VEX IQ Robotics Camp Handbook

Table of Contents

Within each activity, sections have been designated as "cl" for materials intended for "camp leaders" and "cs" for materials intended for "camp or students."

- Content material
- Build Instructions
- Written exercises
- Sample Program Breakdown

When you see these icons, refer to your Kit Documentation for more information.

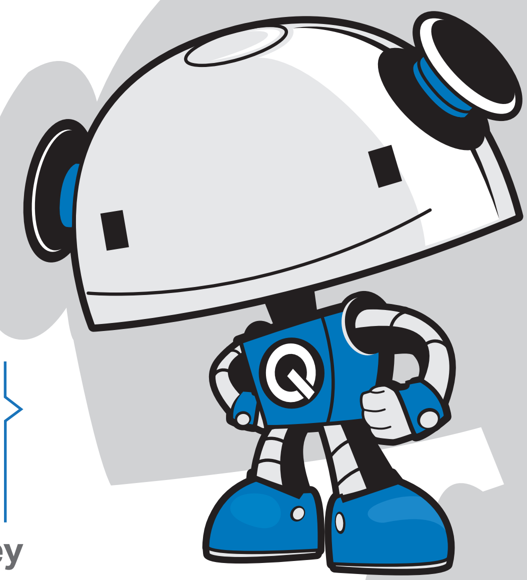
Build Instructions

Control System User Guide

Quey *[Q-E]*

I'm Quey, your personal VEX IQ helper, and I'll be with you every step of the way as you learn how use VEX IQ! Watch for me on the sidelines - I'll try to offer some tips and tricks that should help you out when things get tough.

vexiq.com/askquey **@AskQuey**



Using this Handbook

Using This Handbook

The VEX IQ Robotics Camp Handbook enables you to hold a robotics camp, regardless of your experience level. This handbook gives educators, competition teams, and community volunteers the necessary resources they need to provide a great robotics experience for students of all ages as well as provides in-depth training materials for teachers and adults. Covering a variety of topic options, and providing support for multiple camp structures this handbook is adaptable to your specific needs & circumstances.

Features of the Handbook:

- For beginner to intermediate users
- Flexible one, three, and five-day camp options.
- Instruction through open-ended challenges
- Options for both mechanical & programming focus

The five sections of this Handbook Include:

- The Get Ready Guide: This section guides camp leaders through the planning process and considerations for organizing your camp, including what VEX IQ kits to purchase to run the camp.
- The Camp Overview: This section guides camp leaders and campers through the different Learning Activities and paths that can be chosen to construct your camp experience.
- Open-Ended Challenges: This section provides four challenges that can be used as part of the camp experience. Lots of tips, options, and even sample programs are also provided.
- Real-World Extension Activities: This section provides four extension activities that can be used to help campers apply their work more directly to the world around them.
- Learning Activities: Providing new experiences and samples while leveraging the content from the VEX IQ Curriculum (vexiq.com/curriculum) and Robotics Education Guide (P/N: 228-3319) this section provides fifteen Learning Activities that teach core concepts and principles. Camp leaders can choose which learning activities best fit the camp experience they are providing.

Associated Resources

In addition to this Handbook, the following resources are available to help plan and round out your camp experience:

VEX IQ KITS
vexiq.com/products

Kit Documentation - see your VEX IQ kits for Build Instructions and much more

VEX IQ Curriculum
vexiq.com/curriculum

VEX IQ Robotics Education Guide & Teacher Supplement
P/N: 228-3319 & P/N: 228-4339

VEX IQ Resources Page
vexiq.com/documents-downloads

VEX IQ Youtube Playlist
vexiq.com/videos

Get Ready Guide

Get Ready Guide

Camp Considerations

The tips and suggestions in the "**Get Ready Guide**" have been provided to help guide camp leaders through the thought process and considerations required when starting up a new camp.

As with any new venture, it is important to do your research, consult local governing bodies and authorities, and complete a survey of other similar camps in your area.

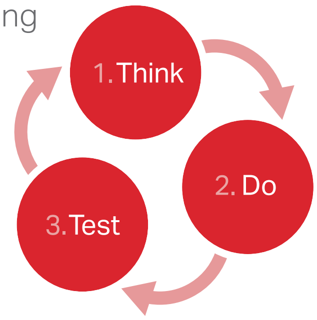
Consideration #1: Set The Tone For Your Camp

Having a clear idea of the focus and tone of your camp will help you outline and advertise what your VEX IQ Camp has to offer. Here are a few points to help get you started.

- VEX IQ is primarily designed for participants ages 8-14.
- Learning Activities are hands-on, engaging, and focus on the engineering design process of Think-Do-Test.

  "Think-Do-Test" is an iterative (repeated) process engineers use to help solve a problem and design a solution for something. The basic process is a cycle that generally starts with a problem and ends with a solution, but steps can vary. When simplified, the design process can be seen as a three-step loop.

  The Learning Activities outlined in this guide have been developed with this process in mind.



- This guide outlines opportunities for 1, 3, and 5-day camp experiences.
- Learning Activities provide opportunities for building, designing, and programming robots in:
  - Modkit for VEX Robotics
    www.modkit.com/vex
  - ROBOTC for VEX Robotics – Graphical
    http://www.robotc.net/graphical/

When outlining the day-to-day routines for your camp, make sure to include (and highlight!) social, outdoor, creative time, or time for team-building Learning Activities. Campers of all ages need time to get to know each other and time to move.

A sample schedule for a 5-day camp has been provided to assist with planning. See the section called "**Sample Camp Schedule**."

A list of team-building activities has also been provided to help campers get to know and work with each other. See the section called "**Sample Team-Building Activities**."

Consideration #2: Choose A Location

Throughout the school year, religious or community centers are often available and open to hosting day camps or workshops. Fees and conditions on use of the space will vary, so make sure to survey several locations and agreements.

Local colleges or schools may also be available throughout the summer months.

Features to look for in a location:

- Nearby parks or play areas for recreation time
- Accessible public transportation to and from the location at convenient times
- Nearby parking for easy drop off and pick up by parents and guardians
- Accessible spaces for campers of all abilities
- Convenient restrooms for camper and staff to use throughout the day
- Building owners or managers who are enthusiastic and encouraging of "camp spirit." Campers will be active, and at times, noisy, so it is important that staff and administration at the location are aware of and embrace this.

Consideration #3: Set Pricing

When deciding on a price per camper, make sure to include:

- Advertising costs
- Staff costs
- Site fees
- Insurance fees
- Consumables such as photocopies, paper, pencils, tape, scissors, or art supplies
- Beverages, snacks or meals (if they are to be included)
- Transportation costs
- Cost of VEX IQ kits and components (See the section called "Suggested VEX® IQ Components and Costs")

- When setting price, consider other camps in the area.
- Keep the pricing reasonable for families in the area.
- Look for and get involved with organizations to help support campers that may not be able to afford to attend otherwise.
- Depending on the location, needs, and interests of the community, some parents and guardians may appreciate a pricing option that includes the cost of a Super Kit to keep at the end of a camp session.

Consideration #4: Advertise

Make sure to include the following on brochures, postings, or websites:

- Age of campers
- Kind of camp
- Camp location
- Cost of camp
- How to register or get more information (email, phone, website)

It is a good idea to also include the following:

- Pictures of projects and campers (with consent from parents or guardians, or purchased from stock photography) , staff, location
- Experience, certification, or education of staff
- Licensing information may also be required by local authorities

Where to advertise:

- Schools
- Parent Teacher Organizations
- Childcare Organizations
- Parenting email lists or forums
- Word of mouth

Social Media:

Social media is a great tool to promote the camp brand by allowing students to share their excitement and experiences with others in their community.

- Campers can share pictures or videos of their robots in action.
- Consider a "theme" or suggested hash-tag for each day that campers can share with parents and friends via social media. (e.g., "#itsalive" to showcase their first autonomous program on the Clawbot).
- Investigate, define, and clearly communicate policies about what is acceptable to share.

When to advertise:

- For summer camps, it is a good idea to advertise in late winter or early spring.
- For workshops and camps throughout the school year, it is a good idea to advertise at least three months before the camp will run.

Other Considerations To Investigate:

- Incorporation and general set-up
- Licensing
- Liability and accident insurance
- Medical insurance
- Background/criminal/reference checks on all staff
- Safety, fire, medical, emergency plans
- Waivers and legal concerns

Suggested VEX® IQ Components

The VEX IQ Classroom Bundle is a great option to supply a group of 12 campers with their own kits, or approximately 24 campers with one kit for every two campers. For greater numbers, individual Super Kits can be purchased to reach the desired number.

Alternatively, campers can purchase their own kits in advance of the camp.

Kits and components are available at www.vexiq.com/robotics-camp-handbook.

VEX IQ Super Kit

(P/N: 228-2500)

The Super Kit is an all-encompassing introduction to STEM and robotics. Students can use the familiar hand-held Controller to drive robots right out of the box or program them to run autonomously using the additional Smart Sensors. While the included Clawbot IQ instructions help students easily build their first robot, the intuitive snap-together parts mean the sky is the limit for their VEX IQ creations.

- Over 850 Structural & Motion Components
- 4 Smart Motors, 7 Sensors, Robot Brain, Controller & Batteries Included
- Storage Bin & Tray included for organized storage of all parts

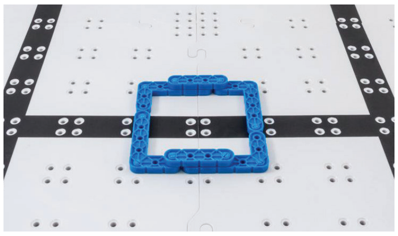
VEX IQ Classroom Bundle

(P/N: 228-4000)

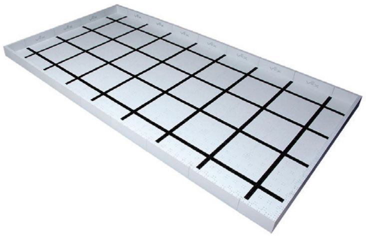
- 12x VEX IQ Super Kits
- 2x 3-Inch Cube Kits

Perfect for a group of 24 students (working in pairs), this discounted bundle contains everything needed to start building robots in your classroom, after school club, or summer STEM camp.

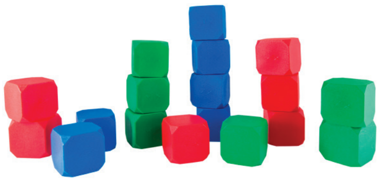
The Super Kit is the most complete introduction to VEX IQ; a set of 12 means that in just a few hours, your classroom could be crawling with 12 robots drivable by wireless control, running autonomously using advanced sensors, or a combination of both. Plus, when you use 3-Inch Cubes to set up a simple game, it fuels a competitive atmosphere and engages students right away.



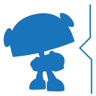
VEX IQ Cube-Base Kit
(P/N: 228-3452)
- 1x Red Set of Parts
- 1x Blue Set of Parts
- 1x Green Set of Parts



VEX IQ Challenge Full Field Perimeter & Tiles
(P/N: 228-2550)
The full field perimeter and tiles for the VEX IQ Challenge.
- Official IQ Challenge perimeter and tiles.
- Snap-together construction allows for assembly in minutes.



3-inch Cube kit
(P/N: 228-3325)
A half set of the cubes used in the 2014-2015 VEX IQ Challenge game, Highrise.
- 6x Red 3-inch Cubes
- 6x Blue 3-inch Cubes
- 6x Green 3-inch Cubes



Note: A full Highrise field uses (2) 3-inch Cube Kits.


Home Products Education Competition Software
Home / VEX IQ / Documents & Downloads
VEX IQ Documents & Downloads
Check back often as more updates are added!
VEX IQ Firmware Updates
- Visit the VEX IQ Firmware page for more information

User Guides and other Documents
- 228-3428-750 - Clawbot IQ Build Instructions
- 228-3427-750 - Control System User Guide
- 228-2500-753 - Super Kit Contents Poster
- 228-3060 - Starter Kit with Controller Contents Poster
- 228-3080 - Starter Kit with Sensors Contents Poster
- 228-2500 - Super Kit Pre-Summer 2014 User Guide (note: this is only applicable to pre-Summer 2014 Super Kits with maroon gears and pins)
- 228-2531 - Foundation Add-On Kit User Guide
- 228-3600 - Competition Add-On Kit User Guide
- 228-2550 - VEX IQ Challenge Field Specs
- VEX IQ Troubleshooting Flowcharts

Build Instructions

VEX IQ Documents and Downloads
www.vexiq.com/documents-downloads

VEX Robotics offers and extensive collection of resources online that are completely free of charge.
- Firmware Updates
- User Guides and other Documents
- Build Instructions
- Free VEX IQ Classroom Curriculum
- VEX IQ CAD Files
- A range of VEX IQ Videos
- VEX IQ - YouTube Playlist
  www.vexiq.com/videos

Sample Camp Schedule

	Monday	Tuesday	Wednesday	Thursday	Friday
8:00-8:20	- Welcome - Introductions - Expectations and routines	- Morning meeting - Prepare for day - Find materials	- Morning meeting - Prepare for day - Find materials	- Morning meeting - Prepare for day - Find materials	- Morning meeting - Prepare for day - Find materials
8:30-9:30	Block 1	Block 1	Block 1	Block 1	Block 1
9:30-9:50	Break/Snack/Outdoor or Team-Building Activity				
10:00-11:00	Block 2	Block 2	Block 2	Block 2	Block 2
11:00-12:00	Block 3	Block 3	Block 3	Block 3	Block 3
12:00-12:50	Lunch and Outdoor/Free Play Time				
1:00-2:00	Block 4	Block 4	Block 4	Block 4	Block 4
2:00-3:00	Block 5	Block 5	Block 5	Block 5	Block 5
3:00-3:20	Break/Snack/Outdoor or Team-Building Activity				
3:30-4:30	Block 5	Block 5	Block 5	Block 5	Block 5
4:30-4:50	Clean Up	Clean Up	Clean Up	Clean Up	Clean Up

Sample Team-Building Activities

The activities in this section have been included as a starting point to help campers to get up and moving and get to know each other a little better. They are a great way to start the day, or they can be worked into the schedule throughout the day when it seems that campers need a change of pace.

- "Two truths and a Lie" – Ask campers to think of three interesting facts about themselves to tell the group. Two of the facts should be true (but possibly surprising) and one of the facts should be false (but seem possible). Have the campers tell the group their facts and have the group guess which facts are true or false. This can be done verbally, or by writing the facts on small scraps of paper for a partner or leader to read if campers are shy.
- "Connected Campers" – Use a ball of yarn or string. Have campers stand in a circle and have one camper hold one end of the string. Have campers toss the ball of yarn or string to other campers at random, saying their names, (facts they've learned about their fellow campers) out loud to let that person know they should catch the string. Each camper should hold the string as they catch it and cannot let go. When the last camper has caught the string, give the campers the challenge of untangling themselves by climbing over or moving under the lengths of string that connect them.
- "I am a robot." – Ask campers to work in partners for this activity. Have one student act as a robot and the other two campers act as "drivers." Provide the campers with a basic task, such as picking up a pencil or taking a picture on a smart device. The robot can only do exactly as their partners dictate (literally). The operator must give precise, step-by-step instructions. (e.g., Lift your left arm 45 degrees. Take one, 24-inch step forward. Open your hand, then rotate your wrist 180 degrees so your palm is facing down.). This is also a great "pseudo code" activity to help campers understand the level of detailed thinking required to program effectively.
- "I say, you build." – Have campers work in pairs and have them sit back-to-back in a position where one camper can see an image or example of a simple model (created by the camp leader, or use the sample basket or tower model examples provided in the camp Learning Activities). One camper should have a copy of the kit documentation, VEX IQ Super Kits Contents and Build Tips poster and one camper should have access to a VEX IQ Super Kit. As one camper describes the components and process to build the simple model, the other camper follows and uses the kit to build. They can ask "yes" or "no" questions for clarification, but cannot show their partner what they are building. When the activity is done, campers can look at the example model to see how closely they were able to replicate it as a team. This is a great way to get to know the components in the VEX IQ Super Kit using accurate terminology.
- When campers have started to work in pairs for different challenges, it is a good idea to give them time to come up with a team name, motto, and if time allows, an original "high-five" or handshake combination and cheer. As campers move into different pairs for different challenges or work together on multi-team challenges, "high-fives", handshakes, and cheers can be combined and expanded, incorporating elements from each team involved.

Tips and Tricks for Camp Management and Organization

Set up

- Allot one kit per two campers. Assign campers to their kit and let them know they are responsible for their kits and all its components throughout the camp.
- Try to limit camp sections to 24 students maximum per instructor at most, with 12 VEX IQ Super Kits. It's also a great idea to have the smallest Camper to Instructor ratio possible. Additional Instructors or even student helpers who have some experience will keep work moving along while individuals are being helped.
- It is a good idea to have 1 or 2 extra VEX IQ Super Kits on hand for additional challenges, to replenish parts, or in case a camper is very keen to work on their own.
- Make sure kits are complete and all components are functional before the start of each camp.
- Make sure batteries are charged and ready before the start of each day.
- Provide a large, flat open work surface for campers (e.g., a 6-foot work desk or a group of smaller desks pushed together).
- Have extra pens, pencils, tape measures, etc. on hand.
- For the first day or two, it is a good idea to provide campers with name tags so they can get to know each other.

Rules and Routines:

- Keep it simple (e.g., Be safe. Be respectful. If you make a mess, clean it up. Stay in the areas of location that have been designated for the camp.)
- Hold a morning meeting every day to set expectations and let campers know what they'll be doing that day.
- Factor in 10-15 minutes of transition time between activities and breaks.
- Factor in 20-30 minutes of clean up time every day.

Organization during Hands-On Activities

- VEX IQ is primarily designed for participants ages 8-14. Campers in this age range can be quite diverse in maturity and ability, so it is a good idea to consider how you would like to group them together into camp sections or pairs within a section.
- Begin every activity with a quick reminder of the task and any specific expectations.
- Circulate often and remind campers that they are responsible for their own kits.
- Reinforce that an organized kit and workstation makes it much easier to find the components you need when you need them.
- Keep a container or bucket by the door for spare components that end up on the floor at the end of the day. These can be accessed throughout the day as needed, or distributed to make sure kits are complete at the end of camp.
- Give campers notice when an activity block has 30, 15, and 5 minutes left to help them keep their eye on the clock.

Keep Parents and Guardians Informed

- Make sure camp documentation includes clear instructions for parent or guardian parking, pick up, and drop off.
- Encourage campers to take photos of their work (challenges and successes!) and share them with their parents.
- Be aware of privacy concerns when it comes to photos or videos being taken of minors. Obtain parent consent before distributing or posting any images of campers or focus documentation on the robots and programs they create.

Camp Overview

Camp Overview

VEX IQ Robotics Camp Learning Activities

The following activities range in length from one (1) to two (2) hours. Extension ideas have been included for each activity to support campers who are ready for an additional challenge.

Learning Activities can be conducted one after the other or mixed and matched to tailor camp programs to requirements for timing or interest.

Three suggested pathways through the activities have been provided, including challenges and drop-in real world extension ideas to tailor the experience to your campers.

1-day camp, 3-day camp, and a 5-day camp outlines have been provide on the following pages.

Modification Examples:

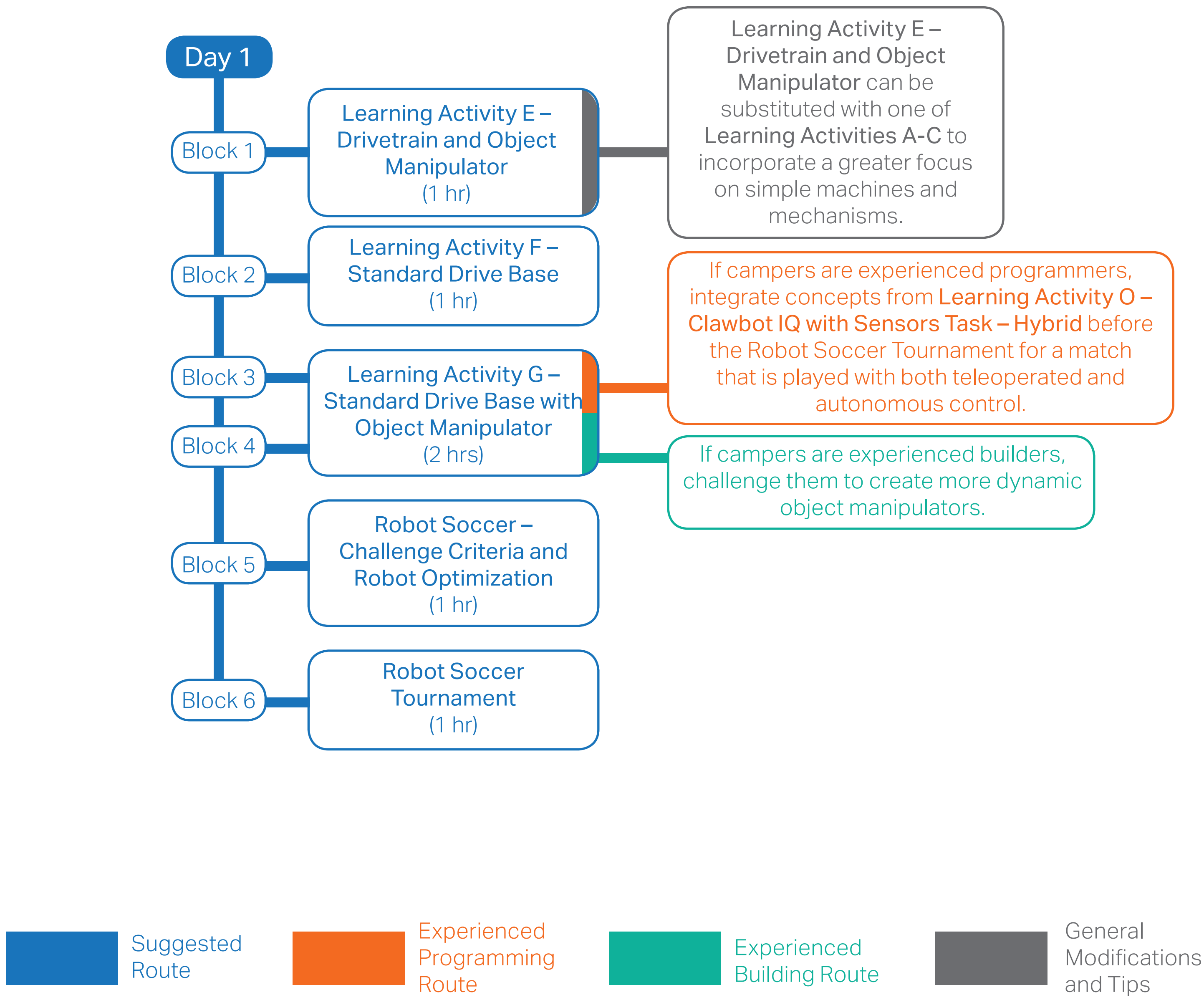
- Learning Activities A-D can be dropped individually into ongoing camp programs as morning "warm up" or "brain break" activities throughout the day.
- In Learning Activities I-L, campers may wish to select particular sensors to work with or "divide and conquer" with teammates to investigate different functions.

1-Day Camp Objectives:

Campers will use VEX IQ components to experiment with essential structures and basic object manipulators. They will build a Standard Drive Base that functions in teleoperated mode (remote control) and add original object manipulator assemblies to participate in a team-based competitive challenge (**Robot Soccer**).

This stream provides a very basic introduction to robotics and teleoperated control. It is a great option for a workshop or one day "robotics in the classroom" event.

> The timing suggested in this 1-Day camp schedule is approximate and may vary based on camper maturity or experience levels, as well as the experience level of staff or the number of camp staff available to support the campers.
>
> In general, more open-ended activities work well when there is a higher staff to camper ratio, while more structured activities will support campers who need a little more guidance when staff numbers or experience levels are lower.
>
> If time or resources are limited, it may be a good idea to focus on building the Clawbot IQ and completing the Robot Soccer Challenge without an in-depth investigation of basic structures and manipulators.



3-Day Camp Objectives:

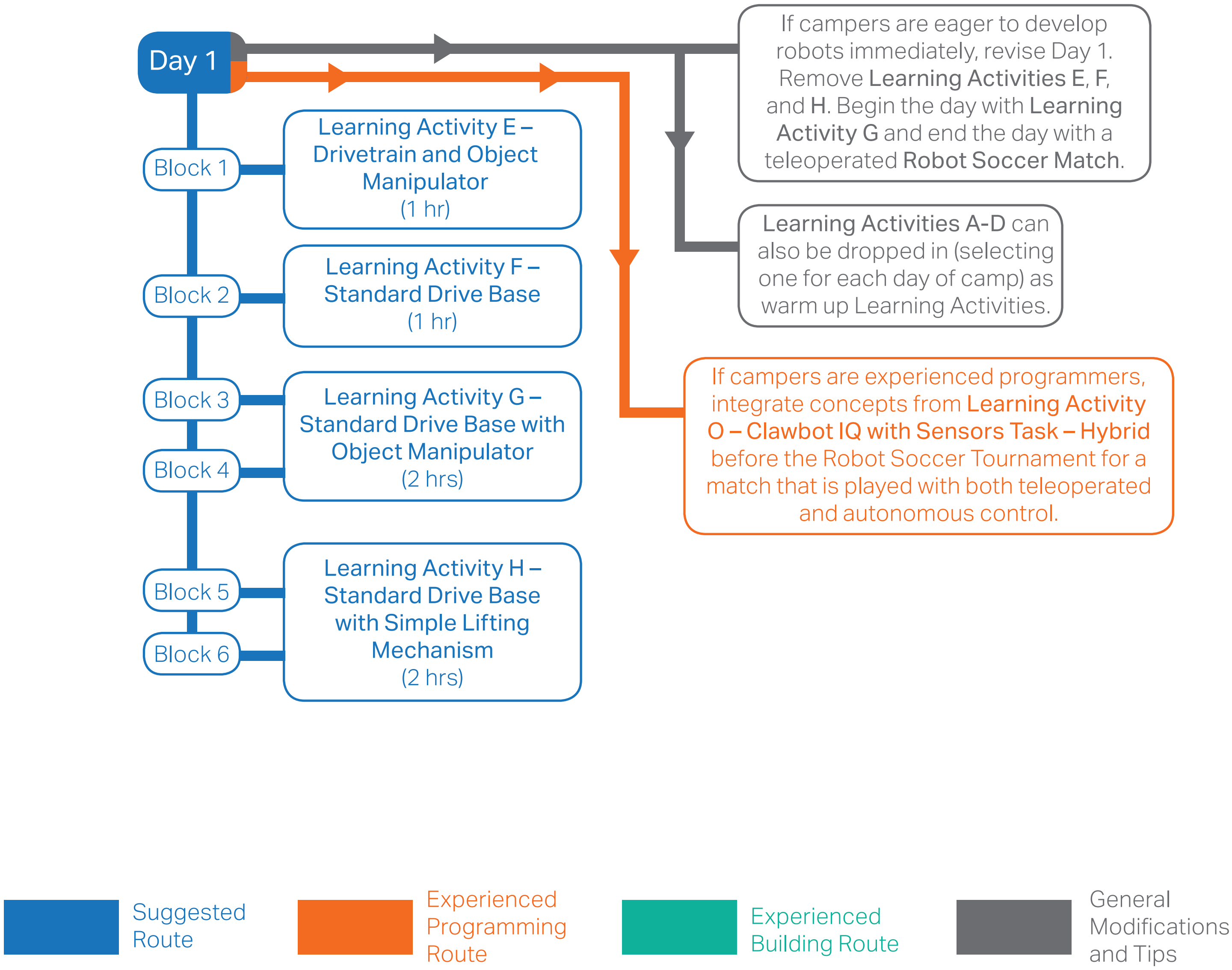
As in the 1-day Camp, campers will use VEX IQ components to build essential structures and object manipulators. They will also construct the Standard Drive Base, but will complete day one with a challenge to build a lifting mechanism for their robot.

In the following days, campers will experiment with pre-programmed sensor functions and gain experience operating a robot using both a controller and simple autonomous programming. The camp will end with robot optimization to participate in a teleoperated or hybrid (teleoperated and autonomous) version of the Highrise challenge (HIghrise Teamwork or Robot Skills Competition).

The timing suggested in this 3-Day camp schedule is approximate and may vary based on camper maturity or experience levels, as well as the experience level of staff or the number of camp staff available to support the campers.

In general, more open-ended activities work well when there is a higher staff to camper ratio, while more structured activities will support campers who need a little more guidance when staff numbers or experience levels are lower.

If time or resources are limited, it may be a good idea to focus on two activities each day using the Clawbot IQ or Clawbot IQ with Sensors.



3-Day Camp Objectives (cont.)

Day 2

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

- Block 1, Block 2: Learning Activity I – Autopilot Robot (2 hrs)
- Block 3, Block 4: Learning Activity M - Clawbot IQ (2 hrs)
- Block 5, Block 6: Learning Activity N – Clawbot IQ with Sensors Task – Autonomous (2 hrs)

Learning Activity M has been included as an opportunity to complete an in-depth investigation of prototyping and building in Engineering Design. Depending on camper interest, this can be extended for additional testing and improvement iterations. It may also be placed after Learning Activities N and O as a final step in optimizing the robot design before completing the Highrise Challenge.

Day 3

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

- Block 1, Block 2: Learning Activity I – Autopilot Robot (2 hrs)
- Block 3, Block 4: Learning Activity M - Clawbot IQ (2 hrs)
- Block 5, Block 6: Learning Activity N – Clawbot IQ with Sensors Task – Autonomous (2 hrs)

Note: Campers may complete the Highrise Challenge using the Clawbot IQ, Clawbot IQ with Sensors, or with a custom-created robot. If custom robots are used, additional building and programming time should be allotted.

Suggested Route

Experienced Programming Route

Experienced Building Route

General Modifications and Tips

5-Day Camp Objectives:

Campers will spend time building and modifying simple machines with a goal of combining or automating them to complete a Chain Reaction Challenge. With that foundation, they will move on to build a basic robot that functions in teleoperated mode (remote control) and add original object manipulator assemblies to participate in a team-based competitive challenge (**Robot Soccer**).

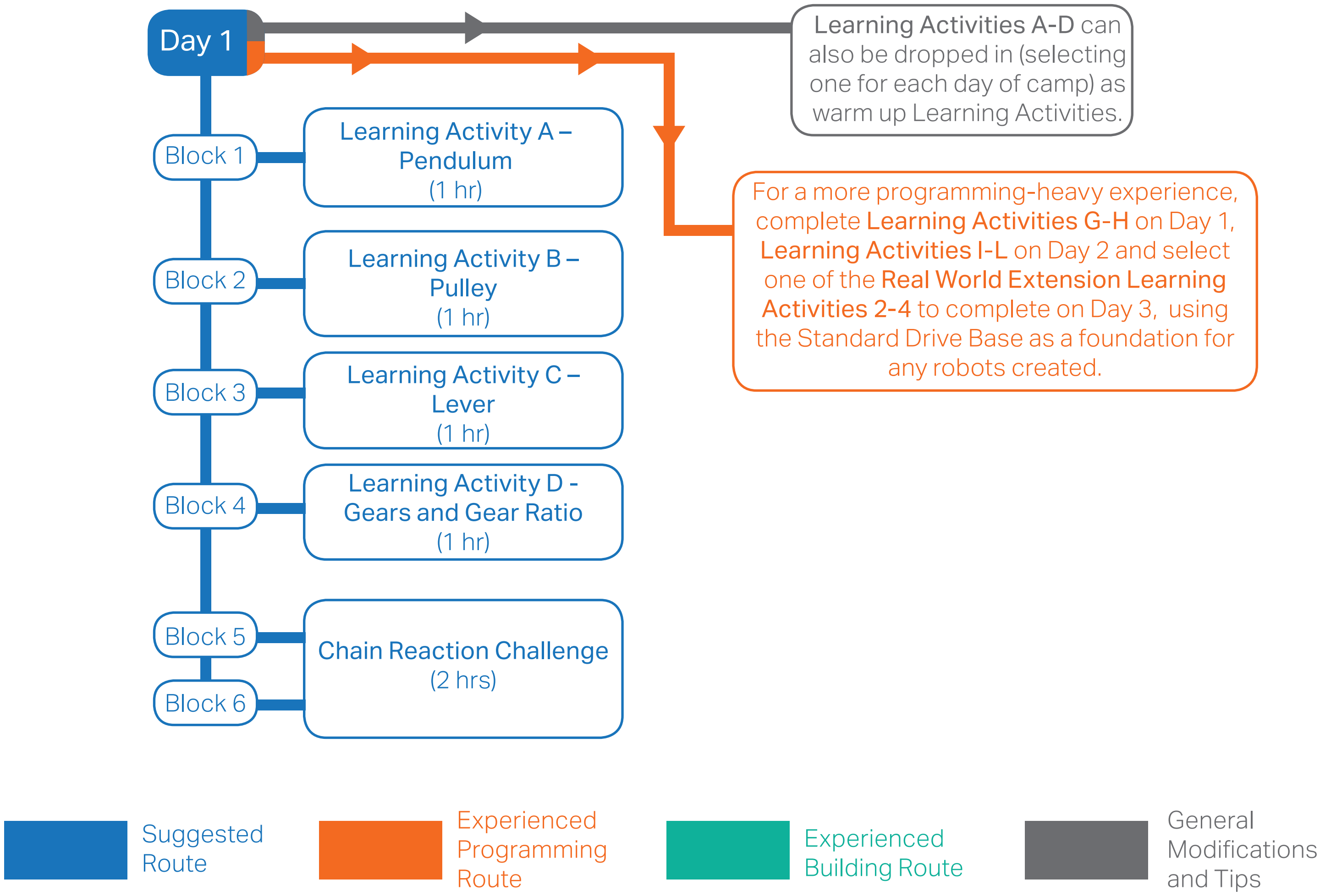
In the following days, campers will experiment with pre-programmed sensor functions and then complete individual programming challenges to customize sensor functions. They will then build and operate the full Clawbot IQ with Sensors and experiment with both a controller and autonomous programming to control it.

On their final day of camp, students will focus on the **Highrise Challenge**, completing a teleoperated mode (remote control) version of the challenge in the morning, then optimizing their robot and programming throughout the day to compete in an autonomous mode **Highrise Programming Challenge** at the end of their camp experience.

The timing suggested in this 5-Day camp schedule is approximate and may vary based on camper maturity or experience levels, as well as the experience level of staff or the number of camp staff available to support the campers.

In general, more open-ended activities work well when there is a higher staff to camper ratio, while more structured activities will support campers who need a little more guidance when staff numbers or experience levels are lower.

If time or resources are limited, it may be a good idea to focus on two activities each day using the Clawbot IQ or Clawbot IQ with Sensors. As time allows,



5-Day Camp Objectives (cont.)

Day 2

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

- Block 1, Block 2: Learning Activity G – Standard Drive Base with Object Manipulator (2 hrs)
- Block 3, Block 4: Learning Activity H – Standard Drive Base with Lifting Mechanism (2 hrs)
- Block 5: Robot Soccer – Challenge Criteria and Robot Optimization (1 hr)
- Block 6: Robot Soccer Tournament (1 hr)

Day 3

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

- Block 1, Block 2: Learning Activity I – Autopilot Robot (2 hrs)

Learning Activity I has been placed at the beginning of the day on Day 3 so that campers have a chance to experiment with a robot that "just works" before a deep dive into sensors and programming. This order can also be reversed, with **Learning Activity I** being completed at the end of the day on Day 3.

- Block 3, Block 4, Block 5, Block 6: Choose two of:
  - Learning Activity J – Distance Sensor Default and Programming Task (2 hrs)
  - Learning Activity K – Color Sensor Default and Programming Task (2 hrs)
  - Learning Activity L – Gyro Sensor Default and Programming Task (2 hrs)

Some campers who are less competition-focused may enjoy creating a robot that is more humanoid. If this is the case, on Day 3, begin with **Learning Activities J-L** and end with **Real World Extension #1: Interactive Robot.**

Suggested Route | Experienced Programming Route | Experienced Building Route | General Modifications and Tips

Day 4

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

- Block 1, Block 2: Learning Activity M - Clawbot IQ (2 hrs)
- Block 3, Block 4: Learning Activity N – Clawbot IQ with Sensors Task – Autonomous (2 hrs)
- Block 5, Block 6: Learning Activity O – Clawbot IQ with Sensors Task – Hybrid (2 hrs)

Learning Activity M has been included as an opportunity to complete an in-depth investigation of prototyping and building in Engineering Design. Depending on camper interest, this can be extended for additional testing and improvement iterations. It may also be placed after Learning Activities N and O as a final step in optimizing the robot design before completing the Highrise Challenge.

Day 5

Learning Activities A-D can also be dropped in (selecting one for each day of camp) as warm up Learning Activities.

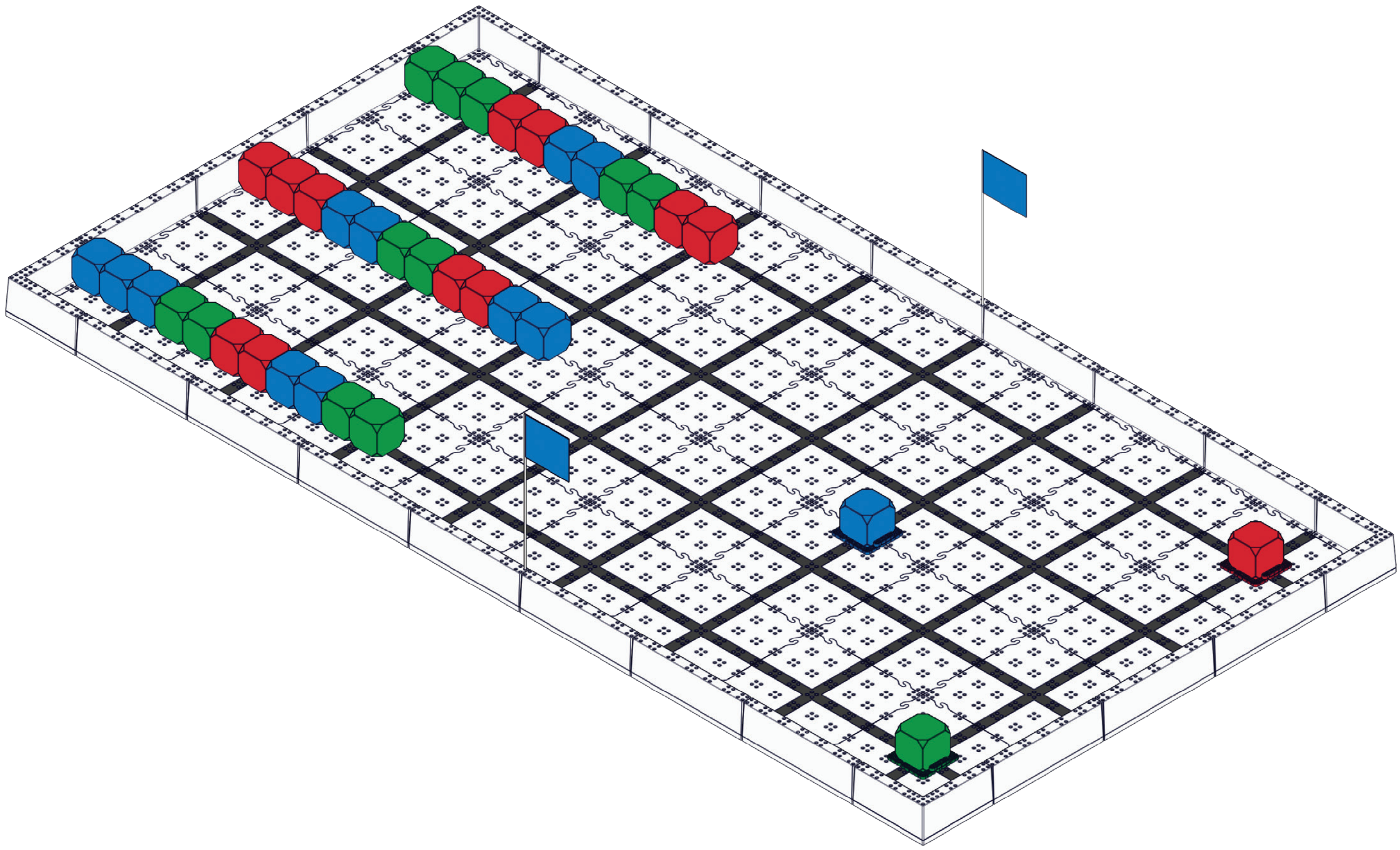
- Block 1, Block 2: Highrise Challenge – Exhibition Games (2 hrs)
- Block 3, Block 4: Highrise Programming Challenge – Criteria and Robot Optimization (2 hrs)
- Block 5, Block 6: Highrise Programming Challenge (2 hrs)

Note: Campers may complete the Highrise Challenge using the Clawbot IQ, Clawbot IQ with Sensors, or with a custom-created robot. If custom robots are used, additional building and programming time should be allotted.

If campers have a strong interest in applied Engineering Design, incorporate one of the open-ended Real-World Extension Activities into the last day of the camp experience. Allow campers or groups to choose their challenge from the options provided."

Suggested Route | Experienced Programming Route | Experienced Building Route | General Modifications and Tips

Open-Ended Challenges



Open-Ended Challenges

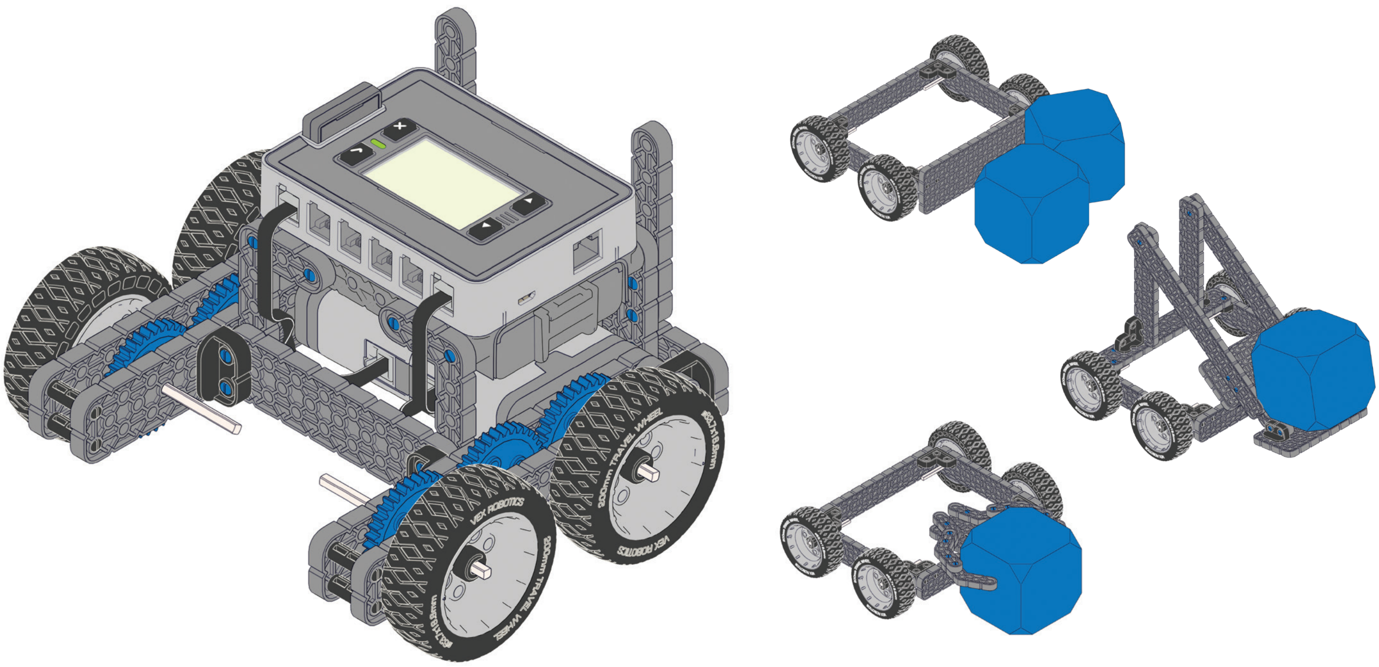
1. Robot Soccer Tournament

(1 hour + 1 hour Tournament Description and Robot Optimization)

Suggested Robot: Standard Drive Base with Object Manipulator

Number of Robots: 4 working in teams of 2

Number of Drivers: 8 total with 4 driving at a time



The Tournament Description:

The Robot Soccer Tournament is played on the 4x8' Highrise Challenge Field. One team (of two robots working collaboratively) compete against another team to score goals.

The object of the game is to be the first team to push two Highrise Cubes from the center of the Challenge Field, into an opposing scoring zone.

The Details:

There are a total of four (4) Highrise Cubes (of any color) placed in a row at the center of the field. There are two scoring zones, one at each end of the Challenge Field that are 6 inches wide and created from Highrise Cube-Base Kit components. The scoring zone should back onto the edge of the field and be open on the opposite side to allow a Highrise Cube to be pushed through the goal, into the scoring zone.

Each robot begins the match on one side of the scoring zone at the opposite end of the game field from the two robots on the opposing team. Robot Soccer matches are one (1) minute long, with drivers on a given team, switching at the 30 second mark. The match is over when two Highrise Cubes have been scored in one goal, or when time runs out.

Encourage campers to choose team names and develop team cheers or logos. The tournament can be played in assigned rounds for points, in random matches with time to revise robots and strategy between matches, or as an organized elimination-based series.

Robot Optimization:

Campers should consider the following when determining their game strategy:

- How will your two teams work together?
- Will one robot play offense as the other plays defense?
- Is one robot design better suited to one role or the other?
- How will changing drivers part way through the match affect your strategy or practice time?

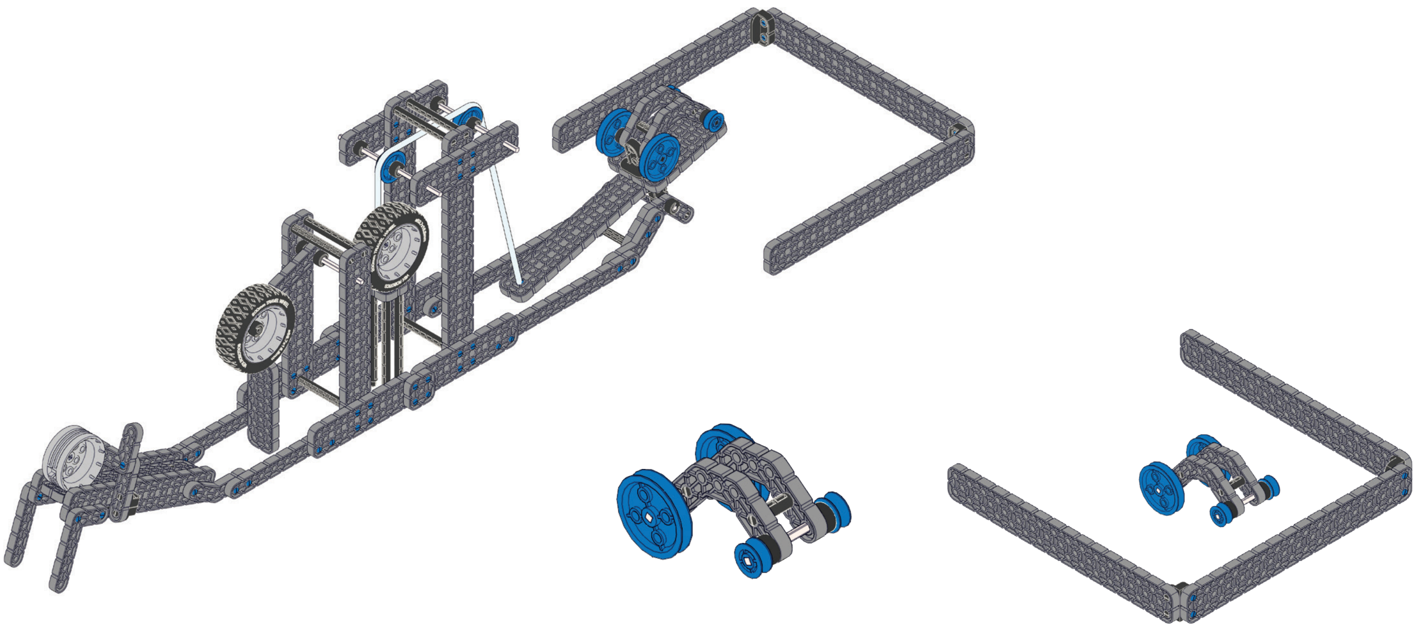
Campers should test their robots on the game field (or a similar set up) to practice their driving skills, collaboration with team members and optimize their robot design as needed for the following:

- Optimize the size of the robot for the confines of the game field with obstacles and three other robots.
- Revised object manipulators for scoring objects or revise robot design for defensive tasks.
- Revise the design of both robots for more effective collaboration.

2. Chain Reaction Challenge

(1 hour + 1 hour Challenge Description and Device Design)

Suggested Device: Combination of components and mechanisms from Learning Activities A, B, C and/or D.



The Challenge Description:

The Change Reaction Challenge is completed by combining a series of simple machines and/or assemblies to create Chain Reaction Devices. Each individual simple machine or assembly is known as a "stage" of the overall device. Campers should aim to include between two (2) and four (4) stages in their device, as well as a Trigger Mechanism to activate the operation/chain reaction of the device.

The goal is for teams of two campers to create a device that will accomplish a basic task in an interesting "hands-free" way. The object of the challenge is to "park" a small car made from VEX IQ components in a designated space. Encourage campers to give their device a name and be prepared to explain how it works.

Encourage campers to focus on one or two stages to start and make sure they are aware of time limitations. Additional stages can be added later. For younger campers, it may be a good idea to begin with just a triggering mechanism and one (1) stage to park the car. They can then share their devices with other campers and incorporate additional stages or combine devices as time allows.

The Details:

This challenge may be completed as a dynamic, non-motorized machine, or with the addition of motors and sensors. Examples of each option can be found in the VEX IQ Robot Curriculum.

Detailed Chain Reaction Challenge Rules from the VEX IQ Curriculum are available at:
http://www.vexrobotics.com/vexiq/education/chain-reaction-challenge/

Device Optimization:

Campers should consider the following when designing their devices:

- What is the objective of the device?
- How does the car move? What are the limitations to its movement?
- How does the shape and size of the parking space compare with the size and shape of the car?
- What simple machines or mechanisms are you familiar with? Which one is best suited to move the car into the parking space?
- Is one of these mechanisms well-suited to creating a triggering mechanism?
- How will the mechanisms used for each stage connect together?

Campers should test and optimize their device design as needed for the following:

- Make sure that the car is consistently parked in its space.
- Make sure that the device is stable and does not fall over or fall apart when it is triggered.
- Revise the design for a greater "wow" factor.

