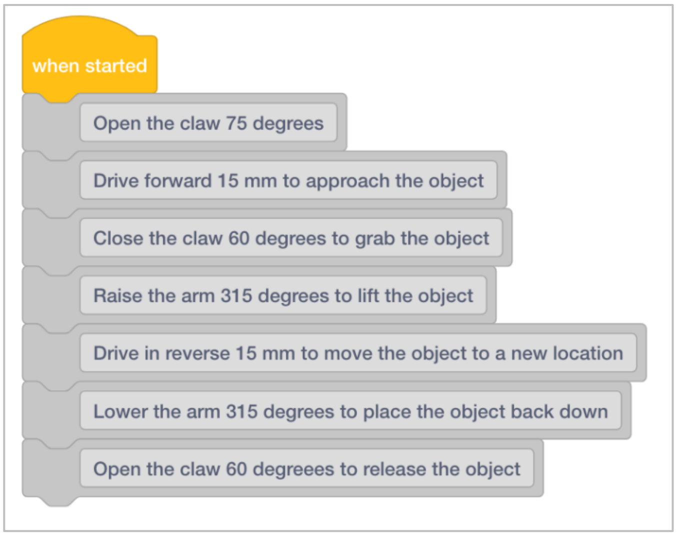


- Rename and save the project as **Sequence**.



- For help renaming and saving a project, view the Naming and Saving Your Project tutorial in VEXcode IQ Blocks.

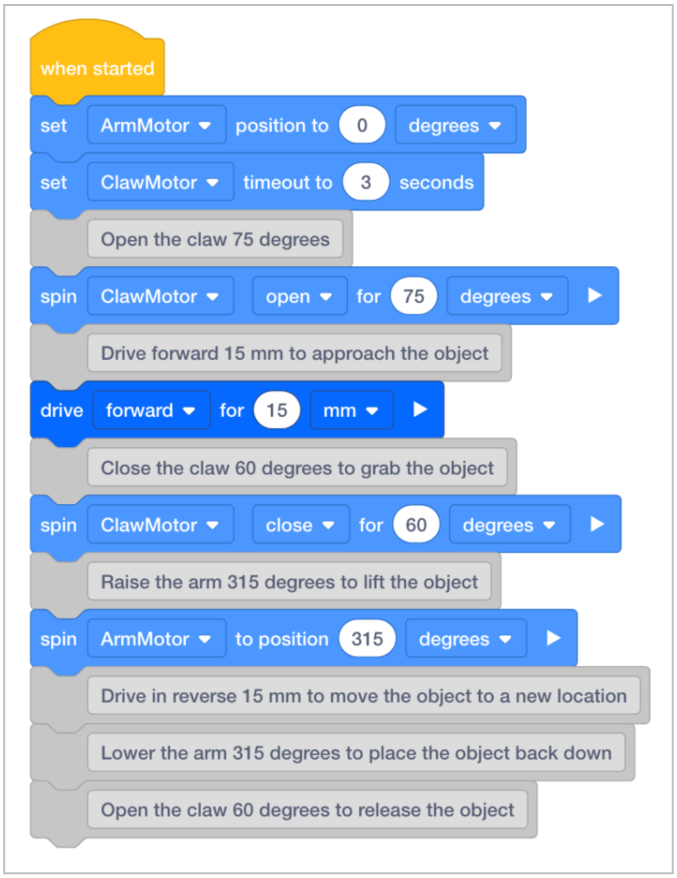
- Now, create your project using your pseudocode. First outline your project using comments to insert your pseudocode.
  The example shown below is a reference. Degree and distance measurements may be different depending on how far away your object is and its size.



- Use the drive, *spin for*, and *spin to position* blocks to create the project based on the pseudocode.

  Don't forget to reset the Arm Motor's position to 0 and to include a 3-second timeout for the Claw Motor.

  Use the image below as an example of how to organize the project. The following project is not complete but you should completely program yours.



- After your project is complete, predict what it will do. Write your predictions in your engineering notebook.

4. Test the Project!

Now that you have created a project to approach, grab, and lift and object - let's test it!

- Download and Run your Sequence project. For help, view the Download and Run a Project tutorial video in VEXcode IQ Blocks. It will have the following icon:

Did your project run as intended? Write your observations in your engineering notebook comparing your pseudocode to your final project and answer the following questions:

- Did your project have your robot to grab, lift, and move an object?

- Why is this sequence of movements important?

SPARK
APPLY

Become a 21st century problem solver
by applying the core skills and concepts
you learned to other problems.

Warehouse Robots

Robots working in a warehouse

Meeting Consumer Needs

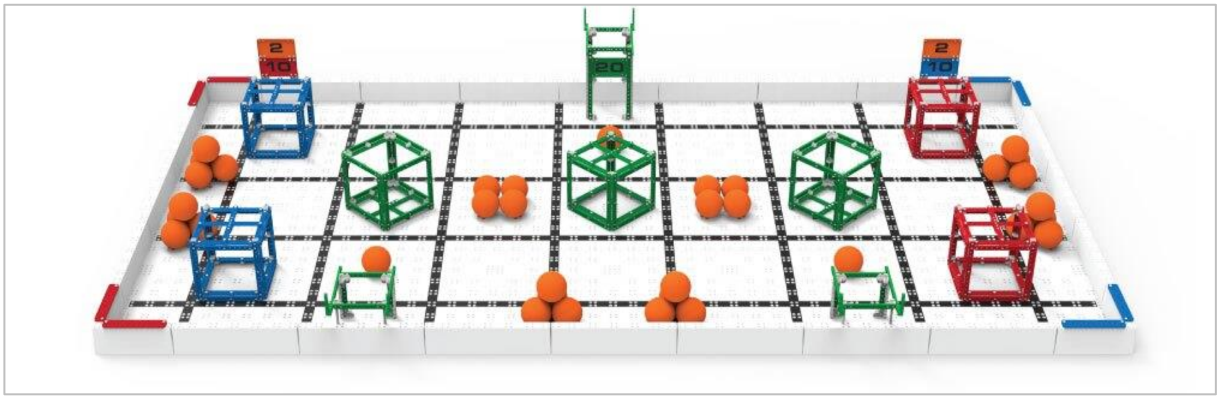
As more and more consumers shop online with guarantees of quick delivery, robots are being used to help fulfill the demand. The quicker customers' requests are fulfilled after a purchase, the happier they are and the more profit the company makes. This makes using robots to assist with orders at the warehouse a great benefit. Some companies use robots to bring the shelf stacks to human workers to select the correct product, while others use robots to travel to identified areas to grab the needed items.
Robot developers are continuing to improve the process. One company is developing robotic arms that are capable of handling fragile objects without having to give the robot detailed information on the object's size or shape. Another company is exploring "swarm robotics," where several robots work as a team by communicating together to complete delivery tasks.