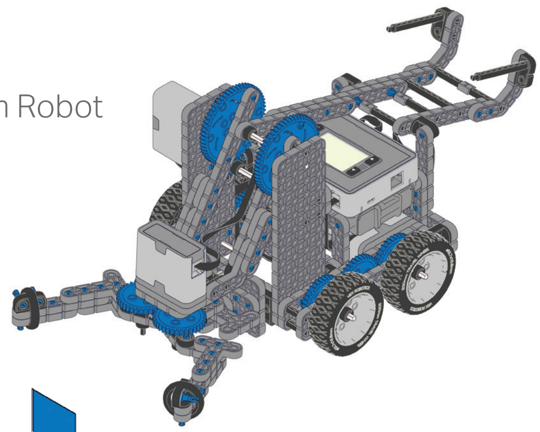
3. VEX IQ Camp Highrise Challenge

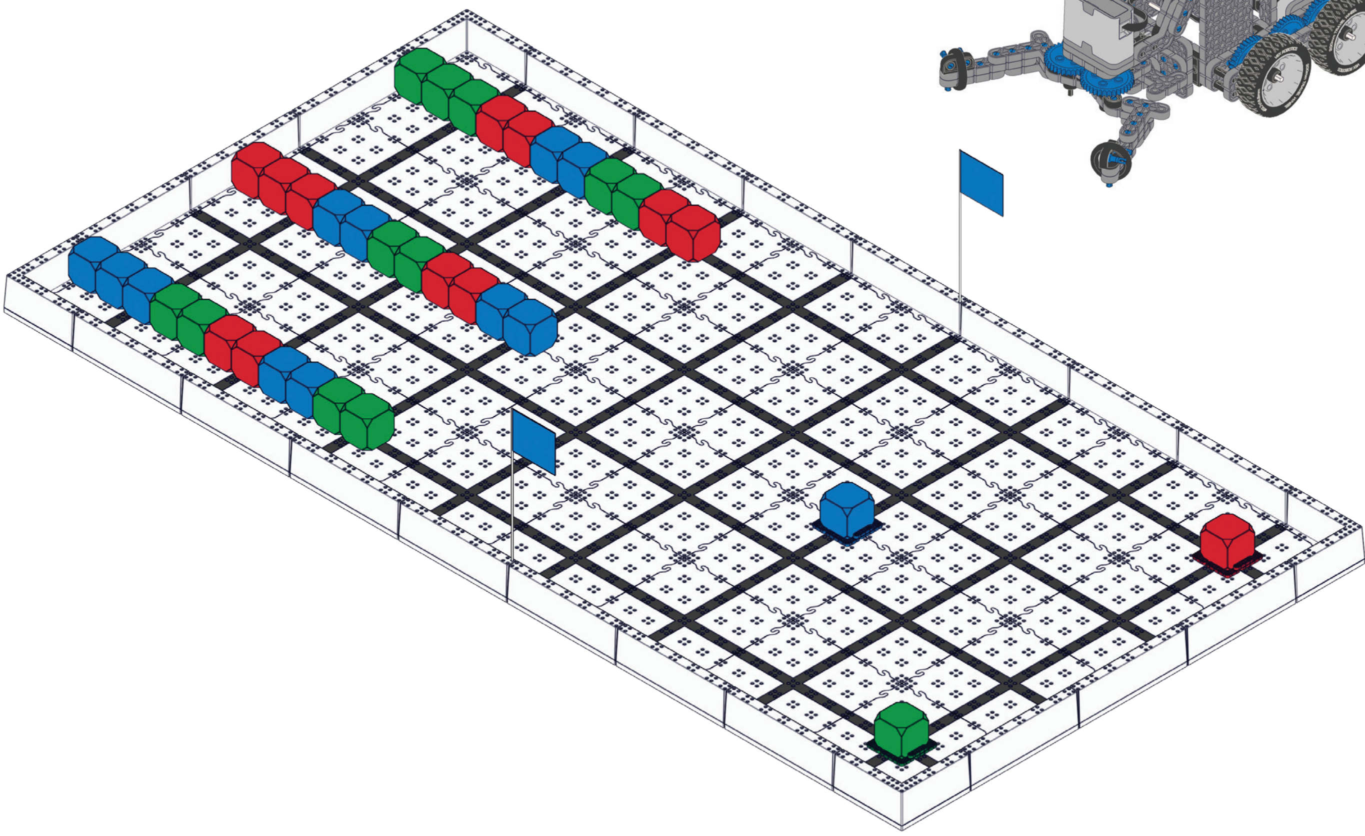
(2 hours + 2 hours Challenge Description and Robot Optimization)

Suggested Robot: Clawbot IQ, Clawbot IQ with Sensors, or Custom Robot

Number of Robots: 2

Number of Drivers: 4 drivers working in teams of 2





The Challenge Description:

VEX IQ Camp Highrise Challenge is played on a 4'x8' rectangular field configured as seen above. Each team has two Drivers that operate their team's Robot with the VEX IQ Controller, one at a time.

Campers will play two matches for the challenge. The Drivers will switch their Controllers between :25 and :35 remaining in the Match. Scores from the two matches will be combined to determine the winners.

Match 1: Teamwork Challenge

Two teams will collaborate and work together in a 60-second match to maximize their score. Teams will move Highrise Cubes to the Scoring Zone and build single-color Highrises that match the color of each Highrise Base Cube. Both teams receive the same total points at the end of each Teamwork Challenge Match.

Match 2: Robot Skills Challenge

Two teams will compete to score as many points as possible for their own team in a 60-second Match. Teams will move Highrise Cubes to the Scoring Zone and build single-color Highrises that match the color of each Highrise Base Cube. Each team will receive a score based on their individual efforts.

The Details:

Campers may complete this challenge using the Clawbot IQ, Clawbot IQ with Sensors, or with a custom-created robot. If custom robots are used, additional building and programming time should be allotted.

Detailed descriptions of **Field Set-Up**, **Robot Operation**, and **Scoring**, as well as guidelines for **Robot Inspection**, **Sizing**, and **Construction**, are available for download. For game manual downloads and other information please visit www.vexiq.com/Highrise.

- **VEX IQ Challenge Highrise Game Manual** – Teamwork Challenge Matches, and Robot Skills Challenge Matches
  http://link.vex.com/Highrise-Game-Manual
- **VEX IQ Challenge Highrise Game**
  http://link.vex.com/Highrise-Game-Appendix
- **VEX IQ Challenge Highrise Field Appendix**
  http://link.vex.com/Highrise-Field-Appendix

  A quick and easy app for scoring is available for free:
  - Google Play Store - VEX IQ Highrise by DWAB Technology
  - iTunes App Store - VEX IQ Highrise

Robot Optimization:

Campers should consider the following when determining their game strategy:

- Do you understand the scoring system and rules of the game? Ask questions!
- Are some colors of Highrise Cubes more accessible than others?
- How does the location of the Highrise Cubes affect the order in which you should try to collect them?
- Are there advantages (e.g., clearing navigation space, points) to focusing on one color or another?
- How can you reduce the number of trips that are needed to gather and stack Highrise Cubes? Is there a way to transport more than one Highrise Cube at once?
- How can you work effectively with your partner team?
- How might you and your partner team interfere with or support each other's goals for the challenge?
- How can you maximize the number of points gained in each match? Run through different scenarios and consider possible outcomes.

Campers should test their robots on the game field (or a similar set up) to practice their driving skills, collaboration with team members and optimize their robot design as needed for the following:

- Does your robot meet the design specifications in terms of size and function?
- Does your robot design allow you to stack Cubes easily?
- Which tasks are the most time consuming or difficult to complete?
- Are there any maneuvers or tasks that your robot completes very efficiently or effectively? How can these strengths be worked into your game strategy?
- Does your robot interfere with the Highrise Cubes, stacks, or other robots on the Challenge Field?

General Tips:

- Start with a simple robot and add functionality as you test and evaluate your robot's performance. This is true for programming as well. Decide what your team's and robot's strengths are and focus on them when developing strategies for game play.
- Use online resources and forums to search for ideas and strategies. (www.vexiqforum.com)
- Download the free app and test out different strategies.

  Google Play Store - VEX IQ Highrise by DWAB Technology

  iTunes App Store - VEX IQ Highrise
- Document your successes AND your failures as you test and revise your robot. These will help you with future designs. Good documentation will also help your team to quickly recall what worked well (or did not) as you revise your robot's program and build. Keeping good notes (written, pictures, videos) will save time in the long run.

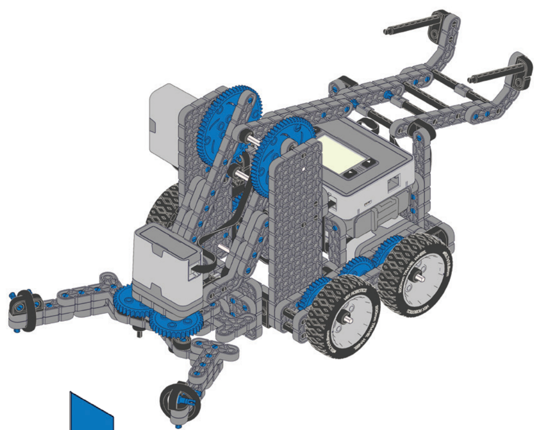
4. VEX IQ Camp Highrise Programming Challenge

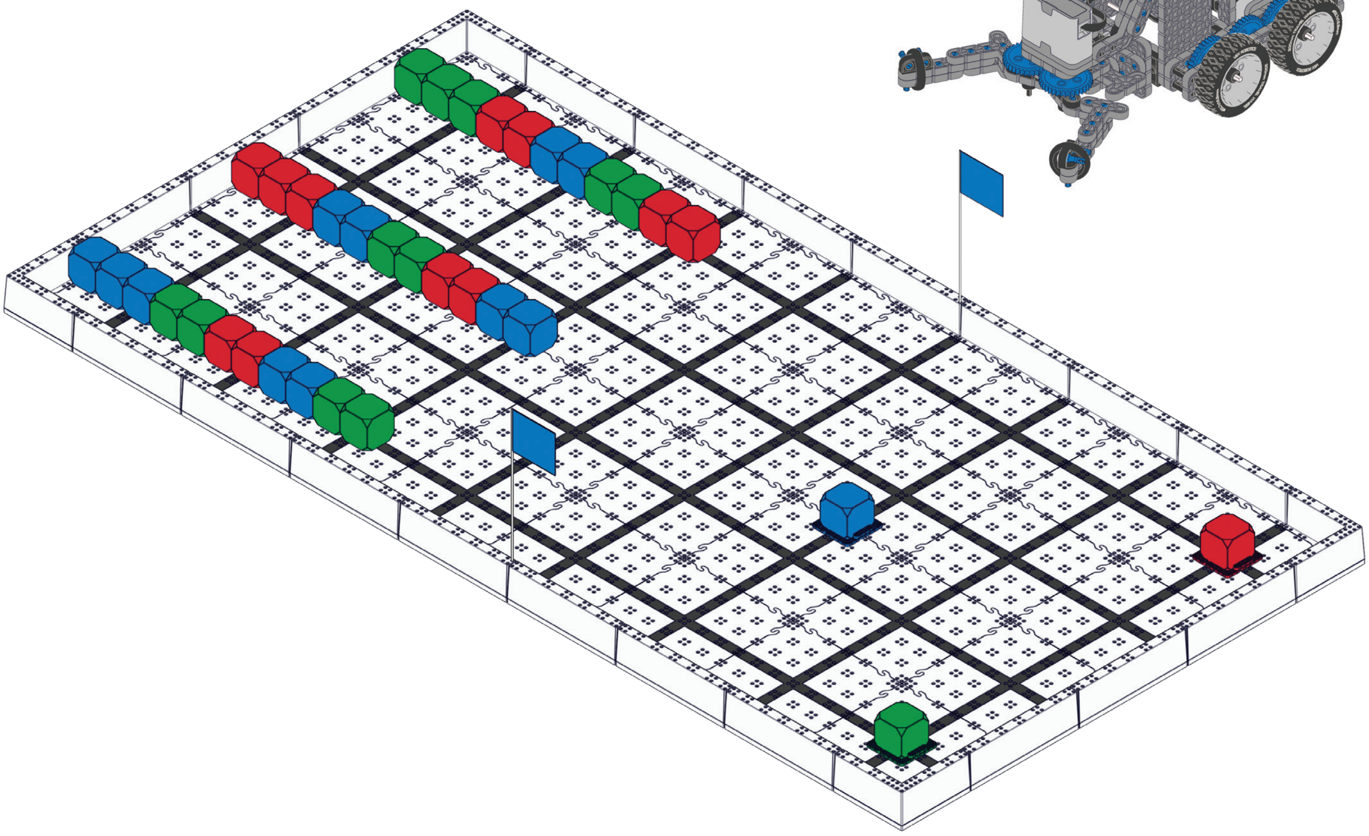
(2 hours + 2 hours Challenge Description and Robot Optimization)

Suggested Robot: Clawbot IQ with Sensors or Custom Robot

Number of Robots: 2

Number of Drivers: 4 drivers working in teams of 2





The Challenge Description:

VEX IQ Camp Highrise Challenge is played on a 4'x8' rectangular field configured as seen above.

Campers will play up to three matches with time to revise their robot or programming in between matches. The Drivers switch their Controller between :25 and :35 remaining in the Match.

Programming Skills Challenge Matches

Two teams will compete to score as many points as possible using only sensors and preprogrammed instructions stored in the VEX IQ Robot Brain. Robots move Highrise Cubes to the Scoring Zone and build mixed-color and/or single-color Highrises upon Highrise Base Cubes. Each team will receive a score based on their individual efforts.

The Details:

Campers may complete this challenge using the Clawbot IQ, Clawbot IQ with Sensors, or with a custom-created robot. If custom robots are used, additional building and programming time should be allotted.

Detailed descriptions of **Field Set-Up**, **Robot Operation**, and **Scoring**, as well as guidelines for **Robot Inspection, Sizing**, and **Construction**, are available for download. For game manual downloads and other information please visit www.vexiq.com/Highrise.

- **VEX IQ Challenge Highrise Game Manual** – Programming Skills Challenge Matches
  http://link.vex.com/Highrise-Game-Manual
- **VEX IQ Challenge Highrise Game Appendix**
  http://link.vex.com/Highrise-Game-Appendix
- **VEX IQ Challenge Highrise Field Appendix**
  http://link.vex.com/Highrise-Field-Appendix

Robot Optimization:

- Do you understand the scoring system and rules of the game? Ask questions!
- Are some colors of Highrise Cubes more accessible than others?
- The field is set up in the same way for every Match. How can you use this to your advantage?
- Is your robot well-suited to more offensive or defensive strategies? How can you use this to your advantage?
- Look for the weaknesses in your robot or programming design. How will competing teams be able to work these to their advantage? Consider what another team might look for in your design or strategy and practice solutions on the Challenge Field.
- What kinds of programs will help you get from the starting position to the Cubes most effectively? Test a few different programs to see which programs yield the most accurate and consistent results.
- How does the location of the Highrise Cubes affect the order in which you should try to collect them?
- Are there advantages (e.g., clearing navigation space, points) to focusing on one color or another?
- How can you reduce the number of trips that are needed to gather and stack Highrise Cubes? Is there a way to transport more than one Highrise Cube at once?
- How can you work with your partner team?
- How might you and your partner team interfere with or support each other's goals for the challenge?
- How can you maximize the number of points gained in each match? Run through different scenarios and consider possible outcomes.
- How will the requirement to change drivers part way through the match affect your team's strategy?

Campers should test their robots on the game field (or a similar set up) to practice their driving skills, collaboration with team members and optimize their robot design as needed for the following:

- Does your robot meet the design specifications in terms of size and function?
- Does your robot design allow you to stack Cubes easily?
- Which tasks are the most time consuming or difficult to complete?
- Are there any maneuvers or tasks that your robot completes efficiently and effectively? How can these strengths be worked into your game strategy?
- Does your robot interfere with the Highrise Cubes, stacks, or other robots on the Challenge Field? Are there any competitive advantages to this?

General Tips:

- Start with a simple robot and add functionality as you test and evaluate your robot's performance. This is true for programming as well. Decide what your team's and robot's strengths are and focus on them where possible.
- Use online resources and forums to search for ideas and strategies. (www.vexiqforum.com)
- Download the free app and test out different strategies.
  - Google Play Store - VEX IQ Highrise by DWAB Technology
  - iTunes App Store - VEX IQ Highrise
- Take advantage of exhibition matches and practice time. Take notes, videos, or discuss what happened as soon as possible, making sure to talk about what went well and what did not in a constructive way.
- Document your successes AND your failures as you test and revise your robot. These will help you with future designs or to remember what worked well (or did not) as you revise your robot's program and build. Keeping good notes (written, pictures, videos) will save time in the long run.

Sample Programming for Modkit for VEX Robotics

VEX IQ Camp Highrise Programming Challenge

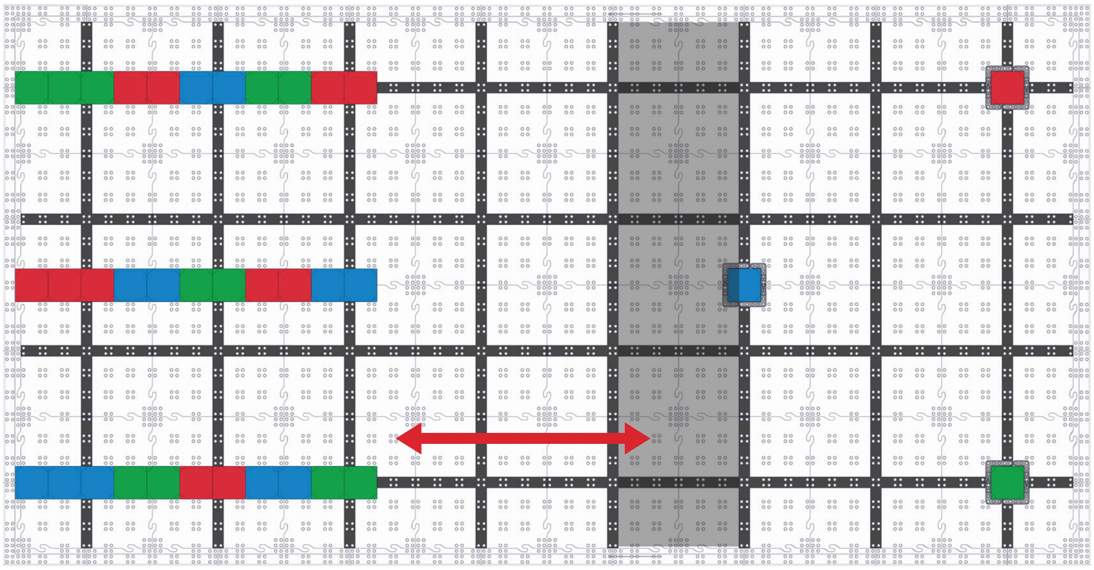
The programs shown in this section can be used as a starting point for exploring programming strategies using the Clawbot IQ with Sensors to complete the Highrise Programming Challenge. Please note that the sample programs can also be downloaded from www.vexiq.com/robotics-camp-handbook.

There are countless solutions to effectively complete the required tasks. These programs have been broken down into three basic tasks that could be helpful to complete the Challenge:

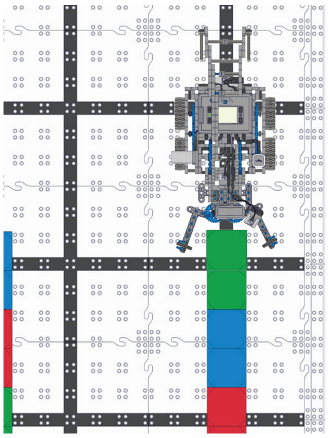
- Task 1: Navigate between the starting position and the Highrise Cubes.
- Task 2: Pick up Highrise Cubes.
- Task 3: Retrieve and stack Highrise Cubes from the Holder.

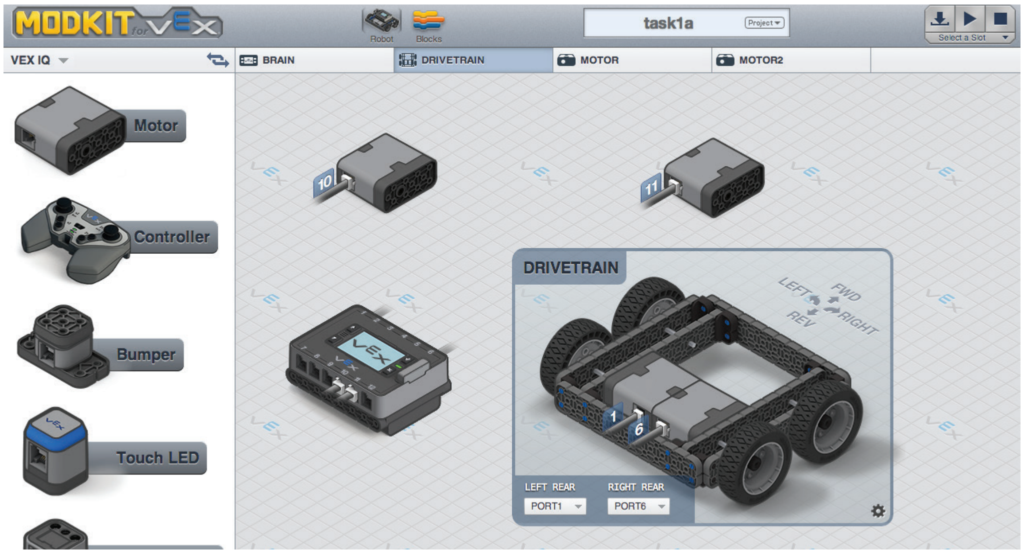
Summary of Program for Task 1 – Navigate between the starting position and the Highrise cubes.

This program is a basic example of a solution to help campers complete one part of the challenge. In this example, the robot moves between the green Highrise Cubes and the starting position.



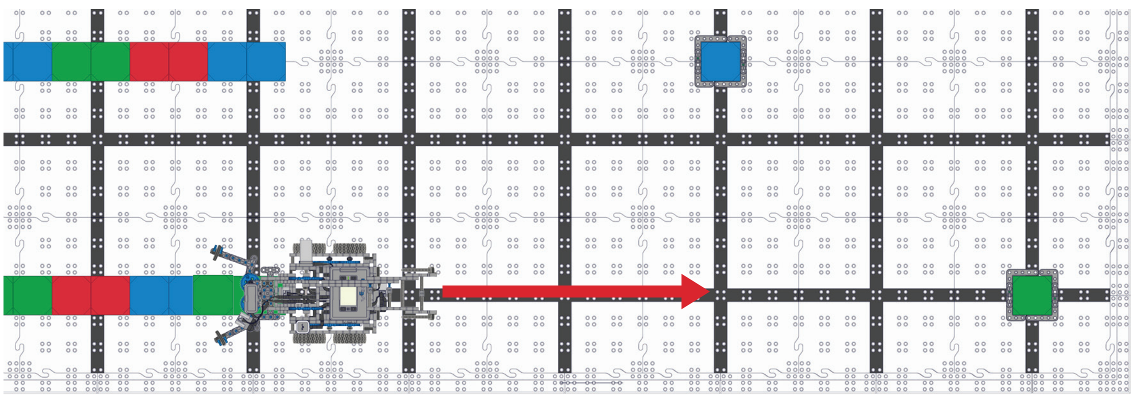
Task 1A: In this scenario, the robot is positioned facing the front green Highrise Cube with the claw open. The first green Cube is inside the open claw.





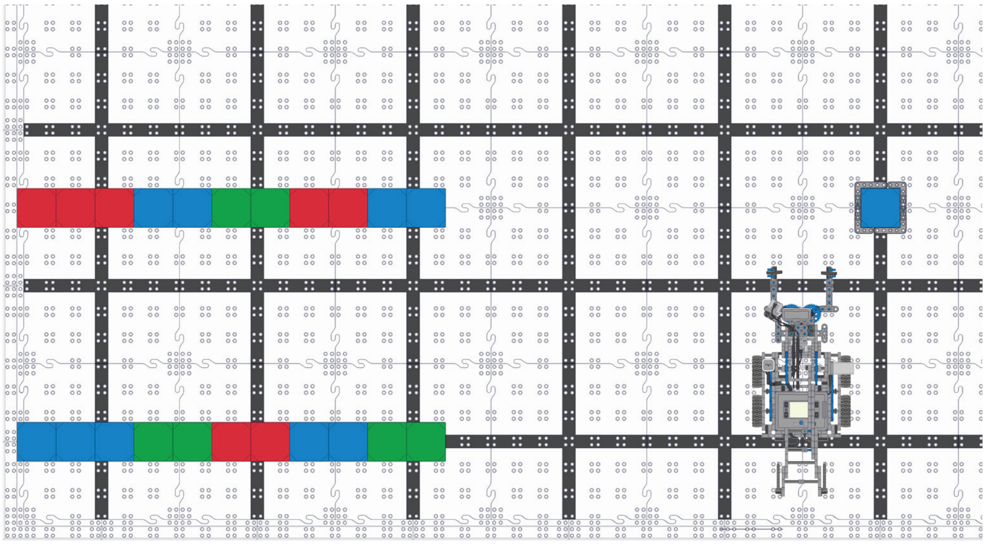
STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.

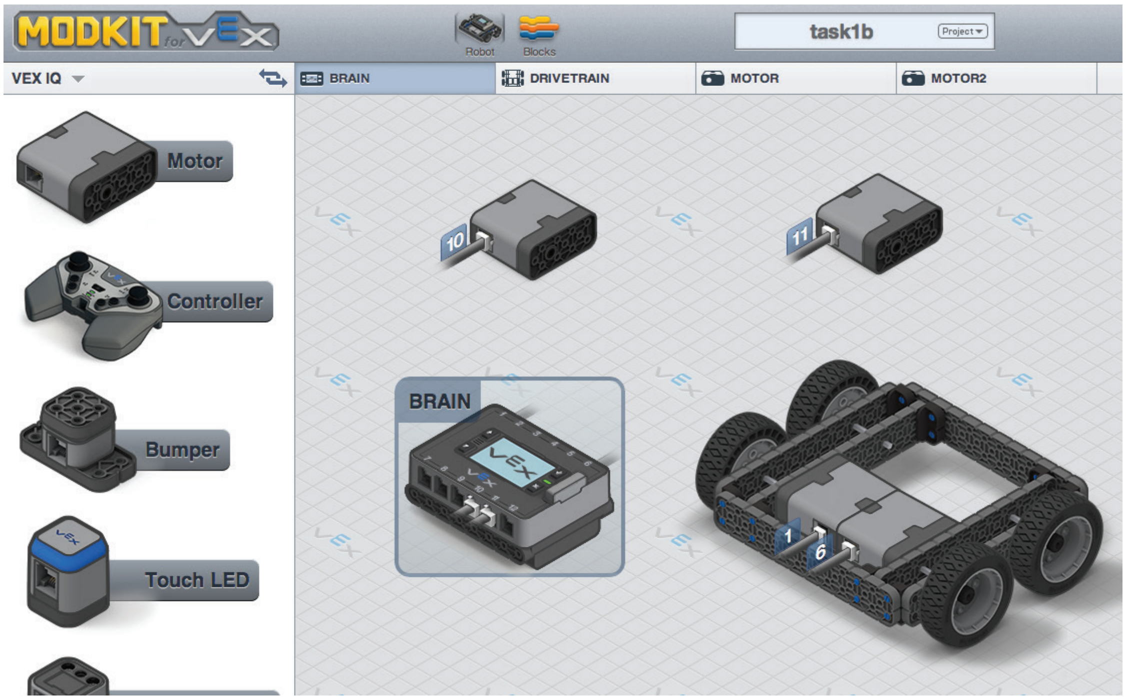
STEP 2: Select BLOCKS view and the DRIVETRAIN tab. Create and save this program. Then download the program to a chosen slot on the VEX IQ Robot Brain.



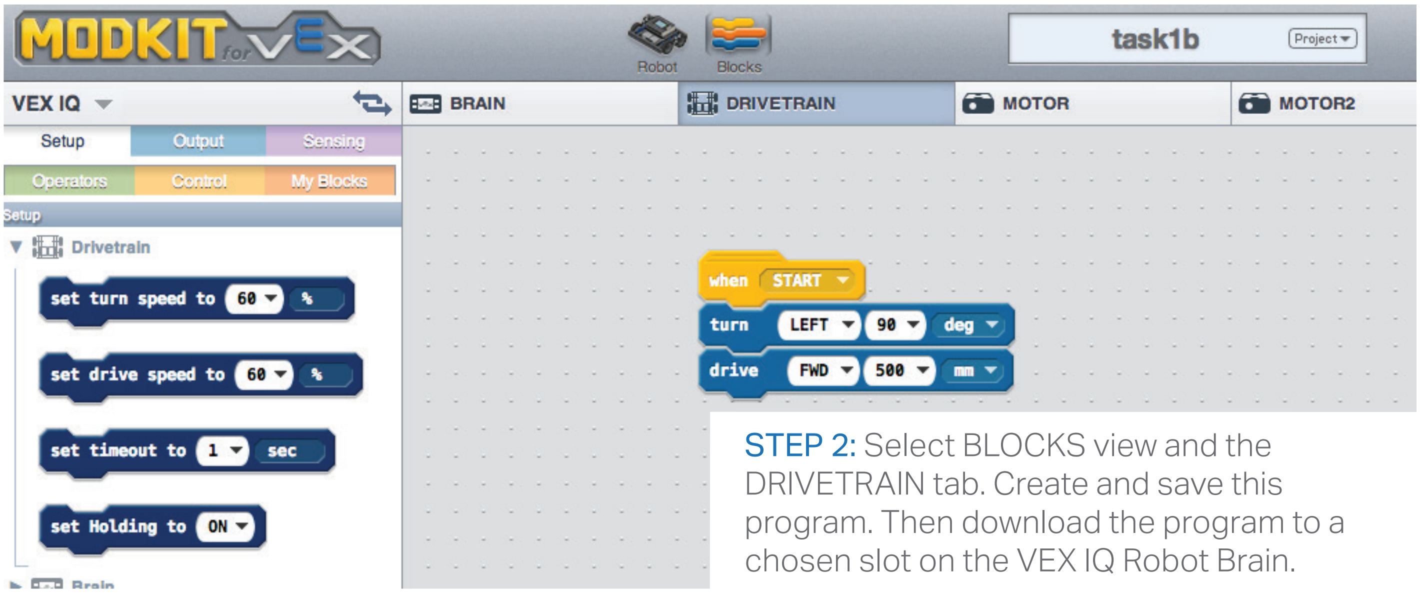
STEP 3: Place the robot in the correct starting position and run it to drive the robot to the starting position from the green cubes as shown above. The robot will back up 500mm, and turn to the right. The robot will end this program in the approximately same location and orientation as it would begin the Highrise Challenge Match in.

Task 1B: In this scenario, the robot is positioned in the match starting position on the highrise field and will navigate to the green blocks in the nearby row.

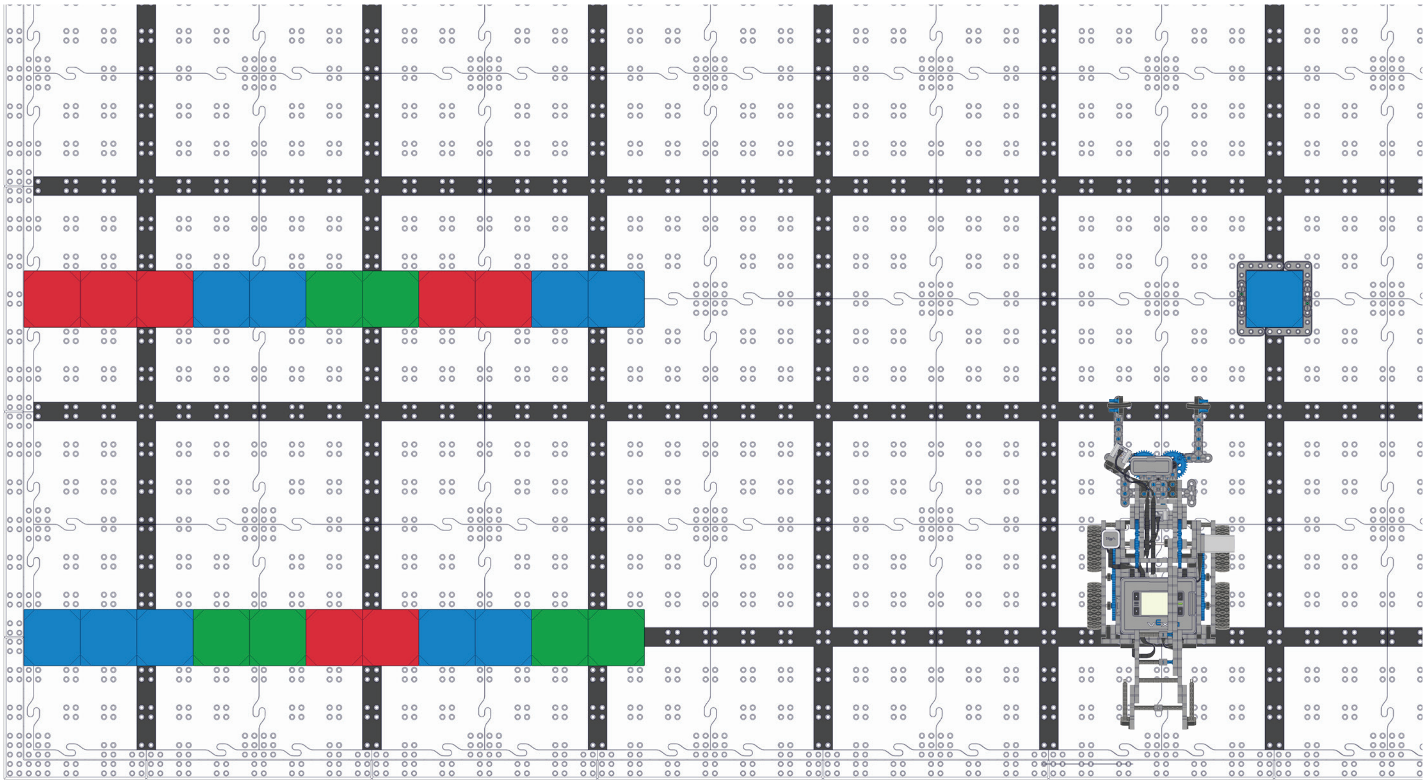




STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.



STEP 2: Select BLOCKS view and the DRIVETRAIN tab. Create and save this program. Then download the program to a chosen slot on the VEX IQ Robot Brain.

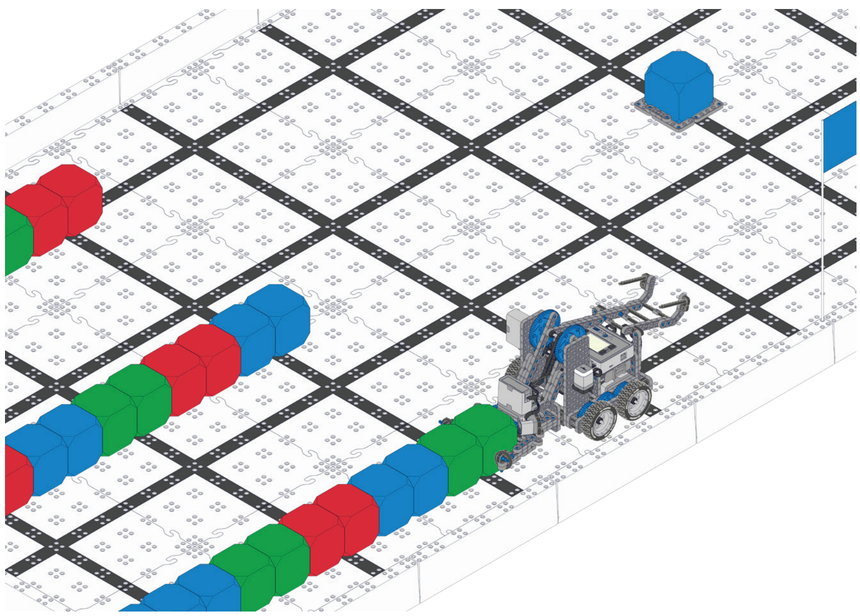


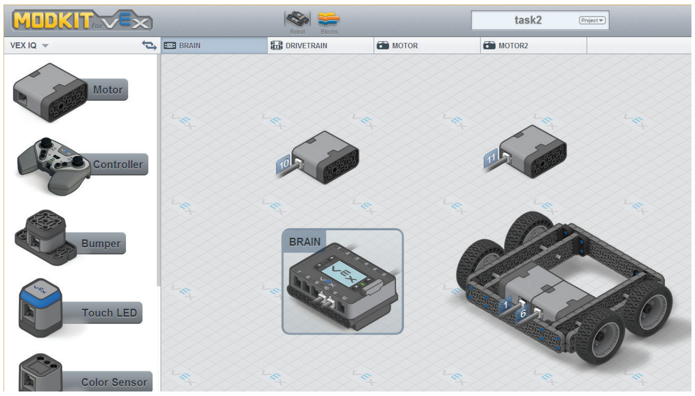
STEP 3: Place the robot in the correct starting position and run this program to drive the robot from the starting position on the same side as the row beginning with the green Highrise Cubes.

Summary of Program for Task 2 – Pick up the Highrise Cubes

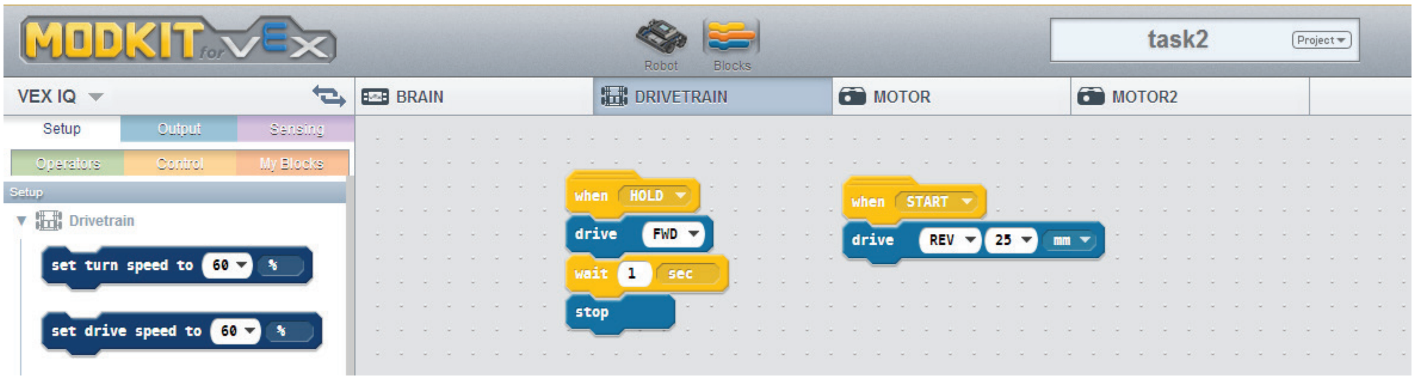
This program is a basic example of a solution to help campers complete one part of the challenge. In this example, the robot, when placed in front of a row of Highrise Cubes, will pick up the first Cube, store it in the Holder, and grasp another Cube in its Claw.

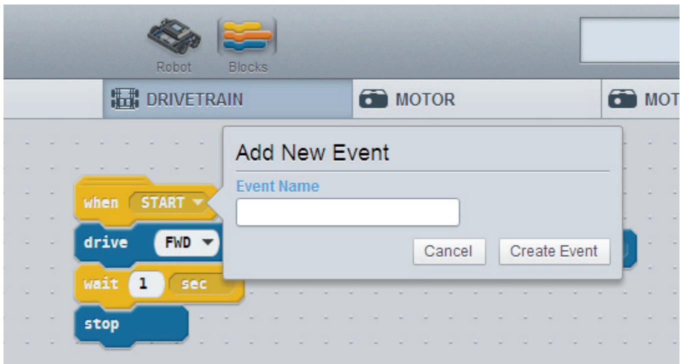
Task 2: In this scenario, the robot is positioned facing the front green Highrise Cube with the claw open. The first green Cube is inside the open claw.





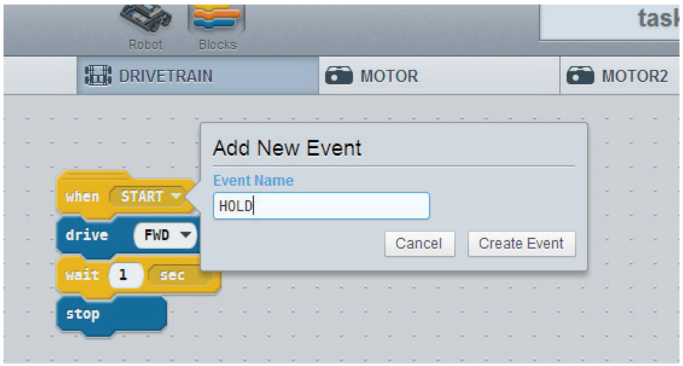
STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.





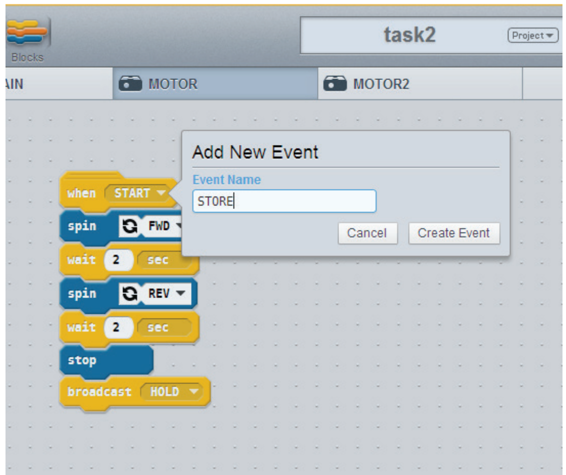
STEP 2: Select BLOCKS view and the DRIVETRAIN tab. Create and save these two blocks of code. Block 1 drives the robot forward upon broadcast of the HOLD event. Block 2 drives the robot backwards 25mm at the start of the program.

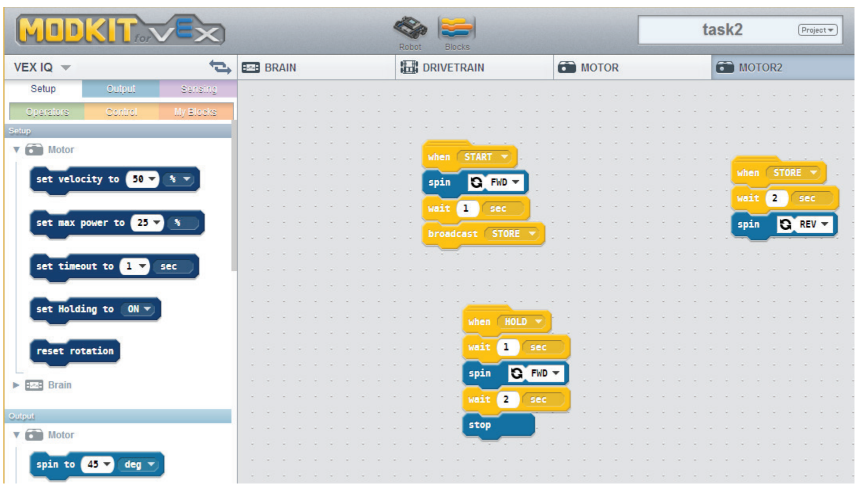
Note: You must create the "HOLD" event by choosing "new" from the drop down of the "WHEN" control block then typing in the new event name.



STEP 3: Select BLOCKS view and the MOTOR tab. Create and save this block of code that controls the robot arm movement.

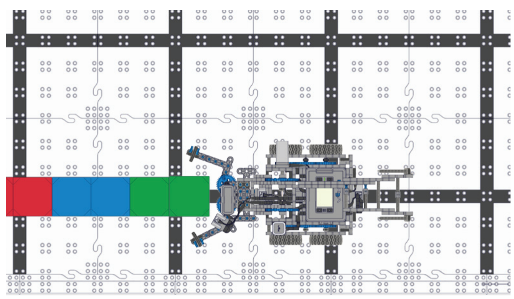
Note: You must create the "STORE" event by choosing "new" from the drop down of the "WHEN" control block then typing in the new event name.





STEP 4: Select BLOCKS view and the MOTOR2 tab. Create and save these blocks of code that controls the robot claw movement. Then download the program to a chosen slot on the VEX IQ Robot Brain.

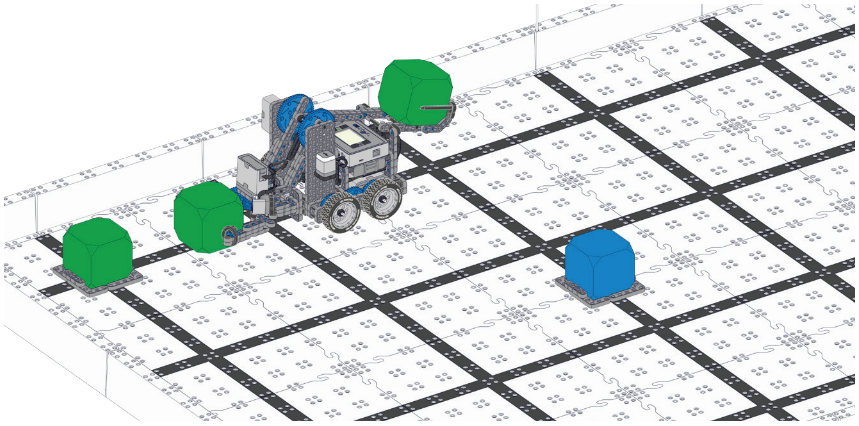
STEP 5: Place the robot in the correct position and run this program to pick up Highrise Cubes.



Summary of Program for Task 3 – Retrieve and Stack Highrise Cubes from the Holder

This program is a basic example of a solution to help campers complete one part of the challenge. In this example, the robot begins with one (1) Highrise Cube in its Holder and one (1) Highrise Cube in its Claw (not yet lifted). The robot will begin facing a Base Cube. It will stack the Highrise Cube in its Claw, retrieve the next Highrise Cube from the Holder, and stack it as well.

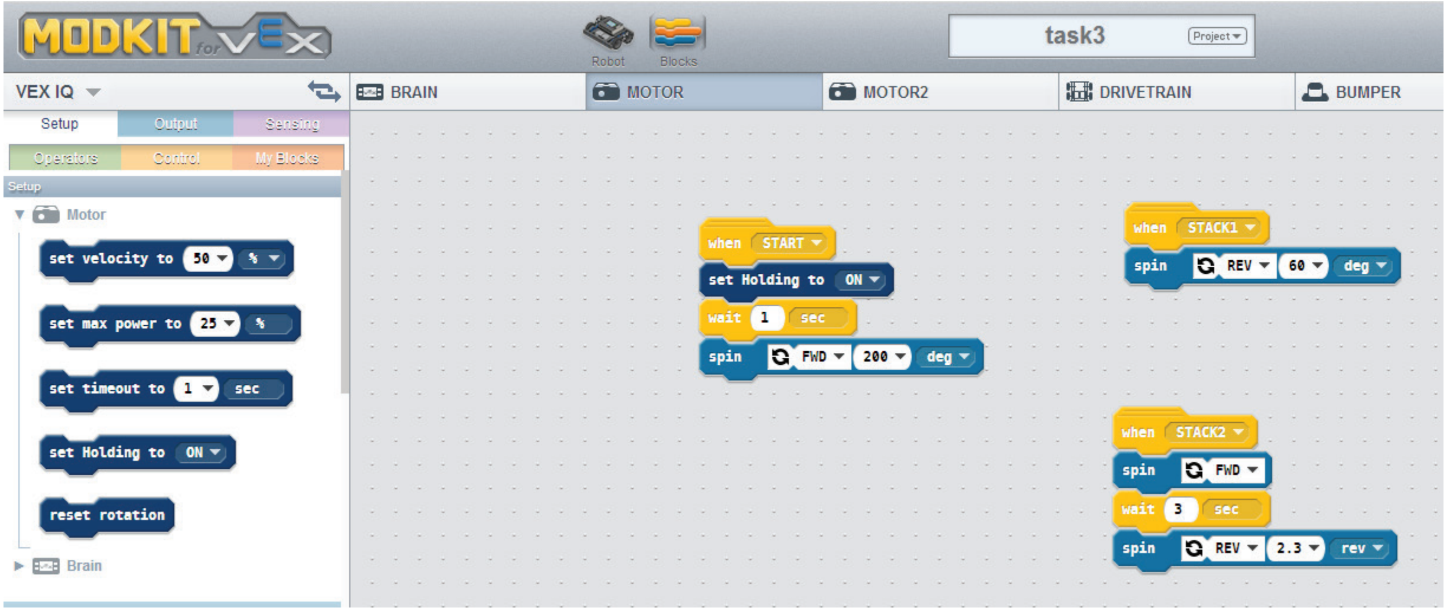
Task 3: In this scenario, the robot is positioned facing the green Highrise Base Cube.



STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.

STEP2: Select BLOCKS view and the DRIVETRAIN tab. Create and save these two blocks of code. Block 1 drives the robot forward upon broadcast of the HOLD event. Block 2 drives the robot backwards 25mm at the start of the program.

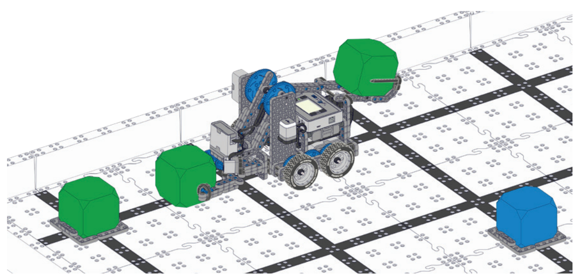
Note: You must create the "STACK1" and "STACK2" events by choosing "new" from the drop down of the "WHEN" control block then typing in the new event names.



STEP3: Select BLOCKS view and the MOTOR tab. Create and save these blocks of code that control the robot arm movement.

STEP4: Select BLOCKS view and the MOTOR2 tab. Create and save these blocks of code that control the robot claw movement. Then download the program to a chosen slot on the VEX IQ Robot Brain.

STEP5: Place the robot in the correct position and run this program to stack Highrise Cubes.



Sample Programming for ROBOTC for VEX Robotics

VEX IQ Camp Highrise Programming Challenge

The programs shown in this section can be used as a starting point for exploring programming strategies using the Clawbot IQ with Sensors to complete the Highrise Programming Challenge. There are countless solutions to effectively complete the required tasks.

Programs have been broken down into three basic tasks that could be helpful to complete the Challenge:

- Navigate between the starting position and the Highrise Cubes.
- Pick up Highrise Cubes.
- Retrieve and stack Highrise Cubes from the Holder

Note: Alternatively, campers can complete their programming in ROBOTC for VEX Robotics – Graphical programming software. This can then be exported to text-based programming if needed.

Summary of Program for Task 1 – Navigate between the starting position and the Highrise Cubes.

This program is a basic example of a solution to help campers complete one part of the challenge. In this example, the robot moves between the green Highrise Cubes and the starting position.

Highrise Programming Challenge 1.rbg

```
// Move backward for 1.6 seconds at -100 speed
backward ( 1.6 , seconds , -100 );
// Reset the gyro sensor to 0 degrees
resetGyro ( gyroSensor );
// While the gyro is greater than -90 degrees...
while ( getGyroDegrees(gyroSensor) > -90 ) {
    // Turn the robot to the right
    setMotor ( leftMotor , 50 );
    setMotor ( rightMotor , -50 );
}
```

STEP 1: Place the robot in front of the green Highrise Cubes. In this case, the robot is positioned facing the front green Highrise Cube with the claw open. The Cube is inside the open claw.

Create and run this program to drive the robot to the starting position on the same side as the green Highrise Cubes and green Base Cube.

Reset the Gyro Sensor. This will help to accurately calculate degrees of rotation and eliminate any error that may have been introduced with prior Gyro Sensor usage in the program

The robot will move backwards for 1.6 seconds (speed 100) and turn 90 degrees to the right (speed 50).

Highrise Programming Challenge 1-2.rbg

```
// Move backward for 1.6 seconds at -100 speed
backward ( 1.6 , seconds , -100 );
// Reset the gyro sensor to 0 degrees
resetGyro ( gyroSensor );
// While the gyro is greater than -90 degrees...
while ( getGyroDegrees(gyroSensor) > -90 ) {
    // Turn the robot to the right
    setMotor ( leftMotor , 50 );
    setMotor ( rightMotor , -50 );
}
// Reset the gyro sensor to 0 degrees
resetGyro ( gyroSensor );
// Move forward for 1.9 seconds at 100 speed
forward ( 1.9 , seconds , 100 );
// Move backward for 0.3 seconds at -50 speed
backward ( 0.3 , seconds , -50 );
```

STEP 2: Create and run this program to drive the robot from the starting position on the same side as the green Highrise Cubes and Base Cube to the row of green Highrise Cubes.

Reset the Gyro Sensor again.

The robot will open the claw and turn 90 degrees to the left (speed 50).

Reset the Gyro Sensor again.

The robot will travel 1.9 seconds forward towards (speed 100) the row of green Highrise Cubes, and then move backwards to a position(speed 50) that will allow the Arm to lift the Highrise Cube effectively.

The robot will travel forwards at full power and then move backwards at half speed. Slowing the robot down will help to make it more accurate.

Summary of Program for Task 2 – Pick up Highrise Cubes.

This program is a basic example of a solution to help campers complete one part of the challenge. In this example, the robot, when placed in front of a row of Highrise Cubes, will pick up a Highrise Cube, store it in the Holder, and grasp another Cube in its claw.

Highrise Programming Challenge 2.rbg

```
// Move the Claw Motor for 0.3 seconds at 100 speed
moveMotor ( clawMotor , 0.3 , seconds , 100 );
// Move backward for 0.2 seconds at -100 speed
backward ( 0.2 , seconds , -100 );
// Move the Arm Motor for 1.8 seconds at 100 speed
moveMotor ( armMotor , 1.8 , seconds , 100 );
// Move the Claw Motor for 0.1 seconds at -100 speed
moveMotor ( clawMotor , 0.1 , seconds , -100 );
// Move the Arm Motor for 1.8 seconds at 100 speed
moveMotor ( armMotor , 1.8 , seconds , -100 );
// Move forward for 0.5 seconds at 100 speed
forward ( 0.5 , seconds , 100 );
// Move the Claw Motor for 0.5 seconds at 100 speed
moveMotor ( clawMotor , 0.5 , seconds , 100 );
```

STEP 1: Create and run this program to pick up a Highrise Cube.

The robot will wait 0.3 seconds to make sure the claw is holding the first Highrise Cube (speed 100).