Some of the benefits of using warehouse robots instead of humans are:

- Better accuracy in selecting the correct items
- More efficient (speed)

- Reduction of utility costs like air conditioning
- Less workplace theft
- Reduction of labor cost (fewer workers needed)

Game Strategy



2019-2020 VEX IQ Challenge - Squared Away

Squared Away

One of the more challenging aspects of the VEX IQ Challenge is that a new game design is introduced for each competition season. This allows students to use their previous game experience as they tackle the new objects and goals of the game while giving both experienced and new teams an equal starting place.

Each year, the challenge requires new strategies and new techniques for scoring as many points as possible. Some teams strategize to move the closest objects first, some strategize to move multiple objects at a time, and some teams do both. The best strategy for scoring the most points in the least amount of time depends on that season's layout of the challenge field and the team's design of their VEX IQ Robot, like which manipulators it has and how the robot moves best.

For the Squared Away Challenge, some strategies could include collecting multiple balls at a time first before placing them in or on a cube. Another strategy could be to place balls inside of the cube, move the cube to the scoring zone, then place balls on top so that transporting the cube does not knock any of the balls off of the top.

A team should focus on their own robot's strategy, however, teams should be flexible with their strategy since they will be randomly assigned a teamwork partner for the challenge. Many different combinations can be explored for the best way to score points. Some teams may strategize how many balls they can score inside of a cube or focus on placing the green cube on the platform.

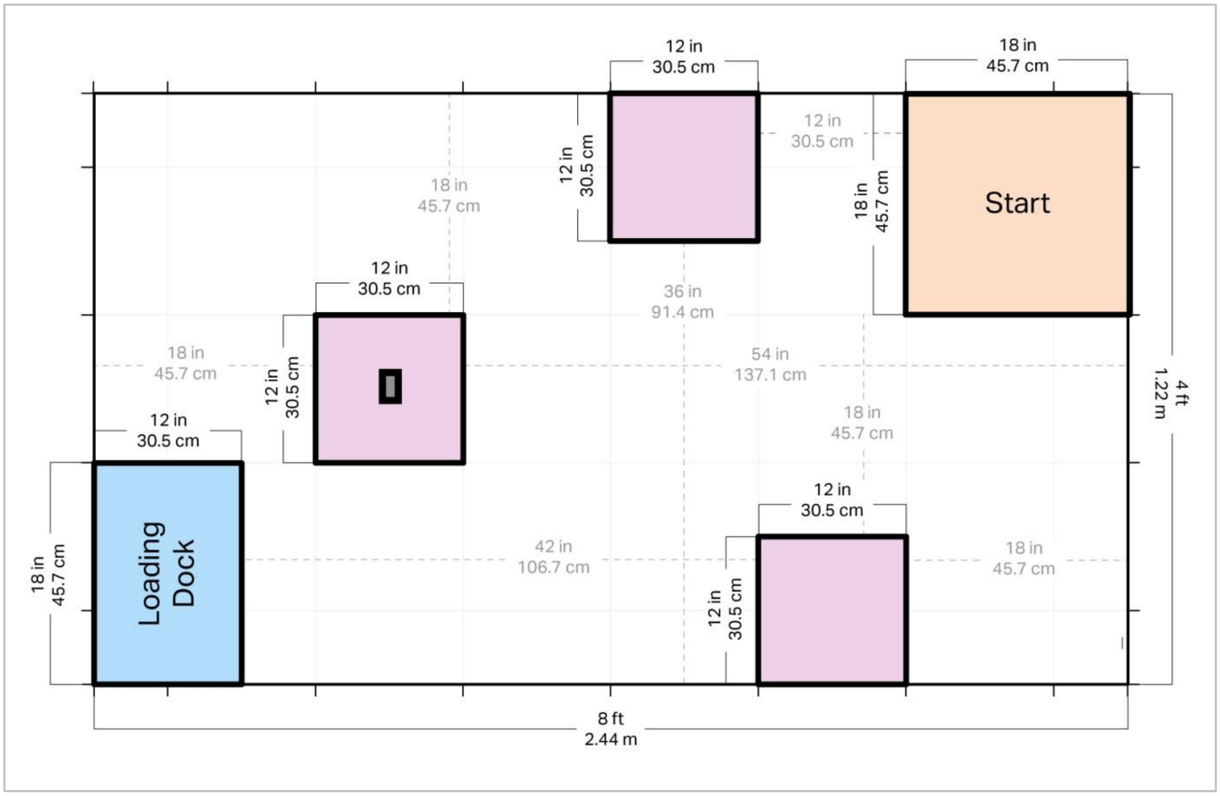
Some common strategies to keep in mind when analyzing a VEX IQ Challenge:

- Read, study, and understand the game manual.
- List all of the different ways to score points and the point values associated with these methods.
- Consider different components of the robot that would be useful for scoring points.
- Record any size or part restrictions for the robot.
- Identify clear objectives or tasks that you want the robot to accomplish in order to score points.

SPARK RETHINK

Is there a more efficient way to come to the same conclusion? Take what you've learned and try to improve it.

Prepare for the Package Dash Challenge



Package Dash Preparation Layout

Prepare for the Package Dash Challenge

In this challenge, you will program your robot to pick up a package and bring it to the Loading Dock as fast as possible! You will use the same skills that you used in the Play section to outline and program a sequence of events to complete a specific task.

To successfully complete this challenge, you need to create a project that drives the robot to a specific place (the pink squares) in the warehouse, picks up a package (aluminum can), and drop it onto the loading dock.