The robot will then back up for 0.2 seconds (speed 100) and stops the left and right motors from moving. It will then lift the Arm (speed 100) for 1.8 seconds to position the Arm to store the first Highrise Cube in the Holder.

The robot will wait for 0.1 second to stop the Arm and release the claw (speed 100), which releases the first Highrise Cube in the Holder.

The robot will wait 1.8 seconds to lower the Arm (speed 100) in front of the robot.

The robot will travel forward (speed 100) for 0.5 seconds to position itself directly in front of the second Highrise Cube in the row and close the claw (speed 100) for 0.5 seconds to ensure the claw closes far enough to grasp the Highrise Cube effectively.

Summary of Program for Task 3 – Retrieve and stack Highrise Cubes from the Holder.

This program is a basic example of a solution to help campers complete one part of the challenge. In this example, a robot begins with one (1) Highrise Cube in its Holder and one (1) Highrise Cube in its claw (not yet lifted). The robot will begin facing a Base Cube. It will stack the Highrise Cube, retrieve the next Cube from the Holder, and stack it as well.

Highrise Programming Challenge 3.rbg

```
// Move the Claw Motor for 0.3 seconds at 100 speed
moveMotor ( clawMotor , 0.3 , seconds , 100 );
// Move the Arm Motor for 0.5 seconds at 100 speed
moveMotor ( armMotor , 0.5 , seconds , 100 );
// While the Bumper Switch is not pressed...
while ( getBumperValue(bumpSwitch) == 0 ) {
    // Move the robot forward at power level 50
    setMotor ( leftMotor , 50 );
    setMotor ( rightMotor , 50 );
}
// Move backward for 0.65 seconds at power level -50
backward ( 0.65 , seconds , -50 );
// Move the Arm Motor for 0.3 seconds at power level -50
moveMotor ( armMotor , 0.3 , seconds , -50 );
// Move the Claw Motor for 0.3 seconds at power level -100
moveMotor ( clawMotor , 0.3 , seconds , -100 );
```

STEP 1: Create and run this program to stack two Highrise Cubes on the Base Cube.

This program begins with one Highrise Cube in the robot's Holder and the other in the robot's claw.

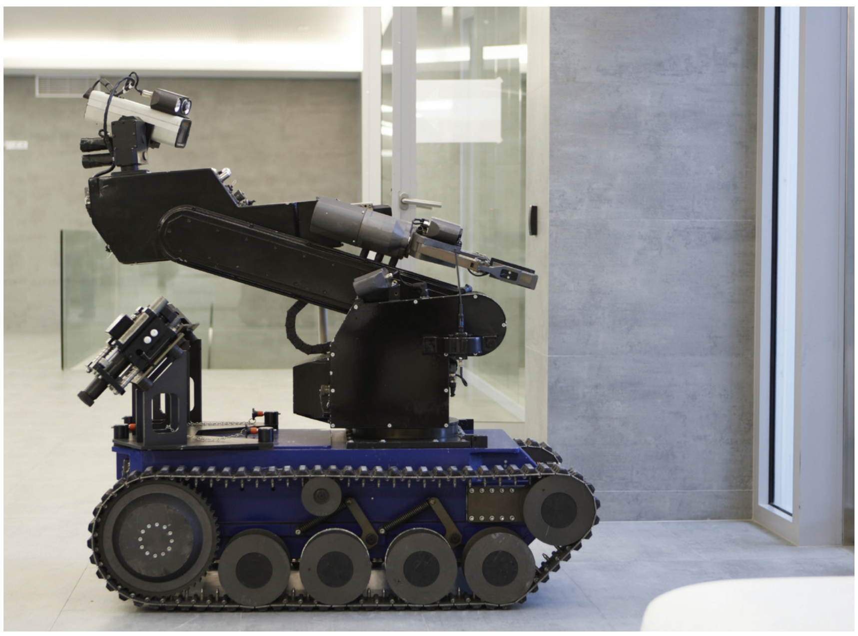
The robot will wait 0.3 seconds to hold the Highrise Cube in its claw (speed 100) and lift the Arm (speed 100) for 0.5 seconds.

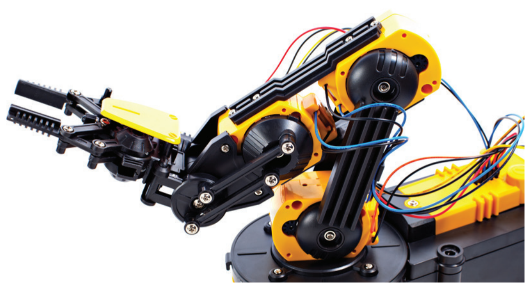
If the Bumper Switch has not been pressed by contact with the Base Cube, the robot will travel forward (speed 50) while keeping the Arm in the same position.

After the Bumper Switch has been pressed by contact with the Base Cube, the robot will move backwards (speed 50) for 6.5 seconds to position itself for stacking, and stop moving.

The robot will lower the Arm (speed 50) for 0.3 seconds to position the Arm so it can stack the Highrise Cube.

Real-World Extension Activities

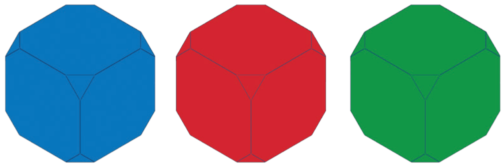




Real-World Extension Activities

These activities are for intermediate to advanced users and can take quite a bit of time. Simply discussing these real-world problems and possible solutions is a great time saving alternative to completing the full activity.

Extension #1: Interactive Robot

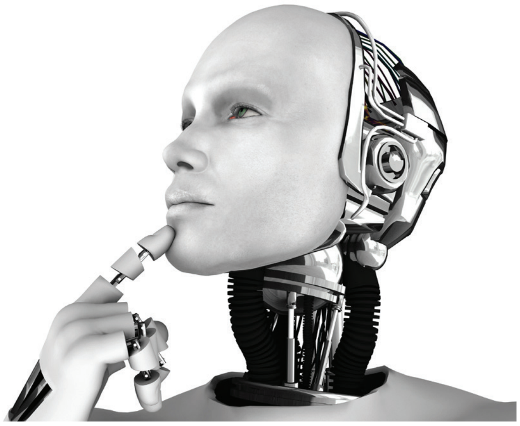


Imagine that you are an engineer designing a companion robot. The robot is powered by Highrise Cubes and has a "preference" for certain kinds of Cubes over others, based on the amount of energy they provide.

Red Highrise Cubes provide the greatest amount of energy.

Blue Highrise Cubes provide half the energy of red cubes.

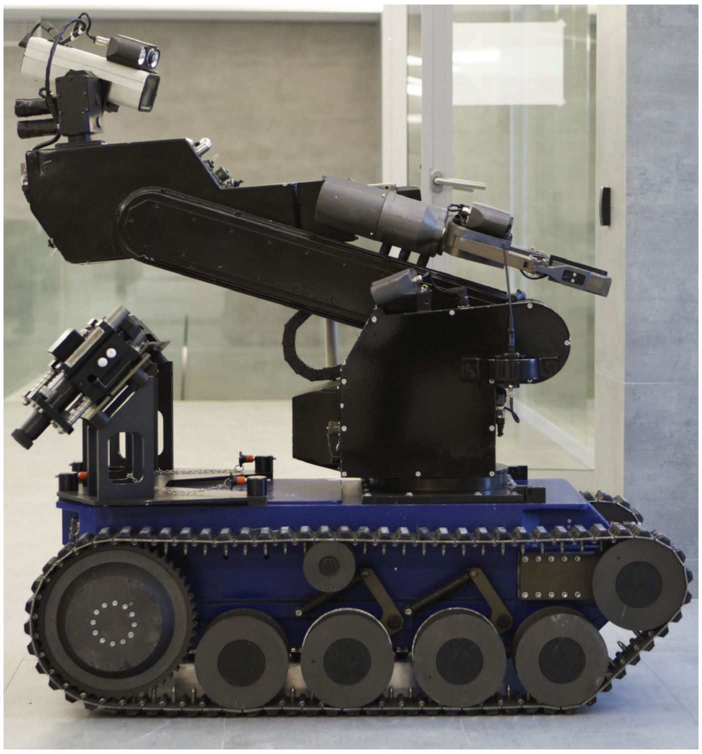
Green Highrise Cubes do not provide any energy.



Build the head and neck or torso of a robot and add mechanisms or sensors that will help the robot identify which kind of Cube it is being offered. Program the robot to respond and move with a head shake, nod, message on the LCD screen of the Robot Brain, or communicate in another way to indicate whether it will accept the Cube being offered, ask for more, or politely decline it.

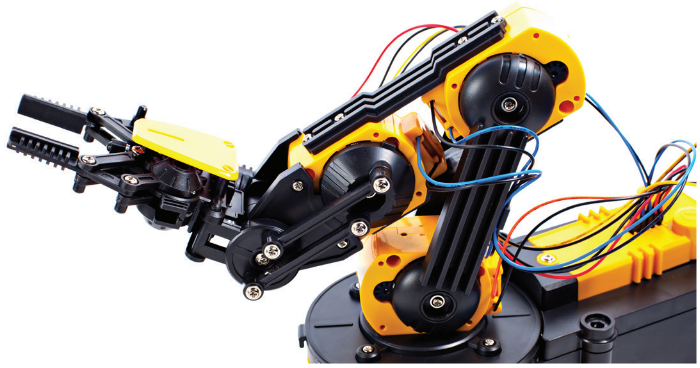
Extension #2: Warehouse Management

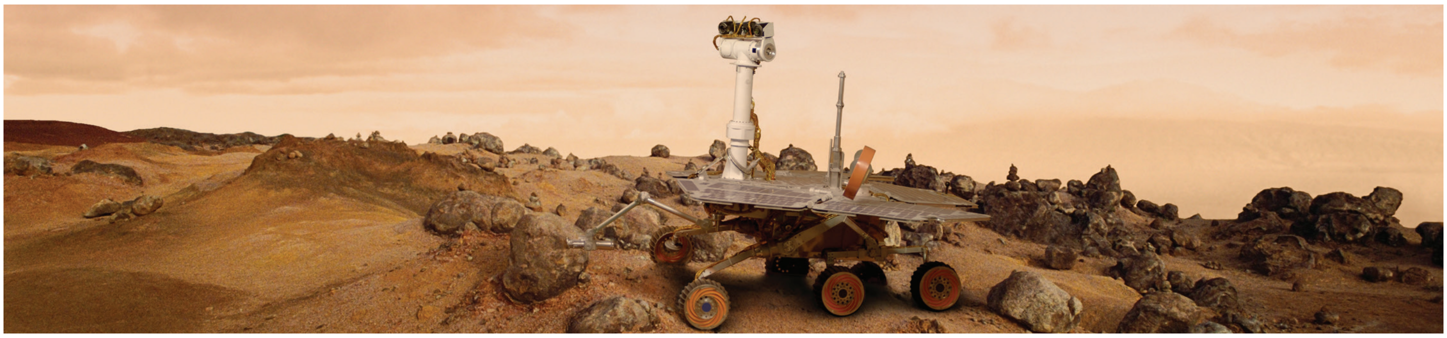
Imagine that you are an engineer in a large factory. You need the robots in the factory to carry goods along a complicated path that travels up and down ramps, around obstacles or over different kinds of surfaces, without endangering factory workers or getting lost. Build a robot that can easily and safely navigate, from one room to another, using pathways (lines) that the robot can detect.



Extension #3: Articulated Arm

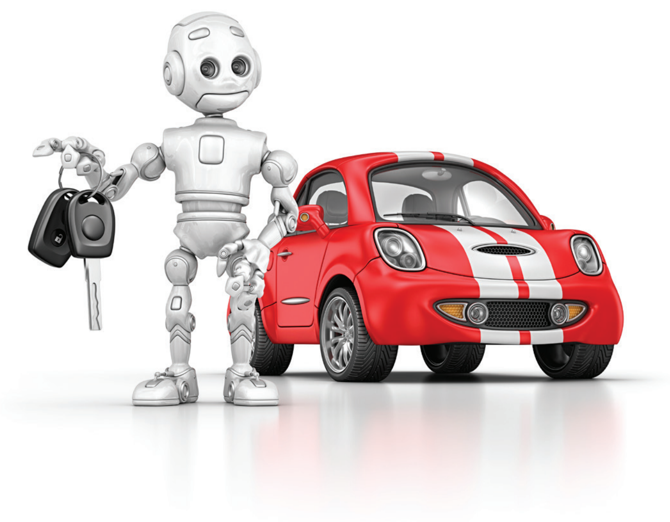
Imagine that you are an aerospace engineer in the space program. Build a robot that could be mounted onto a space rover that will pick up rock samples of different shapes and sizes, from different levels of terrain. The robot should have an articulated arm that can be controlled easily via remote control with some autonomous functionality to control the robot's movement.



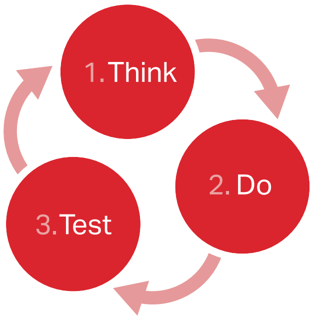


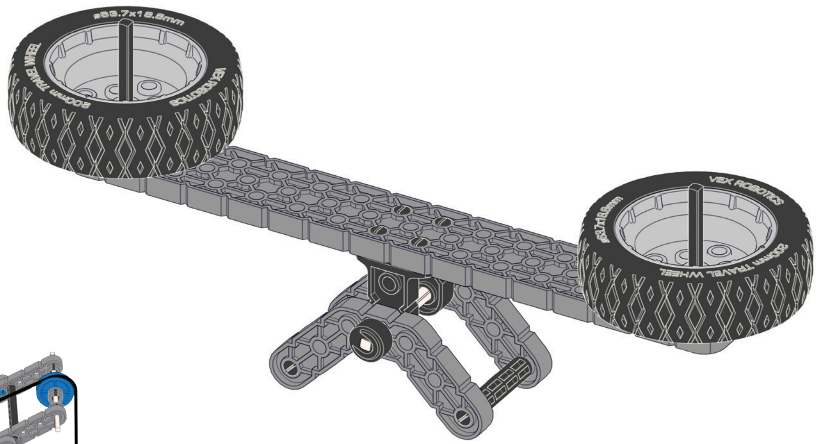
Extension #4: City Navigator

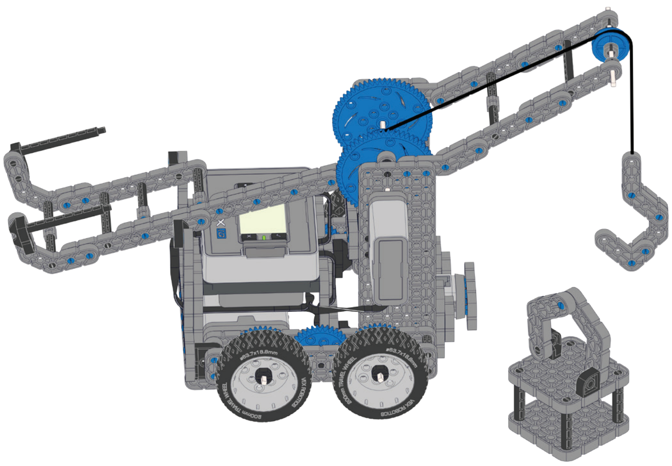
Imagine that you are an engineer assisting in the development of an automated transportation system that will need to navigate from one destination to another through busy city streets. Build a robotic vehicle (car) and add sensors and programming that can take a passenger through a maze of twists and turns in a bustling downtown block. As an urban engineer, you can also make changes to the city itself to help make the automated car system function efficiently for its residents.

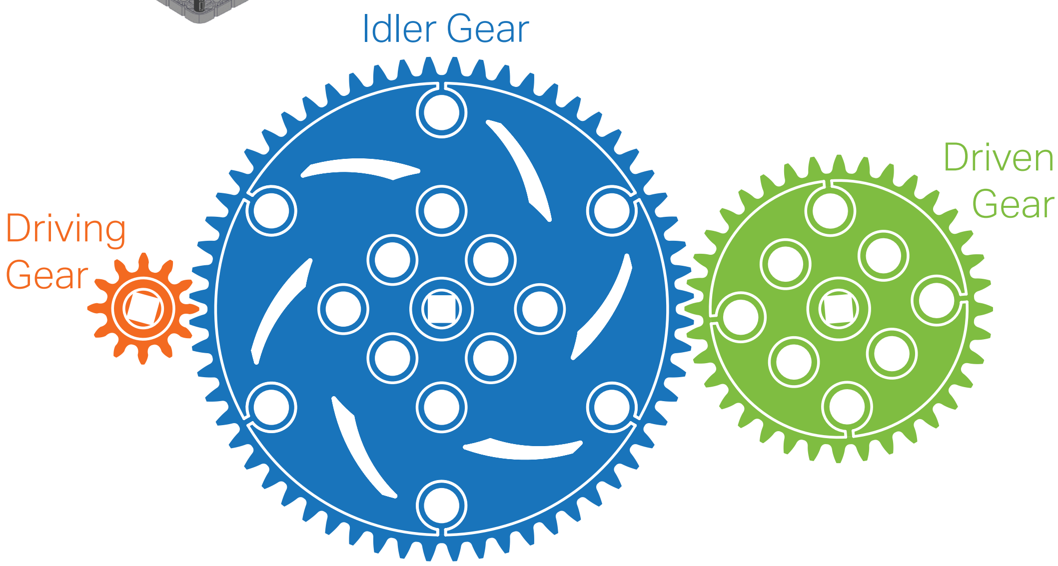


Learning Activities



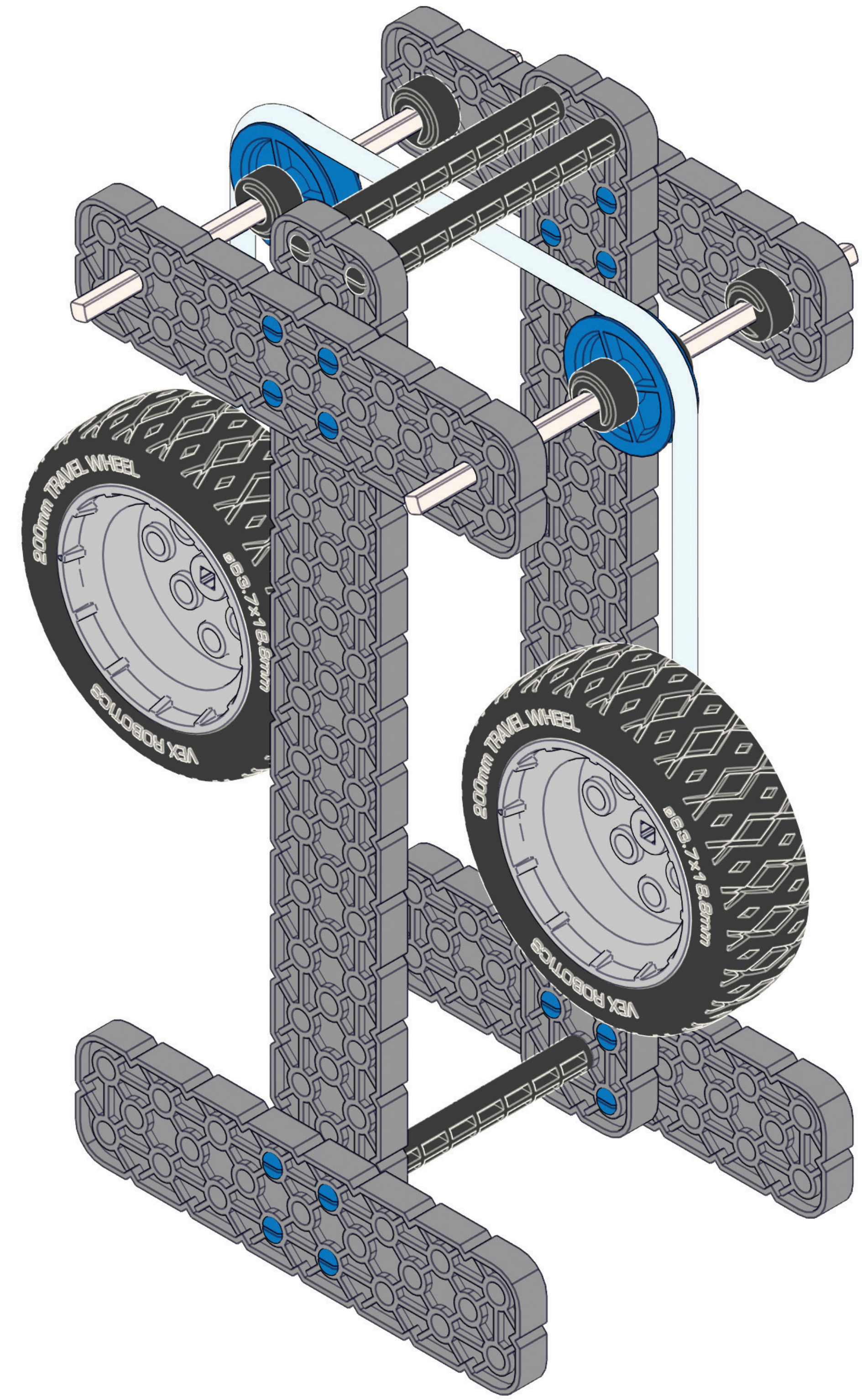


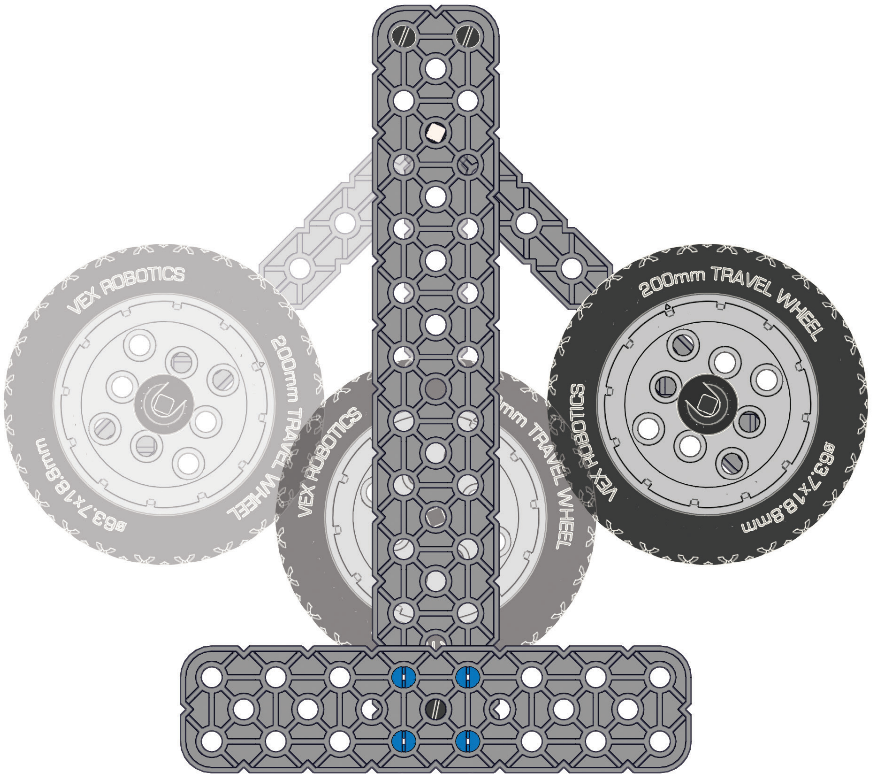




A

Pendulum


200mm TRAVEL WHEEL
VEX ROBOTICS
200mm TRAVEL WHEEL
VEX ROBOTICS



A.cl1

Overview of Activity:

Campers will build a Pendulum and consider situations in real life in which this kind of machine is used, such as old clocks, swings of all shapes and sizes, or as friction pendulums inside buildings where earthquakes are common. Campers will be asked to knock over a stack of Highrise Cubes using a pendulum as a "wrecking ball." They will modify the pendulum design for strength and stability when completing the task and document their changes with drawings or a mobile device.

Learning Objectives:

- Campers will become familiar with the basic building components of the VEX IQ Super Kit.
- Campers will use appropriate terminology to describe basic VEX IQ building components.
- Campers will be able to identify different kinds of simple machines visually.
- Campers will modify an existing assembly for a new purpose.
- Campers will document their machine and its use.

Suggested Timing:

Up to 1 hour

A.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- 2-3 VEX IQ Highrise Cubes
- VEX IQ Super Kit Contents and Build Tips Poster
- VEX IQ Build Instructions - Standard Drive Base (Kit Documentation) 1
- Think, Do, Test – Pendulum
- Simple Motion: The Pendulum

Optional Materials:

- Plastic cups, playing cards or dominoes to stack and knock down with the pendulum.

Pre-learning Suggestions:

- If possible, provide campers with some time to inventory and sort their VEX IQ Super Kits.
- This activity will help campers become more familiar with some of the basic construction elements of the kit.
- It is important to reinforce the habit of an organized workspace and kit.

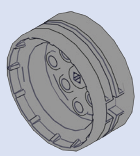
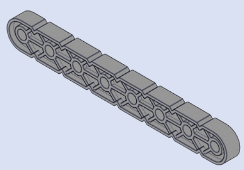
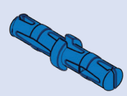
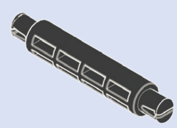
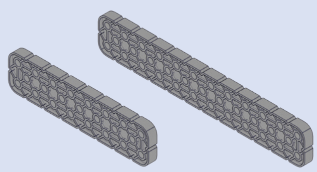
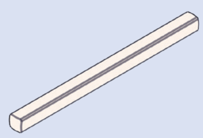
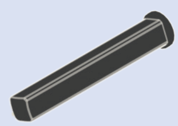
Based on VEX IQ Robotics Education Guide:

- Simple Motion: The Pendulum (D.4)

A.cl2

Detailed Directions

1. Provide each pair of campers with the VEX IQ Super Kit and the kit documentation, VEX IQ Super Kit Contents and Build Tips poster.
2. Ask campers to locate and identify the following components:

			
Wheel Hub	1x8 Beam	1x1 Connector Pin	4x Pitch Standoff
			
Rubber Shaft Collar	2x8 Beam and 2x12 Beam	6x Pitch Shaft	2x Pitch Plastic Capped Shaft

A.cl2 cont.

3. Tell campers that they are going to use the components in front of them to build an old but powerful machine. Ask them the questions below to have them guess their building challenge. Encourage campers to pick up and manipulate the components as they consider the questions.

   Questions to ask campers:

   Think about some of the common simple machines you know.

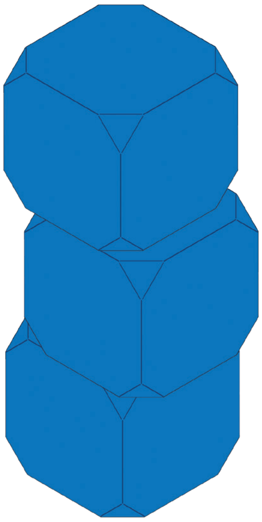
   - Which of these simple machines can be used in construction or demolition?
   - Which of these simple machines can be used for balance?
   - Which of these simple machines uses gravity to make work easier to do?

   How have you seen pendulums used in the real world?
   Examples of pendulums in everyday life:

   - Old fashioned clocks
   - Swings on a playground
   - Wrecking ball
   - Friction pendulums inside buildings in areas where earthquakes are common

   If needed, provide campers with the handout, "Simple Motion: The Pendulum" Background Information.

   This document, drawn from the VEX IQ Curriculum, offers background information on simple harmonic motion and pendulums.



4. Provide campers with the Think, Do, Test - Pendulum handout and Build Instructions – Pendulum. Tell them that their challenge is to create a pendulum, using VEX IQ components, that is strong and stable enough to knock over a stack of Highrise Cubes.
5. Provide campers with time to build the basic pendulum model and complete at least one cycle of testing and improving their pendulums. Encourage campers to use the Think, Do, Test - Pendulum handout to document their improvements and challenges.

   Questions to ask campers:

   Does this pendulum use enough force to knock over at least one cube?

   - If not, how can you modify the pendulum to do this? (e.g., taller with longer string to increase the size of the arc on its swing, use different components to create a heavier weight at the end of the pendulum?)

   Is the pendulum stable enough to withstand the force of impact when the wheel (tire and hub) connects with the Cubes?

   - How can the pendulum be stabilized? (e.g., create a broader base, attach to the Highrise Challenge Field)

   What happens if there is a wall of 3x3 Highrise Cubes instead of just a stack? Is the pendulum still effective?
6. Have campers build their designs and complete tests with different configurations and combinations of Highrise Cubes.

A.cl2 cont.

Tips and Best Practices:

- Make sure the collar shaft is not too close to the arm, or the swing of the pendulum may be slowed by friction.

Extension:

- Create a device that will make the pendulum swing without touching it.
- Have campers create challenges for each other's pendulums using some of the optional materials suggested in the Overview for this lesson (e.g., a plastic cup full of cotton balls, a pyramid of cups, a line of dominoes or house of cards to knock over). These can be made into videos using smart devices to show parents at the end of the day.
- Create a metronome out of the pendulum by increasing the length of the pendulum. (e.g., increasing the length of the pendulum to 25cm) will result in approximately one swing per second. Campers can experiment with different lengths for different timing.

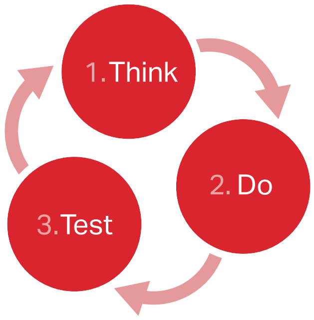
A.cs1

Think, Do, Test - Pendulum

Student Name(s): ______________________________

Instructions:

After you've completed building the **Pendulum Assembly**, set up a stack of three Highrise Cubes.



Test your pendulum.

- Can the pendulum knock the blocks over?
- Is the pendulum stable?
- How can the pendulum be improved?

1. "THINK" - Create a plan to improve your pendulum.

____

____

Draw your idea or problem here, too, if it helps you to describe it.

2. "DO" – What modifications did you make to your pendulum? List the changes you made and how you think they will affect the pendulum's performance.

____

____

____

____

3. "TEST" – After your "DO" step is done, test your design. Write down your observations in a notebook, or use a smart device to record your observations as video or audio:

____

____

Does your Standard Drive Base drive and function like is expected? YES NO

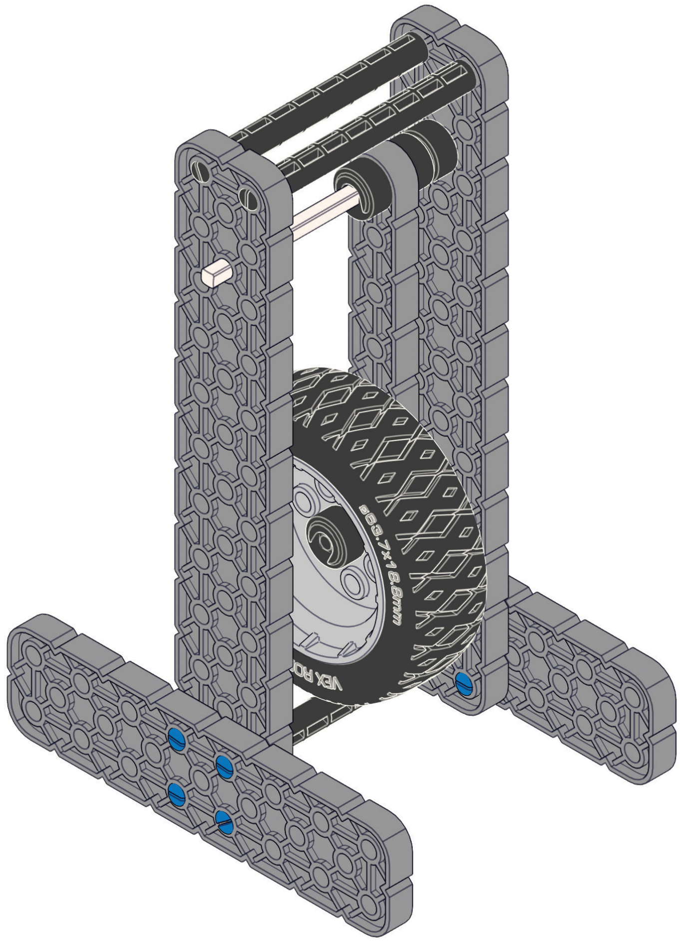
If you answered "YES" - Congratulations! You will score well on the Build Rubric. You may now move on to build the rest of the Clawbot IQ or move on to other lessons.

If you answered "NO" - Use your observations above and the Build Rubric to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process, until your Robot Base functions correctly.

Problems ARE NOT failures. They are an expected part of the design process!

A.cs2

Simple Motion: The Pendulum

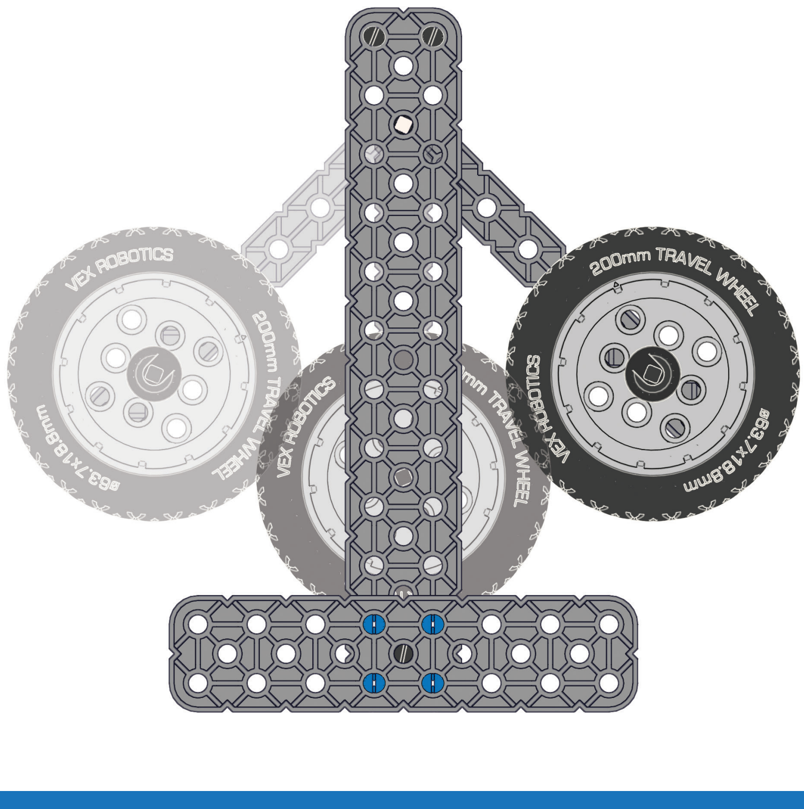


Simple Motion (more fully known as **Simple Harmonic Motion**) is what happens when an object moves in a non-complex periodic way. This means that:

- The object experiences a force that moves it
- The movement occurs, reaching some maximum value
- The object returns to its "original" conditions
- The process repeats

Let's take the example of a **pendulum** and consider what happens. A **pendulum** is defined as a body suspended from a fixed point so that it can swing back and forth under the influence of gravity as a **force**.

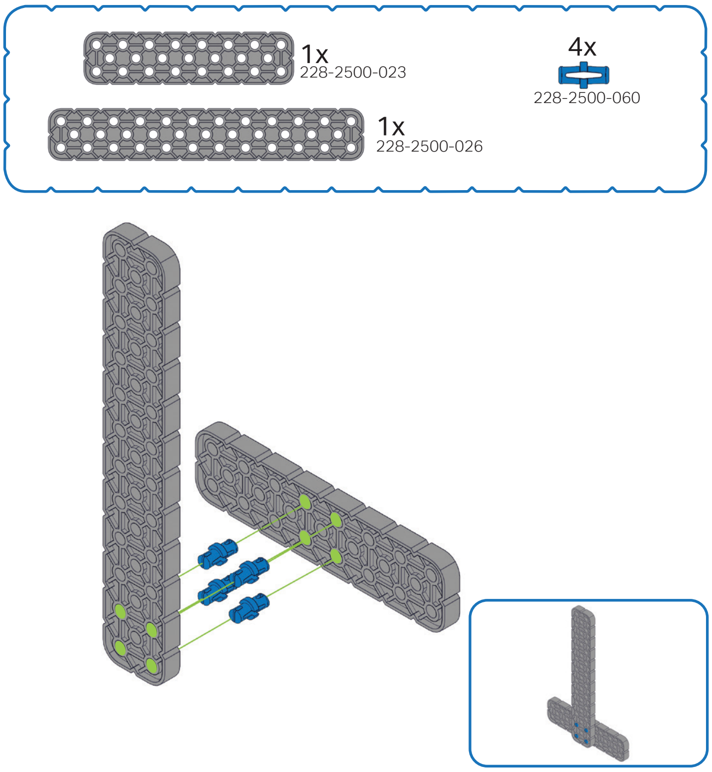
When a **pendulum** is started, it swings (accelerates) down under the influence of **gravity**. **Gravity** is the attraction between two masses, such as the earth and an object on its surface. At the bottom of its arc, the **pendulum** then swings up on the other side. It continues to move up (and decelerate) until it stops. The pendulum then begins to swing back down, reaching some maximum velocity at the bottom of its arc before swinging back up to where it began. The pendulum has gone through one complete cycle of its motion, and because it is a repetitive cycle, it can be said to be **simple harmonic motion**. Friction (the force that resists motion through the rubbing of one object against another) will eventually stop the **pendulum**, but not before several cycles have passed.



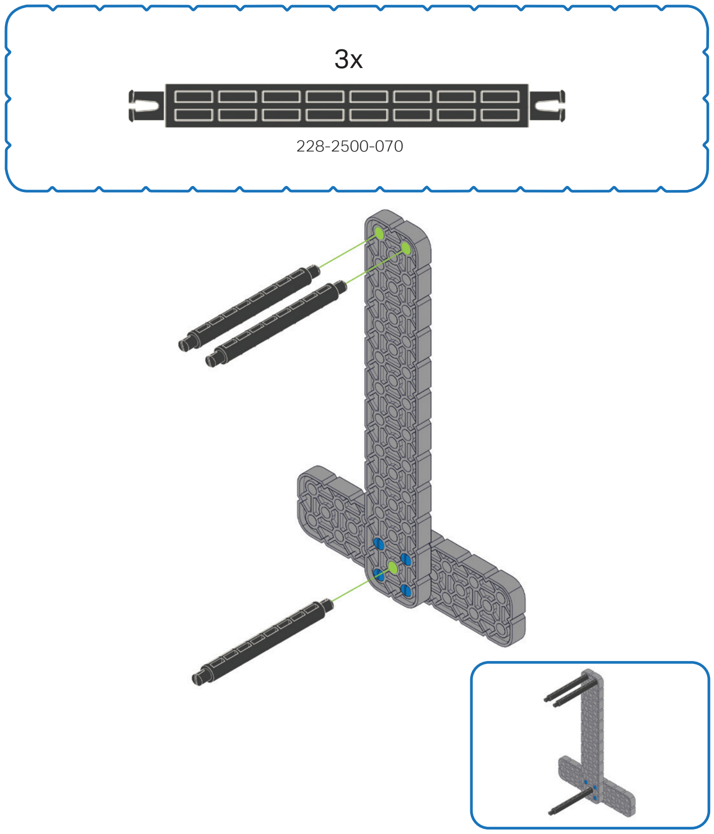
A.cs3

Pendulum Assembly

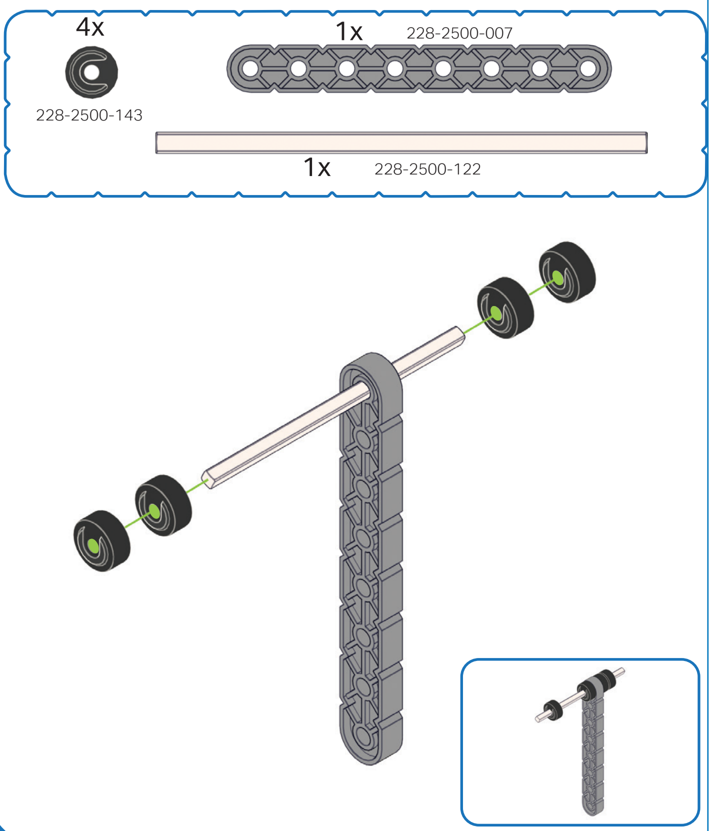
1



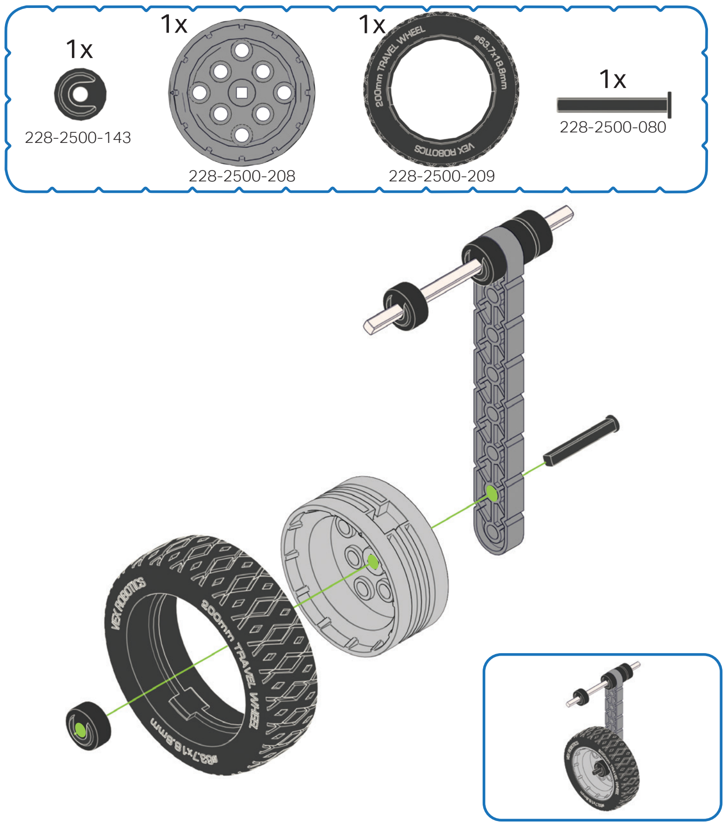
2



3

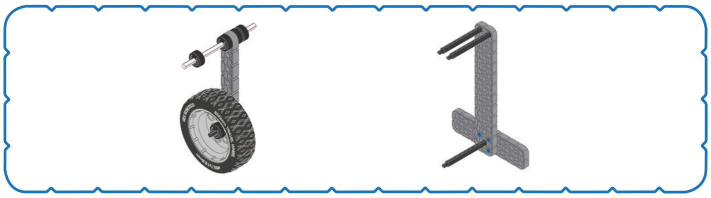


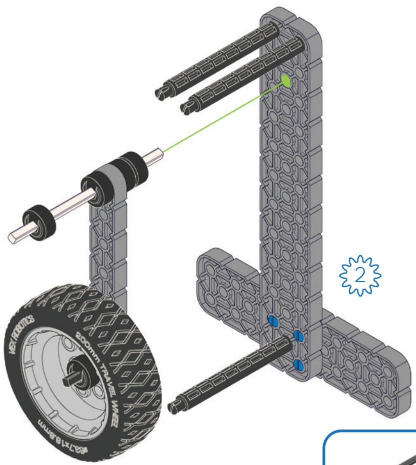
4

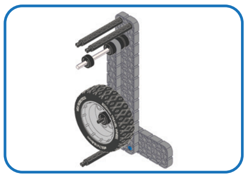


A.cs3 cont.

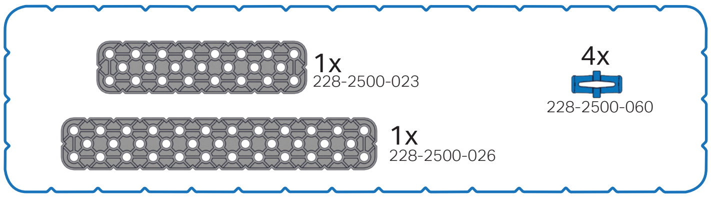
5

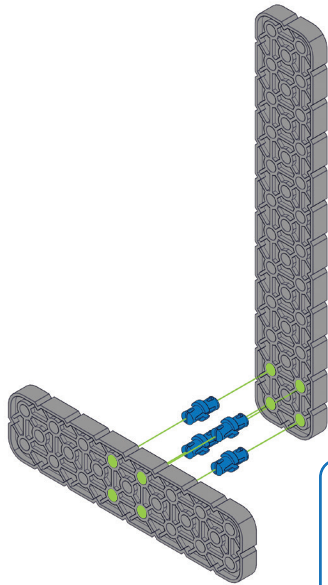


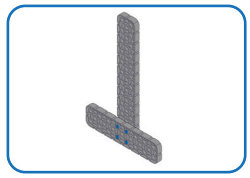




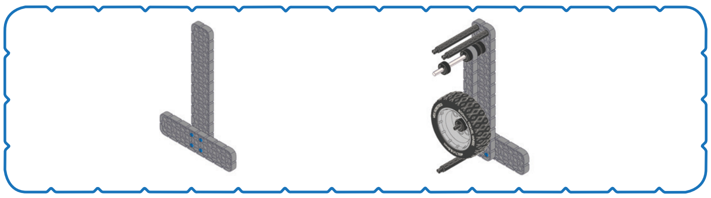
6

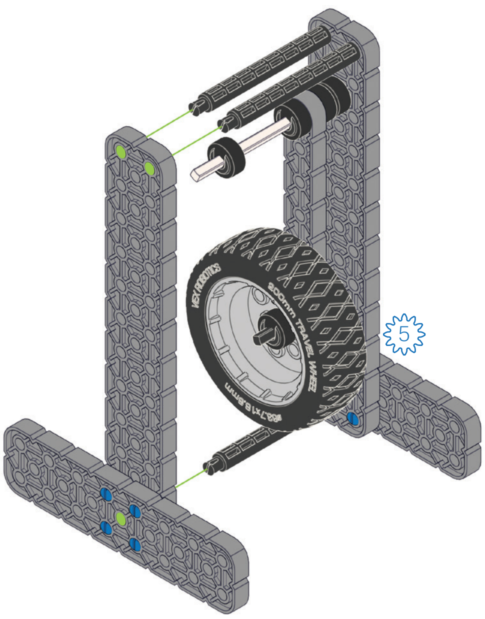


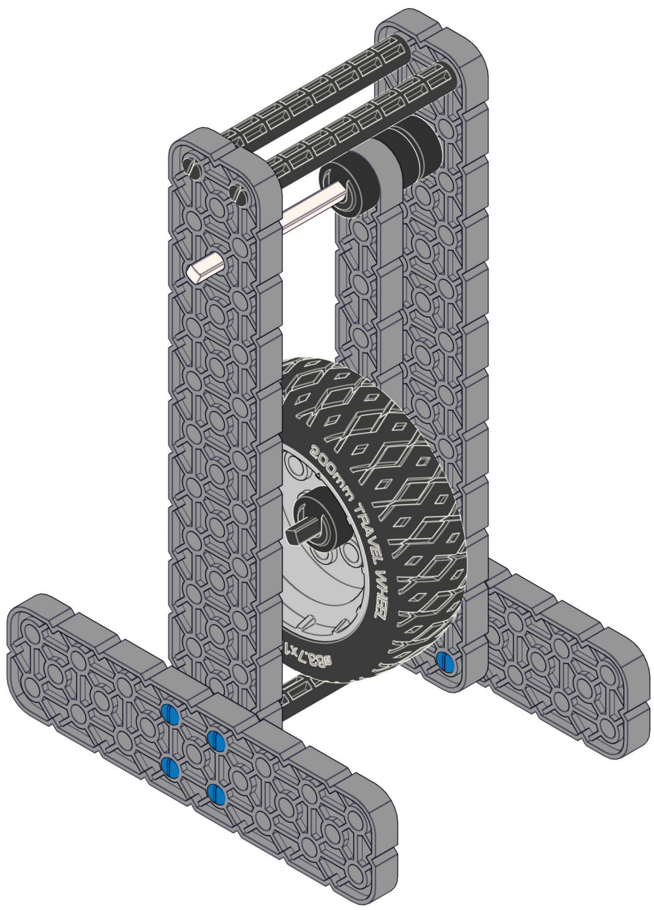




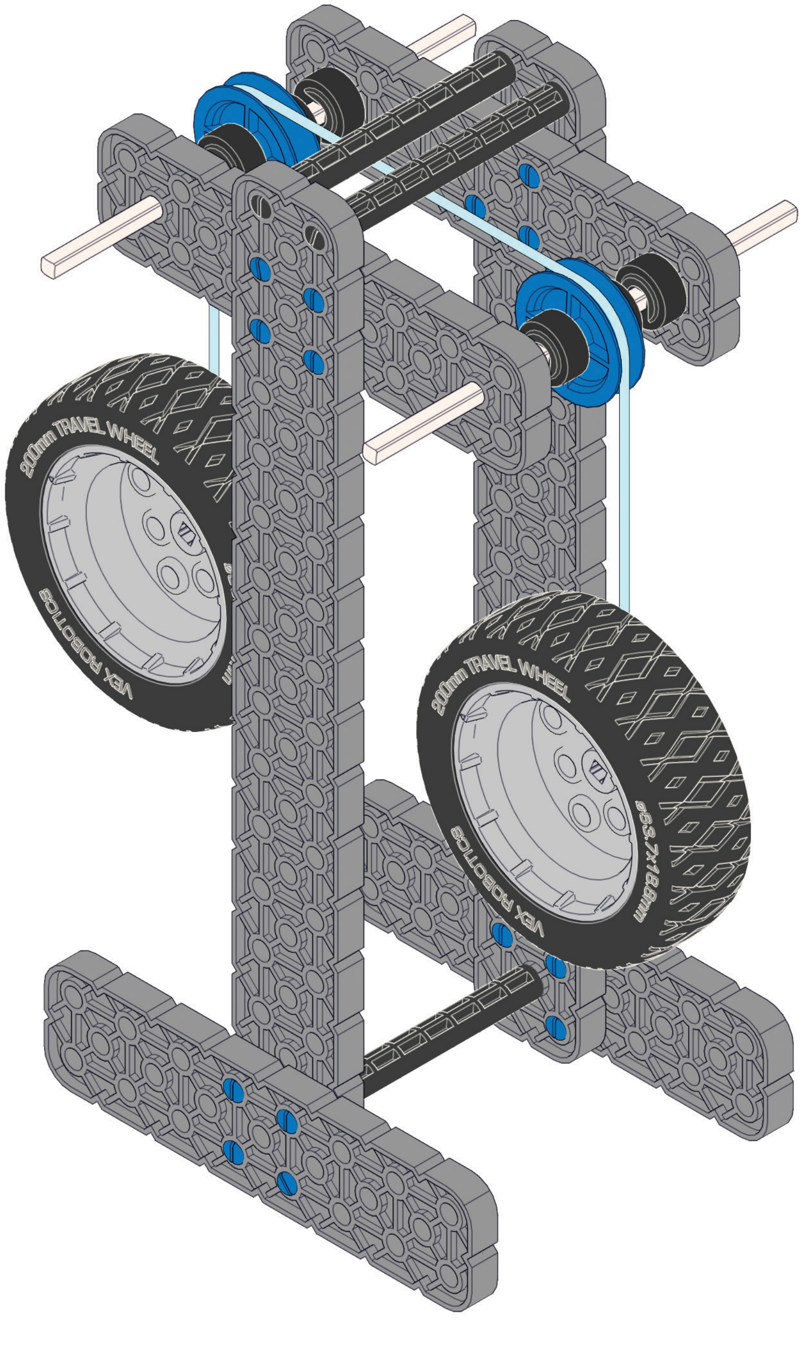
7

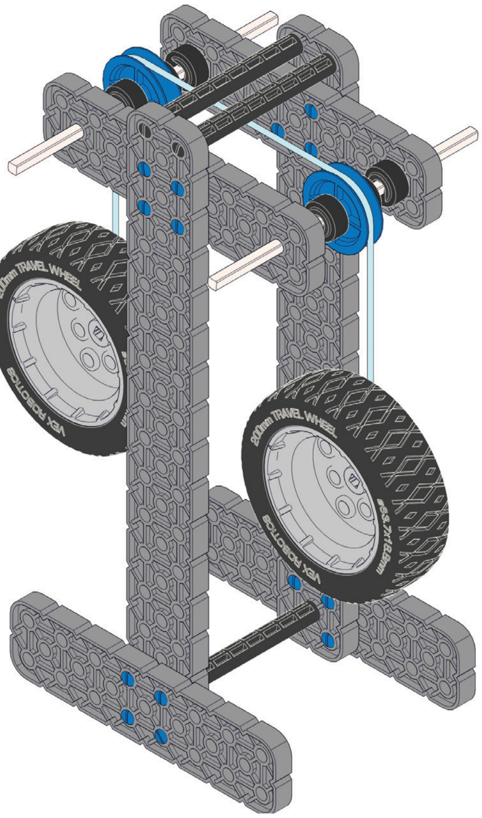






B
Pulley


200mm TRAVEL WHEEL
VEX ROBOTICS
200mm TRAVEL WHEEL
VEX ROBOTICS
ø63.7x18.8mm



B.cl1

Overview of Activity:

Campers will build a Pulley and consider situations in real life in which this kind of machine is used, such as a flag pole or elevator. They will build an example of a pulley system and then modify the build to create an elevator that can lift a small object or bundle of small objects from the VEX IQ Super Kit. Campers will document their modifications with drawings or a mobile device.