Ask your teacher if you or your group should set up the Package Dash Challenge. Also, check with your teacher for which pink squared-off area will contain the package, or if the package will be placed on the left pink square as shown in the layout above.

When the Challenge field is ready, you should measure all of the driving distances and the diameter of the can so that you can precisely plan and program.

To complete the challenge you will need:

- 4 x 8 feet or 1.22 x 2.44 m open area
- Optional: VIQC Field
- Roll of tape
- 3 aluminum cans
- A ruler or meter stick to measure distances
- Stopwatch

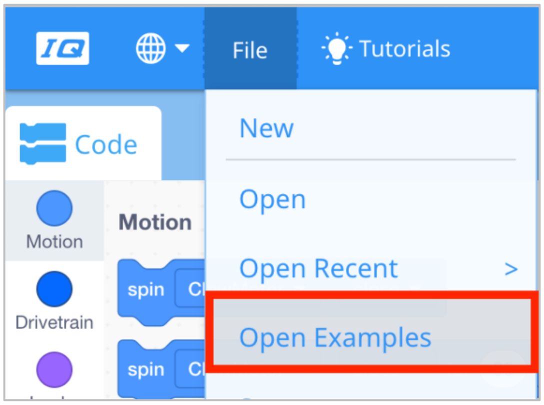
Design, Develop, and Iterate on your Project

Follow the steps below as you create your project:

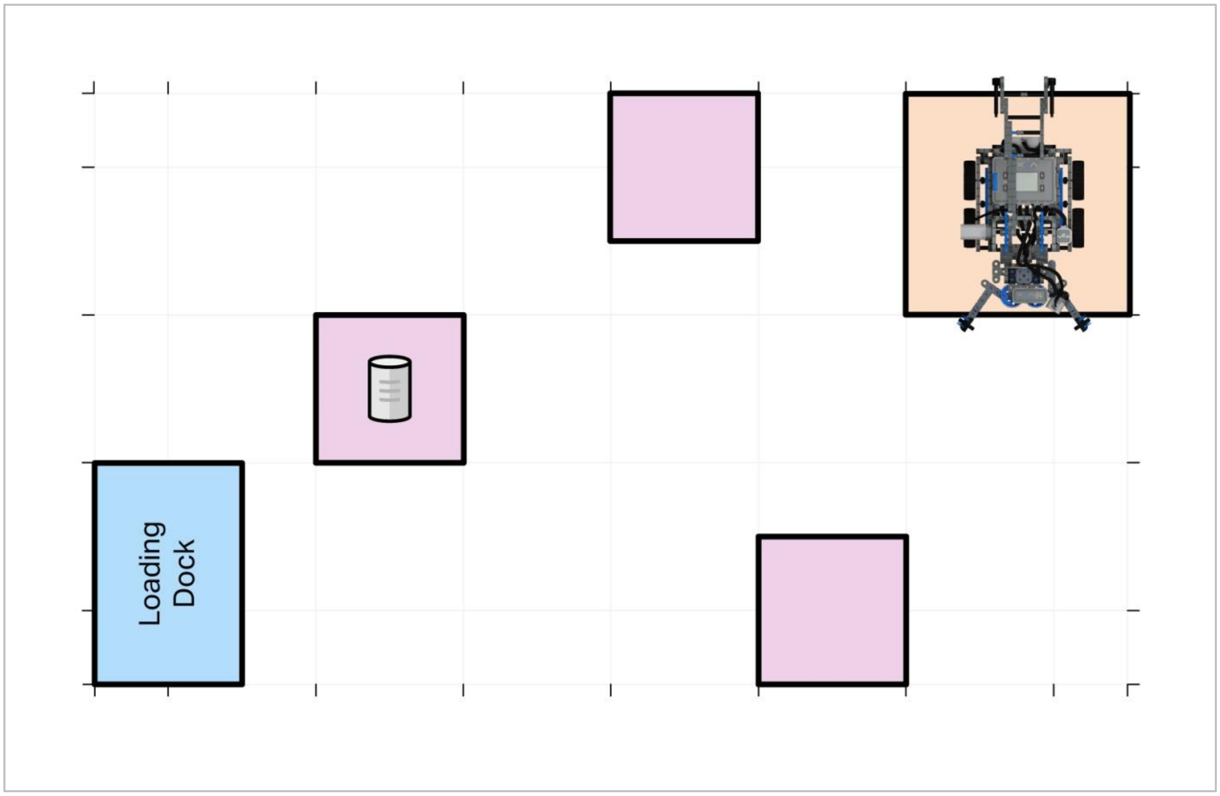
- Plan out the path you want to program your robot to take using drawings and pseudocode.
- Use the pseudocode you created in the Play section to develop your project using blocks.
- Test your project often and iterate on it using what you learned from your testing.

If you're having trouble getting started, review the Example Projects below within VEXcode IQ Blocks:

- Forward (Inches or mm)
- Backward (Inches or mm)
- Left Turn (Degrees)
- Right Turn (Degrees)
- Claw and Arm
- Use the Claw
- Use the Arm



The Package Dash Challenge



Starting the Package Dash Challenge

Package Dash Challenge

In this challenge, you will program your robot to pick up a package and bring it to a loading dock as fast as possible!

Challenge rules:

- The robot must begin the challenge in the Start Zone.
- The package (aluminum can) can only come in contact with the squared-off area, the Clawbot's claw, and the Loading Dock.
  - If a package is dropped on the warehouse ground, you must reset the field and start over again.
- The time for each run starts as soon as the robot moves.
- The time stops as soon as the package is dropped in the loading dock.
- When resetting the field, everything should be returned to the exact location as it started.

- Have fun!

Bonus Challenge: Add sounds for when the robot is backing up and lights from the Touch LED to indicate when the robot has picked a package up and placed it in the loading dock.