Note: This activity can be integrated with Learning Activity H: Standard Drive Base with Lifting Mechanism.

Learning Objectives:

- Campers will become familiar with the basic building components of the VEX IQ Super Kit.
- Campers will use appropriate terminology to describe basic VEX IQ building components.
- Campers will be able to identify different kinds of simple machines visually.
- Campers will modify an existing model for a new purpose.
- Campers will document their machine and its use.

Suggested Timing:

Up to 1 hour

B.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Small objects to lift with an elevator (such as bundles of small VEX IQ Beams, Shafts, or individual sensors)
- Think, Do, Test – Pulley
- "The Six Types of Simple Machines" Background Information
- VEX IQ Build Instructions - Pulley Assembly

Optional Materials:

- String (to complete Pulley Assembly)
- Coins, sticks of gum, car keys, or other small objects can also be used as objects to lift in the elevator. Using objects of different sizes, shapes and weights will present different challenges for its design.

Pre-learning Suggestions:

- If possible, provide campers with some time to inventory and sort their VEX IQ Super Kits.
- This activity will help campers become more familiar with some of the basic construction elements of the kit,
- It is important to reinforce the habit of an organized workspace and kit.

Based on VEX IQ Robotics Education Guide:

- **The Six Types of Simple Machines** (D.3)

B.cl2

Detailed Directions:

1. Provide each pair of campers with the **VEX IQ Super Kit** and the **Build Instructions - Pulley Assembly**.
2. Have students build the **Pulley Assembly** as outlined in the build instructions and brainstorm a list of objects they use or see every day that use a mechanism like this.

   If needed, provide students with the handout, "**The Six Types of Simple Machines**" Background Information. This document, drawn from the VEX IQ Curriculum, offers background information on different kinds of simple machines, including pulleys.

   Examples of Pulleys in everyday life:

   - Flag pole
   - Old-fashioned wells
   - Blinds or curtains
   - Exercise machines
   - Elevators
3. Have campers experiment for a few minutes with the pulley and describe how it works with their partner or teammate.

B.cl2 cont.

4. Provide campers with the criteria for designing an elevator and let them know what kind of “freight” they will be transporting in their elevators. The criteria for the challenge have been outlined on the **Think, Do, Test - Pulley** handout.
5. Have campers build their designs and complete tests with different objects. Results can be documented with pen and paper or made into videos using smart devices to show parents at the end of the day.

Tips and Best Practices:

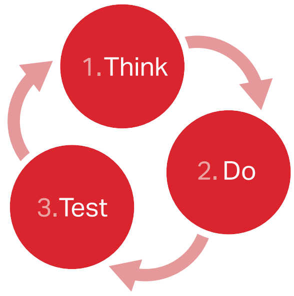
- To create a more stable base for the elevator, attach the assembly to the Highrise Challenge Field.

Extension:

- Add a motor to the model and use the buttons on the VEX IQ Robot Brain or VEX IQ Remote Control to raise and lower the platform.
- Integrate sensors into the model, such as the VEX IQ Touch LED or Bumper Switch to start and stop the movement as a braking system.
- The current design of the Pulley focuses on changing the direction of force. Experiment with different modifications to the Pulley to reduce the effort needed to lift the objects in the elevator.

B.cs1

Think, Do, Test - Pulley



Student Name(s): ____

Instructions:

After you've completed building the **Pulley Assembly**, gather a small group of items (as instructed by your camp leader). Look at these objects.

Begin with the **Pulley Assembly** you created earlier in this activity. Your job is to "imagine" a version of this assembly, or design an original assembly you can build that will allow you to transport these objects between two levels.

Criteria for your design:

- The elevator must lift 3-4 small objects at a time or a single object of equal weight.
- The elevator must use a pulley.
- The elevator must include a platform or container to hold the object(s) being lifted.
- The elevator should include a mechanism to lift and lower the platform manually (e.g., crank handle).

1. "THINK" - Think carefully about the collection of objects and how they will need to be transported by the elevator. Make observations about their physical characteristics (size, weight, shape, how can they be contained or moved).

Draw the objects here and write notes or labels to describe your observations about them

Draw your plan for the lifting platform of container

Draw your plan for the elevator

B.cs1 cont.

2. "DO" – What modifications did you make to the pulley assembly? List the changes you made and describe how the objects will be lifted or lowered.

3. "TEST" – After your "DO" step is done, test your design. Write down your observations in a notebook, or use a smart device to record your observations as video or audio:

Could your elevator complete the task? YES NO

If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until your elevator completes the tasks.

Problems ARE NOT failures. They are an expected part of the design process!

B.cs2

The Six Types of Simple Machines:

Wheel & Axle - Makes work easier by moving objects across distances. The wheel (or round end) turns with the axle (or cylindrical post) causing movement. On a wagon, for example, a container rests on top of the axle.

Inclined Plane - A flat surface (or plane) that is slanted, or inclined, so it can help move objects across distances. A common inclined plane is a ramp.

Wedge - Instead of using the smooth side of the inclined plane to make work easier, you can also use the pointed edges to do other kinds of work. When you use the edge to push things apart, this movable inclined plane is called a wedge. An ax blade is one example of a wedge.

Lever - Any tool that pries something loose is a lever. Levers can also lift objects. A lever is an arm that "pivots" (or turns) against a **fulcrum** (the point or support on which a lever pivots). Think of the claw end of a hammer that you use to pry nails loose; it's a lever. A see-saw is also a lever.

Pulley - Instead of an axle, a wheel could also rotate a rope, cord, or belt. This variation of the wheel and axle is the **pulley**. In a pulley, a cord wraps around a wheel. As the wheel rotates, the cord moves in either direction. Attach a hook to the cord, and now you can use the wheel's rotation to raise and lower objects, making work easier. On a flagpole, for example, a rope is attached to a pulley to raise and lower the flag more easily.

Screw - When you wrap an inclined plane around a cylinder, its sharp edge becomes another simple tool: a **screw**. If you put a metal screw beside a ramp, it may be hard to see similarities, but a screw is actually just another kind of inclined plane. One example of how a screw helps you do work is that it can be easily turned to move itself through a solid space like a block of wood.

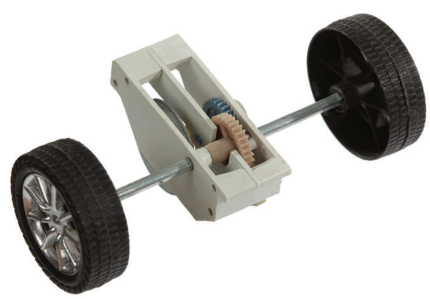
Inclined Plane

Lever

Wedge

Screw

Pulley

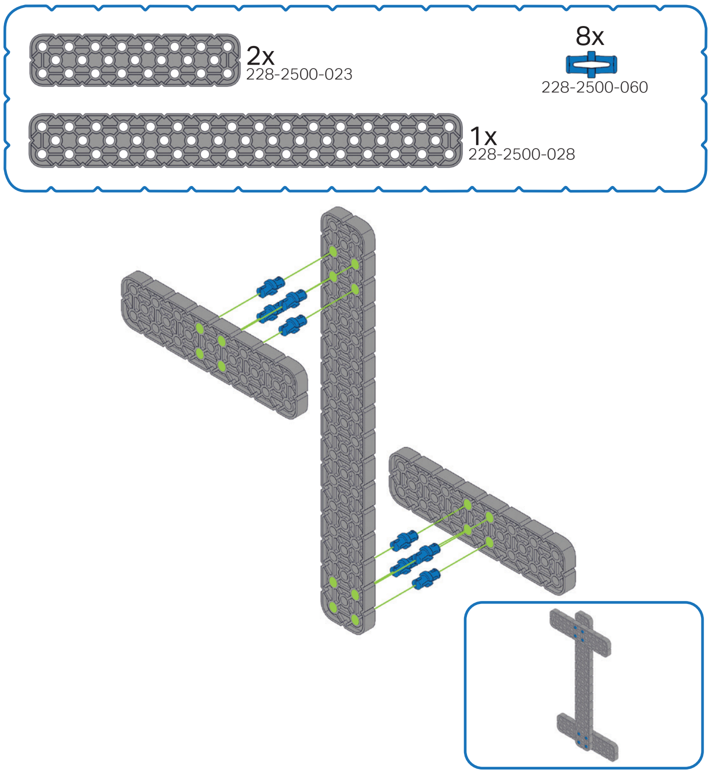


Wheel and Axle

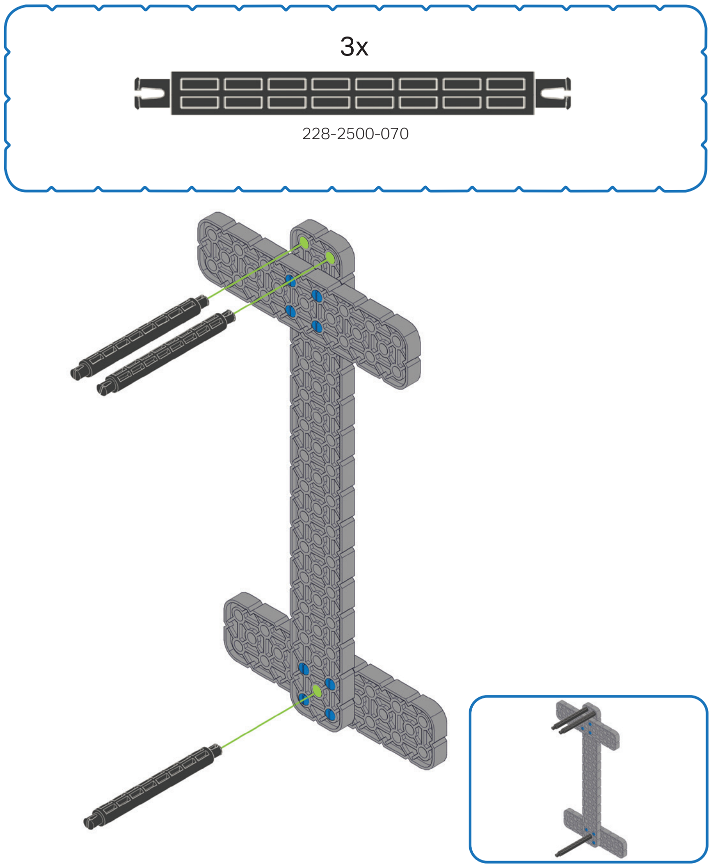
B.cs3

Pulley Assembly

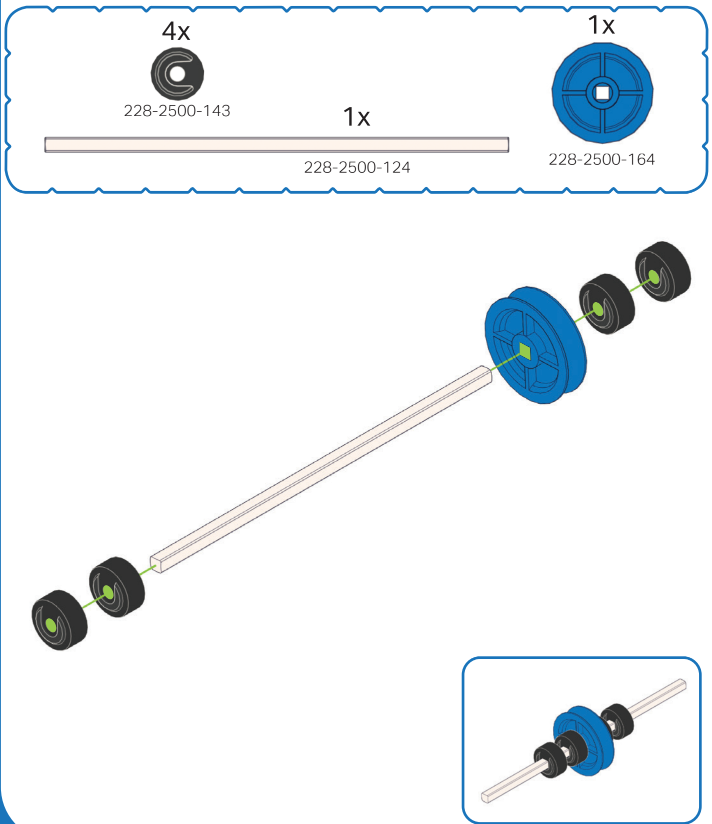
1



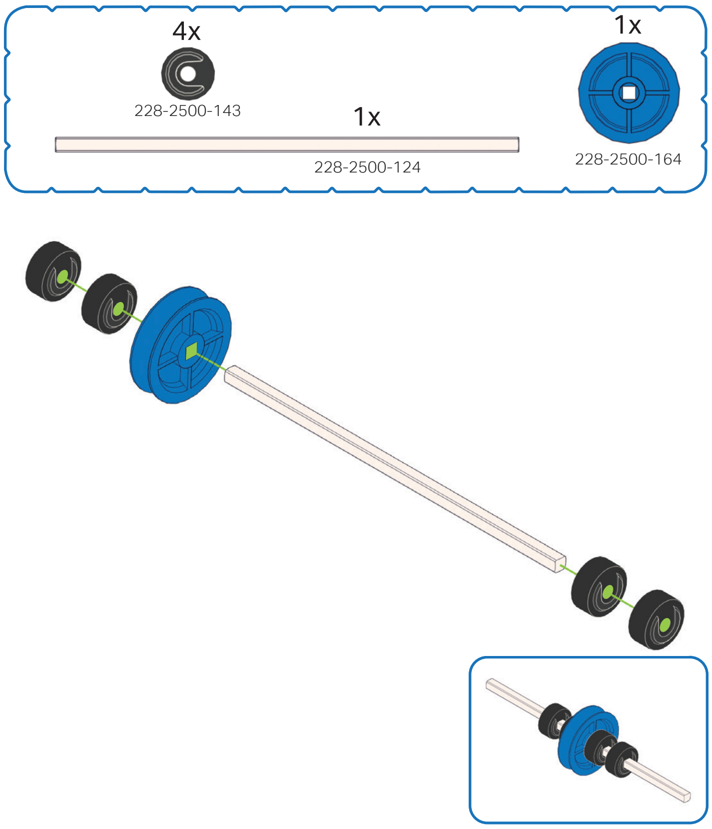
2



3

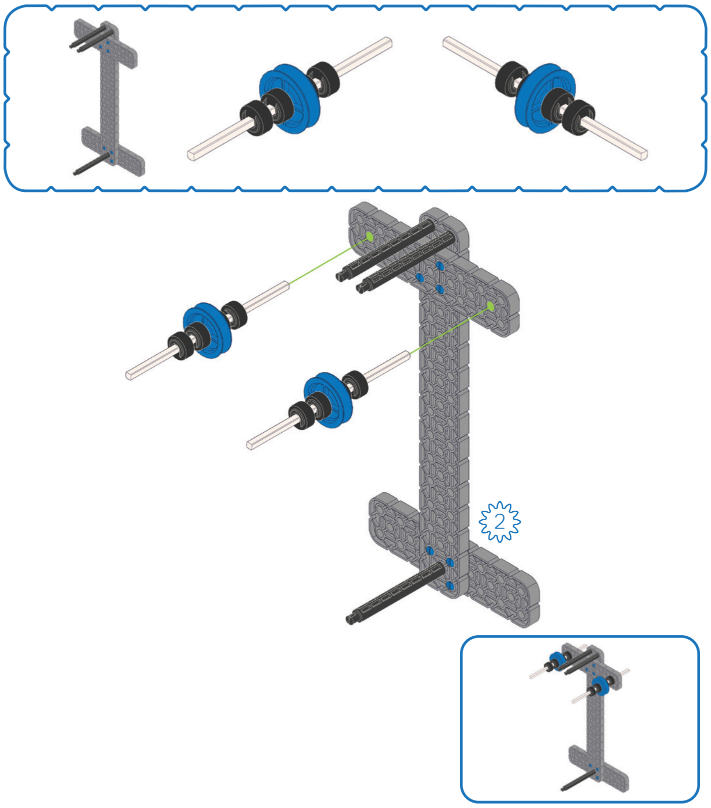


4

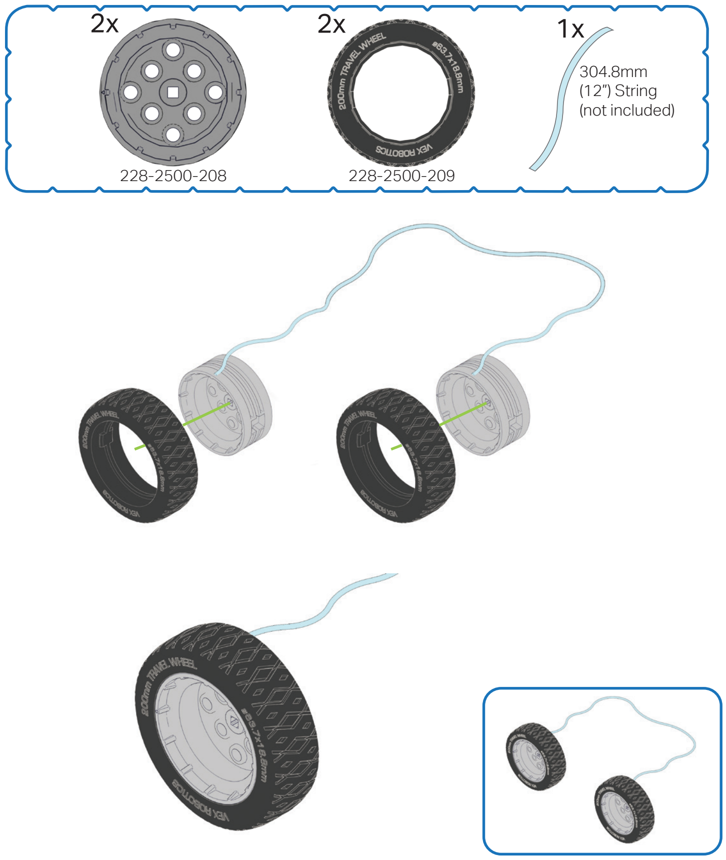


B.cs3 cont.

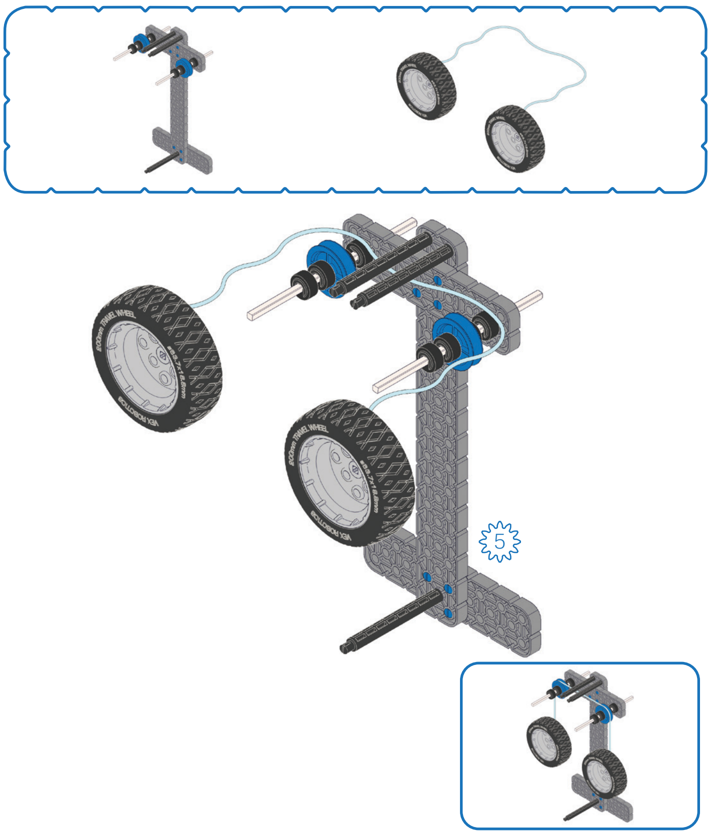
5



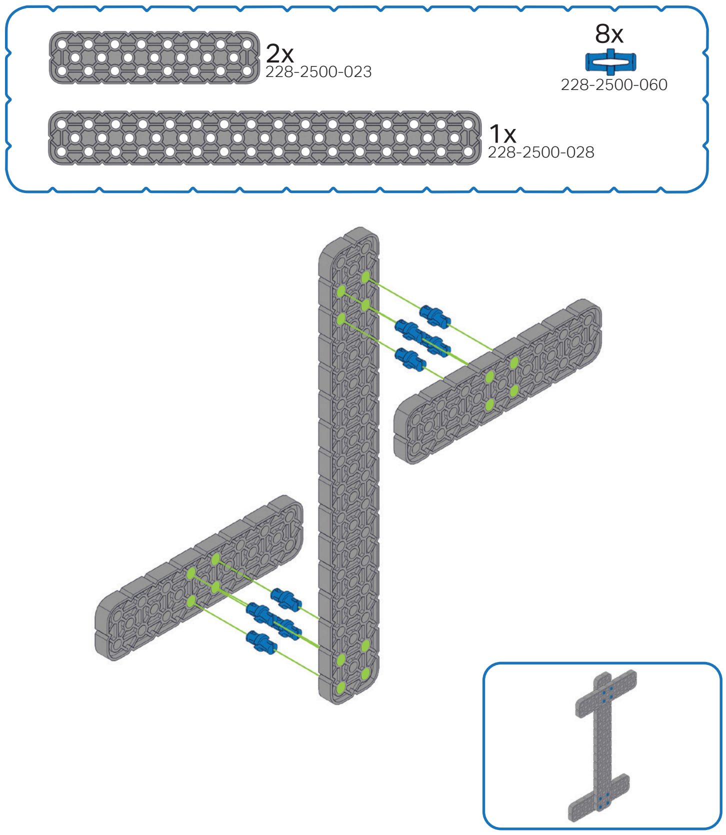
6



7

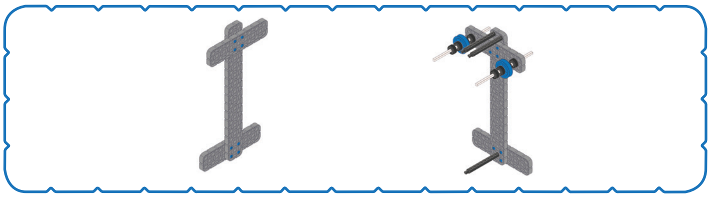


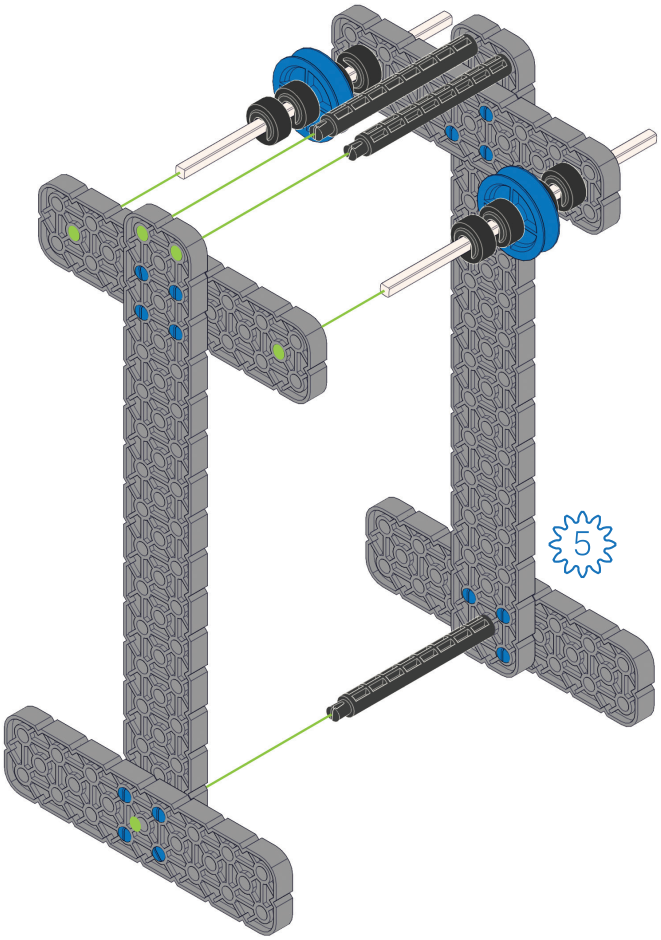
8

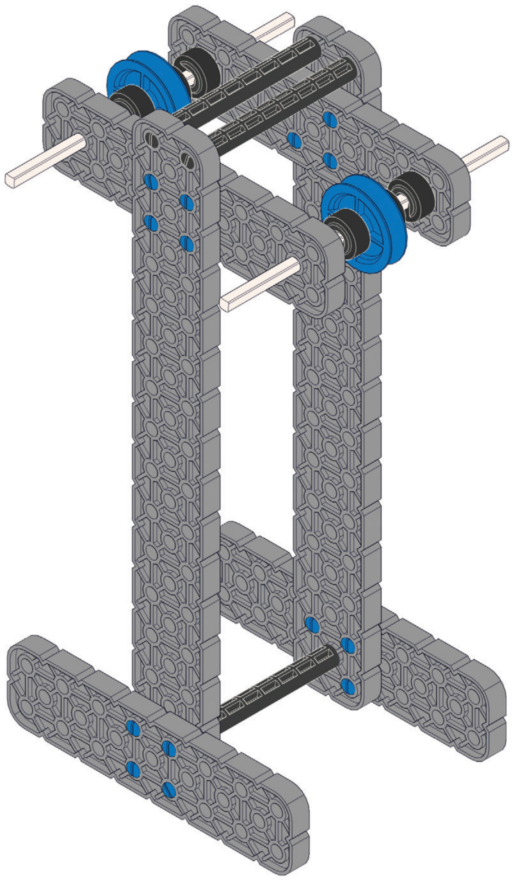


B.cs3 cont.

7

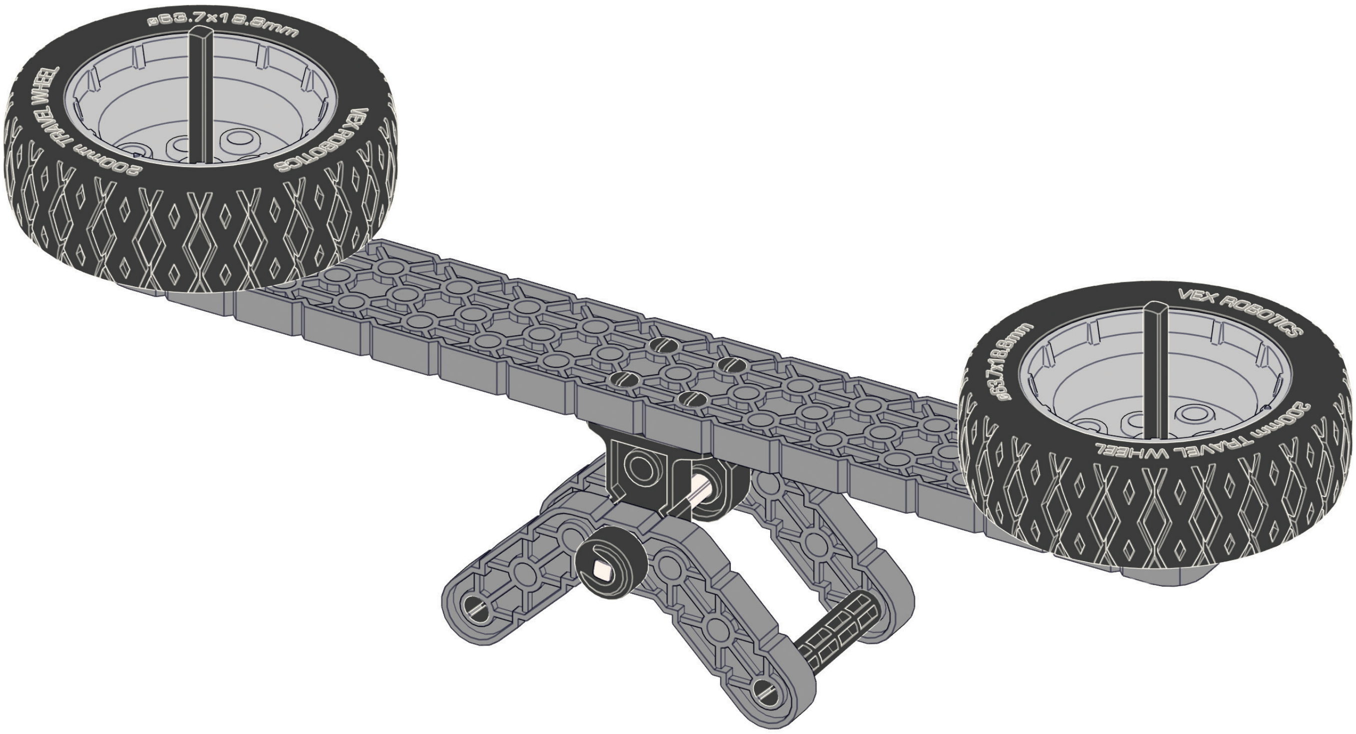






C

Lever



C.cl1

Overview of Activity:

Campers will build a Lever and consider every day uses for this simple machine. They will modify the assembly for use as a catapult that can launch an object a pre-determined distance. Campers will experiment with length, height, and force to improve accuracy and/or distance. They will document their changes with drawings or a mobile device.

Learning Objectives:

- Campers will become familiar with the basic building components of the VEX IQ Super Kit.
- Campers will use appropriate terminology to describe basic VEX IQ building components.
- Campers will be able to identify different kinds of simple machines visually.
- Campers will modify an existing assembly for a new purpose.
- Campers will document their machine and its use.

Suggested Timing:

Up to 1 hour

C.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Extra VEX IQ Pulleys (30 mm or 20mm) to launch with the catapult
- Think, Do, Test – Lever
- The Six Types of Simple Machines
- VEX IQ Build Instructions - Lever Assembly

Optional Materials:

Quarters, cleaning sponge cut into cubes, or other objects of different shapes and weights to launch with the catapult. It is OK if objects bounce a little bit, but should not generally roll upon landing.

Pre-learning Suggestions:

- If possible, provide campers with some time to inventory and sort their VEX IQ Super Kits.
- This activity will help campers become more familiar with some of the basic construction elements of the kit.
- It is important to reinforce the habit of an organized workspace and kit.

Based on VEX IQ Robotics Education Guide:

- The Six Types of Simple Machines (D.3)

C.cl2

Detailed Directions

1. Provide each pair of campers with the VEX IQ Super Kit and the handout, Build Instructions - Lever Assembly.

2. Have students build the assembly as outlined in the build instructions and brainstorm a list of objects they use or see every day that use a mechanism like this.

   If needed, provide students with the handout, "The Six Types of Simple Machines" Background Information. This document, drawn from the VEX IQ Curriculum, offers background information on different kinds of simple machines, including levers.

   Examples of Levers in everyday life:

   - Hammer claw to remove nails that are stuck in a hard surface (Class 1 Lever)
   - Catapult (Class 1 Lever)
   - Scissors (Two Class 1 Levers)
   - Wheelbarrow (Class 2 Lever)

3. Have campers experiment for a few minutes with the lever and describe how it works with their partner or teammate.

4. Provide campers with the Think, Do, Test – Lever handout and let them know that they will be constructing a mechanism that acts as a catapult to launch small objects a short distance with some accuracy. Show them what kinds of objects they will be launching. The criteria for the catapult have been outlined on the handout.

   Questions to ask campers:

   - What might make this object easy or difficult to launch? Will it land smoothly? Will it bounce or roll away? Is it light or heavy? How will its weight and size affect its flight?
   - What kind of lever would work best to create a catapult?
   - How could you adapt this lever to create a catapult?
   - How can you secure or mount the designated object on your catapult so it can be launched?
   - What kinds of VEX IQ components can be used to increase the force with which the catapult fires?
   - How can you stabilize the catapult as it fires?
   - How can you test your catapult? How will you track the results of your test?

5. Have campers build their designs and complete tests with different configurations to create a catapult that is at least consistent and ideally both consistent and accurate. Results can be documented with drawings, photos, or made into videos using smart devices to show parents at the end of the day.

Tips and Best Practices:

- Experiment with different kinds of levers to build different kinds of catapults.
- Testing can be completed on the VEX IQ Challenge Field, where grid lines and holes can be used to measure accuracy and distance.
- Because VEX IQ components are very light in weight, it is a good idea to use rubber bands to provide force on the catapult (instead of weight). Encourage campers to experiment with the placement of the fulcrum and make observations about how this affects the effort needed to launch their catapult.
- Make sure to test completed catapults at least ten (10) times and to measure consistency and accuracy. Complete this process after each modification is made.

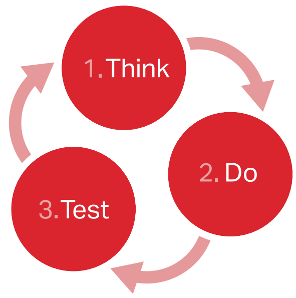
Extension:

- Create a precise target for the catapult using a cup or shoe box. Successful catapults will be able to launch an object into the target at least one (1) out of five (5) times.
- Launch objects over an obstacle, such as a stack of books.

C.cs1

Think, Do, Test - Lever

Student Name(s): ____



Instructions:

After you've completed building the **Lever Assembly**, think about how this simple machine could be used to launch an object, like a catapult.

Criteria for catapult design:

- The catapult must be constructed entirely of VEX IQ components.
- The catapult must use a lever of some kind (Class 1, Class 2, and Class 3).
- The catapult must launch the designated object at least one foot (1') in distance. It must achieve this two (2) out of three (3) times in a group test.

1. "THINK" - Think carefully about the object you will be launching. Make observations about its physical characteristics (size, weight, shape) and how these characteristics will affect its flight or landing.

Draw the objects here and write notes or labels to describe your observations about them and the potential challenges you'll face

2. "DO" – Draw and describe the assembly you will build from your lever to transform it into a catapult. Name it, label its parts, show and describe how it would work to contain and launch the object.

Draw, name, and label your catapult and how it will work.

C.cs1 cont.

3. "TEST" – Build your assembly and test it. Write down your observations in a notebook, or use a smart device to record your observations as video or audio. Make sure to test your catapult many times to make sure it performs consistently and accurately.

Make a note of any changes you make along the way.

Did the catapult consistently launch the object a distance of at least 1 foot? YES NO

If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until your catapult completes its task.

Problems ARE NOT failures. They are an expected part of the design process!

C.cs2

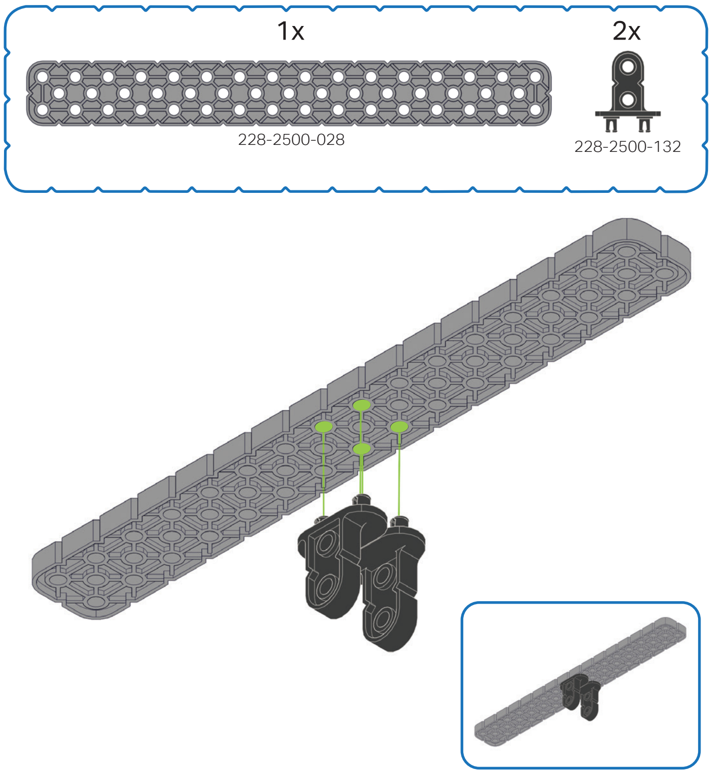
The Six Types of Simple Machines:

See B.cs2 for materials and instructions.

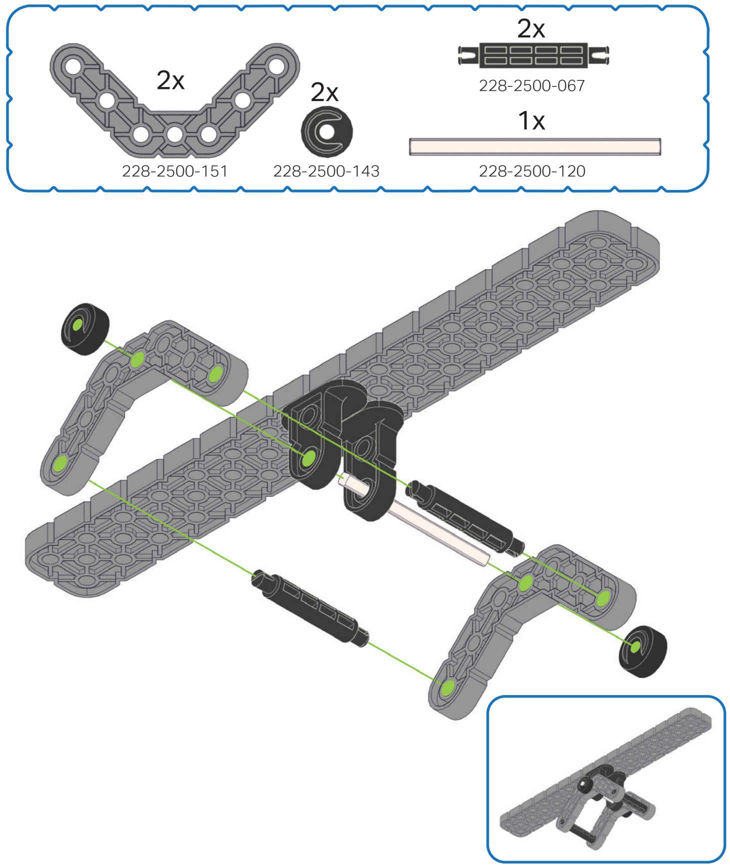
C.cs3

Lever Assembly

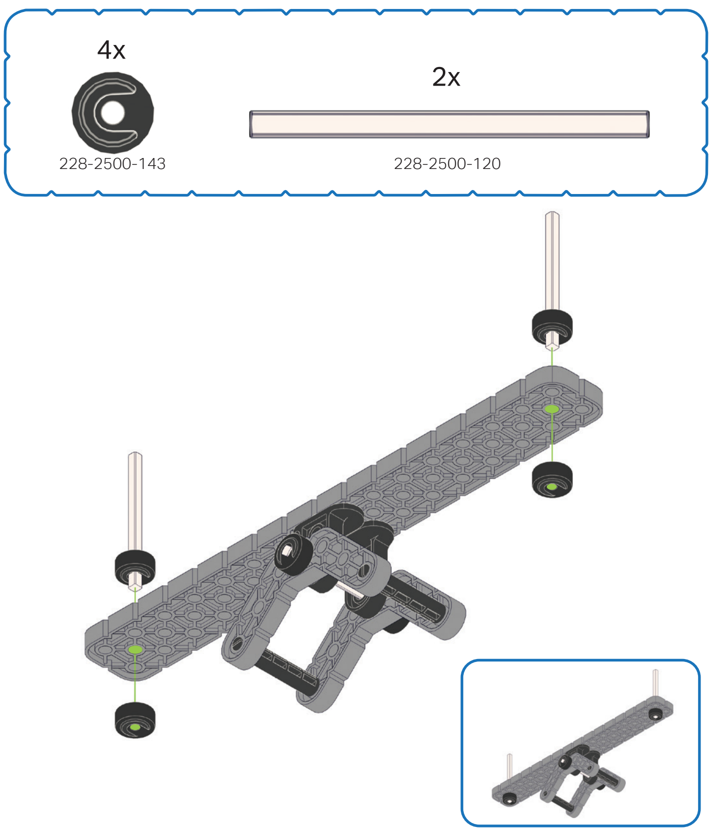
1



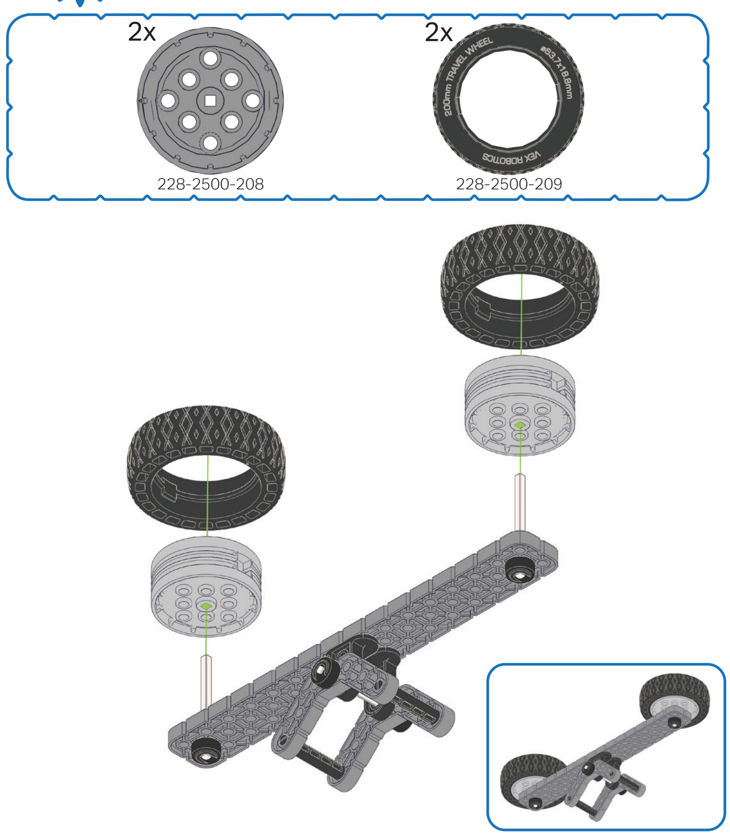
2



3

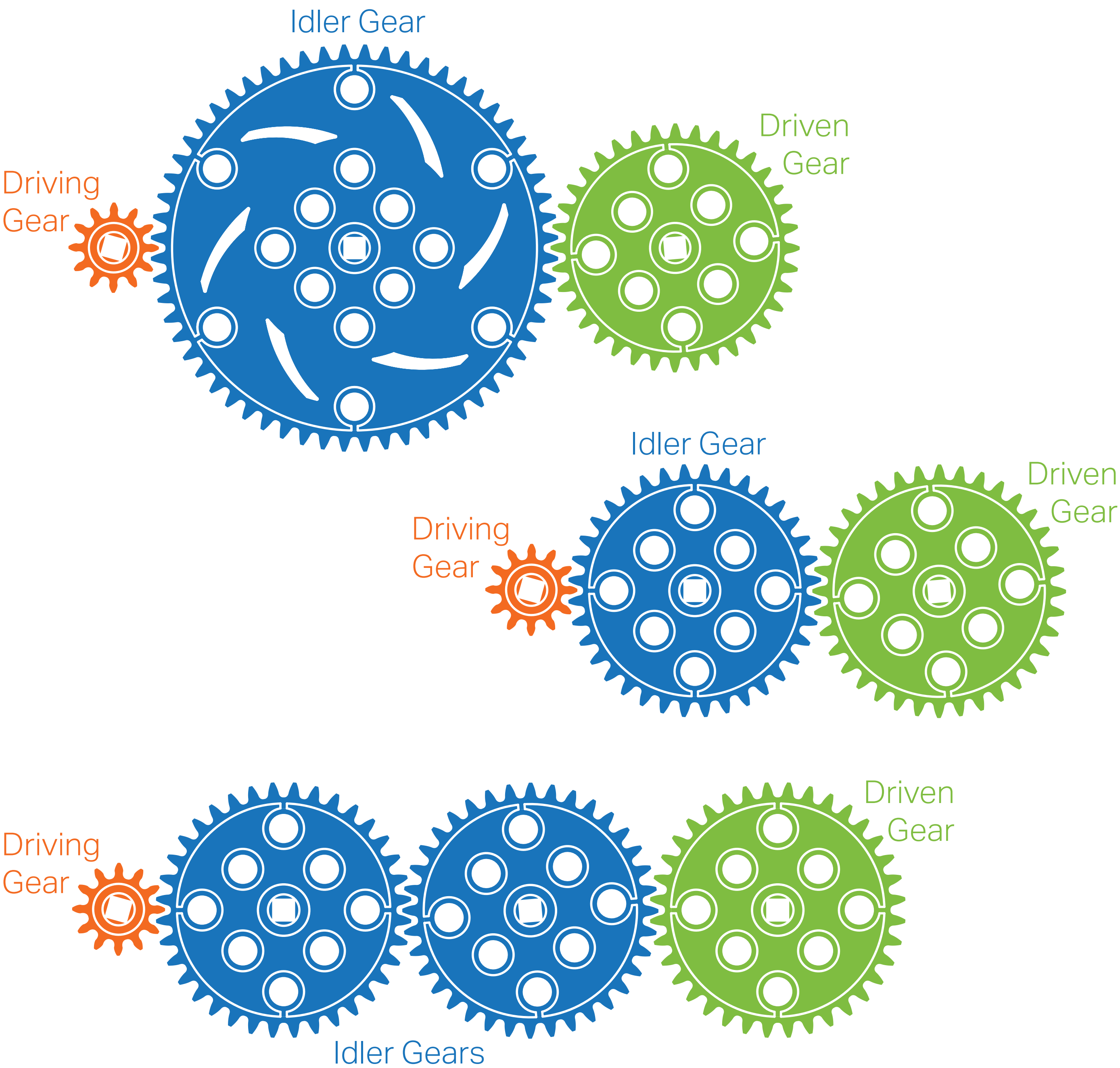


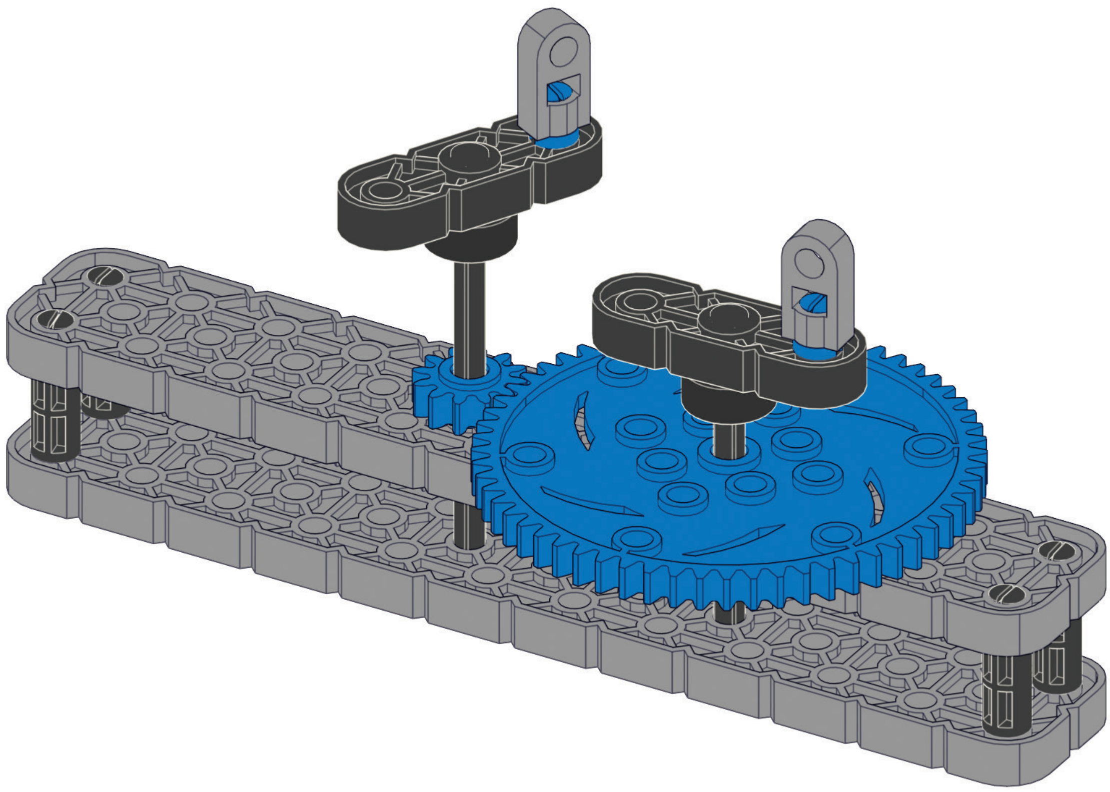
4



D

Gears and Gear Ratio





D.cl1

Overview of Activity:

Campers will build a gear ratio simulator and complete a small series of short challenges to increase speed or torque by trying out different combinations of driven, driving and idler gears. Each group will select one challenge randomly and cannot receive the next challenge until they have completed the first.

Learning Objectives:

- Campers will become familiar with the basic building components of the VEX IQ Super Kit.
- Campers will use appropriate terminology to describe basic VEX IQ building components.
- Campers will learn about and apply basic knowledge of Gear Ratio (Gear Train, Driving Gear, Driven Gear, Idler Gear, and Gear Ratio).
- Campers will be able to identify different kinds of simple machines visually.
- Campers will document their machine and its use.

Suggested Timing:

Up to 1 hour

D.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Extra VEX IQ Pulleys (30 mm or 20mm) to launch with the catapult
- Gear Ratio Challenge Cards
- "Speed, Torque and Power" Background Information
- "Mechanisms: Gear Ratio" Background Information
- VEX IQ Build Instructions - Gear Ratio Simulator Assembly

Online Resources:

- VEX IQ Videos – Curriculum & Misc. – Gear Ratios - optional
  www.vexiq.com/documents-downloads/
- VEX IQ – YouTube Playlist – VEX IQ Mechanisms – Gear Ratio - optional
  www.vexiq.com/videos
- VEX IQ – YouTube Playlist – VEX IQ Key Concepts – Speed, Torque, and Mechanical Advantage - optional
  www.vexiq.com/videos

Optional Materials:

Envelopes in which to place the challenge cards. If using these, each team should receive 4 envelopes, each with a different challenge. This adds to the excitement of the activity as they race to get their envelopes and open them quickly.

Pre-learning Suggestions:

- If possible, provide campers with some time to inventory and sort their VEX IQ Super Kits.
- This activity will help campers become more familiar with some of the basic construction elements of the kit.
- It is important to reinforce the habit of an organized workspace and kit.

Based on VEX IQ Robotics Education Guide:

- Speed, Torque, and Power (F.4)
- Mechanisms: Gear Ratio (G.3)
- VEX IQ Gear Ratio Simulator Assembly Instructions – Basic Gear Assembly, Assembly with Idler Gear, Assembly with Compound Gear Reduction (G.8)

D.cl2

Detailed Directions

1. Provide each pair of campers with the VEX IQ Super Kit and the VEX IQ Super Kit Contents and Build Tips poster. Have them use the poster to identify and locate a selection of gears of different sizes.
2. Have campers experiment with and make observations about the gears and share their observations with a partner.

   Questions to ask campers:

   - How do these gears compare with each other in size? (60 Tooth, 36 Tooth, 12 Tooth)
   - Where have you seen gears used in everyday life?
   - How can gears help us in robotics?

   Examples of Gears in everyday life

   - Old-fashioned clocks
   - Bicycles
   - Food mixers in the kitchen
   - Cars

   How gears help us in robotics:

   - Change direction of movement or rotation
   - Lift heavier objects
   - Turn more precisely
   - Move further or faster without increasing effort

   If needed, provide campers with the handouts, "Speed, Torque and Power" Background Information, and "Mechanisms: Gear Ratio" Background Information.

   Alternatively, if resources and time allow, have campers watch the following VEX IQ Videos. These are also available for download in the Documents and Downloads section of the VEX IQ Curriculum website.

   - VEX IQ Videos – Curriculum & Misc. – Gear Ratios - optional
     www.vexiq.com/documents-downloads/
   - VEX IQ – YouTube Playlist – VEX IQ Key Concepts – Speed, Torque, and Mechanical Advantage - optional
     www.vexiq.com/videos
   - VEX IQ – YouTube Playlist – VEX IQ Mechanisms – Gear Ratio - optional
     www.vexiq.com/videos
3. Encourage campers to use the gears they have pulled from the VEX IQ Super Kit to create different combinations of Driving Gears (on the left) and Driven Gears (on the right) and consider how this would affect speed and torque.

D.cl2 cont.

4. Provide campers with the handout, Build Instructions - Gear Ratio Simulator Assembly and give them time to construct it. Tell campers that they will use this simulator to complete a series of challenges to learn more about the work that gears help us do.

   For each challenge, they will need to:

   - Read the challenge card carefully.
   - Make modifications to their simulator to accomplish the task on the challenge card.
   - Think of and explain a scenario in robotics where this would be helpful. (e.g., making a drivetrain turn quickly and precisely, on an arm that moves a heavy object, slowing down or speeding up the robot's travel time without putting extra stress on the motor, or moving all the wheels on a drive train at once).
   - Show their camp leader their modifications and provide an example of a useful scenario. When the task has been confirmed as completed, campers will receive their next challenge.

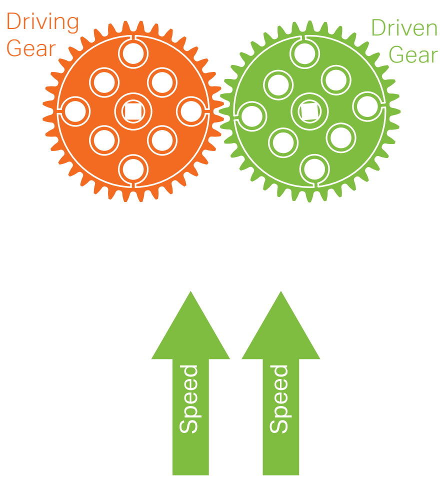
     If time allows, campers can draw a quick sketch and describe their ideas or take a photo or movie using a smart device. This can be shared with parents or each other.

5. Provide each pair of campers with their first challenge card (see D.cs1) face down or in an envelope and have them wait for your signal to reveal them. Begin the challenge, handing out new challenge cards as the previous challenge is completed.

D.cl2 cont.

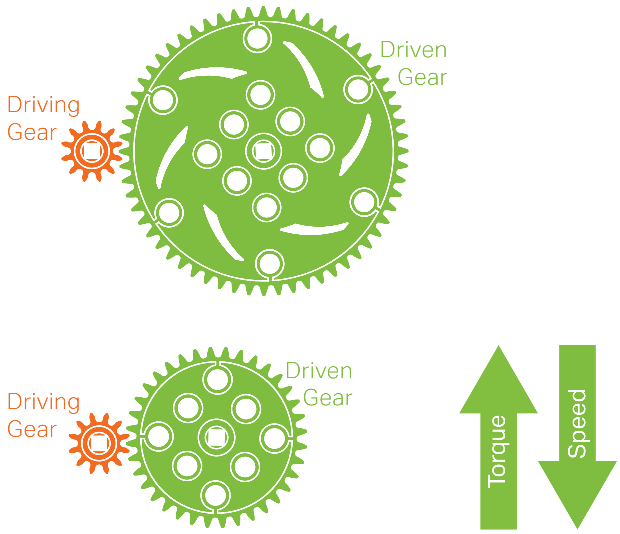
These sample answers provide some of the possible solutions to the Challenge Cards. The sample answers can also be used with campers who are "stuck" to show them one possible answer for the challenge, then ask them to come up with another solution that is also correct.

Challenge Card A Solution



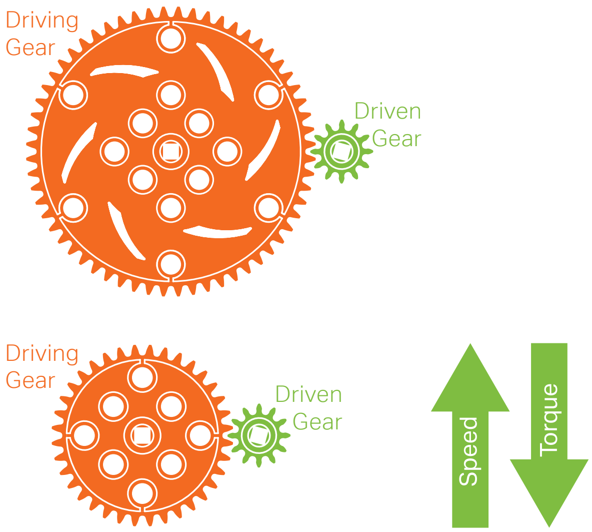
How is this useful?
Driving and Driven Gears of equal size are helpful when you want both gears to rotate at the same speed and torque, but in opposite directions.

Challenge Card B Solution



How is this useful?
A smaller Driving Gear is helpful when you are trying to move slower mechanically, lift heavier objects, and/or have more pushing ability.

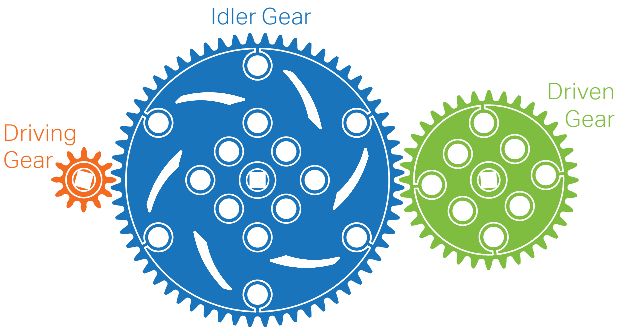
Challenge Card C Solution

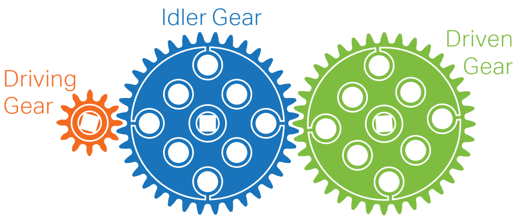


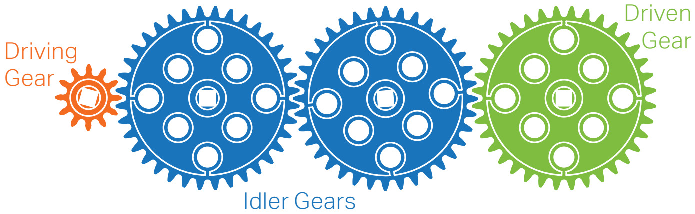
How is this useful?
A larger Driving Gear is helpful when you are trying to lift or move faster mechanically, you don't require the ability to lift heavy objects, and/or you favor agility over pushing ability in a drivetrain.

D.cl2 cont.

Challenge Card D Solution







An Idler gear does not affect the torque or speed.

Assemblies with two gears spin in opposite directions of each other.

In a gear train with one idler gear, they spin in the same direction.

In a gear train with two idler gears, they will spin in opposite directions again.

How is this useful?
Idler gears can be used to change the direction of rotation from a driven gear, or to reduce the size of the input/output gears without changing the spacing between the axles to which they are attached.

Tips and Best Practices:

- Align the gray standoff connectors before beginning simulation. Note which gear spins faster than the other by counting the number of revolutions until the standoff connectors are re-aligned.
- Different sized gears have different diameters. Campers will need to consider this when positioning the axles for different configurations

Extension:

- Create a gear assembly with three gears that increases speed and decreases torque, or vice versa.
- Modify the gear simulator assembly to function as a hand held lifting mechanism. Experiment with lifting objects manually using gears to demonstrate torque.
- Have students experiment with Compound Gears and Compound Gear Reductions and create challenge cards for each other.

D.cs1

Gear Ratio Challenge Cards

Challenge Card A

1. Create an assembly with two gears where input and output are equal for speed and torque.
2. Think of a situation in robotics where this would be useful.

Challenge Card B

1. Create an assembly with two gears where input and output result in increased torque and decreased speed.
2. Think of a situation in robotics where this would be useful.

Challenge Card C

1. Create an assembly with two gears where input and output result in increased speed and decreased torque.
2. Think of a situation in robotics where this would be useful.

Challenge Card D

1. Create a gear train assembly with more than two gears where there is no increase in speed or torque.
2. Think of a situation in robotics where this would be useful.

D.cs2

Speed, Torque, and Power

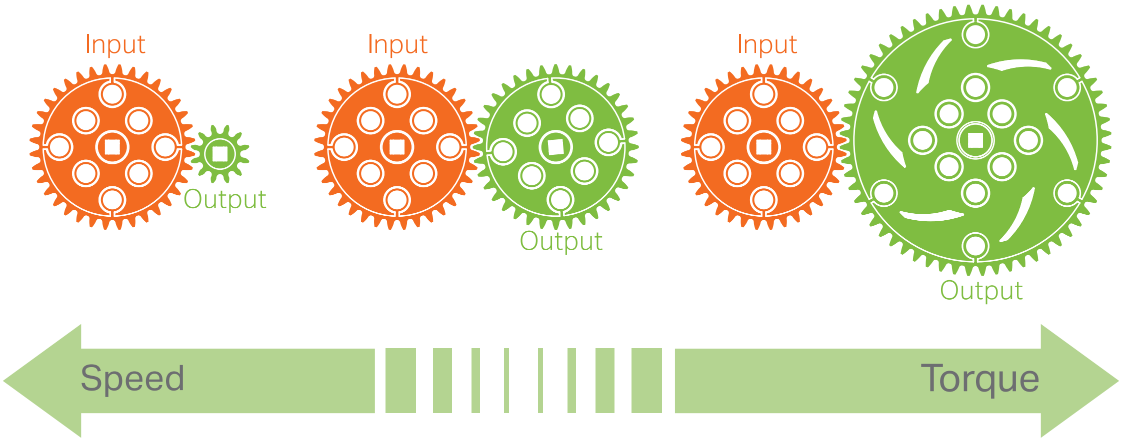
Speed is a way of measuring how fast an object is moving. Speed measures how far an object will travel over a given period of time. This measure is given in units of distance per time such as Miles per Hour or Feet per Second.

Torque is a force directed in a circle, most often rotating an object. Torque is a spinning force. When torque is spinning an object, the object will create a linear (straight line) force at its edge, such as an axle spinning a tire and causing the tire to move in a straight line along the ground. Torque is measured in units of force × distance, such as Inch-Pounds or Newton-Meters.

Power is the rate at which work is done. With VEX IQ, Smart Motors convert electrical energy into mechanical energy and produce power for a mechanical system. Power is most commonly measured in Watts.

The physical principles of Speed, Torque, and Power all fit together in what engineers call Classical Mechanics. In Classical Mechanics, speed and torque have an inverse (or opposite) relationship – as one increases the other decreases. Higher speed means lower torque, and higher torque means lower speed.

D.cs2 cont.



Also, **the amount of power supplied has an effect on how much speed and/or torque can be produced in a mechanical system.**

D.cs3

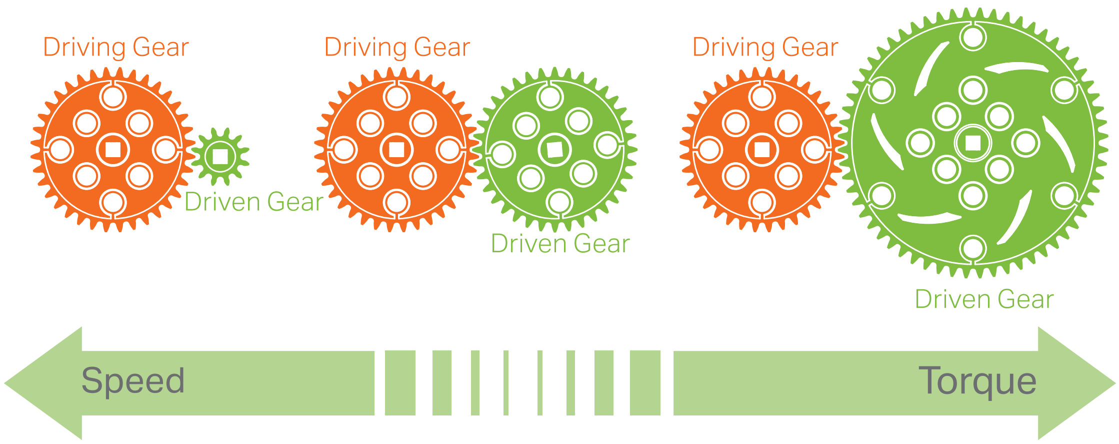
Mechanisms: Gear Ratio

Gear Ratio Basics

As you learned in other lessons, making a **Gear Ratio** change is one of the easiest ways to change Mechanical Advantage in a mechanism or system to achieve desired speed and/or torque. Gear Ratio expresses the relationship between a **Driving Gear** (the gear connected to the input power source, such as a motor) and a **Driven Gear** (the gear connected to the output, such as a wheel or mechanism) in a system.

When you have a system with a Driving Gear that is SMALLER than the Driven Gear you will increase Torque and decrease Speed:

Making this kind of change to Mechanical Advantage is helpful when you are trying to move slower mechanically, lift heavier objects, and/or have more pushing ability.



When you have a system with a Driving Gear that is LARGER than the Driven Gear you will increase Speed and decrease Torque:

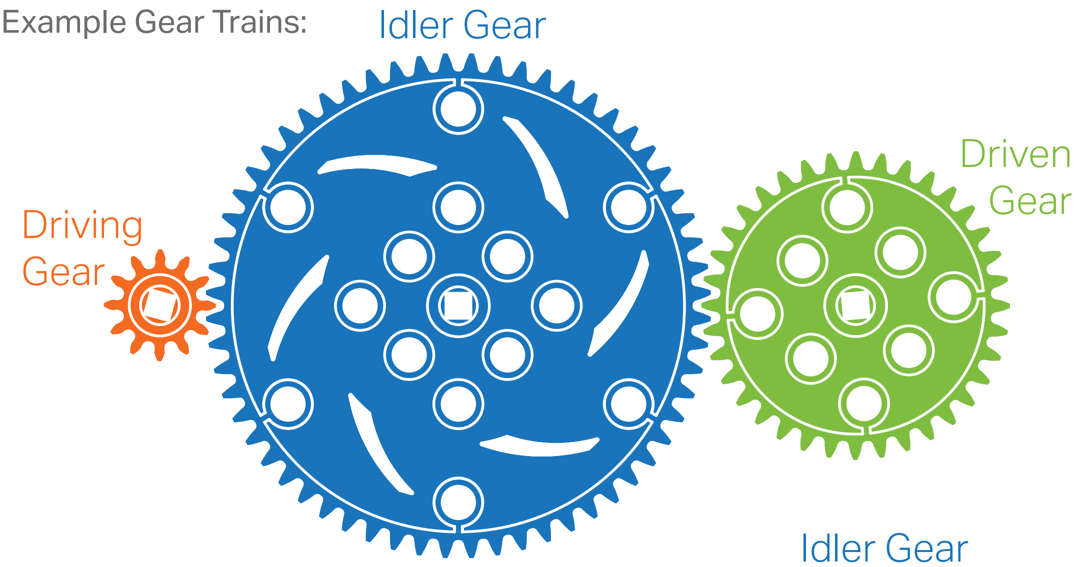
Making this kind of change to Mechanical Advantage is helpful when you are trying to lift or move faster mechanically, you don't require the ability to lift heavy objects, and/or you favor agility over pushing ability in a drivetrain.

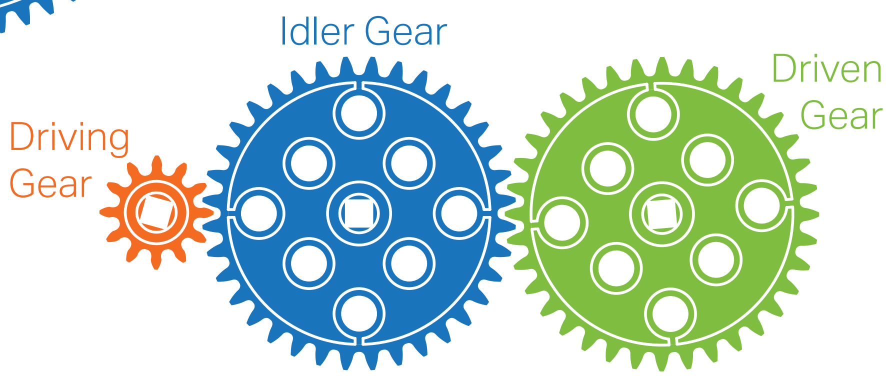
D.cs3 cont.

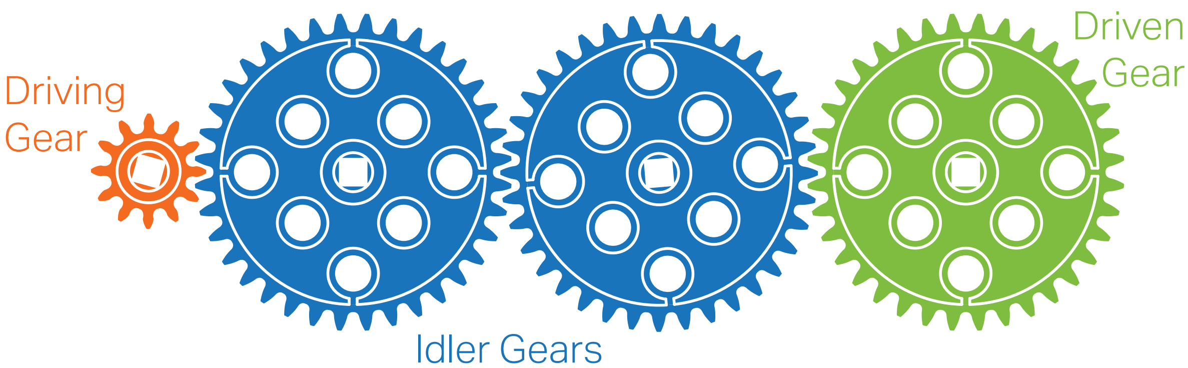
Gear Trains and Idler Gears

A simple **Gear Train** is a connected set of rotating gears that transmits power from an input (like a Driving Gear connected to a motor) to an output (like a Driven Gear connected to a wheel or mechanism). Simple Gear Trains can have any number of gears in a single row. All gears in between the Driving Gear and the Driven Gear that only transmit power are known as Idler Gears. Idler Gears have NO EFFECT on Gear Ratio or Gear Reduction, regardless of the number of teeth they have.

Example Gear Trains:







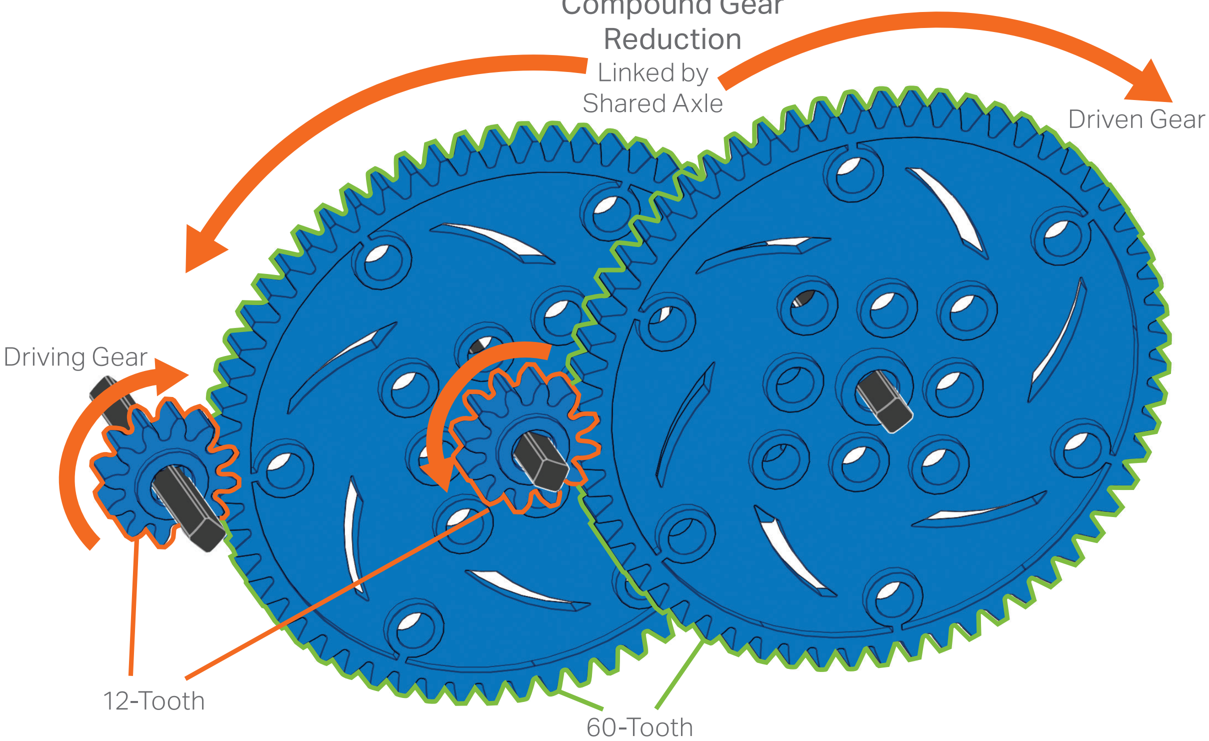
In all three of these example **Gear Trains**, the **Driving Gear** is 12-teeth and the **Driven Gear** is 36-teeth, thus the **Gear Ratio** for all three examples is the exact same - 12:36 . Size and number of **Idler Gears** have no effect on **Gear Ratio** or **Gear Reduction**, they just transmit power!

D.cs3 cont.

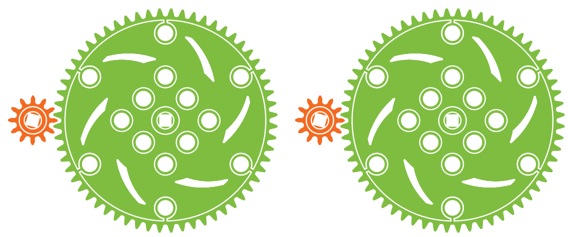
Compound Gears and Compound Gear Reductions

In certain situations, a design may require more mechanical advantage than a single gear ratio can provide or is otherwise impractical. For example, if a VEX IQ robot design requires a 12:500 gear ratio it is a problem because there is no 500-tooth gear available. In this situation, a designer can use multiple gear reductions in the same mechanism. This is called a Compound Gear Reduction.

In a **Compound Gear** system, there are multiple gear pairs. Each pair has its own **Gear Ratio**, and a shared axle connects the pairs to each other. The resulting Compound Gear system still has a **Driving Gear** and a **Driven Gear**, and still has a **Gear Reduction**. However, it's now called a **Compound Gear Reduction** and is calculated by multiplying the gear reductions of each of the individual gear pairs.



For the above example with 12-tooth and 60-tooth gears, the overall Gear Reduction is calculated this way:



Gear Reduction

(60 / 12) x (60 / 12)

(5 / 1) x (5 / 1) = 25 / 1

Say "25 to 1 Compound Gear Reduction"

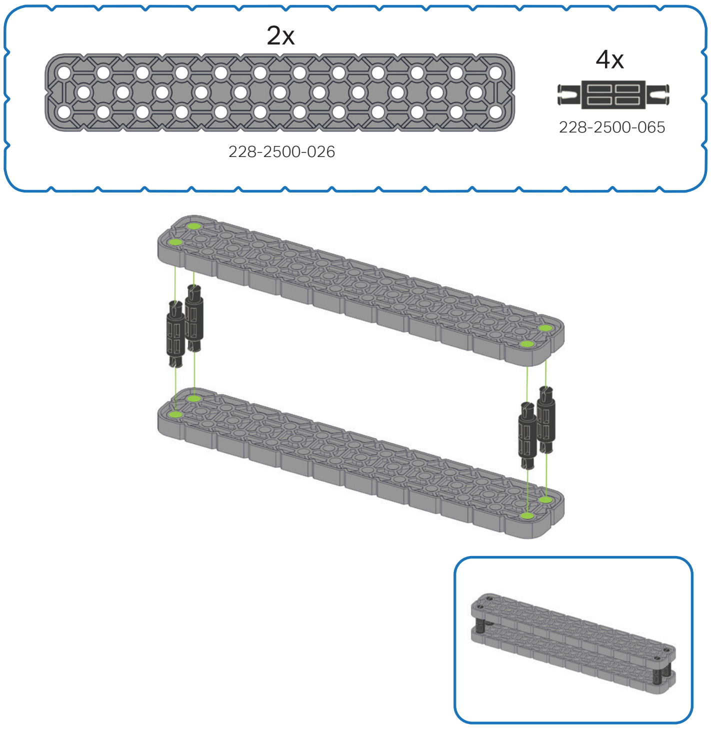
That means the output (**Driven Gear**) shaft is 25 times slower than the input (**Driving Gear**) shaft, and has 25 times as much torque. **Compound Gear Reductions** add up quickly!

Note: The VEX IQ Gear Ratio Simulator (G.8) and Gear Ratio Exercises (G.9) from the VEX IQ Robotics Education Guide can be used to help understand this section.

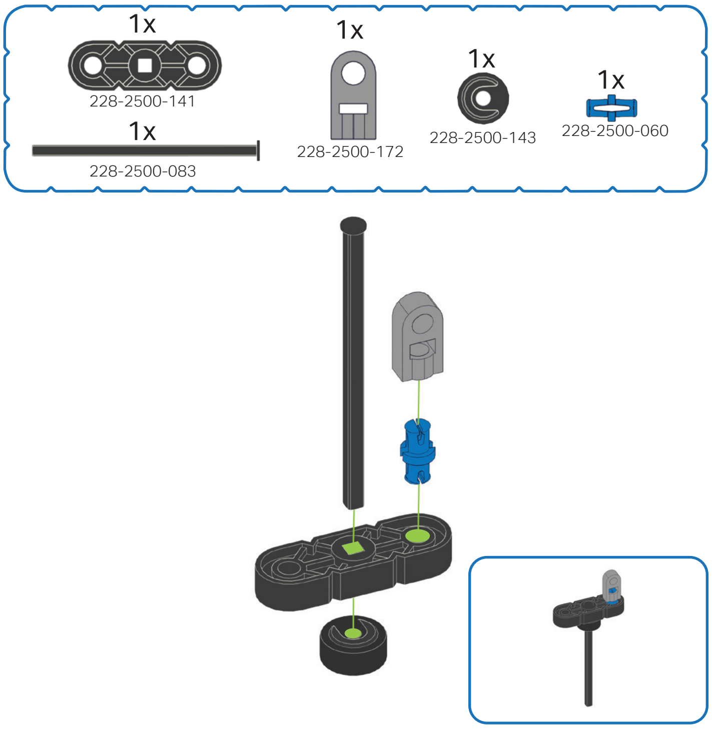
D.cs4

VEX IQ Gear Ratio Simulator Assembly Instructions

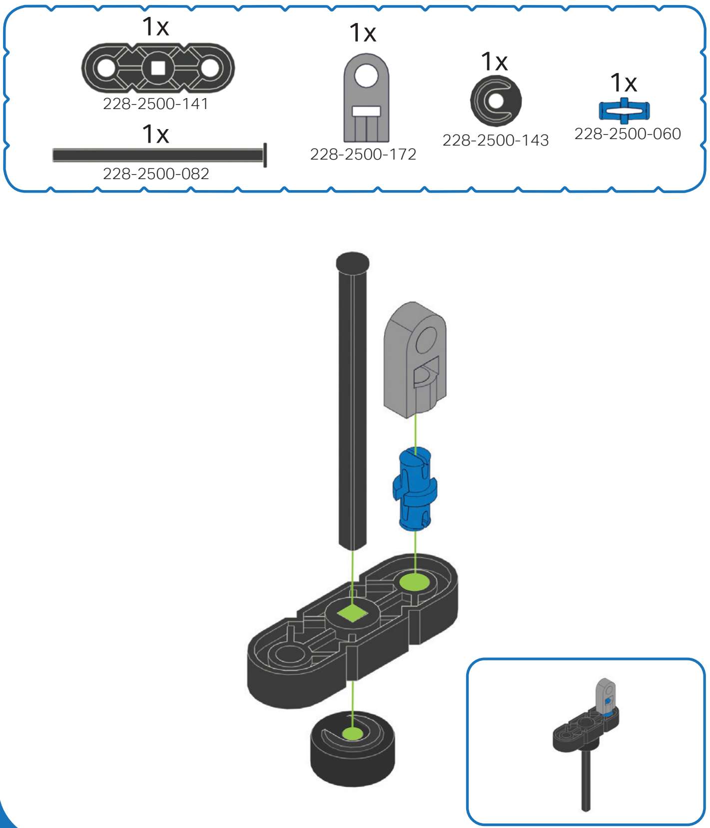
1



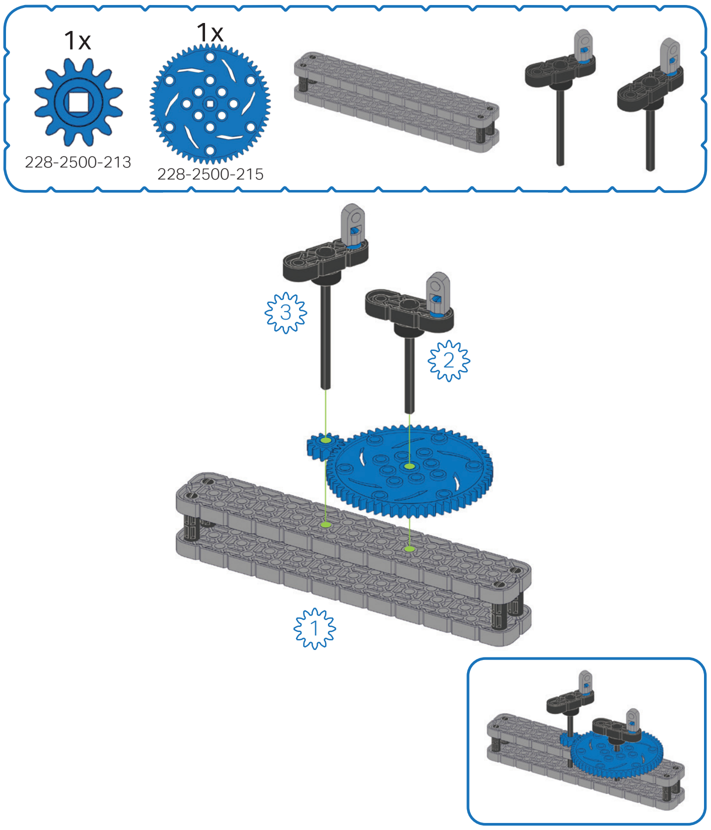
2



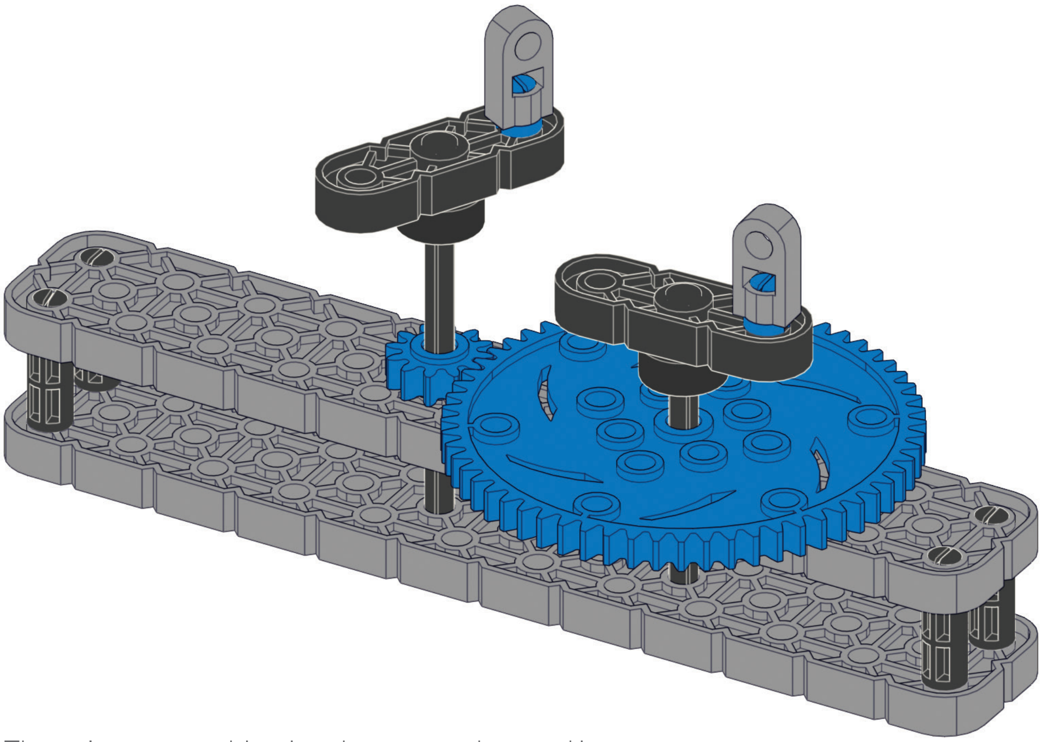
3



4

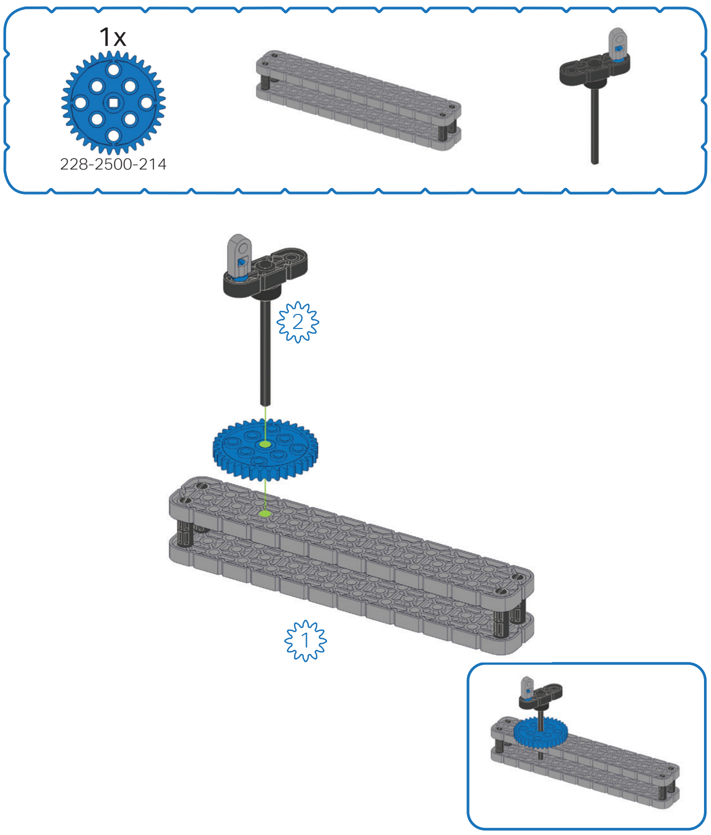


Basic Gear Assembly

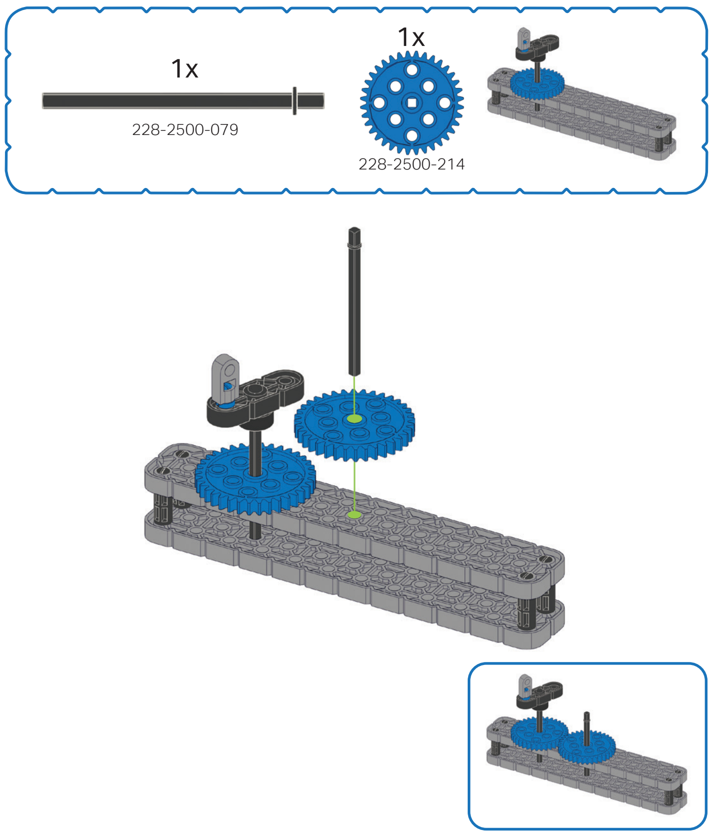


Note: The only gear combination that cannot be used is a 12-tooth gear on both input and output shafts.

5

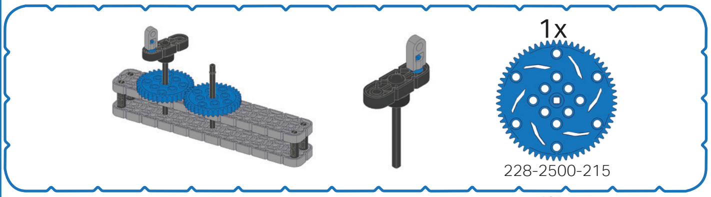


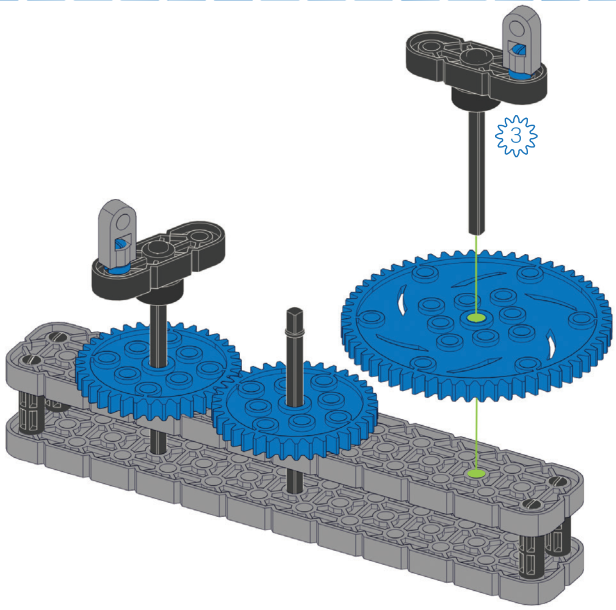
6



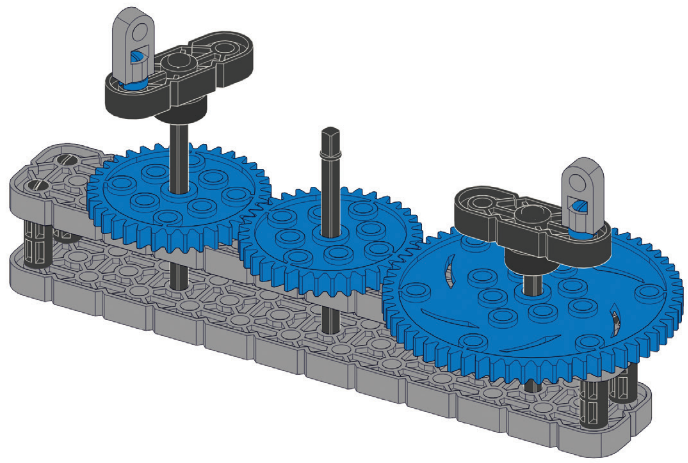
D.cs4 cont.

9

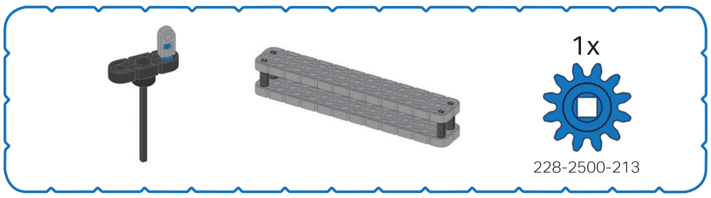


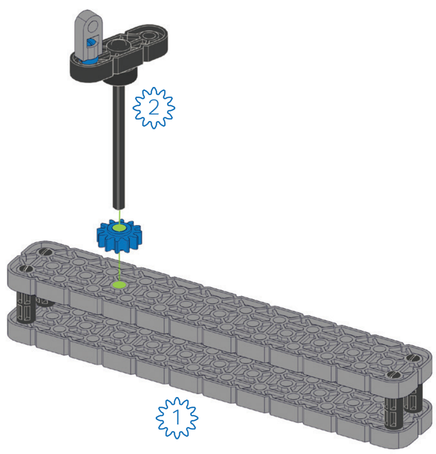


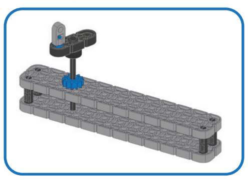
Assembly with Idler Gear



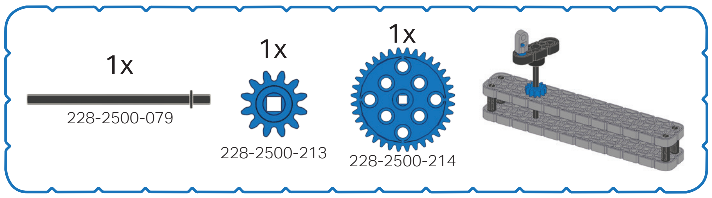
8

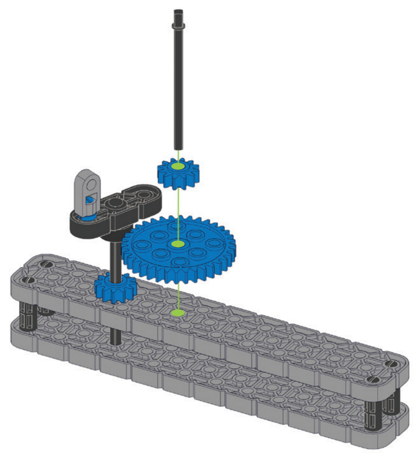


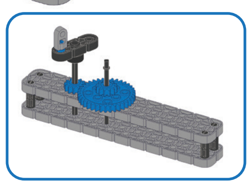




9

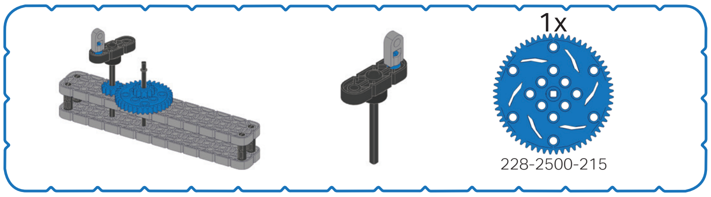


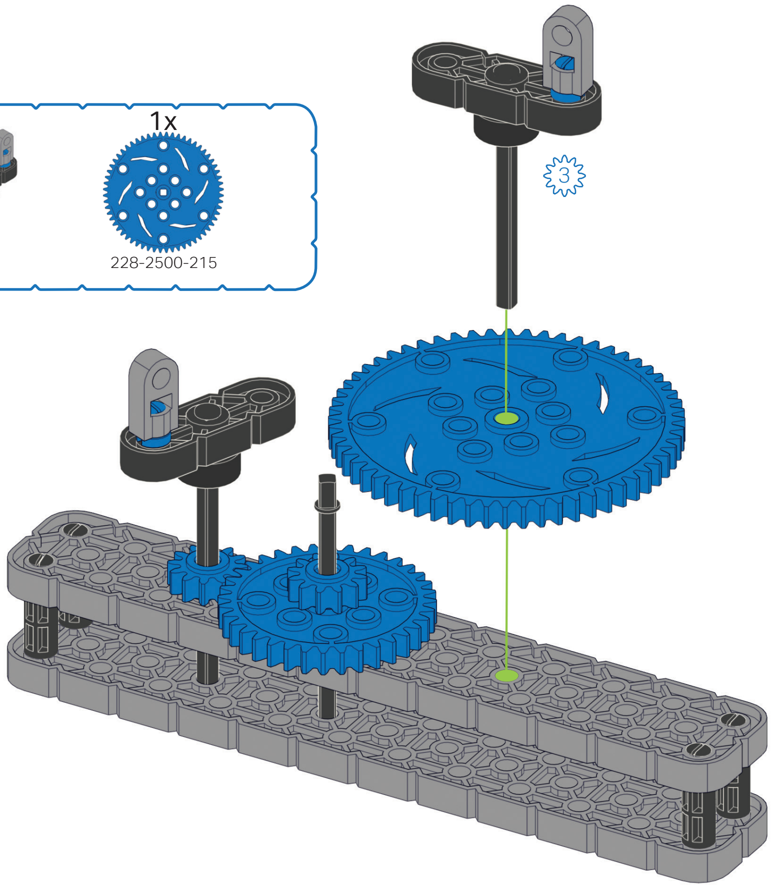




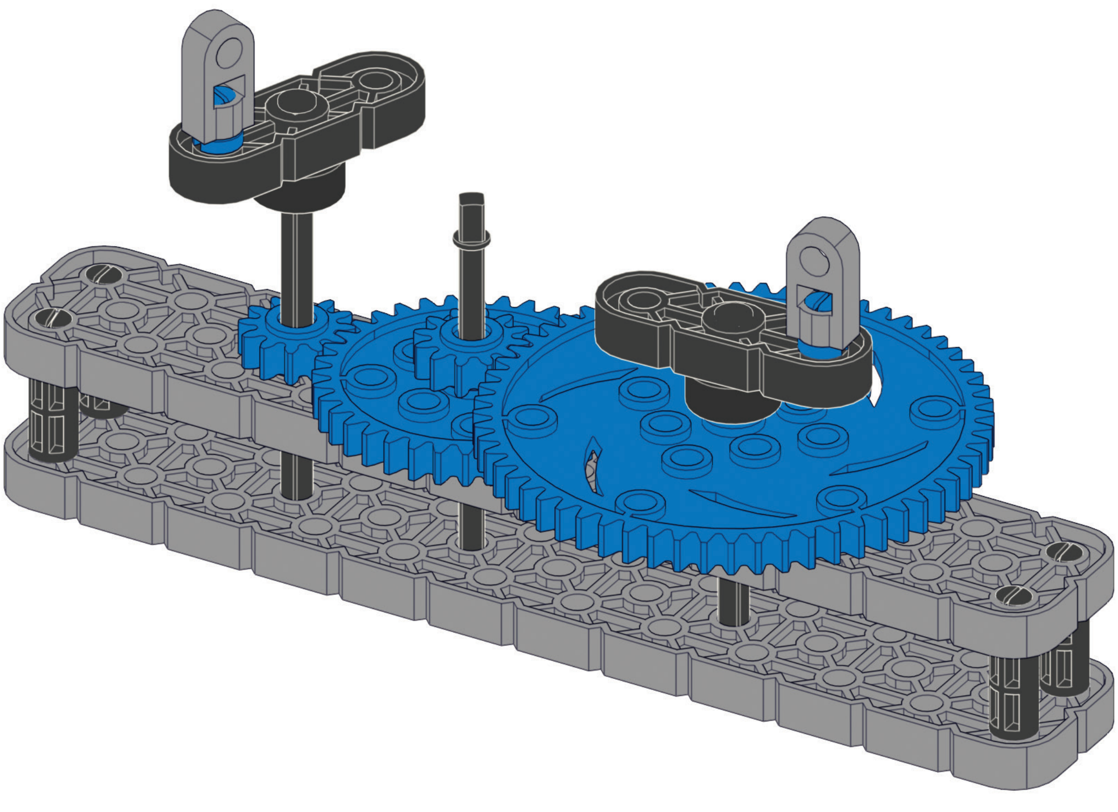
D.cs4 cont.

10



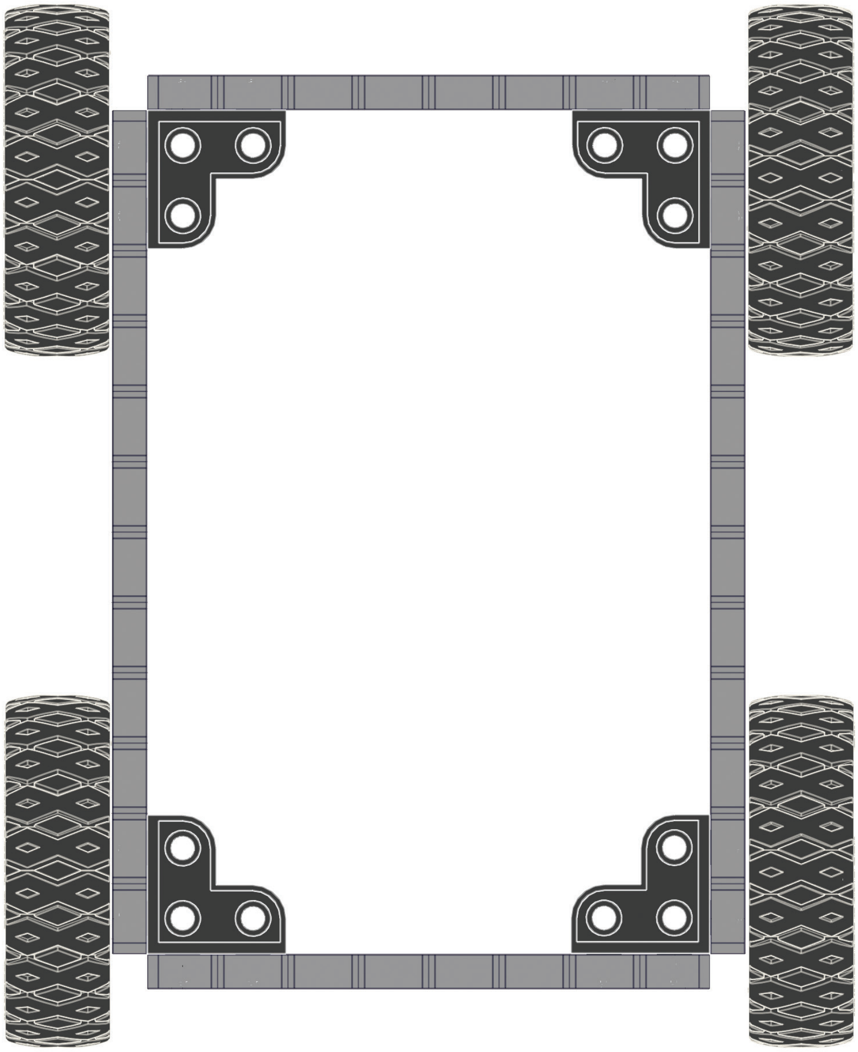


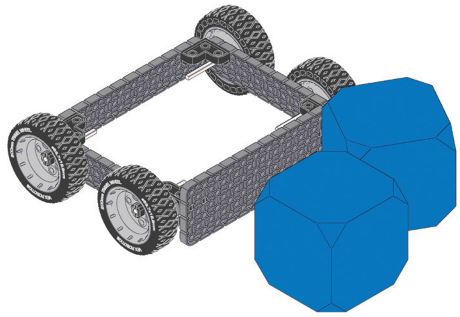
Assembly with Compound Gear Reduction

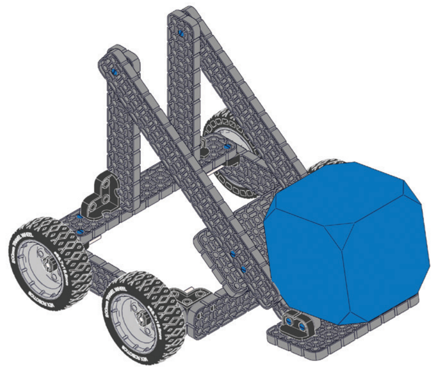


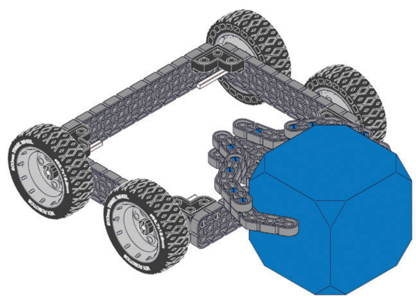
E

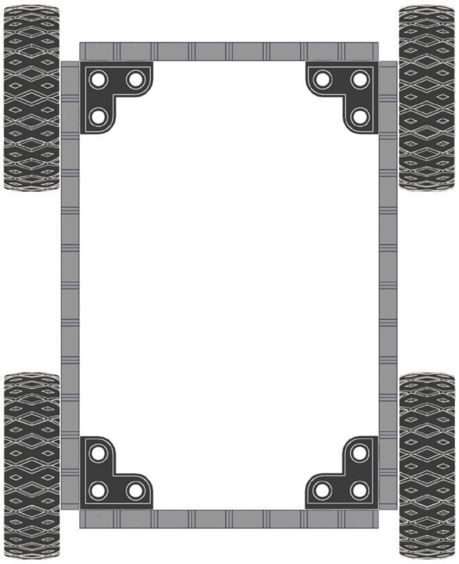
Drivetrain and Object Manipulator

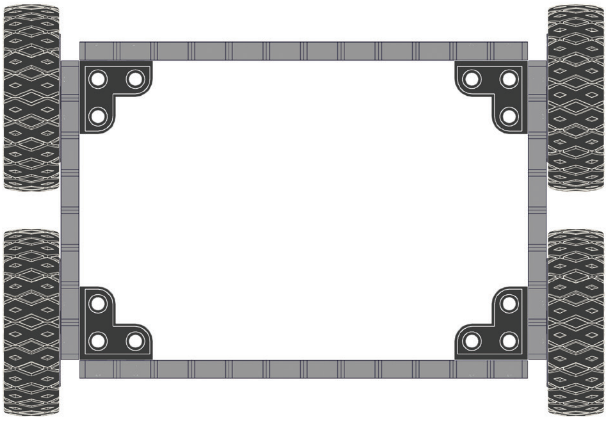












E.cl1

Overview of Activity:

Campers will construct a basic Drivetrain using two different widths of Wheelbase, first one and then the other. These will be used to learn about Turning Scrub and how it affects a robot's ability to turn. They will then build on this Drivetrain and experiment with different non-motorized object manipulators (Plow, Scoop, Friction Grabber) to accomplish the basic task of moving a stack of Beams, a Tire, or a Highrise Cube from one location to another. They will evaluate the advantages and disadvantages of each manipulator for different kinds of objects.