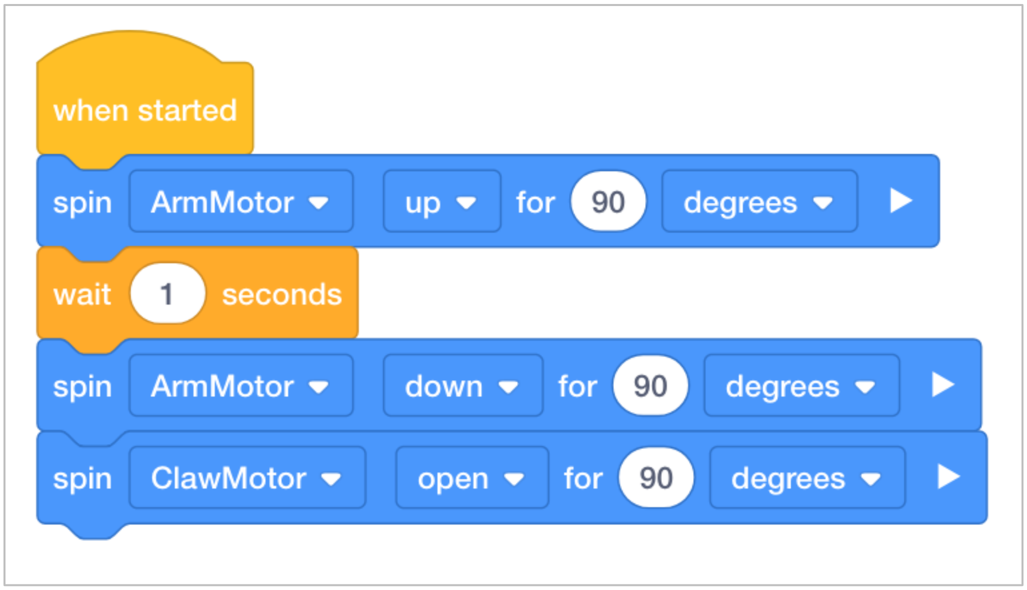
Increase complexity: Add more packages (cans) that the robot must pick up! Multiple rounds can be played.

SPARK
KNOW

Understand the core concepts and how to apply them to different situations. This review process will fuel motivation to learn.

Review

1. **In the following project, what will the robot do?**

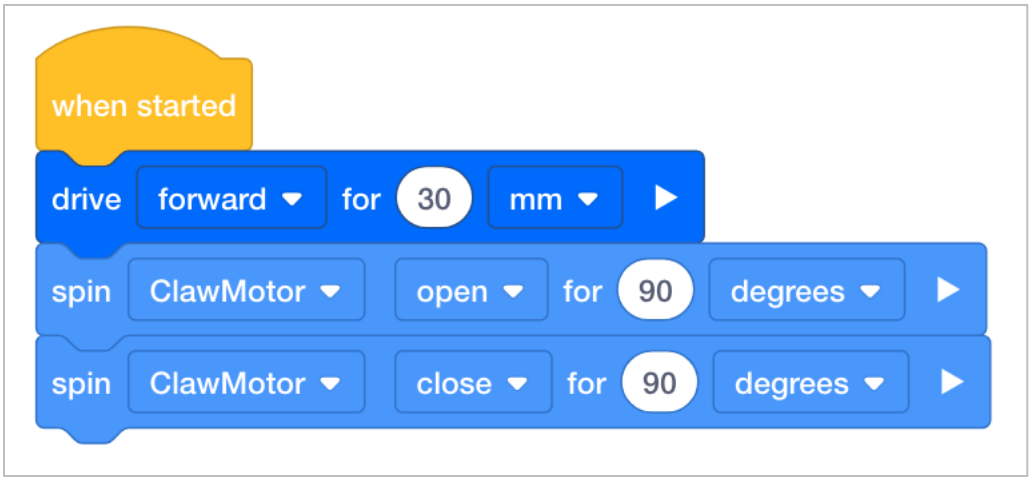


- ○ Open the claw, wait 1 second, close the claw, and then lift the arm.
- ○ Lift the arm, wait 1 second, lower the arm, and then open the claw.
- ○ Lift the arm, wait 1 second, lower the arm back to a position below where it started, and then open the claw.
- ○ Open the claw, wait 1 second, close the claw, and then reopen the claw again.

2. **Which of these is the correct definition of a *spin to position* block?**

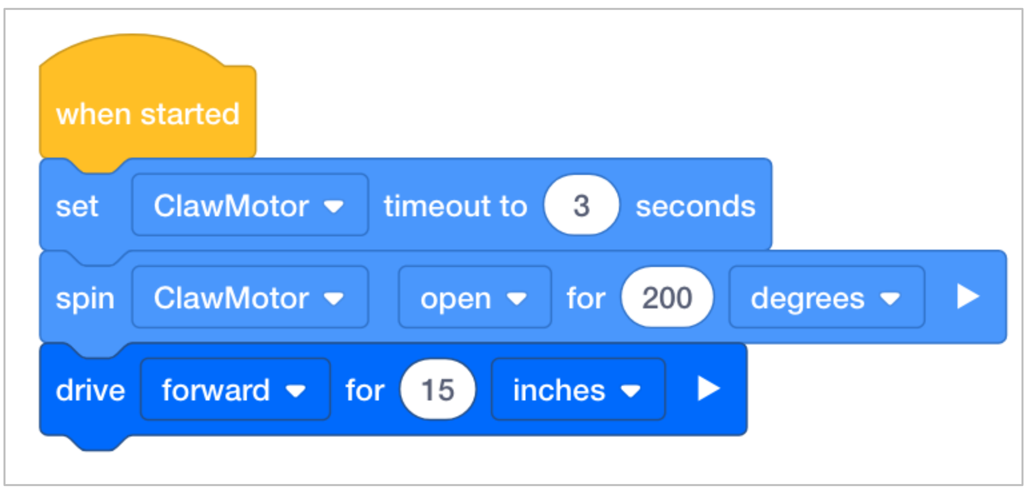
- ○ A block that spins the motor to reach to a specific position measured in degrees based on its full range of motion
- ○ A block that spins the motor in a specific direction for a specific distance
- ○ A block that resets the motor's position
- ○ A block that sets a timeout for the motor

3. **The robot was supposed to drive forward and grab an object. What is the biggest error in this project?**



- The project makes the robot first drive in reverse.
- The Claw Motor should only be programmed using *spin to position* blocks.
- The claw should open before the robot drives forward.
- The Claw Motor needs to be programmed to *spin for* distances greater than 180 degrees.