Learning Objectives:

- Campers will learn about Drivetrains and how features in their design affect Turning Scrub.
- Campers will learn about Object Manipulation.
- Campers will think critically about how design choices affect a mechanism's efficacy for a given task.

Suggested Timing:

Up to 2 hours

E.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Assorted objects to move (e.g., VEX IQ beams, tires, or Highrise cubes)
- Drivetrain and Course Handout
- Think, Do, Test – Drivetrains and Object Manipulators
- Mechanisms: Drivetrains
- Mechanisms: Object Manipulation

Optional Materials:

- Images of real world versions of different object manipulators. (e.g., construction vehicles)
- Different objects to move that provide different challenges. (e.g., balls that may roll, a plastic cup filled with beads or cotton balls that may tip over)

Pre-learning Suggestions:

- Time to get familiar with VEX IQ components and organization of the Super Kit.

Based on VEX IQ Robotics Education Guide:

- Mechanisms: Drivetrains (G.4)
- Mechanisms: Object Manipulation (G.5)

E.cl2

Detailed Directions

1. Provide campers with the VEX IQ Super Kit and the VEX IQ Super Kit Contents and Build Tips poster.
2. Show campers the example builds on the Drivetrain and Course Handout, and have them describe what they see with a partner.

   Questions to ask campers:

   - Why is this assembly important in robot design?
   - What purpose does it serve for the robot?
   - Which components are being used?
   - What is another method to make a robot move? (e.g., tracks)
   - Why would each configuration be a reasonable choice?

   If needed, provide campers with the handout "Mechanisms: Drivetrains" Background Information. This document, drawn from the VEX IQ Curriculum, offers background information on Drivetrain design and Turning Scrub.
3. Have campers look carefully at the Drivetrain assemblies and select the components they think can be used to build one.

E.cl2 cont.

4. Have campers use the images included on the **Drivetrain and Course Handout** to build a drivetrain with wheels on its long sides and manually maneuver this drive train around three Highrise Cubes in a line. Encourage them to make observations about how the drivetrain turns. Repeat the same task with wheels on the short sides of the drivetrain.

Note: The greater the Turning Scrub (friction) in a Drivetrain, the harder it is for a robot to turn.

Questions to ask campers:

- Which drivetrain was easier to maneuver around the turns?
- Which drivetrain configuration created less Turning Scrub (friction) as the robot turned?

5. When campers have selected their preferred Drivetrain configurations, tell them that they are now going to give their Drivetrain a job. Assign or have each pair of campers choose a kind of object to move.

   Campers may choose from these objects:

6. Provide campers with the handout "**Mechanisms: Object Manipulation**" Background Information. This document, drawn from the VEX IQ Curriculum, offers images of and background information on basic object manipulators including Plows, Scoops, and Friction Grabbers.
7. Provide campers with the handout, **Think, Do, Test – Drivetrains and Object Manipulators**. Give them time to experiment with, select, and design the best object manipulator for their selected object so that they can create an effective Object Mover. Encourage them to make observations about which manipulators worked well, which didn't, and why. Note that this may be useful to them in future robotics challenges with VEX IQ. The criteria for the object mover have been outlined on the handout.
8. When campers have created their Object Movers have them demonstrate moving their object from one location (on a flat surface) to another location (on a flat surface). Results can be documented with drawings, photos, or made into videos using smart devices to show parents at the end of the day.

Tips and Best Practices:

- If campers are struggling to move their selected or assigned object, have them experiment with another object. What they have designed may not work for their particular object, but may be very well-suited for moving another kind of object.

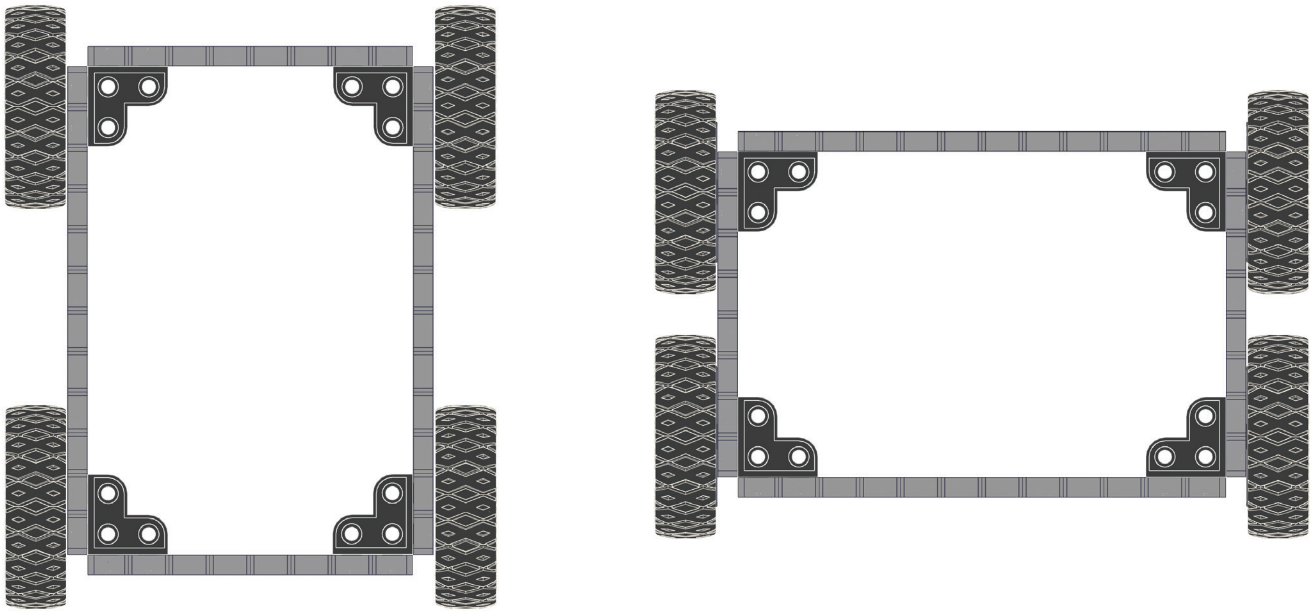
Extension:

- Make the object manipulators more dynamic (e.g., making arms articulate, using handles, levers, or pulleys to operate them).
- Expand the criteria for the challenge to a manipulator that can collect two or more different kinds of objects, or larger or smaller objects.

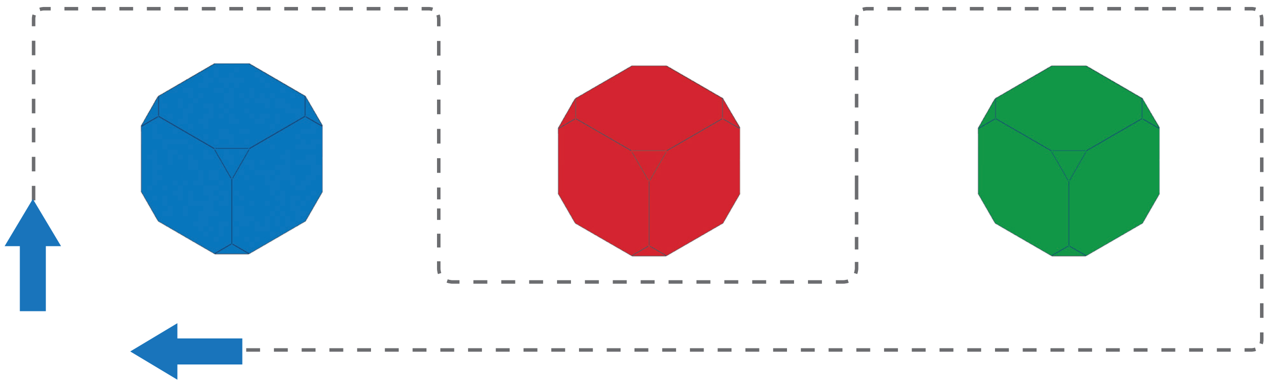
E.cs1

Drivetrain and Course Handout

Drivetrain Construction:

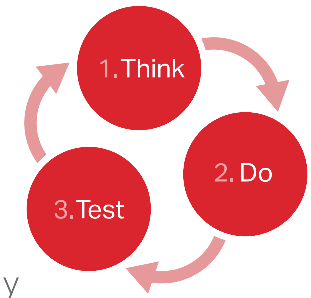


Drivetrain Testing Course:



E.cs2

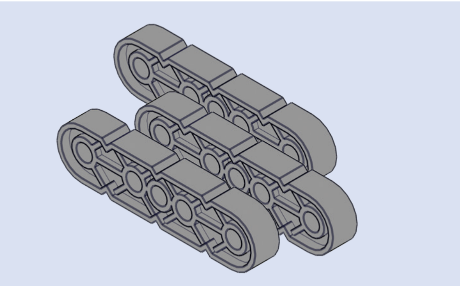
Think, Do, Test - Drivetrains and Object Manipulators



Student Name(s): ____

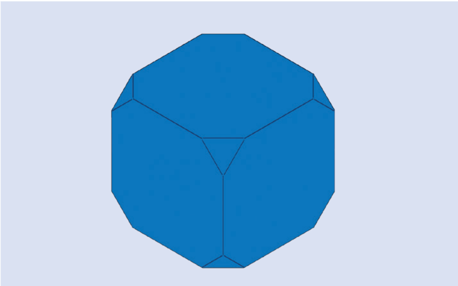
Instructions:

Select an object from the choices below. Create an object manipulator assembly that will attach to your drivetrain to move this object from one location to another on a flat surface.



A stack of VEX IQ Beams

A VEX IQ Tire



A Highrise Cube

Criteria for Object Mover design:

- The design must use VEX IQ components.
- The design must collect one kind of object or objects effectively.
- Campers may drive their assembly using their hands, and may touch parts of their assembly (e.g., opening and closing grabbers) but cannot touch the object they are moving directly.
- Object manipulators may be static (do not move) or dynamic (move).

1. "THINK" - Think carefully about the object(s) you will be moving. Make observations about the object and what might make it easier or more difficult to collect and move from one location to another location.

Draw and label your observations here:

E.cs2 cont.

2. "DO" – Draw and describe the object manipulator you will build and attach to your drivetrain. Show where and how it will attach to the drivetrain. Name it, label its parts, and show how it would work to collect and move the object from one location to another.

Draw, name, and label your object mover and how it will work.

3. "TEST" – Build your object manipulator and test it. Write down your observations in a notebook or use a smart device to record your observations as video or audio.

If you have time, test your Object Mover with different kinds of objects and make a note of the kinds of jobs for which it is well-suited. It may not work well for the object you have selected, but it may work very well for another object!

Did the Object Mover collect and move your object(s) effectively? YES NO

If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until you have an object manipulator that moves at least one kind of object effectively.

Problems ARE NOT failures. They are an expected part of the design process!

E.cs3

Mechanisms: Drivetrains

Mobile and Competition robots will vary greatly depending on the tasks they are designed for. However, one thing common among them is that they usually have some method for moving. The robotic subsystem that provides the ability to move is often known as a **Drivetrain**. Drivetrains may come in many different forms – two examples are wheels or treads (like a tank). The wheeled, rolling drivetrain is the most common one found in competition robotics and one of the most popular in the entire industry.

Drivetrain Design

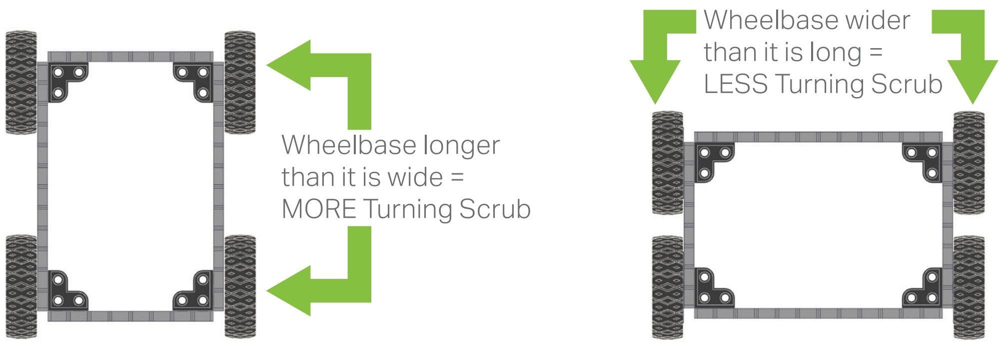
The most basic, multi-functional competition robot **Drivetrain** design consists of:

- A rectangular **Chassis** (the structure of a mobile robot that holds wheels, motors, and/or any other hardware used to make up a **Drivetrain**)
- Two Motors
- Four Wheels
- Gears transmitting Power from the Motors to all Wheels.

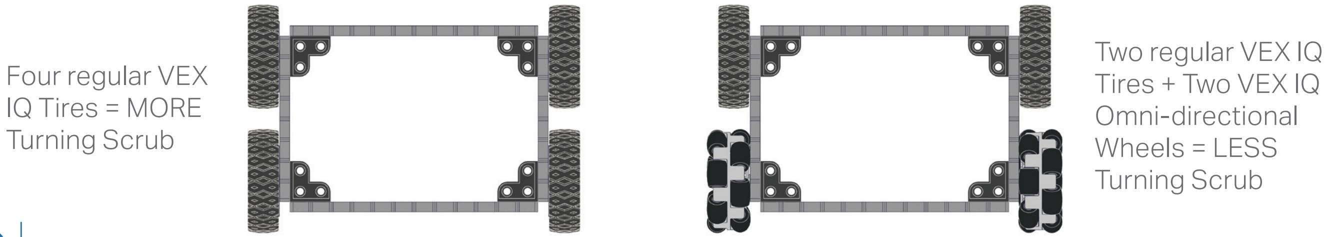
The Clawbot IQ Standard Drive Base is one example that you can build. However, **Drivetrains** can come in all shapes and sizes - some don't provide power to all wheels, use different types of wheels, or are not even a rectangular shape! Whatever the details of your **Drivetrain**, you should always be aware of a property known as **Turning Scrub**.

Turning Scrub is the friction that resists turning. This friction is created from the wheels dragging sideways on the ground as a robot (or other mobile vehicle) turns. The greater the **Turning Scrub** in a **Drivetrain**, the harder it is for a robot to turn. **Turning Scrub** in a basic Drivetrain can be easily managed and minimized in two ways:

1. Make sure that the **Wheelbase** (distance between **Drivetrain** wheels) is wider (side-to-side) than it is long (front-to-back):



2. Use different wheel and/or tire types to reduce the friction of **Turning Scrub**:



Try building the example **Drivetrains** above to see the **Turning Scrub** effect!

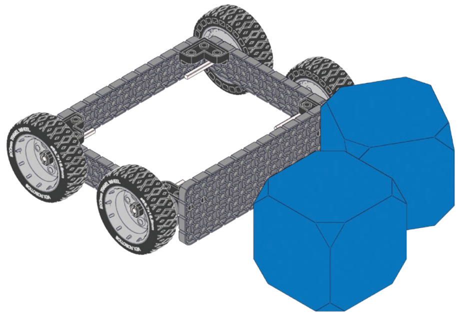
E.cs4

Mechanisms: Object Manipulation

In mobile and competition robotics, an Object Manipulator is a mechanism that allows a robot to interact with objects in its environment. There are three basic categories of **Object Manipulators**: **Plows**, **Scoops**, and **Friction Grabbers**.

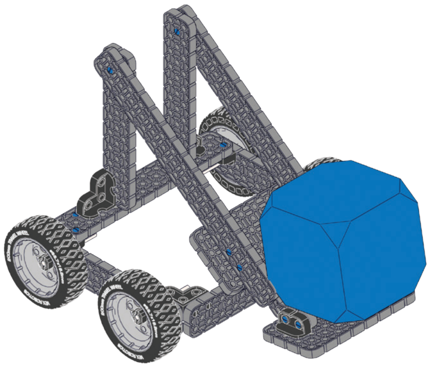
Plows

The first **Object Manipulator** category applies a single force to the side of an object. **Plows** move objects without actually picking them up and are by far the easiest manipulator type to design and build.



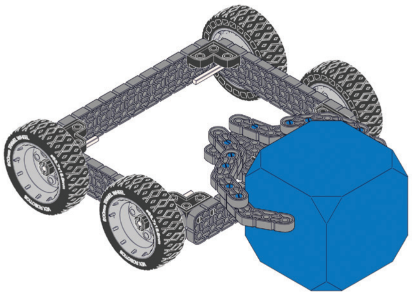
Scoops

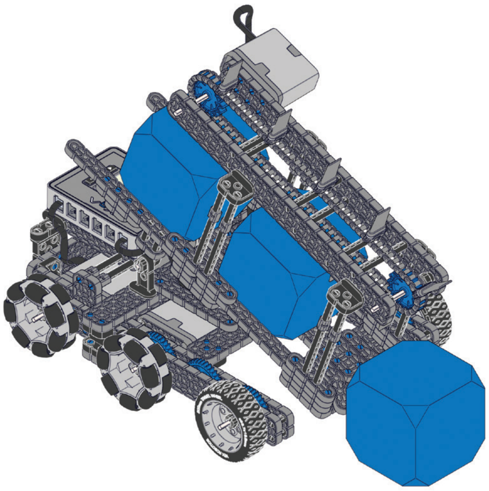
The second **Object Manipulator** category applies force underneath an object such that the object can be elevated and carried. Once an object is on a **Scoop**, it can be lifted and lowered relying on gravity to keep the object on or in the **Scoop**.



Friction Grabbers

The third **Object Manipulator** category applies a force to an object in at least two places, allowing the object to be pinched or grabbed. Thus, **Friction Grabbers** have the ability to hold objects securely and are generally the manipulator type that provides the most control over objects. The most common form of this manipulator type is a pinching claw.

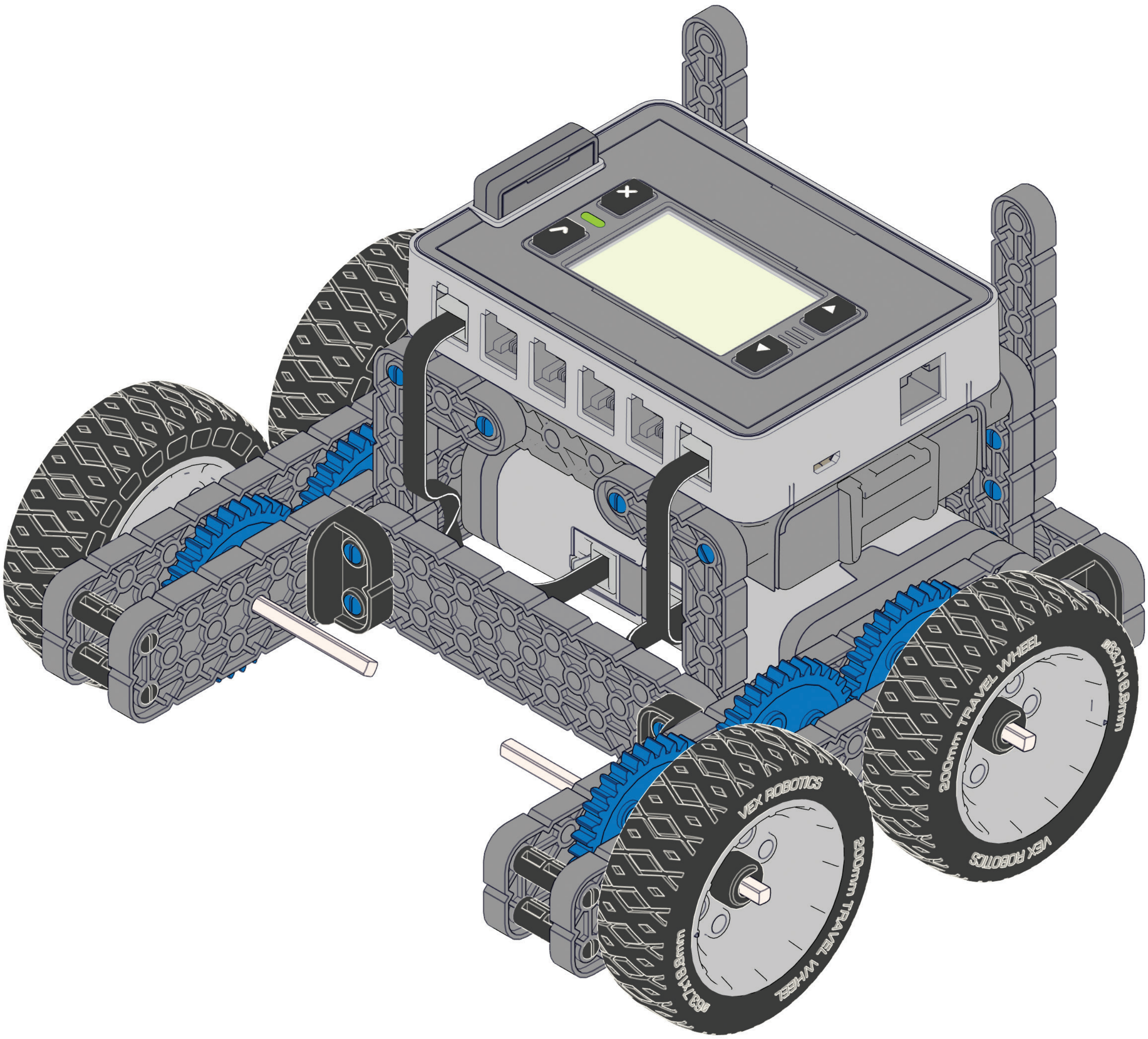


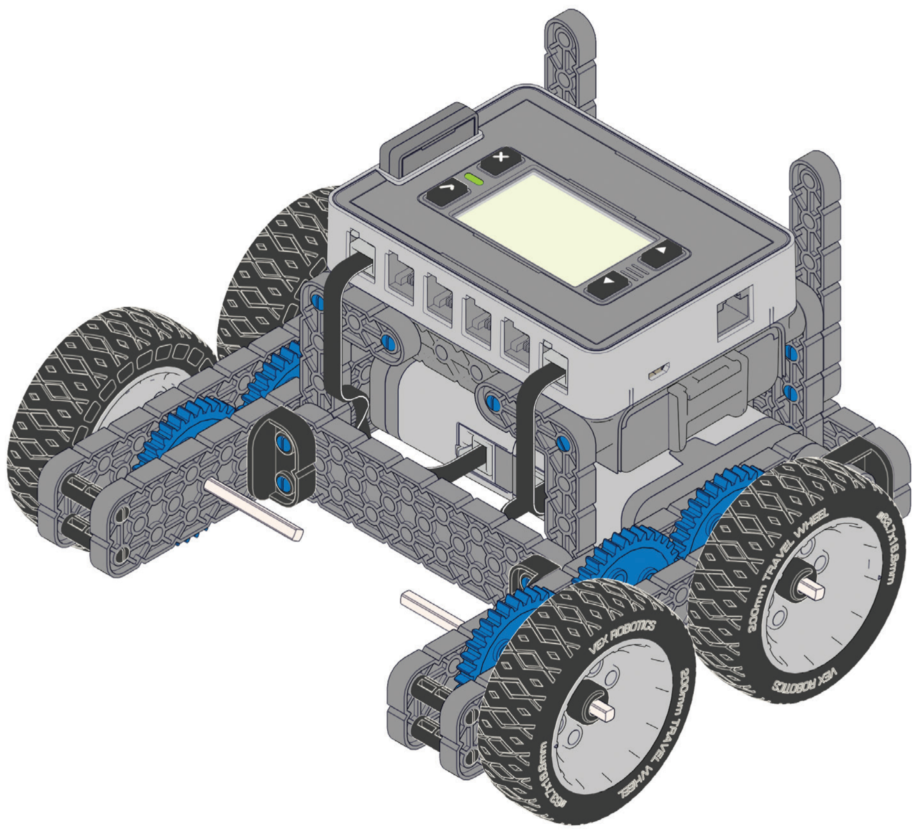


Regardless of which category an **Object Manipulator** fits into, some are designed to handle single objects, while others are designed to collect and hold multiple objects. Any specialized **Object Manipulator** designed to collect and hold multiple objects at one time is known as an **Accumulator**. **Accumulators**, when desired, can allow for greater efficiency of an object manipulation system.

F

Standard Drive Base





F.cl1

Overview of Activity:

Campers will build the Standard Drive Base and learn to control it using the VEX IQ Controller to maneuver around and through obstacles on the VEX IQ Highrise Challenge Field. Three or more teams will then play a game of “Freeze Tag” on the Challenge Field with obstacles in place.

Learning Objectives:

- Campers will build a Standard Drive Base.
- Campers will pair their VEX IQ Controller with the VEX IQ Robot Brain.
- Campers will learn to control their basic robot with a remote control.
- Campers will document success and challenges when driving their robot with the controller.

Suggested Timing:

Up to 1 hour

- The building component of this activity may take more time if campers are young or less experienced with following step-by-step instructions.

F.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- VEX IQ Super Kit Contents & Build Tips Poster
- Highrise cubes to create obstacles
- Highrise Challenge Field or space of similar size
- VEX IQ Build Instructions - Standard Drive Base (Kit Documentation) 1
- VEX IQ Control System User Guide – Charge & Install Batteries, Install Radios & Pair Controller (Kit Documentation) 1+2
- Freeze Tag Playing Field and Rules Handout

Note: VEX IQ Build Instructions, User Guides, and other useful resources are available at www.vexiq.com/vexiq/documents-downloads.

Optional Materials:

- Plastic cups to create additional obstacles
- Thin planks of plywood (to create ramps or obstacles)
- Stopwatches or smart devices with timing apps

Pre-learning Suggestions:

- If possible, complete **Learning Activity E – Drivetrain and Object Manipulator** to provide campers with some experience with basic Drivetrains and a basic understanding of Turning Scrub in robot design.

Based on VEX IQ Robotics Education Guide:

- **Build and Test Clawbot IQ – Build Options – Option 1** (C.2)

F.cI2

Detailed Directions

1. Provide campers with the kit documentation, VEX IQ Build Instructions - Standard Drive Base and the VEX IQ Super Kit Contents & Build Tips poster. Review the directions with campers and have them complete the build to Step 19.
2. Demonstrate how to install the VEX IQ Radio and pair the VEX IQ Robot Brain with the VEX IQ Controller so that the components can communicate with each other wirelessly. Instructions for this process are outlined in the kit documentation,
   VEX IQ Control System User Guide – Install Radios and Pair Controller. 2

   Note: A helpful video demonstration on "Pairing the Brain & Controller" is available at www.vexiq.com/vexiq/documents-downloads.

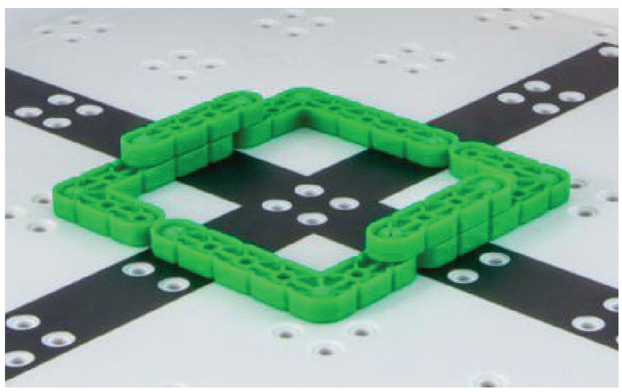
   Once a VEX IQ Robot Brain and VEX IQ Controller have been paired together, they will stay linked even after being turned off and back on.

   To save time, especially with younger or less experienced students, it may be a good idea to install the VEX IQ Radio and pair the VEX IQ Robot Brain with the VEX IQ Controller before campers begin this activity.
3. Tell campers that this robot base will be the foundation for other assemblies and that today they will get comfortable driving with a controller.
4. Provide each pair of campers with four (4) or five (5) Highrise Cubes and a large open space to practice their driving skills. Have partners arrange the Cubes in different configurations for each other and take turns driving the robot around them using the controller.

   To add extra excitement, have campers set time goals for each other to complete the "course" they have created and time each other using smart devices or stopwatches.
5. After some time practicing with a partner, show campers the VEX IQ Highrise Challenge Field or have them help to set up the field in line with the sample configuration in the Freeze Tag Playing Field and Rules handout.

   It is a good idea to secure the Highrise cubes in place with Cube-Base Kit components, or the game can be played with Cubes that move freely for different strategy options.

   See F.cs1 for sample field configuration and rules.
6. Have campers play the game. Encourage them to cheer on other campers. Campers who have already competed can track points or act as referees for campers who compete after them.



Cube-Base Kit components on field

F.cl2 cont.

Tips and Best Practices:

- Before beginning this activity with campers, make sure that all VEX IQ Robot Batteries have been charged or that the Robot AA Batter Holder has been filled with 6x 1.5V AA Batteries.
- Make sure that VEX IQ Robot Controller battery has been changed and installed in the VEX IQ Controller in line with the directions found in the **VEX IQ Control System User Guide**. 1

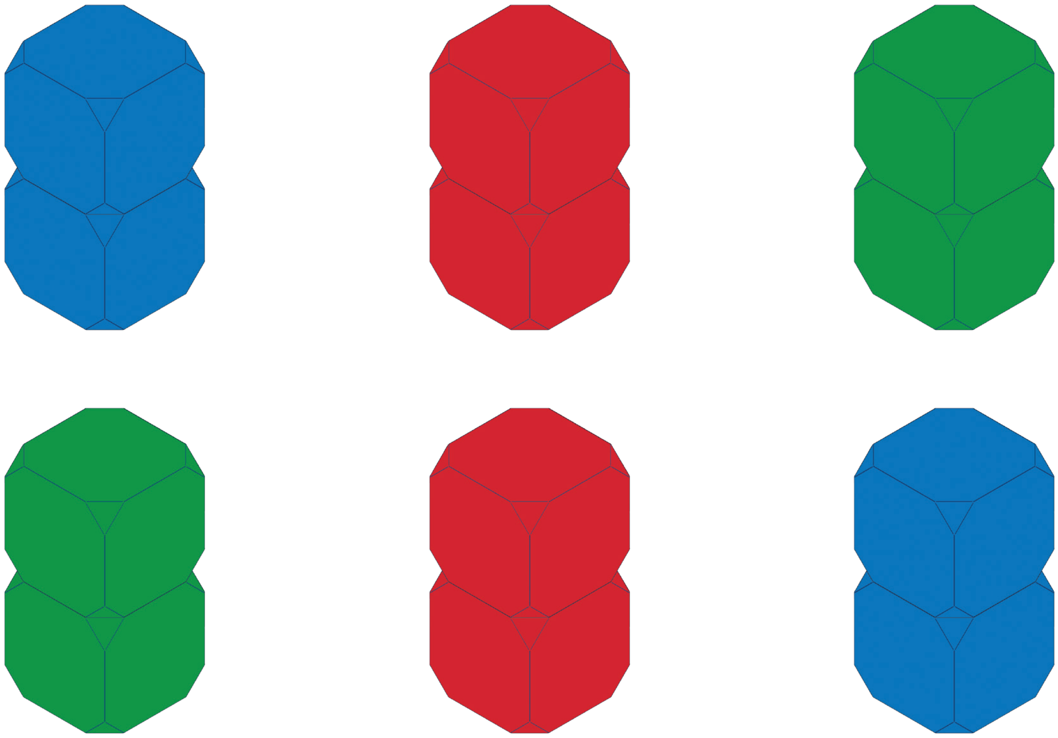
Extension:

- Have campers use everyday materials to create original test courses for other pairs to complete (e.g., cups or containers to create obstacles, planks of wood, cardboard or books to create ramps or tunnels). Campers or pairs can rotate through each other's courses and identify features that were challenging or engaging from each course.
- Add or remove Highrise Cubes from the Challenge Field or assign points to Cubes and use the VEX IQ Highrise Scoring App to track points earned when robots make contact with a particular color during the course of the game.

F.cs1

Freeze Tag Playing Field and Rules

Playing Field:

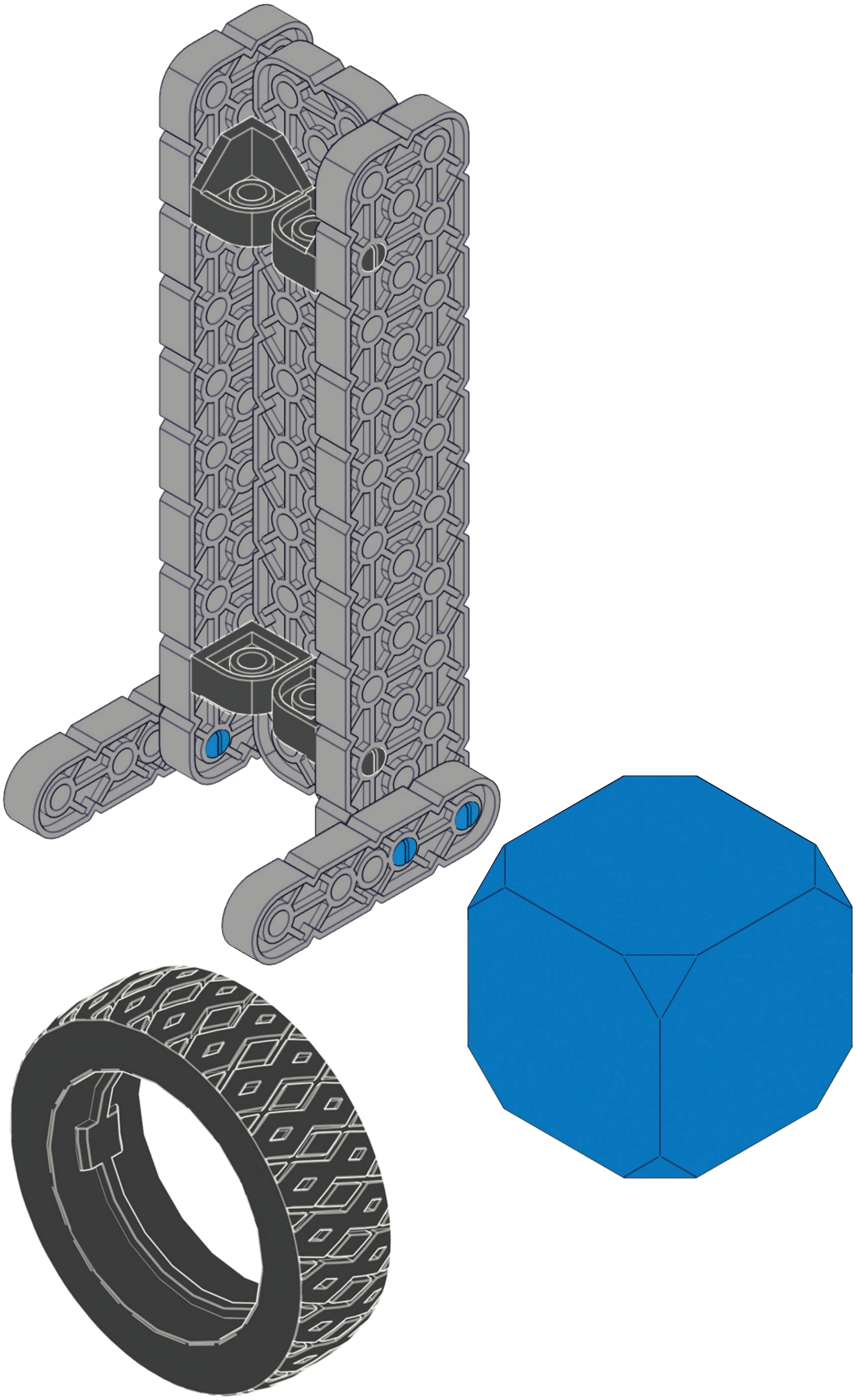


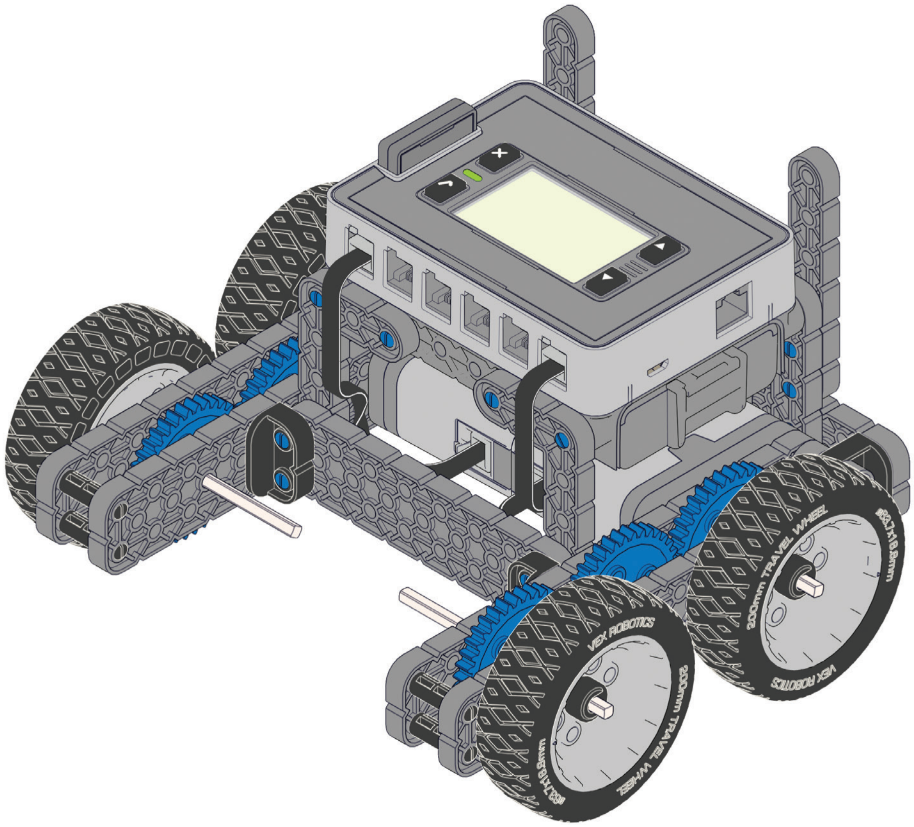
Freeze Tag rules:

- All teams begin with 5 points each.
- Each game of Freeze Tag will be two minutes long, with drivers on the same team switching at the 0:55 mark.
- Three or more robots will play at the same time.
- Robots will begin from a set starting position on the field and the robot who is “It” first will be selected randomly (e.g., pulling a team from a hat).
- If a robot has been tagged, it must freeze for three (3) seconds and will lose one (1) point.
- Each stack has two Highrise Cubes, one on top of the other. Robots may make contact with the stacks without penalty, but will lose one (1) point if the Highrise Cube on the top falls.
- If robots touch the obstacles, no penalties will occur, but if a robot moves an object as a result of this contact, that robot will return to the starting position.
- At the end of the two minute match, points will be totaled to determine the winner.

G

Standard Drive Base with Object Manipulator





G.cl1

Overview of Activity:

Campers will design and construct a non-motorized object manipulator that can be attached to the Standard Drive Base. The object manipulator may be static or dynamic (operated by hand or with a simple machine-based mechanism). The end result should meet the criteria of being able to effectively move a VEX IQ Tire, a VEX IQ Distance Sensor, and a VEX IQ Highrise Cube from one location to another using the robot and Controller.

Learning Objectives:

- Campers will learn about Object Manipulation.
- Campers will plan and implement an original build design.
- Campers will think critically about how design choices affect a mechanism's efficacy.
- Campers will learn to implement the "Think, Do, Test" design process.

Suggested Timing:

Up to 2 hours

- This activity may take less time if campers have already investigated object manipulators in Learning Activity E - Drivetrains and Object Manipulators.
- This activity may take more time if campers have not already built the Standard Drive Base from Learning Activity F - Standard Drive Base.

G.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Standard Drive Base
- VEX IQ Challenge Field (full- or half-size)
- Extra VEX IQ Tires
- VEX IQ Highrise Cubes
- VEX IQ Build Instructions - Standard Drive Base (Kit Documentation) 1
- Think, Do, Test – Standard Drive Base with Object Manipulator
- **Mechanisms: Object Manipulation** (E.cs4)

Optional Materials:

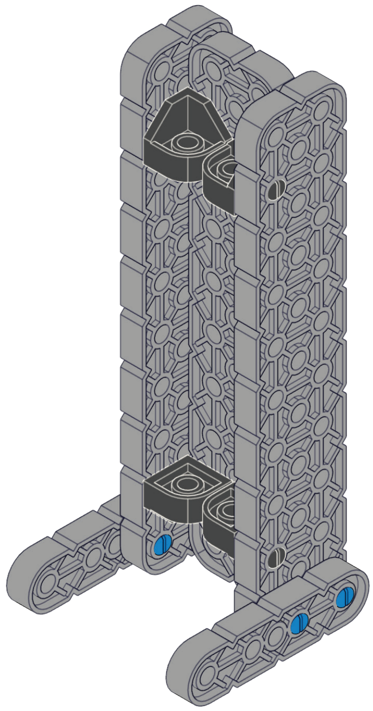
- A small tower created from VEX IQ components

Pre-learning Suggestions:

- If possible, complete **Learning Activity E: Drivetrain with Object Manipulator**.
- Concepts from **Learning Activity E** can be integrated into this activity. Provide extra time to experiment with basic object manipulators.

Based on VEX IQ Robotics Education Guide:

- **Build and Test Clawbot IQ – Build Options – Option** 1 (C.2)
- **Mechanisms: Object Manipulation** (G.5)



Small tower created from VEX IQ components

G.cl2

Detailed Directions

1. Provide each pair of campers with three objects: a small tower created from VEX IQ components, a VEX IQ tire, and a VEX IQ Highrise Cube. Ask them to identify a few physical characteristics for each object that may affect how it can be moved.
2. Provide campers with the handout, **Think, Do, Test – Standard Drive Base with Object Manipulator** to document their ideas.
3. For each object, ask campers to think about its physical characteristics and how these will affect the task of moving it horizontally (Will it tip over if you push it?) and vertically (Is it difficult to grasp and pick up?).

   Encourage them to pick up and move the objects in different ways with their hands or other non-robot parts and record their observations on the handout.

   Questions to ask campers:

   - What are some physical characteristics of each object?
   - Do these characteristics present challenges in moving the object?
     - A small tower created from VEX IQ components may easily tip over.
     - The tire will roll away if pushed while on its edge, but the rubber may cause friction on its side and make it difficult to push.
     - The VEX IQ Highrise Cube is large and may be difficult to grasp.
4. Tell campers they will be trying to move each object to two different locations:
   - From a flat surface to another flat surface.
   - From a flat surface to an elevated surface (1-2 inches).

   Questions to ask campers:

   - What kind of motion could move each object horizontally?
   - What kind of motion could move each object vertically?
5. If it hasn't yet been done, build the foundation for the robot using the kit documentation **VEX IQ Build Instructions – Standard Drive Base**. *1*
6. Provide campers with time to investigate the components in the VEX IQ Super Kit and experiment with building different assemblies.

   If needed, provide campers with the handout, "**Mechanisms: Object Manipulation**" Background Information. This document shows some basic object manipulators, including **Plows**, **Scoops**, and **Friction Grabbers**.

   Questions to ask campers:

   - Will these designs work for one or both of these tasks?
   - How will the assembly attach to the Standard Drive Base?
   - Does attaching this assembly to the Standard Drive Base change the intended movement or function of the assembly? (Limit mobility or range of motion.)
7. Have campers build their designs and complete tests with the different objects, moving them horizontally along the competition field or floor and moving them vertically to a height of an inch or two.

Questions to ask campers:

- Are there certain objects which are easier to move with your robot?
- Can one object manipulator be used for all the objects and both the tasks?
- Can all functions be completed using the controller or do some parts of the solution require manual adjustments (e.g., turning a crank or pulling a handle to lift an arm)?

Tips and Best Practices:

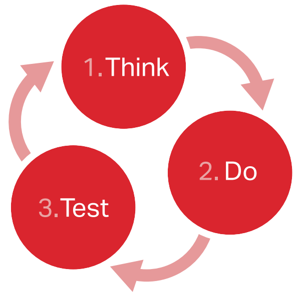
- Remind campers that in engineering, you may not be able to do it all. Certain situations call for a choice between quality and quantity of functions. Think carefully about the time you have to complete a design and decide which functions are worth the time. An idea that doesn't work is not a failure. It may be helpful in a different situation.
- Object manipulators do not need to be motorized or automated. Experiment with gears and crank handles, or a simple lever on a dynamic arm. A driver cannot touch the object itself.

Extension:

- Add a smart motor or sensor to the assembly to complete the required task with the push of a button.

G.cs1

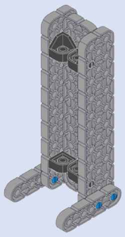
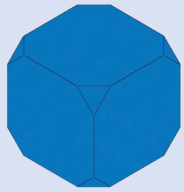
Think, Do, Test - Drivetrains and Object Manipulators



Student Name(s): ____

Instructions:

Look at these objects.

		
Tower	Tire	Highrise Cube

How could you move each object without touching the object yourself?

Begin with the Standard Drive Base or a robot of your own creation. Your job is to "imagine" an assembly you can build and add to the robot to help you move each object .

1. "THINK" - Think carefully about each object. Make observations about its physical characteristics (size, weight, shape, material it is made of) and what might be challenging about the task of moving it horizontally and vertically.

Draw the objects here and write notes or labels to describe your observations about them and the potential challenges you'll face

2. "DO" - Draw and describe the assembly you will build to move the objects. Name it, label its parts and show and describe how it would work for all three objects.

Draw, name, and label your assembly and how it will work for all three objects.

G.cs1 cont.

3. "TEST" – Build your assembly and connect it to your robot. Test your design. Write down your observations in a notebook, or use a smart device to record your observations as video or audio.

Did the assembly work for all of the objects? YES NO

If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until your robot completes the tasks.

Problems ARE NOT failures. They are an expected part of the design process!

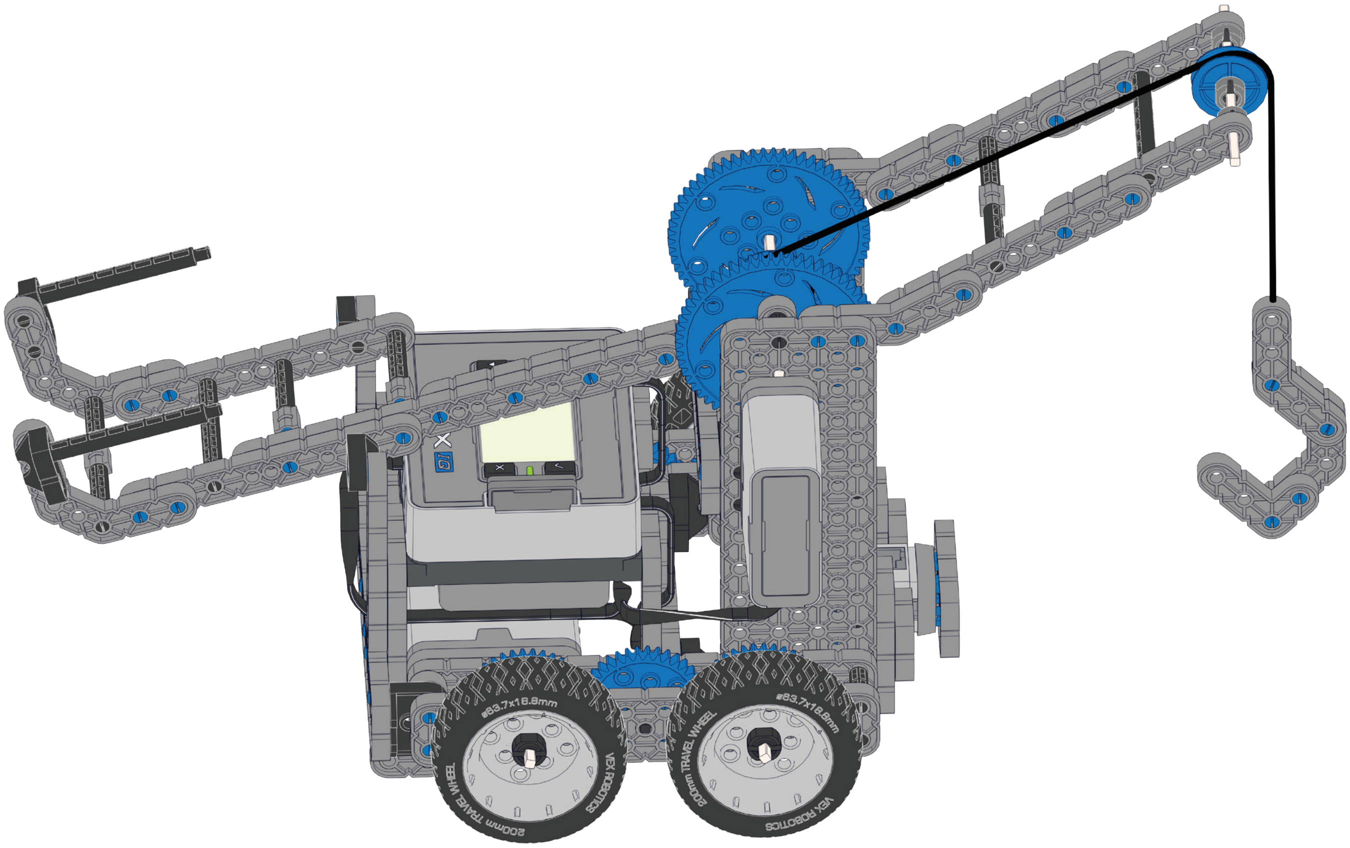
G.cs2

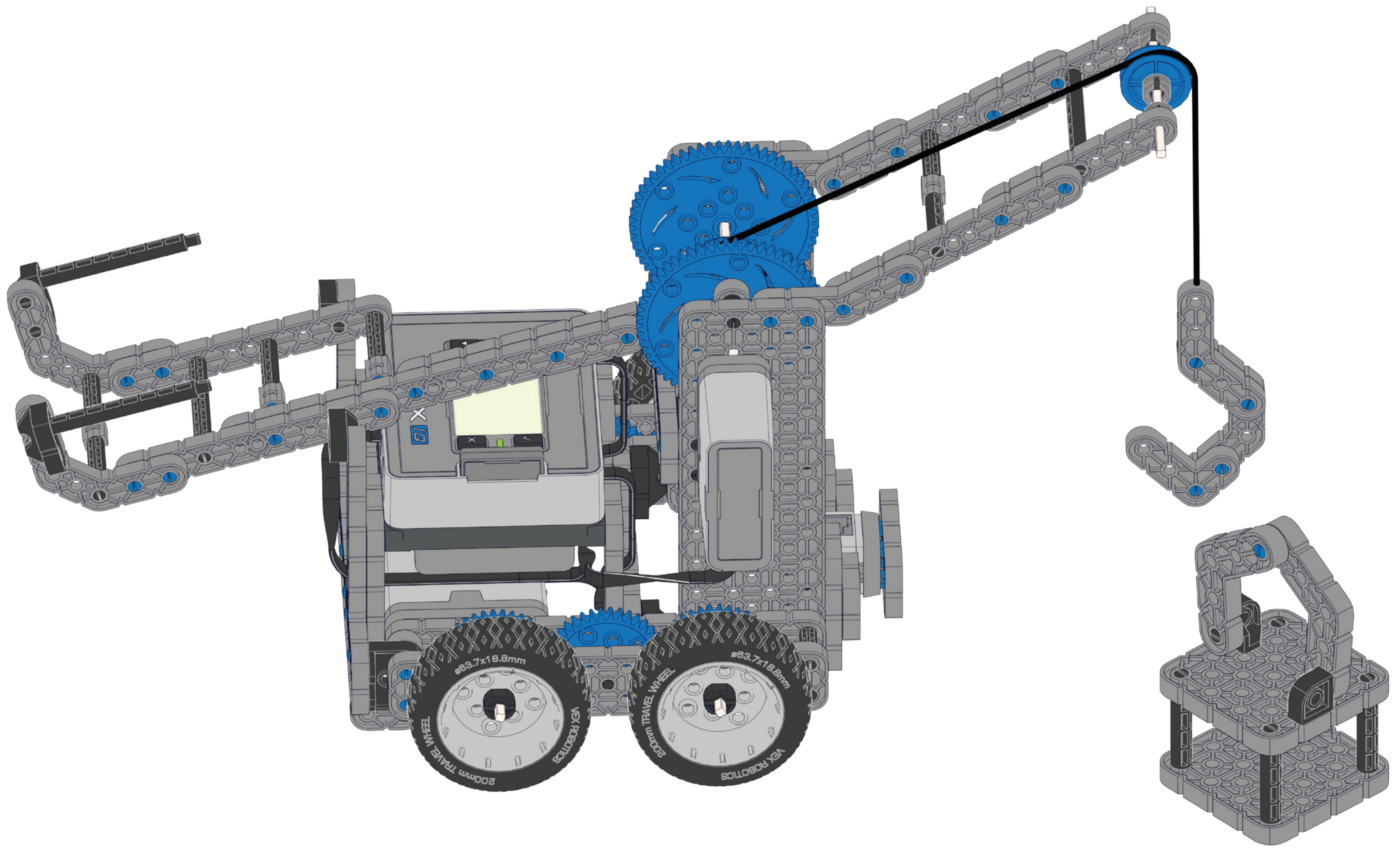
Mechanisms: Object Manipulation

See E.cs4 for materials and instructions.