4. What will this project have the robot do?



- The claw will open 200 degrees in 3 seconds and then the robot will drive forward.
- The arm will lift for 200 degrees within 3 seconds and then the robot will drive forward.
- The robot will open the claw 200 degrees and then drive forward for 15 inches or as far as it can within 3 seconds.
- The robot will drive forward after the claw has opened 200 degrees or reached the timeout limit of 3 seconds.

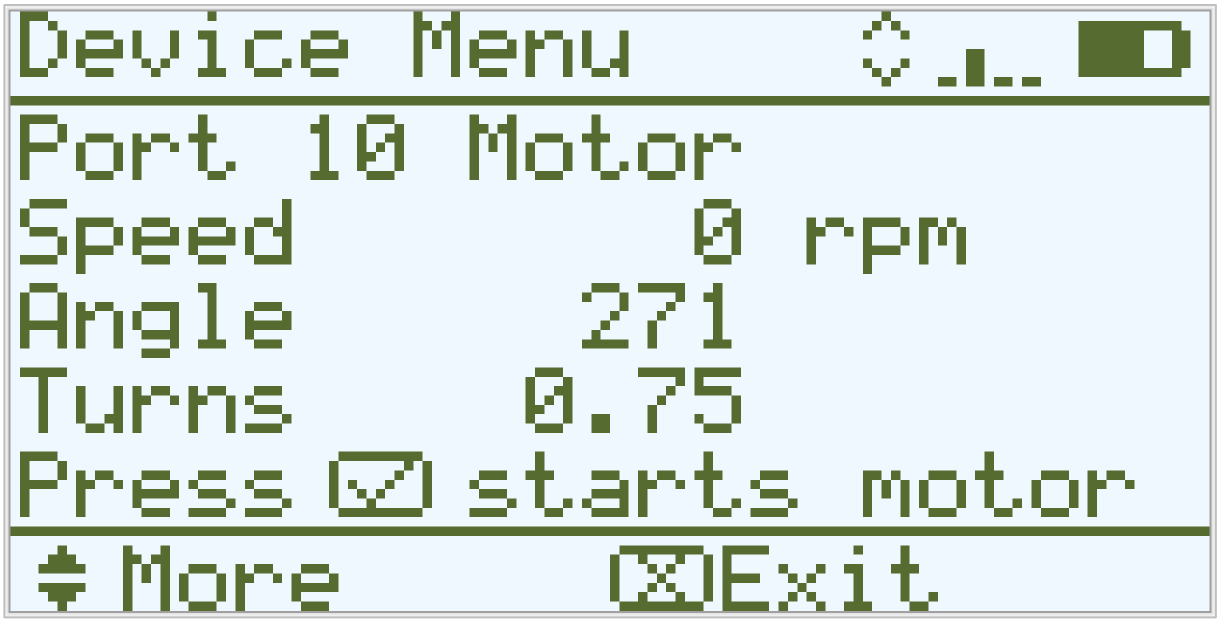
5. *True or False:* A programmer can use a *set timeout* block to make sure that the project does not get stuck if a motor's motion block cannot complete its movement. The timeout lets the project continue to the next block.

- True
- False

6. Which of these lines of pseudocode would be best for programming a robot to grasp and transport an object to a new location?

- Find and pick up an object.
- Open the claw, drive forward 15 mm, raise the arm 360 degrees, drive in reverse 15 mm, lower the arm 360 degrees, and close the claw.
- Open the claw 75 degrees, drive forward 10 mm, close the claw 60 degrees, raise the arm 315 degrees, drive in reverse 10 mm, lower the arm 315 degrees, and open the claw 60 degrees.
- Pick up the object and move it 15 mm.

7. Look at this image of the VEX IQ Brain's Device Info screen. What is the current position of the Arm Motor?



- Its position is 0 degrees.
- Its position is 271 degrees.
- Its position is 0.75 degrees.
- Its position is 271 turns.

APPENDIX

Additional information, resources, and materials.

Knowledge Base Articles

Links to the VEX Robotics Knowledge Base Articles for this STEM Lab:

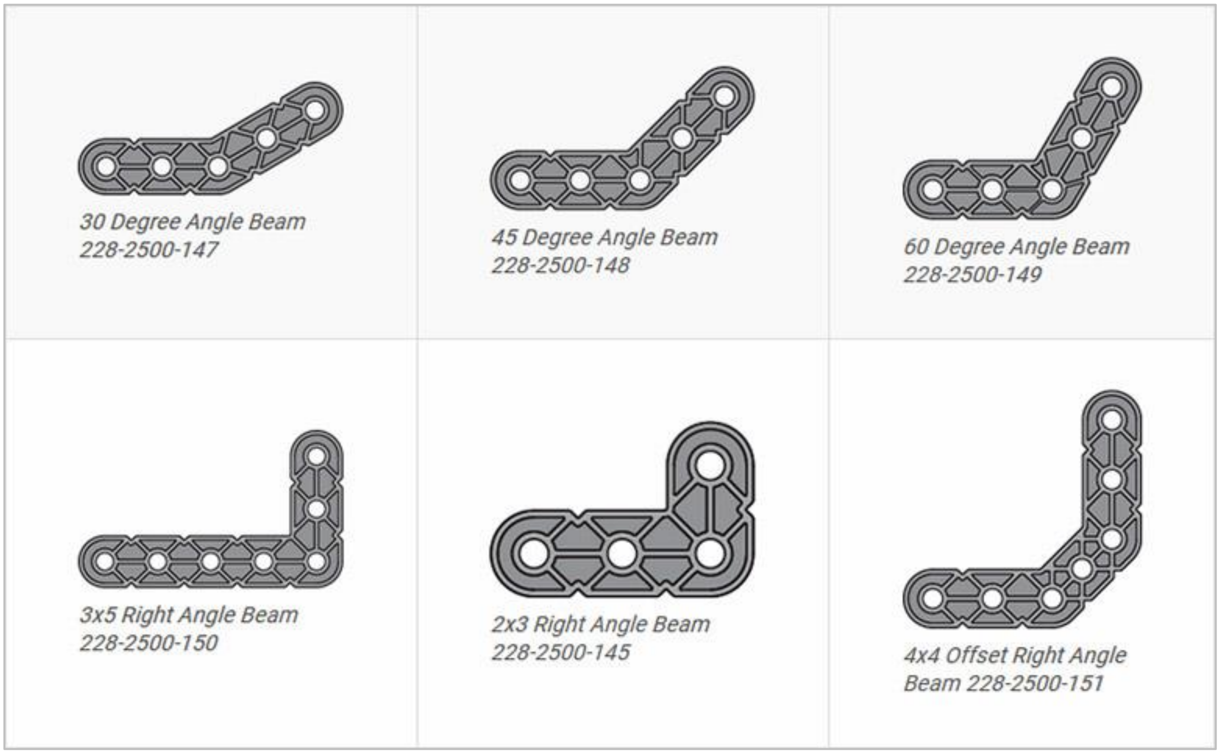
- How to Turn On/Off a VEX IQ Robot Brain
  https://kb.vex.com/hc/en-us/articles/360035952571-How-to-Turn-On-Off-a-VEX-IQ-Robot-Brain
- How to Read Indicator Lights on the VEX IQ Robot Brain
  https://kb.vex.com/hc/en-us/articles/360035590672-How-to-Read-Indicator-Lights-on-the-VEX-IQ-Robot-Brain
- How to Navigate the VEX IQ Robot Brain
  https://kb.vex.com/hc/en-us/articles/360035952331-How-to-Navigate-the-VEX-IQ-Robot-Brain
- How to Connect VEX IQ Devices to Smart Ports
  https://kb.vex.com/hc/en-us/articles/360035952151-How-to-Connect-VEX-IQ-Devices-to-Smart-Ports
- How to Install or Remove the VEX IQ Robot Battery
  https://kb.vex.com/hc/en-us/articles/360035951991-How-to-Install-or-Remove-the-VEX-IQ-Robot-Battery
- How to Charge the VEX IQ Robot Battery
  https://kb.vex.com/hc/en-us/articles/360035955011-How-to-Charge-the-VEX-IQ-Robot-Battery
- How to Use the Autopilot Program in the Demos Folder
  https://kb.vex.com/hc/en-us/articles/360035952031-How-to-Use-the-Autopilot-Program-in-the-Demos-Folder
- Best Practices for Preserving the VEX IQ Robot Battery’s Life
  https://kb.vex.com/hc/en-us/articles/360035953671-Best-Practices-for-Preserving-the-VEX-IQ-Robot-Battery-s-Life
- Ideas for Organizing the VEX IQ Super Kit
  https://kb.vex.com/hc/en-us/articles/360035590332-Ideas-for-Organizing-the-VEX-IQ-Super-Kit
- VEX IQ Brain Status (USB Cable)
  https://kb.vex.com/hc/en-us/articles/360035955411-How-to-Understand-the-VEX-IQ-Brain-Status-Icon-USB-VEXcode-IQ-Blocks

Links to VEXCode IQ Blocks Knowledge Base Articles for this STEM Lab:

- How to Begin a New Project in VEXcode IQ Blocks
  https://kb.vex.com/hc/en-us/articles/360035954551-How-to-Begin-a-New-Project-VEXcode-IQ-Blocks

- How to Download and Run a Project
  https://kb.vex.com/hc/en-us/articles/360035591232-How-to-Download-and-Run-a-Project-VEXcode-IQ-Blocks
- How to Save a Project on Windows
  https://kb.vex.com/hc/en-us/articles/360035954531-How-to-Save-a-Project-on-Windows-VEXcode-IQ-Blocks
- How to Save a Project on macOS
  https://kb.vex.com/hc/en-us/articles/360035954511-How-to-Save-a-Project-on-macOS-VEXcode-IQ-Blocks
- How to Save a Project on Chromebook
  https://kb.vex.com/hc/en-us/articles/360035955351-How-to-Save-on-a-Chromebook-VEXcode-IQ-Blocks
- How to Download to a Selected Slot on the Brain
  https://kb.vex.com/hc/en-us/articles/360035591292-How-to-Download-to-a-Selected-Slot-on-the-Brain-VEXcode-IQ-Blocks

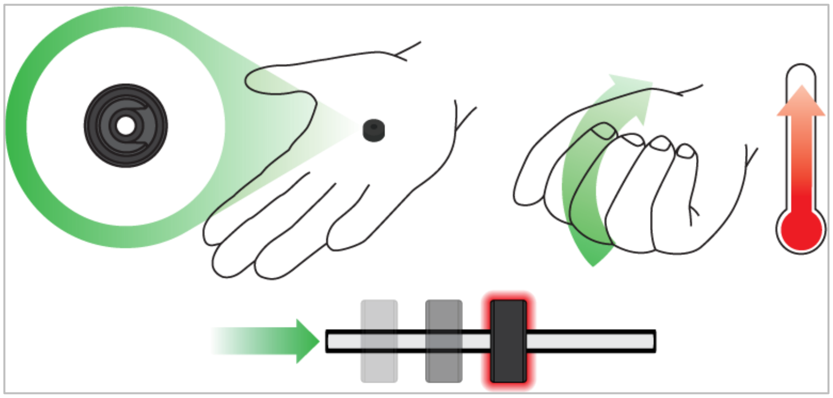
Identifying Angle Beams



How to Identify the Different Angles of the Angled Beams

There are four different types of beams that have a bend at an angle: 30^{o} Angle Beams, 45^{o} Angle Beams, 60^{o} Angle Beams, and Right Angle (90^{o}) Beams. There are also three types of Right Angle Beams: 3x5, 2x3, and Offset. The best way to tell which angles are which is to stack the beams on top of each other. Then you can compare how they look. You can also use a protractor to measure the angle of the beam.

Installing Rubber Shaft Collars

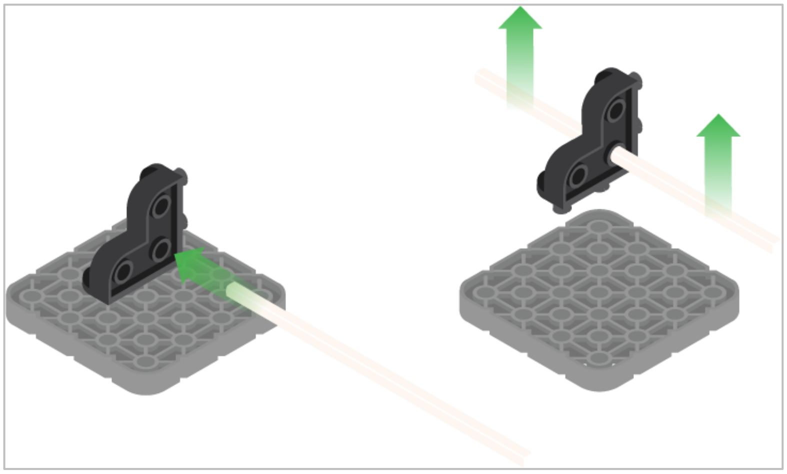


Using your hand to warm a Rubber Shaft Collar

Rubber Softens as it gets Warm

Hold the Rubber Shaft Collars in your hand for 15-30 seconds before you slide them onto a shaft. Holding the Rubber Shaft Collar in your hand will warm and soften the rubber to make it easier to slide onto a shaft.

Removing Connectors from Beams and Plates

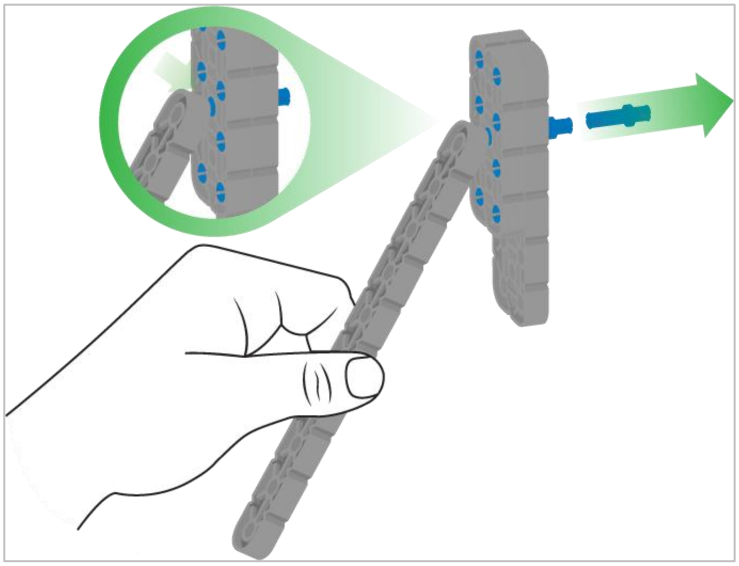


Using a pitch shaft to remove a corner connector

How to Easily Remove Connectors

You can easily remove corner connectors from beams or plates by placing a metal shaft through one of the holes of the corner connector and pulling outward while holding down the beam or plate.

Removing Pins from VEX IQ Beams and Plates



Removing a pin from a plate assembly using a beam

How to Easily Remove Pins from Beams and Plates

You can quickly remove connector pins from beams or plates by pressing a beam against the back of the pin, which partially pushes the pin out, so you can remove it with your fingers. You can use this technique to more easily remove pins from individual plates and beams, or from built structures.

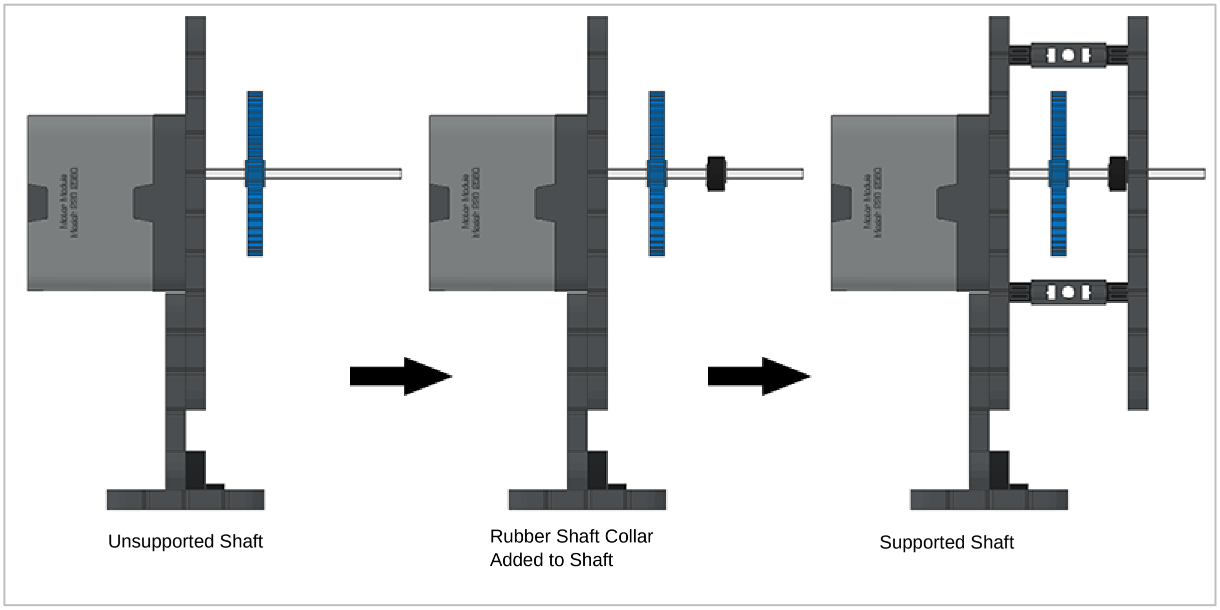
Removing Standoffs from Mini Standoff Connectors

Removal of a standoff from a Mini Standoff Connector

How to Easily Remove Parts from Mini Standoff Connectors

Standoffs and Mini Standoff Connectors can be separated by pushing a shaft through the Mini Standoff Connector. The same technique can be used for parts with similar ends in Mini Standoff Connectors, such as pins.