H

Standard Drive Base with Lifting Mechanism





H.cl1

Overview of Activity:

Campers will design and construct a lifting mechanism (with or without motors) that can be attached to the Standard Drive Base. The end result must be able to pick up a small basket constructed from VEX IQ components and place it on top of a Highrise Cube. If time allows, campers can be encouraged to create both a motorized and non-motorized lifting mechanism. (e.g., using a hand crank or a lever to lift a scoop)

Learning Objectives:

- Campers will apply their knowledge of simple machines and motion.
- Campers will think critically about how design choices affect a mechanism's efficacy.
- Campers will modify an existing assembly for a new purpose.
- Campers will learn to implement the "Think, Do, Test" design process.

Suggested Timing:

Up to 2 hours

- This activity may take less time if campers have explored simple machines and mechanisms with Learning Activities A-D.
- This activity may take more time if campers have not already built robot from Learning Activity F – Standard Drive Base.

H.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Standard Drive Base
- VEX IQ Highrise Cubes
- Sample Basket Assembly Image
- Think, Do, Test – Standard Drive Base with Lifting Mechanism

Optional Materials:

- Objects of different shapes and sizes that present challenges for lifting. (e.g., heavier objects)

Pre-learning Suggestions:

- If possible, complete **Learning Activities A-D**, as well as **Learning Activity F – Standard Drive Base**.

Based on VEX IQ Robotics Education Guide:

- **Build and Test Clawbot IQ – Build Option 1** (C.2)
- **The Six Simple Machines** (D.3)

H.cl2

Detailed Directions

1. Provide each pair of campers with one Highrise Cube. Tell them that their challenge is to design and build a device to attach to the Standard Drive Base that can lift a small basket made from VEX components onto a Highrise Cube.
2. Provide campers with an image of the Sample Basket Assembly Image or have them create their own.
3. Ask partners to brainstorm different ways that this object could be lifted (e.g., sliding or dragging the basket up a ramp, picking it up with grabbers that open and close, adapting an elevator assembly, creating a platform attached to a lever to lift the Cube, or creating a hook and pulley system
4. Have campers make a list of all their ideas and tell them that the main criterion for this challenge is to create an assembly for their robot that can lift the basket on to the Cube. Encourage campers to make note of any ideas that do not involve lifting for possible future designs, but to remove them from the list of ideas for this particular challenge.
5. Provide campers with the handout, Think, Do, Test - Standard Drive Base with Lifting Mechanism to document their ideas and design a lifting mechanism for their robot.

   Criteria for Robot and Lifting Mechanism:

   - The robot must be controlled with the VEX IQ Controller.
   - The lifting mechanism on the robot may be manual or motorized.
   - The new assembly must have some moving parts.
   - The completed robot must LIFT the object onto the Highrise Cube in some way.

   If it hasn't been done yet, build the Standard Drive base using the kit documentation, **VEX IQ Build Instructions – Standard Drive Base**.

H.cl2 cont.

6. Have campers consider different simple machines and object manipulators they have either worked with recently, or seen in real life. Provide them with time to investigate the components in the VEX IQ Super Kit and experiment with building different assemblies.

   If needed, provide students with the handout, "**Mechanisms: Object Manipulation**" Background Information and/or "**The Six Types of Simple Machines**" Background Information. These documents may provide inspiration and ideas for assembly designs.

   Questions to ask campers:

   - What are some different ways you can lift the basket?
   - Which VEX IQ components are available to build your assembly after you have constructed the Standard Drive Base?
   - Will the whole assembly need to move or will some parts move while others remain static?
   - How and where will the assembly attach to your Drive Base?
   - How will you control the movement of your robot?
   - How will you control the movement of your assembly?
   - Is there a way to make the basket itself easier to lift? (e.g., the addition of small feet so that the forks on a lift can slide easily underneath)

7. Have campers build their designs and test them with the basket and the Highrise Cubes. Results can be documented with drawings, photos, or made into videos using smart devices to show parents at the end of the day

Tips and Best Practices:

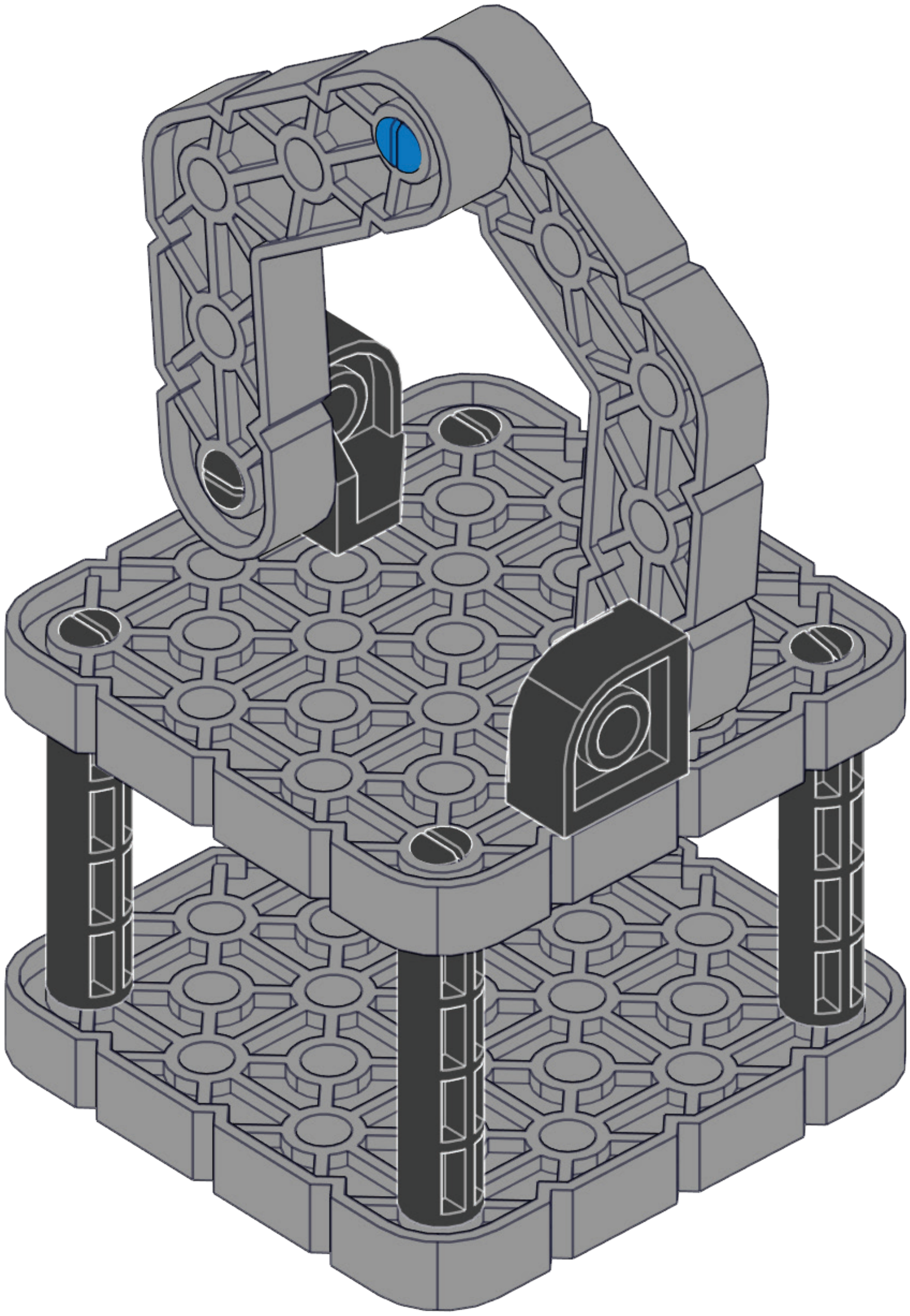
- Concepts from **Learning Activity C - Pulley** can be integrated into this activity. Provide extra time to experiment with different pulley configurations.
- If desired, the basket design can be modified by campers to make it function more effectively with their lifting mechanisms.
- Make note of interesting and creative design ideas. They may not function well for a particular task, but may be perfect in the future for another.
- Encourage campers to share their ideas and designs with other pairs. This activity is not a competition, but an exercise in design.

Extension:

- Have campers create a lifting mechanism that can place the basket on top of two or more stacked Highrise Cubes or more precisely into a container.
- Add a motor and a VEX IQ Touch LED and program the robot to start and stop its lifting mechanism at the touch of a button.
- Have campers create different mechanisms or use different simple machines to move the object onto the Highrise Cube (e.g., an inclined plain to slide the basket onto the Cube). Experiment with moving different shapes and objects that may include balls or plastic cups.

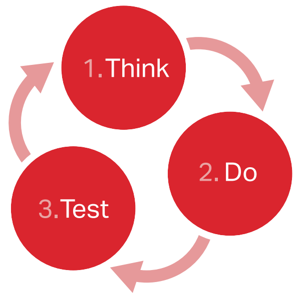
H.cs1

Sample Basket Assembly Image



H.cs2

Think, Do, Test - Standard Drive Base with Lifting Mechanism



Student Name(s): ____

Instructions:

After you've built the Standard Drive Base, think about the kind of mechanism design and add to help your robot lift a basket made from VEX IQ components.

Criteria for lifting mechanism:

- The robot must be controlled with the VEX IQ Controller.
- The lifting mechanism on the robot may be manual or motorized.
- The new assembly must have some moving parts.
- The completed robot must LIFT the basket onto the Highrise Cube in some way.

1. "THINK" - Think carefully about the object you will be lifting and the height you will need to lift it. Make observations about the basket's physical characteristics and the Standard Drive Base onto which you must attach your new lifting mechanism.

____

____

____

Draw the objects here and write notes or labels to describe your observations about them

2. "DO" - Draw and describe the assembly you will build for your robot. Name it, label its parts, and show or describe how it will work.

Draw, name, and label your lifting mechanism and how it will work.

H.cs2 cont.

3. "TEST" – Write down your observations in a notebook, or use a smart device to record your observations as video or audio. Make sure to test your lifting mechanism several times to make sure it performs consistently and accurately.

Make a note of any changes you make along the way.

____

____

____

____

____

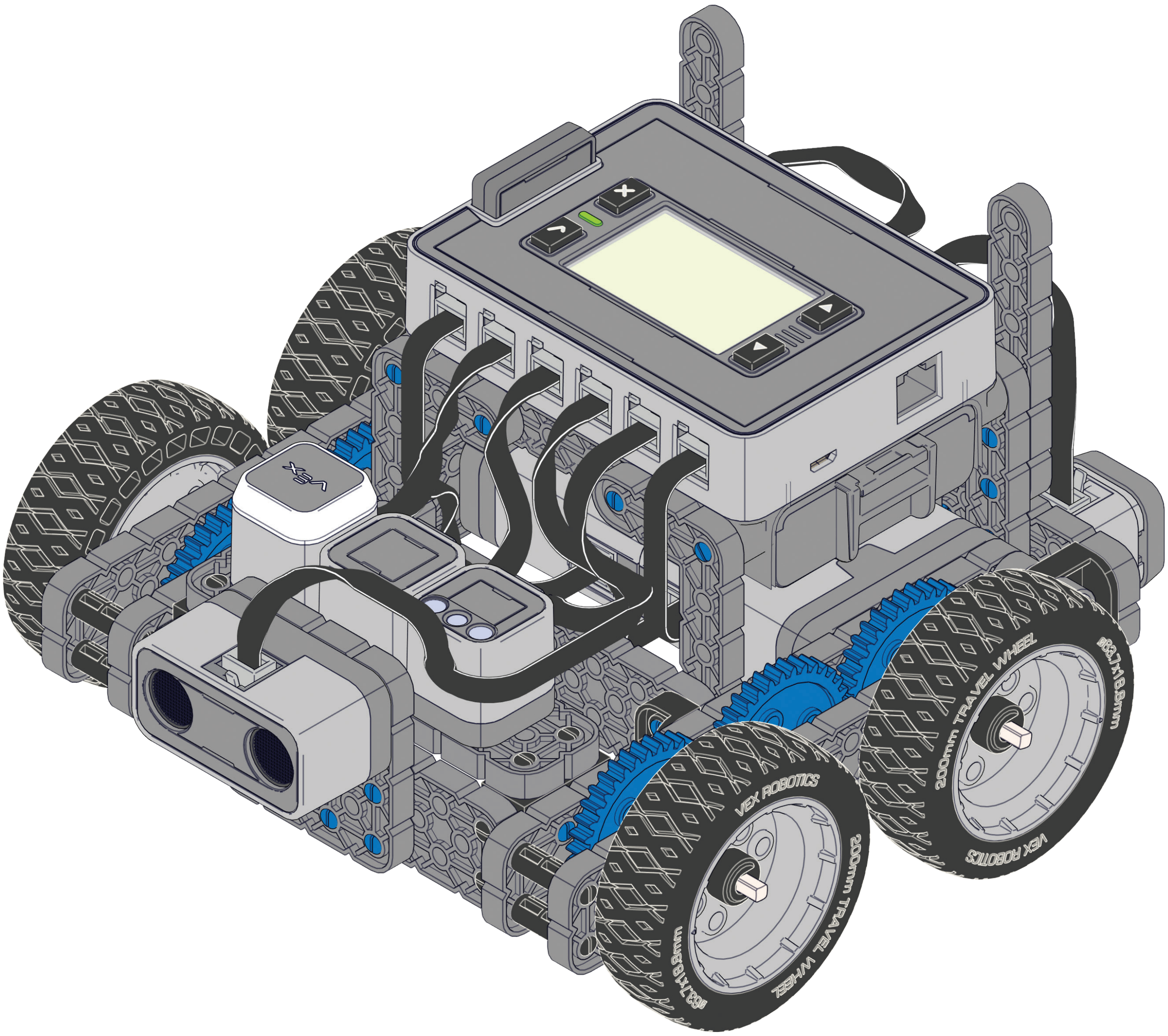
Did the lifting mechanism work on the robot? YES NO

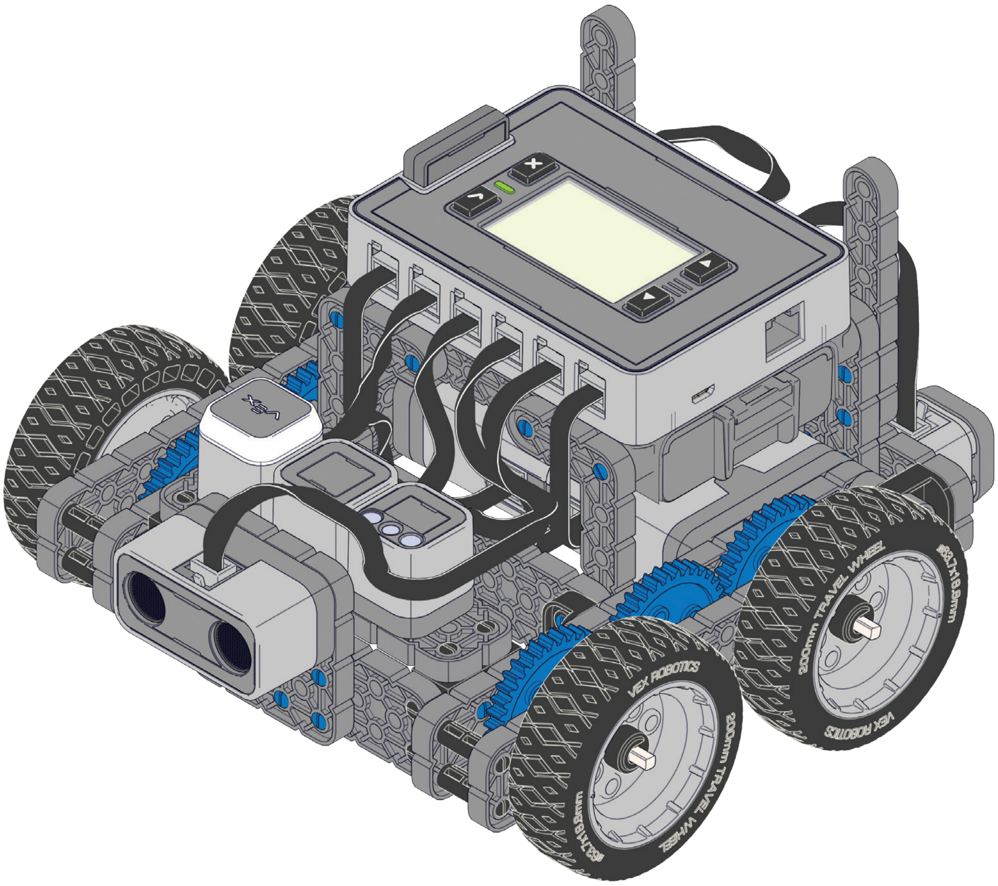
If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until your robot completes its task.

Problems ARE NOT failures. They are an expected part of the design process!

Autopilot Robot





I.cl1

Overview of Activity:

Campers will build the VEX IQ Autopilot robot and run it with particular sensors connected or disconnected to gain an understanding of how sensors affect the "intelligence" of a robot. They will conduct mini investigations into robot behavior using the Autopilot test modes.

Learning Objectives:

- Campers will learn key terminology related to sensors and programming.
- Campers will gain an understanding of how sensors and motors contribute to the capabilities of a robot.
- Campers will learn sensor types and functions.
- Campers will begin to think critically and creatively about how sensors can be incorporated into a range of robotics tasks.

Suggested Timing:

Up to 2 hours

- This activity may take less time if campers have previous experience with sensors and robotics.
- This activity may take more time if campers have not already built the Standard Drive Base from Learning Activity F – Standard Drive Base.

I.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Standard Drive Base
- Highrise Challenge Field (half or full size) and Cubes
- Empty VEX IQ Super Kit box
- VEX IQ Build Instructions - Autopilot Instructions (Kit Documentation) *1+4*
- VEX IQ Control System User Guide – 6.2 Autopilot Program (Kit Documentation) *6.2*
- Autopilot Investigation Handout
- VEX IQ Sensor Overview

Optional Materials:

- Shoes, text books, plastic cups and other objects of different weights to act as barriers or obstacles

Pre-learning Suggestions:

- If possible, complete **Learning Activity F** – Standard Drive Base.

Based on VEX IQ Robotics Education Guide:

- **Running Autopilot Modes** (I.6)

I.cl2

Detailed Directions

1. Provide campers with the kit documentation, VEX IQ Build Instructions - Autopilot Instructions and give them time to complete the construction of their robots.
2. When the robot has been assembled, challenge campers to identify each sensor that has been added and brainstorm a few tasks that sensor could help to achieve. Encourage campers to note their ideas in a notebook for future use.

   If needed, provide campers with "VEX IQ Sensor Overview" Background Information for more specific details about sensor names, specs, use, and default functionality (how it works without any additional programming).
3. Tell campers that they will be experimenting with three different autopilot programs. Have them set up a half-size Highrise Challenge Field or block off a 2'x 4' area using tables, boxes or other barriers to contain the robot. This will be the testing area for a few different investigations.
4. Provide campers with the kit documentation, VEX IQ Control System User Guide – 6.2 Autopilot Program. Tell them that they will need to use this robot and program to complete their investigation.
5. Provide campers with the handout, Autopilot Investigation. They should complete their investigations with the robots on the field (or other designated space) and conduct several tests for each investigation.

6. Conduct the investigations.

INVESTIGATION #1: Explore the Autopilot test modes.

See **I.cs1** for steps.

Question to ask campers:

- Which sensors are being used to create each behavior?

Descriptions of robot behavior when running the Autopilot Program can be found in the kit documentation, VEX IQ Control System User Guide, 6.2 Autopilot Program.

INVESTIGATION#2: Unplug the VEX IQ Distance Sensor.

See **I.cs1** for steps.

Questions to ask campers:

- What tasks is the robot able to complete without the Distance Sensor?
- Is it possible to stop the robot without the Distance Sensor? (yes)
- What stops the robot when the Distance Sensor is disconnected? (Force)
- How does the robot know it needs to back up? Is there a sensor being used or something else? (The smart motors provide feedback based on force.)
- How much force is required to provide the feedback to the robot to stop and change directions? (What is the threshold?)

Note: This is a great point to introduce the concept of threshold, an important point when programming sensors, to campers at a high level.

Exact measurements of force are not required, but campers should consider the weight or amount of friction of an object and how this affects the feedback a robot receives from its sensors or motor. The concept of threshold is used to set ranges for sensor feedback that determine how a robot will behave in response to stimuli. (e.g., If X amount or less of force is detected, the robot will keep moving. If X amount of force or more is detected, the robot will back up.)

INVESTIGATION #3: Unplug the other VEX IQ Sensors.

See **I.cs1** for steps.

Questions to ask campers:

- Will the robot function when each sensor is unplugged?
- Which sensors affect the programming the most?
- How do you think each sensor functions in the Autopilot Program?
- For each sensor function, could a different sensor be used to perform a similar function?

7. If time allows, place a variety of objects (e.g., stacks of Highrise cubes, structures and archways made from VEX components) on the competition field and run the Autopilot program with all sensors for one minute.

Questions to ask campers:

- How many obstacles/structures remain?
- Were any of the modes more effective at avoiding the obstacles?
- Which obstacles were easiest to avoid? Why?

I.cl2 cont.

Tips and Best Practices:

- When selecting objects to place on the Challenge Field, keep in mind that the Color Sensor will automatically detect and respond to certain colors when they are near the Color Sensor. (e.g., red, green, blue).

Extension:

- Move the Color Sensor to a forward-facing position on the robot. Experiment with programming to replicate the movement patterns completed by the Autopilot Program and investigate different light and color thresholds to make contact with objects that are outside a given range and avoid objects within it.

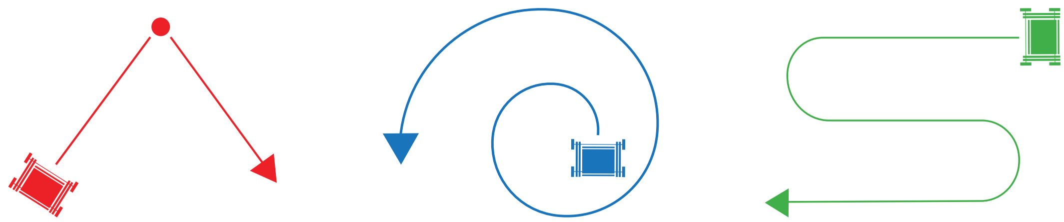
I.cs1

Autopilot Investigation

Student Name(s): ______________________________

Instructions:

Build the Autopilot Robot and run the Autopilot Program to conduct the investigations below.



Random Mode — Spiral Mode — Lawnmower Mode

INVESTIGATION #1: Explore the Autopilot test modes.

1. Place the Autopilot robot on the Challenge Field and start the program.
2. Place your hand in front of the robot. Describe how the robot reacts.
3. Place different colored Highrise cubes in front of the robot. Describe how the robot reacts to each color.

Draw or describe your observations here

I.cs1 cont.

INVESTIGATION#2: Unplug the VEX IQ Distance Sensor.

1. Disconnect the VEX IQ Distance Sensor.
2. Make a prediction about how removing this sensor will affect the robot's ability to stop or change directions.
3. Place the Autopilot robot on the Challenge Field and start the program. Describe how the robot's behavior is the same or different from INVESTIGATION 1.
4. Place different objects in front of the robot as it completes the Autopilot Program.
   - A Highrise Cube. Describe what happens.
   - A running shoe or something of similar weight. Describe what happens.
   - An empty VEX IQ Super Kit box.
5. Place the robot near a wall and run the Autopilot program. Describe what happens.

Draw or describe your observations here

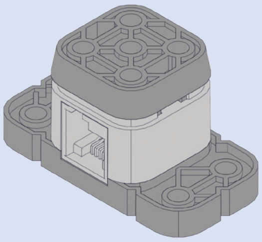
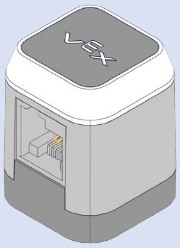
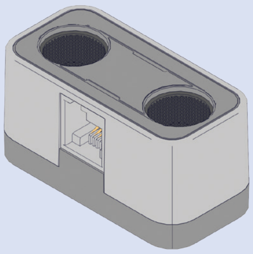
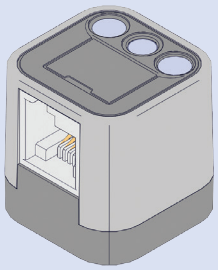
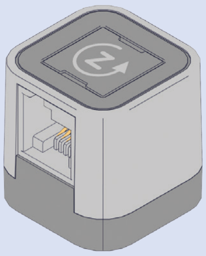
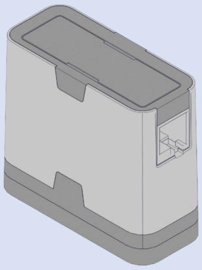
INVESTIGATION #3: Unplug the other VEX IQ Sensors.

1. Reconnect the VEX IQ Distance Sensor.
2. Unplug each VEX IQ Sensor one after another and test what happens to the Autopilot robot's behavior.
3. Describe what happens when each sensor is unplugged.

Draw or describe your observations here

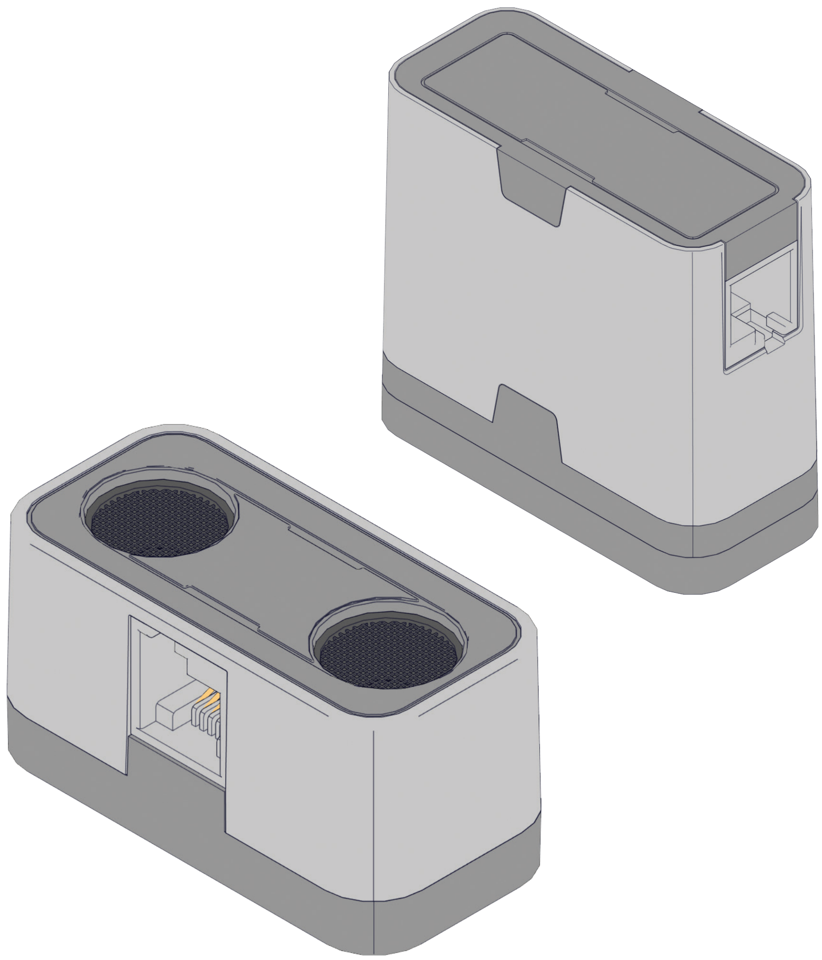
I.cs2

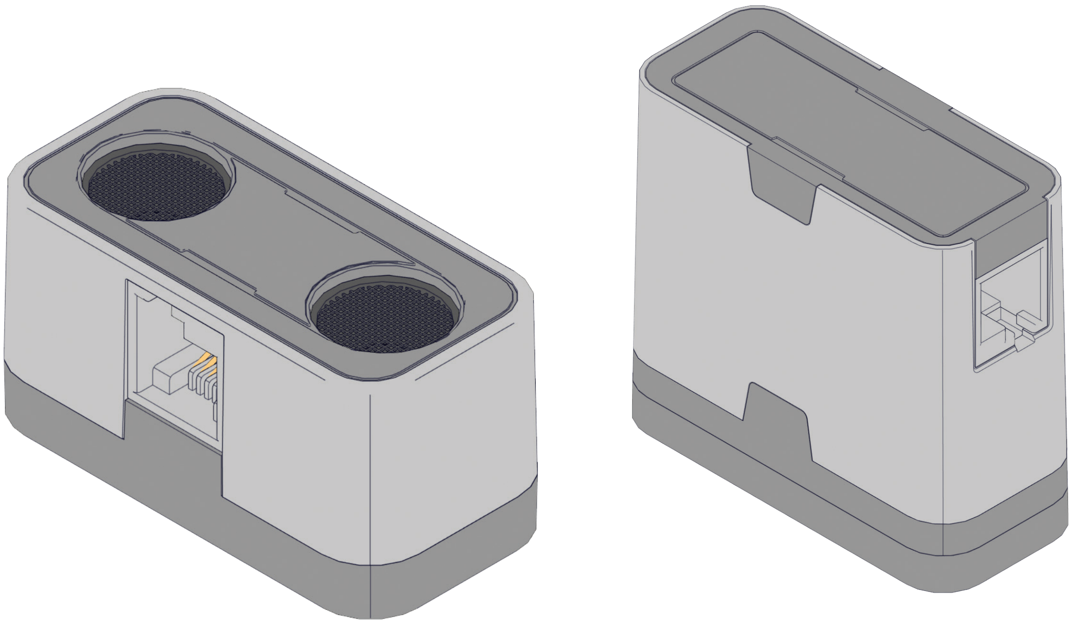
VEX IQ Sensor Overview

Sensor Name	Sensor Image	Sensor Specs & Use	Default Functionality
Bumper Switch		Allows for the sense of touch in the VEX IQ platform. Detect a wall, obstacle, or limit mechanism movement.	With a Bumper Switch in port 2, it disables/enables a Smart Motor in port 4.
Touch LED		Smart Sensor with red, green, blue LEDs. Constant on, off, or blink at any desired rate. Touch sensor with finger for interaction.	Enables and disables Autopilot or similar robot running Driver Control Program when dome of sensor is tapped. Glows green when enabled, red when disabled.
Distance Sensor		Uses ultrasonic sound waves to measure distance. Measures distance from 1 inch to 10 feet. Commonly used to avoid obstacles.	Slows down and eventually stops as Autopilot or similar robot running Driver Control Program as the robot approaches an obstacle.
Color Sensor		Detects the color of objects. Measures independent red, green, and blue in 256 levels each.	Enables and disables Autopilot or similar robot running Driver Control Program when the sensor is “shown” a green card (enabled) or red card (disabled).
Gyro Sensor		Measures turn rate and calculates direction. Frequently used in autonomous robot navigation and turning.	With Autopilot or similar robot running Driver Control Program, it returns a robot to its original direction when driving stops.
Smart Motor		Commands and measures speed, direction, time, revolutions and/or degrees of turn using its Encoder.	No extra default functionality, but Encoders allow for superior motor control with simple programming.

J

Distance Sensor Default and Programming Task





J.cl1

Overview of Activity:

Campers will complete an in-depth "collision avoidance" functionality exercise using the VEX IQ Distance Sensor. Campers will use the Autopilot robot with ONLY VEX IQ Smart Motors and a VEX IQ Distance Sensor to test the basic functionality of the sensor and then make minor changes to the robot's behavior. Campers will then use programming software to create a basic program that can be transferred and run on the robot. They will first mimic the default functionality and then make minor modifications to the programming to make it their own.

Learning Objectives:

- Campers will learn how to use a VEX IQ Distance Sensor.
- Campers will program one sensor using selected software.
- Campers will transfer a program from software to the VEX IQ Robot Brain.

Suggested Timing:

Up to 2 hours

- This activity may take more time if campers have not completed Learning Activity I – Autopilot Robot.
- This activity may take less time if campers have previous experience working with ModKit for VEX Robotics, ROBOTC for VEX Robotics or another programming language.
- This activity may take more time if campers have not had previous experience transferring and running programs on a robot.

J.cl1 cont.

VEX IQ Materials:

- Completed Autopilot Robot
- VEX IQ Robot Controller
- Computer for programming
- Modkit for VEX Robotics or ROBOTC for VEX Robotics programming software
- Program Planning Template
- Programming Support - VEX IQ Distance Sensor
- **VEX IQ Sensor Overview** (I.sc2)

Online Resources:

- VEX IQ Videos – Default Sensor Functionality – VEX IQ Distance Sensor Default Functionality - optional
  www.vexiq.com/documents-downloads/
- VEX IQ – YouTube Playlist – VEX IQ Distance Sensor Default Functionality - optional
  www.vexiq.com/videos

Pre-learning Suggestions:

- If possible, complete **Learning Activity I – Autopilot Robot**.

Based on VEX IQ Robotics Education Guide:

- **Default Sensor Functionality Exercises** (I.7)
- **Simple Programming Exercises Using Programming Software** (I.9)

J.cl2

Detailed Directions

1. Have each pair of campers think carefully about the Autopilot Robot. Ask students to identify the VEX IQ Distance Sensor and remove or disconnect all other sensors from the robot.

   Questions to ask campers:

   - What is the function of the distance sensor?
   - Can you find an example of how a distance sensor would function in a robot in the real world?
   - What kind of tasks would a distance sensor enable this robot to do?

   If needed, provide campers with "VEX IQ Sensor Overview" Background Information for more specific details about sensor names, specs, use, and default functionality (how it works without any additional programming).

   If campers have not yet built the robot, provide them with time to do so. Instructions for its construction can be found in the kit documentation, VEX IQ Build Instructions – Autopilot Instructions.

J.cl2 cont.

2. Have campers test the Distance Sensor Default Functionality by placing their robot on a flat surface, a few feet away from a wall or other flat surface (such as a desk turned on its side).

   Campers should:

   1. Make sure that ONLY the VEX IQ Smart Motors and a VEX IQ Distance Sensor are connected to the VEX IQ Robot Brain.
   2. Turn ON the VEX IQ Robot Brain and VEX IQ Controller.
   3. Select and run the Driver Control program.
   4. Use the VEX IQ Controller to drive the robot towards a wall.
   5. Document what happens.

   When the VEX IQ Distance Sensor "sees" an object that is too close to the robot, it will stop the robot from hitting that object. Campers should note the distance from the wall at which the robot stopped.

   Questions for campers:

   - How could you use this function?
   - What kinds of robotics tasks or situations would this function be useful for?

3. Have campers return the driver control program to its default settings and open their programming software (either ModKit for VEX Robotics or ROBOTC for VEX Robotics). Tell them that they will be creating an original program to control their robot's response to an obstacle such as a wall.
4. Have campers create a program that mimics the default program, where the robot drives autonomously towards the wall, then stops 6 inches from the wall. Challenge them to modify the program in some way to make it their own. (e.g., changing the distance at which the robot stops, changing its reaction to the wall by turning or backing up, or displaying text on the LCD screen.)

   See the Programming Support - VEX IQ Distance Sensor section of this activity for sample programs in Modkit for VEX Robotics and ROBOTC for VEX Robotics.

   Questions to ask campers:

   - How can you change this program by changing only 1 variable? (e.g., stop a different distance from the wall)
   - Instead of stopping, what could the robot do? (e.g., back up, change directions, display a message on the LCD screen)

5. If time allows, have campers make one or more modification to their program and then share it with a few other pairs of campers. They should then choose one modification from another pair and incorporate it into their own program.

   Encourage campers to use the handout, Program Planning Template to document their ideas and completed programs.

J.cl2 cont.

Tips and Best Practices:

- It is a good idea to provide students with some extra time to get to know the programming software they'll be using. They should be able to open the software, save a program, connect the programming computer to a VEX IQ robot. They should also be able to successfully transfer custom programs to the VEX IQ Robot Brain and run a program after it has been started.
- When building the robot, make sure that wires do not cross in front of the sensors. They may affect the distance sensed.
- When programming the robot to travel a specific distance from an object, slow down the speed of the motor. It takes some time to detect and calculate the distance.

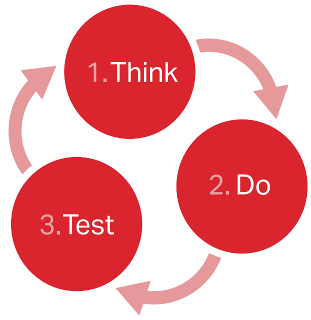
Extension:

- If campers have previous experience with other sensors, brainstorm and test different ways to avoid obstacles as alternatives for a scenario when a Distance Sensor is not available. Investigate which other sensors or mechanisms could be used to stop your robot from crashing into walls (e.g., VEX IQ Bumper Switch, programming timed turns, using a VEX IQ Color Sensor and making walls different colors with programmed avoidance of that color).
- Use the Distance Sensor to program the robot to maneuver around objects in a particular way. (e.g., rotate 90 degrees to the left, move forward, rotate 90 degrees to the right, move forward, etc.)

J.cs1

Program Planning & Troubleshooting

Student Name(s): ____



Instructions:

Before completing this activity, be sure you are familiar with your programming software and how it works. Follow the steps on this page to document, test, troubleshoot, and refine your program. If your program includes more steps than fit on one page, use multiple copies as necessary.

Describe what you want your program to be able to do here:	
Describe the device/robot you will be programming:	List the sensors that will be used in your program:

"THINK" Write step-by-step program instructions here.	"DO" Write your program using programming software and make notes here as you work.	"TEST" Does this step of your program function as expected? What needs improvement (NI)?
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:

If ANY program steps do NOT function as expected, address your "NI" items using as many copies of this page as necessary until all parts of your program function as expected.

Remember: Problems ARE NOT failures, they are an expected part of the design process!

J.cs2

Programming Support - VEX IQ Distance Sensor

Modkit for VEX Robotics

To access helpful sample programs in Modkit for VEX Robotics, follow the link below:

Basic Distance Sensor Programming

http://help.modkit.com/#vex

ROBOTC for VEX Robotics

To access helpful sample programs in ROBOTC for VEX Robotics, open the software and click the File tab and select "Open Sample Programs."

Sample Programs for this activity include:

- Detect Wall

Helpful VEX IQ Documents and Downloads

To access a variety of helpful resources for VEX IQ, follow the link below:

www.vexiq.com/documents-downloads/

VEX IQ Videos

- VEX IQ Distance Sensor Default Functionality

J.cs3

VEX IQ Sensor Overview

See I.cs2 for materials and instructions.

Color Sensor Default and Programming Task

K.cl1

Overview of Activity:

Campers will complete a "red light, green light" functionality exercise using the Color Sensor, where a green object will allow the robot to drive using the VEX IQ Robot Controller and a red object will prevent it from driving.

Campers will use the Autopilot robot with ONLY VEX IQ Smart Motors and a VEX IQ Color Sensor to test the basic functionality of the sensor and then make minor changes to the robot's behavior.

Campers will then use programming software (Modkit for VEX Robotics or ROBOTC for VEX Robotics) to complete an autonomous programming challenge to turn left or right depending on the color of the object being detected. Campers will use colored cubes to "drive" the robot around a Challenge Field without touching it.

Learning Objectives:

- Campers will learn how to use a VEX IQ Color Sensor.
- Campers will program one sensor using selected software.
- Campers will transfer a program from software to the VEX IQ Robot Brain.

Suggested Timing:

Up to 2 hours

- This activity may take more time if campers have not completed Learning Activity I – Autopilot Robot.
- This activity may take less time if campers have previous experience working with ModKit for VEX Robotics, ROBOTC for VEX Robotics, or another programming language.
- This activity may take more time if campers have not had previous experience transferring and running programs on a robot.

K.cl1 cont.

VEX IQ Materials:

- Completed Autopilot robot
- VEX IQ Robot Controller
- Computer for programming
- Modkit for VEX Robotics or ROBOTC for VEX Robotics programming software
- Highrise Cubes (one green and one red for each pair of campers)
- Program Planning Template
- Programming Support - VEX IQ Color Sensor
- **VEX IQ Sensor Overview** (I.cs2)

Online Resources:

- VEX IQ Videos – Default Sensor Functionality – VEX IQ Color Sensor Default Functionality - optional

  www.vexiq.com/documents-downloads/
- VEX IQ – YouTube Playlist – VEX IQ Color Sensor Default Functionality - optional

  www.vexiq.com/videos

Optional Materials:

- Different colors of electrical tape

Pre-learning Suggestions:

- If possible, complete **Learning Activity I – Autopilot Robot**.

Based on VEX IQ Robotics Education Guide:

- **Default Sensor Functionality Exercises** (I.7)
- **Simple Programming Exercises Using Programming Software** (I.9)

K.cl2

Detailed Directions

1. Have each pair of campers think carefully about the Autopilot Robot. Ask students to identify the VEX IQ Color Sensor and remove or disconnect all other sensors from the robot.

   Questions to ask campers:

   - What is the function of the Color Sensor?
   - Can you find an example of how a Color Sensor would function in a robot in the real world?
   - What kind of tasks would a Color Sensor enable this robot to do?

   If needed, provide campers with "VEX IQ Sensor Overview" Background Information for more specific details about sensor names, specs, use, and default functionality (how it works without any additional programming).

   If campers have not yet built the robot, provide them with time to do so. Instructions for its construction can be found in the kit documentation, VEX IQ Build Instructions – Autopilot Instructions.

2. Have campers test the Color Sensor Default Functionality by placing their robot on a flat surface. Have them hold a green Cube (or other object) above the Color Sensor and observe the robot's behavior. Have them hold a red Cube (or other object) above the Color Sensor and observe the robot's behavior.

   Campers should:

   1. Make sure that ONLY the VEX IQ Smart Motors and a VEX IQ Color Sensor are connected to the VEX IQ Robot Brain.
   2. Turn ON the VEX IQ Robot Brain and VEX IQ Controller.
   3. Select and run the Driver Control program.
   4. Use the VEX IQ Controller to try and drive the robot forward.
   5. Document what happens.

   The robot will start in an enabled mode. When the VEX IQ Color Sensor "sees" a green Cube (or other object) campers will be able to drive the robot. When the VEX IQ Color Sensor "sees" a red Cube or other object, the robot will be disabled and campers will NOT be able to drive the robot.

   Questions for campers:

   - For what kinds of robotics tasks or situations would this kind of functionality be useful?

3. Have campers return the driver control program to its default settings and open their programming software (either ModKit or ROBOTC). Tell them that they will be creating an original program that will make their robot drive autonomously (using programming instead of the controller) or stop based on the color of Cube detected by the Color Sensor.

4. Have campers create a program that drives when a green Cube is detected ("green light") and stops when a red Cube is detected ("red light"). Challenge them to create an additional behavior when a blue Cube is detected (e.g., changing speed, turning 90, or driving backwards autonomously).

   See the Programming Support - VEX IQ Color Sensor section of this activity for sample programs in Modkit and ROBOTC.

Questions to ask campers:

- How can you make the robot's response to the Cubes more interesting or useful?
- Instead of just driving and stopping, what could the robot do?
- How could you use this functionality to control not only whether the robot drives, but where it drives, without using a controller?

5. If time allows, have campers create a simple course to navigate on a VEX Highrise Challenge Field. This course may be as simple as navigating around an object by turning left, or as complex as moving around the field in a route, using strategically placed Cubes and autonomous programming to steer it.

   Encourage campers to use the handout, Program Planning Template to document their ideas and completed programs.

Tips and Best Practices:

- It is a good idea to provide students with some extra time to get to know the programming software they'll be using. They should be able to open the software, save a program, connect the programming computer to a VEX IQ robot. They should also be able to successfully transfer custom programs to the VEX IQ Robot Brain and run a program after it has been started.
- Make sure that the robot is not traveling too quickly as the Color Sensor needs time to detect colors.
- The lighting in the room may affect the color detection. Campers may find that the Color Sensor detects green light from white light. It may be easier to detect the colors blue or red, or to have the program detect green last.

Extension:

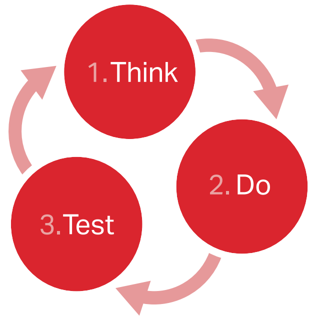
- Reposition the Color Sensor and use colored electrical tape (blue, red, or green) to create a path for the robot to follow on a flat surface. Challenge campers to create a program that will allow the robot to autonomously travel a path using the Color Sensor to detect and stay on the line.
- Practice calibrating and detecting different colors with different behaviors programmed for each color. A color wheel for Sensor Calibration has been provided in the kit documentation on the last page of the VEX IQ Control System User Guide.

K.cs1

Program Planning & Troubleshooting

Student Name(s): ____

Instructions:



Before completing this activity, be sure you are familiar with your programming software and how it works. Follow the steps on this page to document, test, troubleshoot, and refine your program. If your program includes more steps than fit on one page, use multiple copies as necessary.

Describe what you want your program to be able to do here:

Describe the device/robot you will be programming:	List the sensors that will be used in your program:

"THINK" Write step-by-step program instructions here.	"DO" Write your program using programming software and make notes here as you work.	"TEST" Does this step of your program function as expected? What needs improvement (NI)?
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:

If ANY program steps do NOT function as expected, address your "NI" items using as many copies of this page as necessary until all parts of your program function as expected.

Remember: Problems ARE NOT failures, they are an expected part of the design process!

K.cs2

Programming Support - Color Sensor

Modkit

To access helpful sample programs in Modkit for VEX Robotics, follow the link below:

Basic Color Sensor

http://help.modkit.com/#vex

ROBOTC

To access helpful sample programs in ROBOTC for VEX Robotics, open the software and click the File tab and select "Open Sample Programs."

Sample Programs for this activity include:

- Detect Wall
- Detect Color 12
- Color Mode Basic
- Wait for Color

Helpful VEX IQ Documents and Downloads

To access a variety of helpful resources for VEX IQ, follow the link below:

www.vexiq.com/documents-downloads/

VEX IQ Videos

- VEX IQ Color Sensor Default Functionality

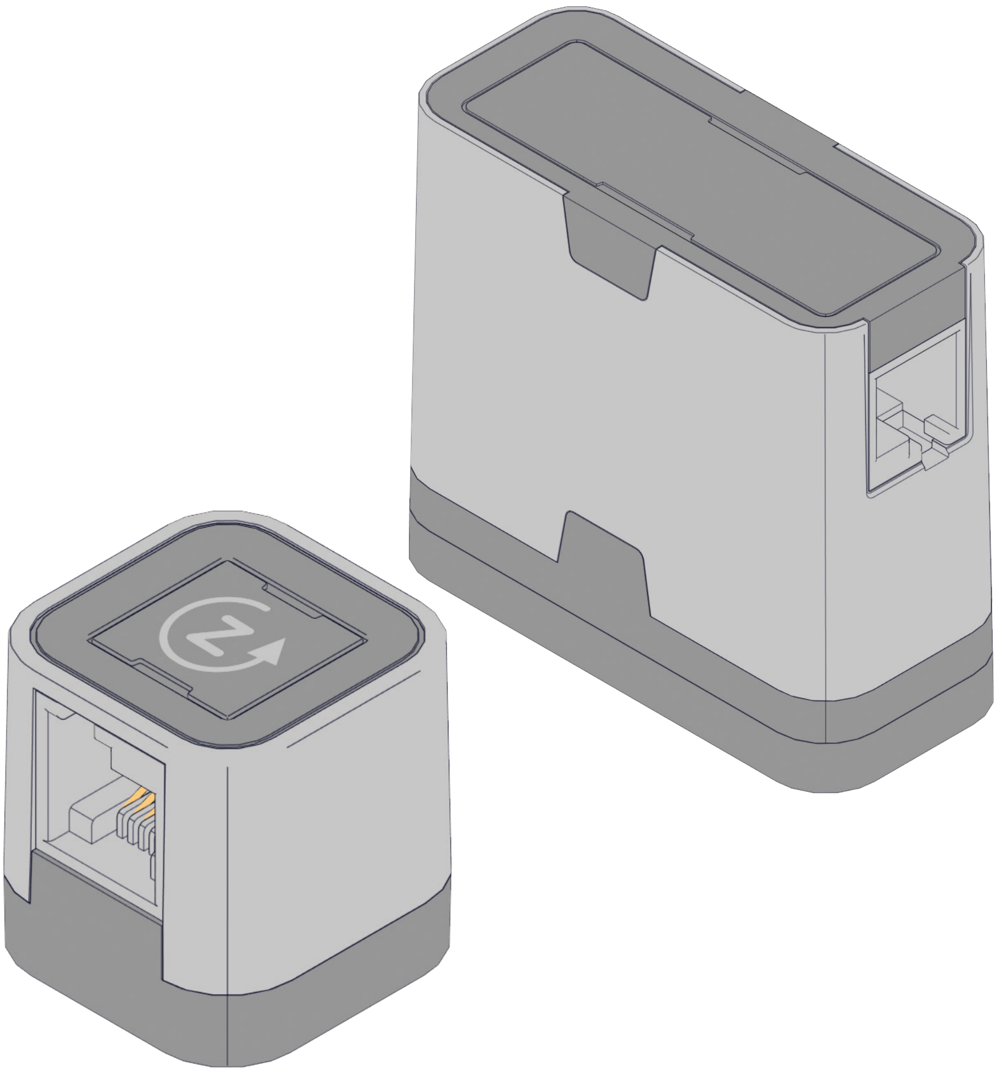
K.cs3

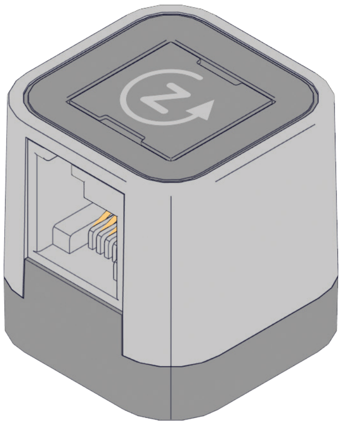
VEX IQ Sensor Overview

See I.cs2 for materials and instructions.