Supporting Shafts using Rubber Shaft Collars

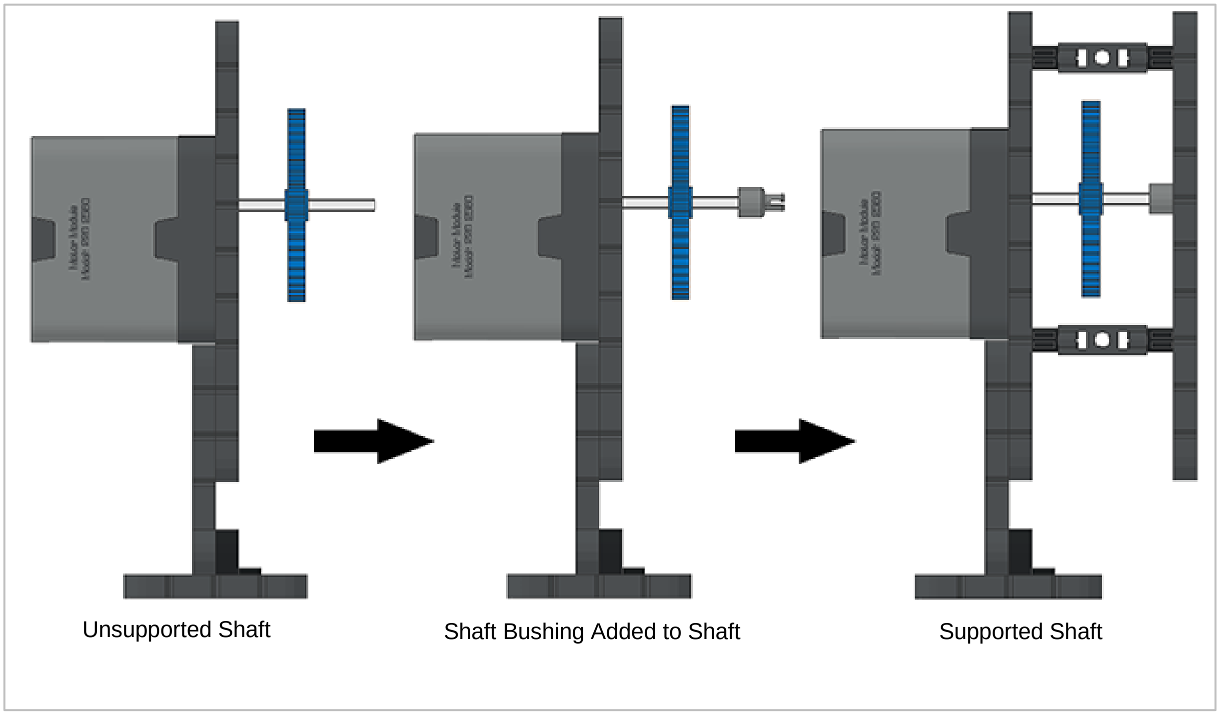


Supporting a shaft with a Rubber Shaft Collar

How to Support Shafts with Rubber Shaft Collars

Shafts can fall out of place or alignment very easily if they aren't supported properly. You can make a shaft more secure and prevent it from falling out of place by putting a Rubber Shaft Collar before the end of it. You can then connect the shaft to a support structure with the shaft collar positioned against it. That will allow the shaft to turn but will prevent it from wobbling or falling out.

Supporting Shafts using Shaft Bushings



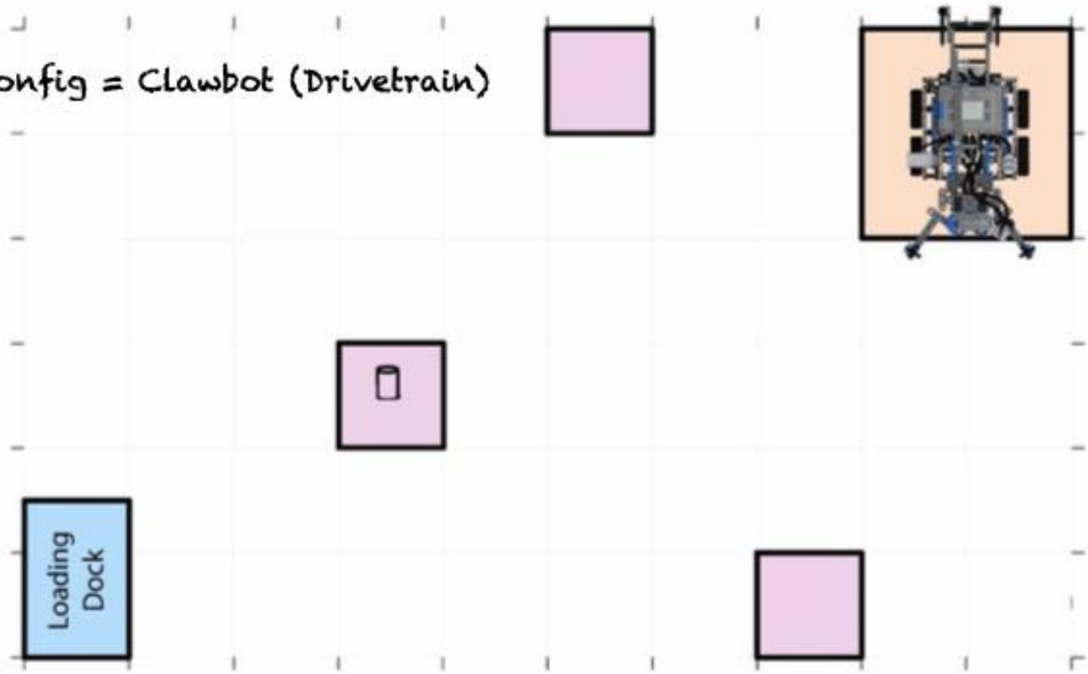
Supporting a shaft with a Shaft Bushing

How to Support Shafts Using Shaft Bushings

Shafts can fall out of place or alignment very easily if they aren't supported properly. You can make a shaft more secure and prevent it from falling out of place by putting a bushing at the end of it. You can then connect that bushing into another beam or additional part. That will allow the shaft to turn but will prevent it from wobbling or falling out.

Example Pseudocode for the Package Dash Challenge

Config = Clawbot (Drivetrain)



Drive forward 1000 mm

Turn right 90 degrees

Open the claw 75 degrees

Drive forward 1350 mm

Close the claw 60 degrees

Raise the arm 315 degrees

Drive in reverse 280 mm

Turn left 90 degrees

Drive forward 600 mm

Turn right 90 degrees

Drive forward 1350 mm

Lower the arm 315 degrees

Open the claw 60 degrees