L

Gyro Sensor Default and Programming Task





L.cl1

Overview of Activity:

Campers will complete a "home direction" functionality exercise using the VEX IQ Gyro Sensor, to keep the robot pointed in the same direction when not being driven by the controller. Campers will use the autopilot robot with ONLY Smart Motors and a VEX IQ Gyro Sensor to test the basic functionality of the sensor and then make minor changes to the robot's behavior.

Campers will then use programming software (Modkit for VEX Robotics or ROBOTC for VEX Robotics) to complete an autonomous programming challenge to complete a "robot dance" with an obvious pattern of movement.

Learning Objectives:

- Campers will learn how to use a VEX IQ Gyro Sensor.
- Campers will program one sensor using selected software.
- Campers will transfer a program from software to the VEX IQ Robot Brain.
- Campers will learn to program basic decision-making logic.

Suggested Timing:

Up to 2 hours

- This activity may take more time if campers have not completed Learning Activity I – Autopilot Robot.
- This activity may take less time if campers have previous experience working with ModKit for VEX Robotics, ROBOTC for VEX Robotics or another programming language.
- This activity may take more time if campers have not had previous experience transferring and running programs on a robot.

L.cl1 cont.

VEX IQ Materials:

- Completed Autopilot robot
- VEX IQ Robot Controller
- Computer for programming
- Modkit for VEX Robotics or ROBOTC for VEX Robotics programming software
- Program Planning Template
- Programming Support - VEX IQ Gyro Sensor
- **VEX IQ Sensor Overview** (I.cs2)

Online Resources:

- VEX IQ Videos – Default Sensor Functionality – VEX IQ Gyro Sensor Default Functionality - optional

  www.vexiq.com/documents-downloads/
- VEX IQ – YouTube Playlist – VEX IQ Gyro Sensor Default Functionality - optional www.vexiq.com/videos

Pre-learning Suggestions:

- If possible, complete Learning Activity I – Autopilot Robot.

Based on VEX IQ Robotics Education Guide:

- **Default Sensor Functionality Exercises** (I.7)
- **Simple Programming Exercises Using Programming Software** (I.9)

Detailed Directions

1. Have each pair of campers think carefully about the Autopilot Robot. Ask students to identify the VEX IQ Gyro Sensor and remove or disconnect all other sensors from the robot.

   Questions to ask campers:

   - What is the function of a gyro sensor?
   - Can you find an example of how a gyro sensor would function in a robot in the real world?
   - What kind of tasks would a gyro sensor enable this robot to do?

   If needed, provide campers with "VEX IQ Sensor Overview" Background Information for more specific details about sensor names, specs, use, and default functionality (how it works without any additional programming).

   If campers have not yet built the robot, provide them with time to do so. Instructions for its construction can be found in the kit documentation, VEX IQ Build Instructions – Autopilot Instructions.

2. Have campers test the Gyro Sensor Default Functionality by placing their robot on a flat surface.

   Campers should:

   1. Make sure that ONLY the VEX IQ Smart Motors and a VEX IQ Gyro Sensor are connected to the VEX IQ Robot Brain.
   2. Turn ON the VEX IQ Robot Brain and VEX IQ Controller.
   3. Select and run the Driver Control program.
   4. Use the VEX IQ Controller to turn the robot to a new direction.
   5. Document what happens.

   When the camper stops driving or turning the robot with the VEX IQ Controller, the robot will automatically turn back to the original direction in which it was facing.

   If the robot is pushed or spun by a hand or contact with another robot, it will use the VEX IQ Gyro Sensor to measure how much it spun and automatically spin back to the original direction in which it was pointing.

   Questions for campers:

   - How could you use this function?
   - Describe some different robotics tasks or situations for which this function would be useful.

3. Have campers return the driver control program to its default settings and open their programming software (either ModKit for VEX Robotics or ROBOTC for VEX Robotics). Tell them that they will be creating an original program that will drive the robot autonomously (using programming instead of a controller) 5 motor revolutions, spin 180 degrees and then stop.

   See the Programming Support – VEX IQ Gyro Sensor section of this activity for sample programs in Modkit for VEX Robotics and ROBOTC for VEX Robotics.

L.cl2 cont.

Questions to ask campers:

- How can you make the robot's movement more interesting or useful?
- Can you make it spin several times before stopping? Which variables do you need to change to do this?
- Can you add one or more pauses to the spin? Which programming functions will you need to add to do this?
- Can you make the robot spin, stop and change directions? Which variables do you need to change to do this?

4. If time allows, have campers create an original robot "dance" using the VEX IQ Gyro Sensor and VEX IQ Smart motors. Encourage them to select music for their robot's choreography. The robot's routine should be a short, obvious pattern of movement, spins and pauses.

Tips and Best Practices:

- Make sure the robot is on a flat surface for its dance.
- It is a good idea to provide students with some extra time to get to know the programming software they will be using. They should be able to open the software, save a program, connect the programming computer to a VEX IQ robot, and successfully transfer custom programs to the VEX IQ Robot Brain, and run a program after it has been started.
- Make sure batteries are fully charged before beginning the activity.
- When programming ranges for the VEX IQ Gyro Sensor, it is a good idea to use "angle" in Modkit for VEX Robotics and "heading" in ROBOTC for VEX Robotics.

Extension:

- Have campers expand on their robot dance by adding assemblies or additional sensors to their robot (e.g., an arm and additional VEX IQ Smart Motors to move or articulate, a VEX IQ Touch LED to trigger different movements as a driver "interacts" with the dancer, a VEX IQ Color Sensor to trigger different movements or start and stop the dance when a Highrise cube is placed on the dance floor).
- As a robot completes its dance, have it describe the dance moves or pattern on the LCD screen of the VEX IQ Robot Brain.
- Have campers create a "home direction" program using the VEX IQ Gyro Sensor.

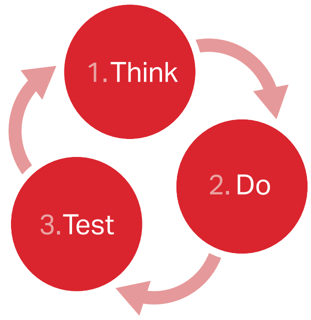
L.cs1

Program Planning & Troubleshooting

Student Name(s): ____

Instructions:

Before completing this activity, be sure you are familiar with your programming software and how it works. Follow the steps on this page to document, test, troubleshoot, and refine your program. If your program includes more steps than fit on one page, use multiple copies as necessary.



Describe what you want your program to be able to do here:	
Describe the device/robot you will be programming:	List the sensors that will be used in your program:

"THINK" Write step-by-step program instructions here.	"DO" Write your program using programming software and make notes here as you work.	"TEST" Does this step of your program function as expected? What needs improvement (NI)?
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:

If ANY program steps do NOT function as expected, address your "NI" items using as many copies of this page as necessary until all parts of your program function as expected.

Remember: Problems ARE NOT failures, they are an expected part of the design process!

L.cs2

Programming Support - VEX IQ Gyro Sensor

Modkit for VEX Robotics

To access helpful sample programs in Modkit for VEX Robotics, follow the link below:

Basic Gyro Programming

http://help.modkit.com/#vex

ROBOTC for VEX Robotics

To access helpful sample programs in ROBOTC for VEX Robotics, open the software and click the File tab and select "Open Sample Programs."

Sample Programs for this activity include:

- 90 degrees turn

Helpful VEX IQ Documents and Downloads

To access a variety of helpful resources for VEX IQ, follow the link below:

www.vexiq.com/documents-downloads/

VEX IQ Videos

- VEX IQ Gyro Sensor Default Functionality

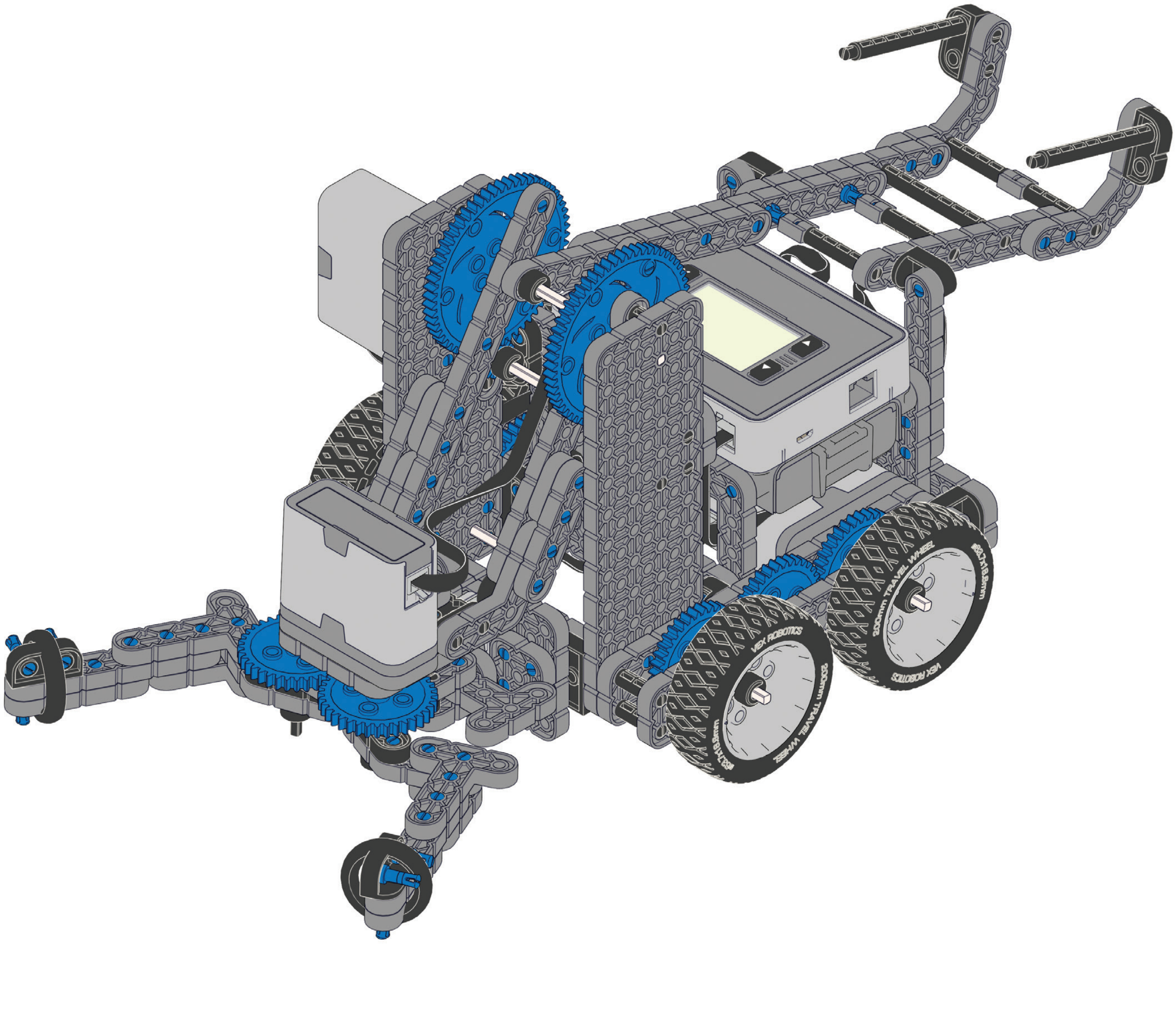
L.cs3

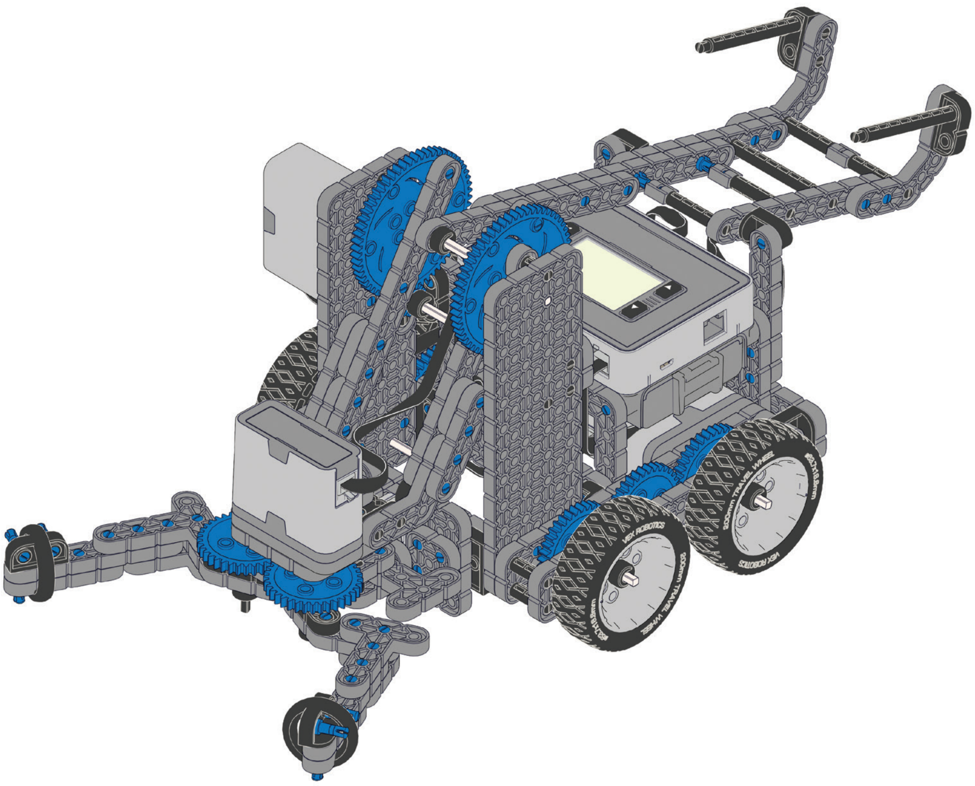
VEX IQ Sensor Overview

See l.cs2 for materials and instructions.

M

Clawbot IQ





M.cl1

Overview of Activity:

From the Standard Drive Base, campers will complete the Clawbot IQ Tower, Object Holder, and Claw. They will practice picking up different objects that may include a basket created from VEX IQ components, a VEX IQ Tire, and a Highrise Cube. They will make observations about how these objects affect their robot as they are transferred to a prototype containment unit (made from a paper template) for the Clawbot Object Holder.

Campers will be challenged to build a containment unit for the Clawbot IQ Object Holder based on experience working from a basic prototype.

Learning Objectives:

- Campers will learn about and apply knowledge of object manipulator: Friction Grabber.
- Campers will learn about and apply knowledge of Center of Gravity.
- Campers will gain experience working with and iterating design testing and improving from a basic prototype.

Suggested Timing:

Up to 2 hours

- This activity will take more time if campers have not yet completed the Standard Drive Base or if they need to remove elements from the Autopilot Robot to return to the Standard Drive Base construction.
- This activity will take less time if campers have already completed the full Clawbot IQ robot.

M.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Standard Drive Base
- VEX IQ Controller
- Extra VEX IQ Tires
- VEX IQ Highrise Cubes
- Tape and scissors
- VEX IQ Build Instructions - Clawbot IQ (Kit Documentation) *1+2*
- Think, Do, Test – Clawbot IQ
- Paper Prototype Containment Unit Template
- Center of Gravity

Optional Materials:

- Plastic cups, action figures, chess pieces, or other objects of different sizes, weight, shape, etc.

Pre-learning Suggestions:

- If possible, complete **Learning Activity H – Standard Drive Base with Lifting Mechanism** and **Learning Activity G – Standard Drive Base with Object Manipulator**.

Based on VEX IQ Robotics Education Guide:

- **Build and Test Clawbot IQ – Build Options – Option 1** (C.2)
- **Your First Robot Idea Book Pages**: Standard Drive Base, Clawbot IQ Tower, Clawbot IQ Object Holder, Clawbot IQ Claw. (C.6)
- **Center of Gravity** (F.3)
- **Mechanisms: Object Manipulation – Friction Grabbers** (G.5)

M.cl2

Detailed Directions

1. Provide each pair of campers with three objects: a basket created from VEX IQ components, a VEX IQ Tire, and a Highrise Cube. Ask them to identify a few physical characteristics for each object that may help determine the kind of object manipulator required to pick them up and carry them.

   Questions to ask campers:

   - What kind of object manipulator would work to lift and move all of these objects?
   - Why is this object manipulator ideally suited to this task?

2. Tell campers they will be building a robot that can not only pick up these objects, but hold them.

   If it hasn't yet been done, build the Standard Drive Base using the kit documentation, VEX IQ Build Instructions – Standard Drive Base.

3. Provide campers with VEX IQ Build Instructions – Clawbot IQ from the kit documentation. Provide them with time to try out their newly constructed robot using the VEX IQ Controller.

   Alternatively, the Clawbot IQ can be controlled, one motor at a time, using the buttons on the LCD screen of the VEX IQ Robot Brain.

   Questions to ask campers:

   - How will this robot work to pick up and place objects in the Object Holder?
   - Can all of these objects be contained in the Object Holder? Which objects can be and which cannot?
   - Where is the center of gravity on the robot?
   - What happens to the robot's center of gravity when different objects are placed in the object holder? Is this a problem? If so, how can it be remedied?

   If needed, provide campers with the handout "Center of Gravity" Background Information. This document describes how collecting, holding or manipulating objects can affect a robot's center of gravity.
4. Provide campers with the Paper Prototype Containment Unit Template and have them assemble and place it in the object holder. Have them conduct a series of tests to determine the unit's efficacy for different kinds of functionality and durability. Encourage campers to test both placing the objects into the prototype containment unit, as well removing them using the Friction Grabbers.

   Questions to Ask Campers:

   - How would a containment unit like this be helpful?
   - Would this prototype be an effective addition to a robot with a task to collect several objects and then place them in a new location? Why or why not?
   - How would you change the shape or size of the containment unit?
   - Could you build a similar or better containment unit using VEX IQ components?
   - How would the containment unit affect your robot's center of gravity?
5. Have campers design, build and test their containment units and share their solutions with each other. Encourage campers to use the handout, Think, Do, Test – Clawbot IQ to document their ideas.

Tips and Best Practices:

- A prototype is an early sample, model or release of a product that has been created to test a concept or process. Some campers may benefit from finding examples online of prototypes used to create their favorite products. This is also a good opportunity to discuss the role of the prototype in the engineering design process.
- It may be a good idea to have pairs of campers focus on collecting and containing a specific kind of object (e.g., Highrise Cubes in preparation for the Highrise Challenge.)
- The Claw may not reach all the way to the bottom of the Holder to retrieve objects. Campers can be challenged to modify the robot design to make this possible.

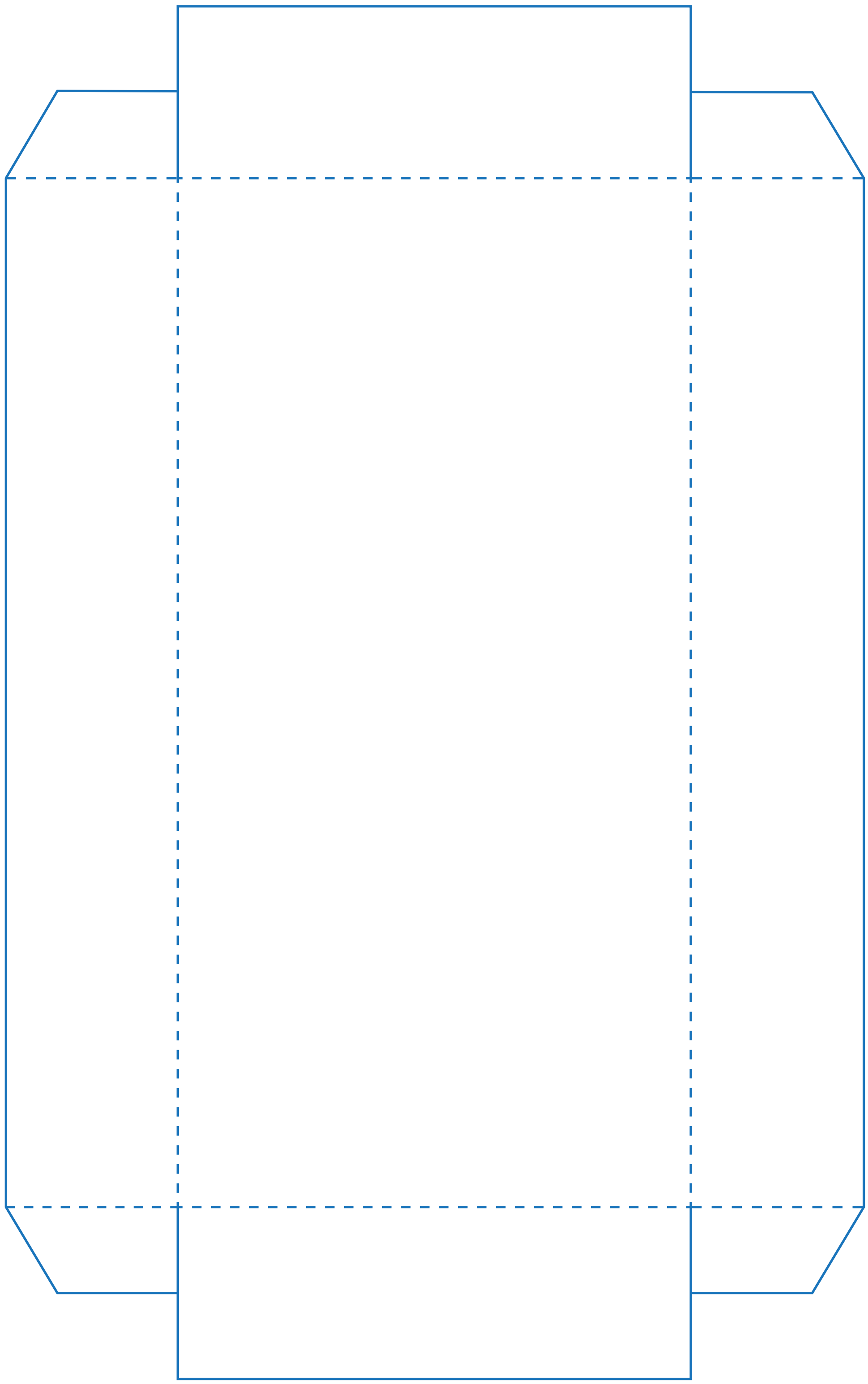
M.cl2 cont.

Extension:

- Have campers test their collection containment units with different kinds of objects (e.g., gathering plastic cups, action figures, etc.) and think about real world scenarios where a robot with these functions could be used. Encourage campers to test and revise their containment units and retrieval mechanisms for a very particular use case or object.
- Modify the Holder on the robot to “dump” the items in its containers after collection and transport for easy pick up and stacking.

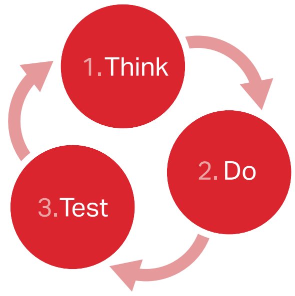
M.cs1

Paper Prototype Containment Unit Template



M.cs2

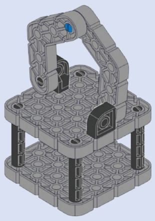
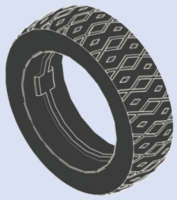
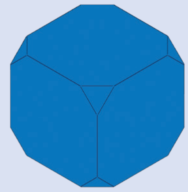
Think, Do, Test - Standard Drive Base with Lifting Mechanism



Student Name(s): ____

Instructions:

Look at the objects below.

		
Basket	Tire	Highrise Cube

Begin with the Standard Drive Base or a robot of your own creation and complete the build for the Clawbot IQ including the Tower, Holder, and Claw. Your job is to “imagine” how the objects shown above could be, not only picked up, but collected and stored using this robot.

A basic, paper prototype for a containment unit will be provided by your camp leader for you to assemble and add to your robot

1. “THINK” - Test the prototype and think carefully about how well it functions for the different objects. Make observations about the materials used, how well it works for different objects, and how you would change the design to improve it.

____

____

____

____

Draw the objects here and write notes or labels to describe your observations about them

M.cs2 cont.

2. "DO" - Draw and describe the assembly you will build out of VEX IQ components to effectively contain the objects. Name it, label its parts, show and describe how it would work to both hold the objects and allow for easy retrieval using the Clawbot IQ claw. Make sure to note any modifications that would need to be made to the rest of the Clawbot IQ construction.

Draw, name, and label your containment unit design and how it will work.

3. "TEST" – Build your containment unit. Test your design. Write down your observations in a notebook, or use a smart device to record your observations as video or audio.

____

____

____

____

____

Did the containment unit function as planned? YES NO

If you answered "YES" – Congratulations! Ask your camp leader for an additional challenge.

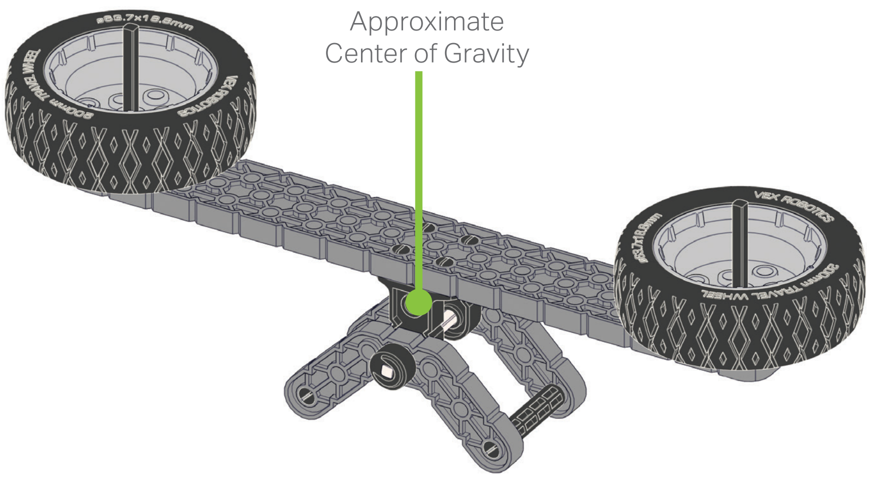
If you answered "NO" – Use your observations above to determine what problem exists, then use another copy of this page to help solve that problem. Keep repeating this "THINK-DO-TEST" process until your robot completes the task.

Problems ARE NOT failures. They are an expected part of the design process!

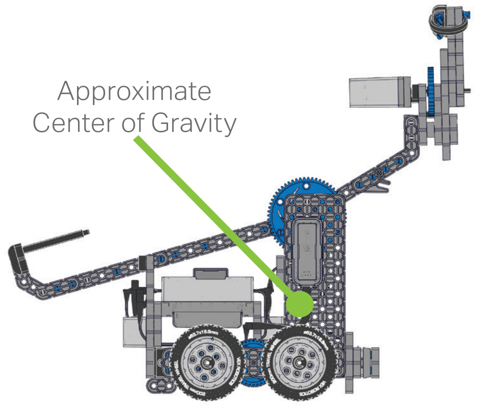
M.cs3

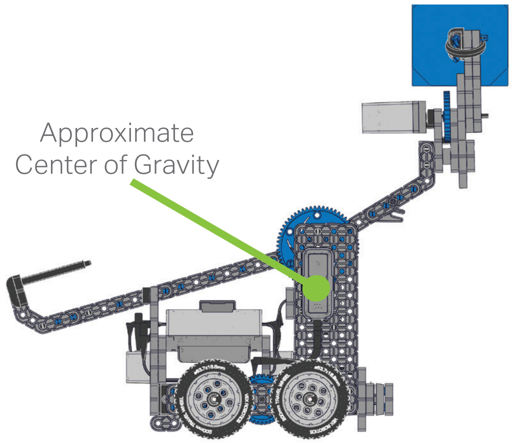
Center of Gravity

Center of Gravity is the place in a system or body (such as a robot) where the weight is evenly distributed and all sides are in balance. An example of center of gravity is the middle of a seesaw when it is balanced.

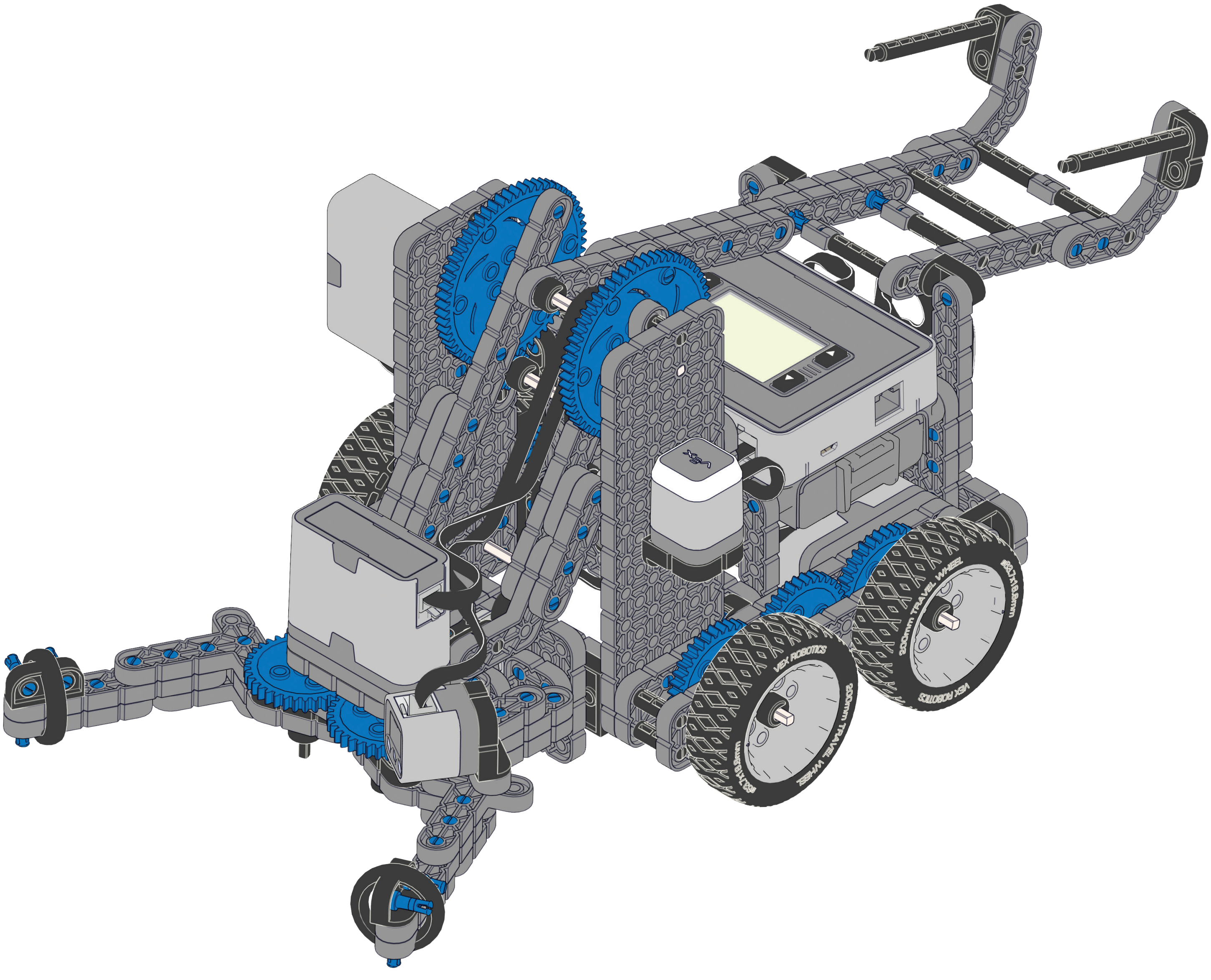


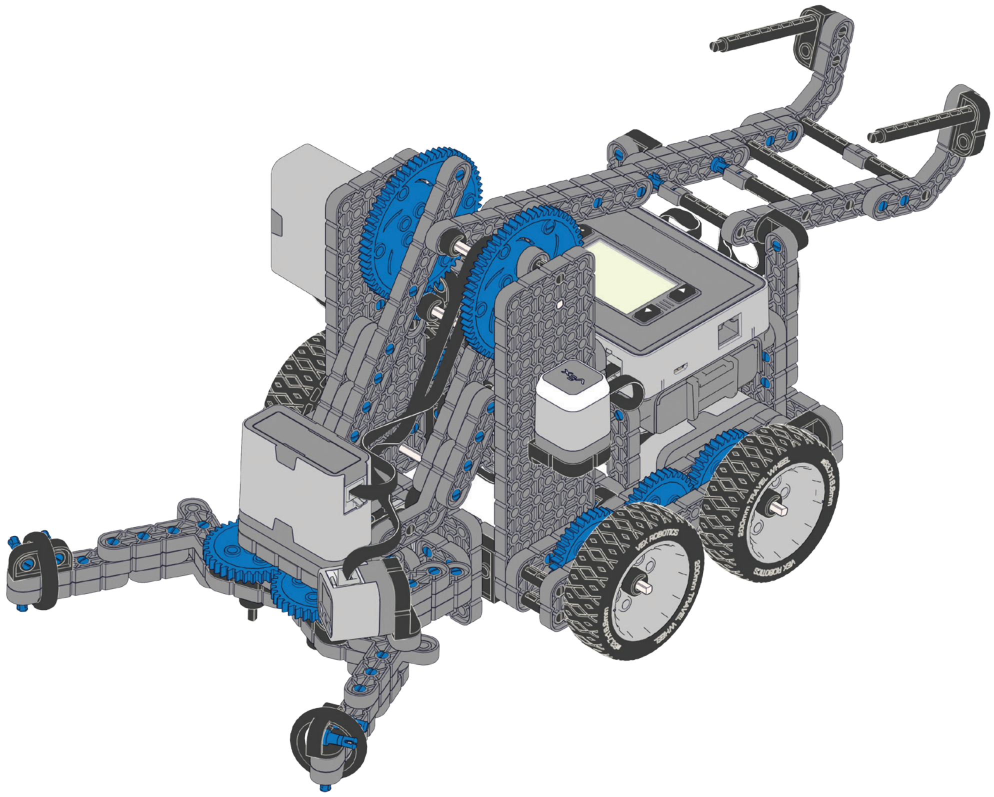
You can think of a robot's center of gravity as the "center position" of all the weight on the robot. Because **Center of Gravity** uses both weight and position, heavier objects have a greater effect than lighter ones in determining where the center of gravity is. For example, if your robot can collect, hold, and/or manipulate objects, those objects change the center of gravity as they are being manipulated because they add weight.





Clawbot IQ with Sensors Task – Autonomous





N.cl1

Overview of Activity:

Campers will add the VEX IQ Touch LED and VEX IQ Gyro Sensor to the Clawbot IQ and test their functionality. The robot will begin the challenge with a small tower created from VEX IQ components in its Claw. Campers will program the robot to run autonomously when the VEX IQ Touch LED is tapped.

The robot should turn 180 degrees to face the opposite direction, travel a designated distance, and drop the tower on a target.

Learning Objectives:

- Campers will learn how to use the VEX IQ Touch LED.
- Campers will create a program for the VEX IQ Gyro Sensor.
- Campers will learn to revise and customize sensor functionality.
- Campers will create an autonomous program to accomplish a specific goal.

Suggested Timing:

Up to 2 hours

- This activity will take more time if campers have not completed Learning Activity I – Standard Drive Base with Object Manipulator, Learning Activity L – Gyro Sensor Default and Programming Task, or Learning Activity M – Clawbot IQ.
- This activity will take more time if campers have not already constructed the Clawbot IQ.

N.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Clawbot IQ
- Computer for programming
- Modkit for VEX Robotics or ROBOTC for VEX Robotics programming software
- VEX IQ Build Instructions - Clawbot IQ with Sensors (Kit Documentation)
- Programming Planning Template
- Sample Tower Image

Optional Materials:

- Sample Program Breakdown for Modkit Clawbot IQ with Sensors Task - Autonomous
- Sample Program Breakdown for ROBOTC Clawbot IQ with Sensors Task - Autonomous
- Electrical or masking tape to mark a target area
- Plastic cups filled with cotton balls can also be used in place of the small tower made from VEX IQ components

Pre-learning Suggestions:

- If possible, complete **Learning Activity I – Standard Drive Base with Object Manipulator**, **Learning Activity L – Gyro Sensor Default and Programming Task**, and **Learning Activity M – Clawbot IQ**.

Based on VEX IQ Curriculum:

- **Build and Test Clawbot IQ – Build Options – Option 2** (C.2)
- **Smarter Machines Unit Challenges** (K.5)

N.cl2

Detailed Directions

1. Have each pair of campers think carefully about the Clawbot IQ they have constructed.

   If campers have not yet built the robot, provide them with time to do so. Instructions for its construction can be found in the kit documentation, VEX IQ Build Instructions – Clawbot IQ.

   Questions to ask campers:

   - What can this robot do?
   - What are its limitations?
   - How can you control it?
   - What kind of information will you need to give the robot so that it can make decisions?
   - What kind of decisions will the robot need to be able to make for itself?

2. Tell campers that they will be adding sensors and creating programs that will help their robot make decisions and complete a task without the use of a controller.

   If needed, provide campers with "VEX IQ Sensor Overview" Background Information for more specific details about sensor names, specs, use, and default functionality (how it works without any additional programming).

3. Provide campers with the kit documentation, VEX IQ Build Instructions – Clawbot IQ with Sensors. Give them time to complete the construction of their robots. Encourage them to identify the sensors being added to the robot and talk about their functionality.

4. Have campers build a small tower from VEX IQ components using the Sample Tower Image or encourage them to create their own design.

5. Tell campers that their job is to create a program that will allow their robot to autonomously (using programming instead of a controller) carry this tower from a starting point to a predetermined target area.

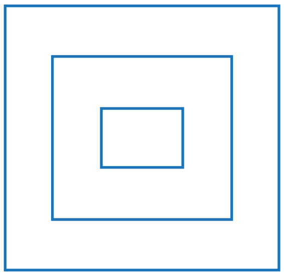
   This task can take place on a Highrise Challenge Field or on any flat surface.

   If using a Highrise Challenge Field, designate one grid line as a "starting line" and one square as the "target." Score robot achievement of the goal as follows:

   - 3 Points: Place the tower at the center of the square.
   - 2 Points: Place the tower completely in the square, but not in the center.
   - 1 Point: Place some part of the tower in the square or on one of the grid lines surrounding the square.

   If the tower falls over, no points will be awarded.

   If using a flat surface, use electrical or masking tape to create a start line and a series of concentric circles or squares a couple feet away from the starting line. Use the same scoring system as above.



Sample concentric squares

   Have campers return the driver control program to its default settings and open their programming software (either Modkit for VEX Robotics or ROBOTC for VEX Robotics). Tell them that they will be creating an original program that will turn the robot 180 degrees to face the opposite direction, travel a designated distance, and drop the tower on a target.

   If needed, see the Sample Program Breakdown for Modkit Clawbot IQ with Sensors Task - Autonomous or Sample Program Breakdown for ROBOTC Clawbot IQ with Sensors Task – Autonomous.

   Questions to Ask Campers:

   - What are the smaller tasks that your robot will need to complete in order to achieve its larger goal?
   - What decisions will the robot need to make?
   - What kinds of information will it need to gather to make those decisions?
   - Which sensors will you use to complete this task?
   - How will you test your program to make sure it functions as expected?
   - How will you document your successes and challenges so that you can determine what the problem was?

6. Provide campers with time to create and test their programs.

N.cl2 cont.

Tips and Best Practices:

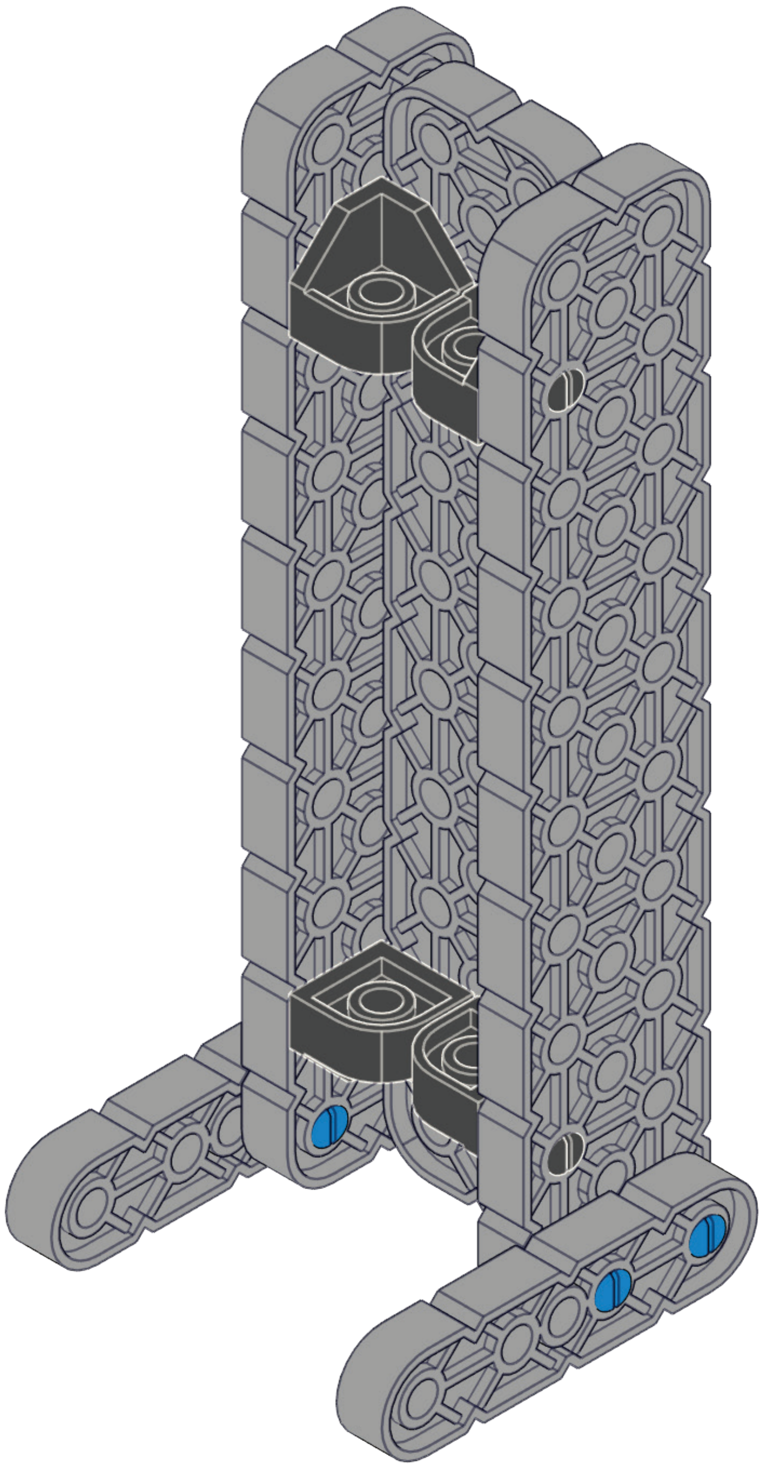
- It is a good idea to provide students with some extra time to get to know the programming software they'll be using. They should be able to open the software, save a program, connect the programming computer to a VEX IQ robot, and successfully transfer custom programs to the VEX IQ Robot Brain, and run a program after it has been started.
- Make sure batteries are fully charged before beginning the activity.

Extension:

- Program the VEX IQ Touch LED to turn red at the conclusion of the program and display a message such as "I am done." Have campers think of a real world situation in which this kind of task would take place and have the message reflect that situation. (e.g., "Here are the parts you ordered." or "Pizza delivered.")
- Reposition the Color Sensor on the robot and program it to follow a line to a target before placing the container there.

N.cs1

Sample Tower Image



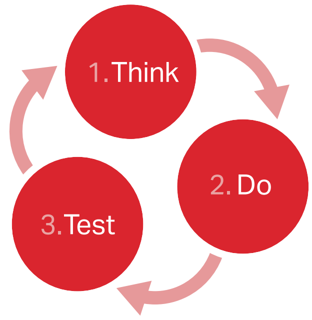
N.cs2

Program Planning & Troubleshooting

Student Name(s): ______________________________

Instructions:

Before completing this activity, be sure you are familiar with your programming software and how it works. Follow the steps on this page to document, test, troubleshoot, and refine your program. If your program includes more steps than fit on one page, use multiple copies as necessary.



Describe what you want your program to be able to do here:	
Describe the device/robot you will be programming:	List the sensors that will be used in your program:

"THINK" Write step-by-step program instructions here.	"DO" Write your program using programming software and make notes here as you work.	"TEST" Does this step of your program function as expected? What needs improvement (NI)?
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:

If ANY program steps do NOT function as expected, address your "NI" items using as many copies of this page as necessary until all parts of your program function as expected.

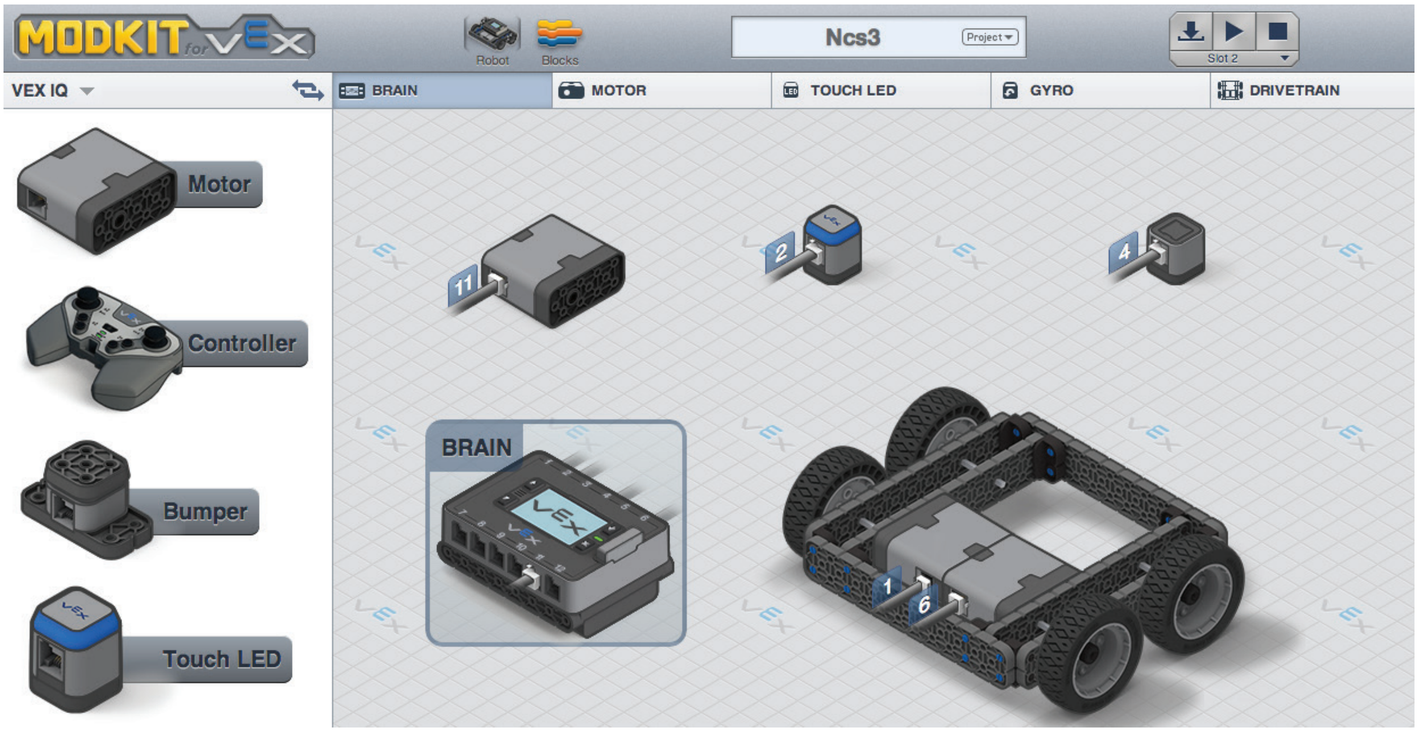
Remember: Problems ARE NOT failures, they are an expected part of the design process!

N.cs3

Sample Program Breakdown for Modkit Clawbot IQ with Sensors Task - Autonomous

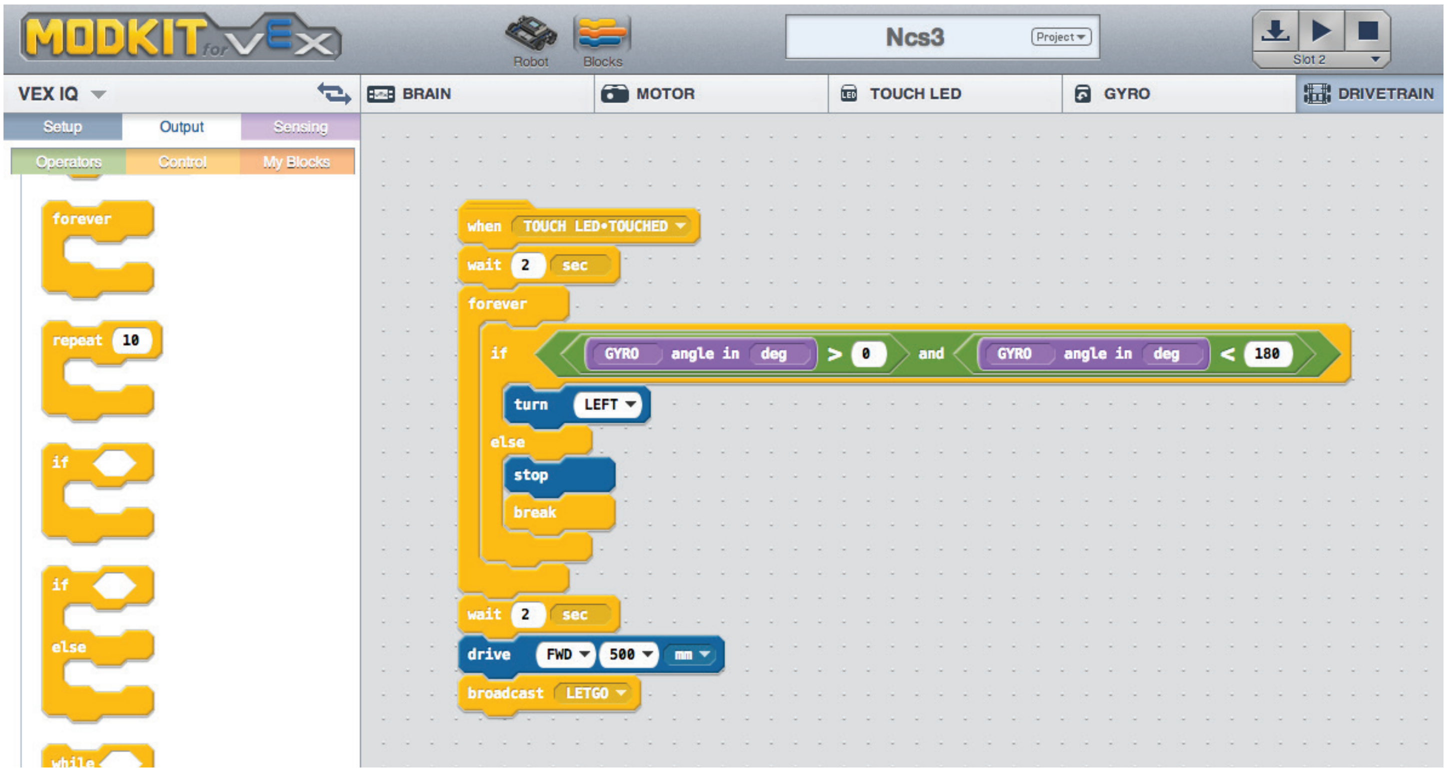
Summary of Program:

This program combines the VEX IQ Touch LED and the VEX IQ Gyro Sensor with Smart Motors and the VEX IQ Robot Brain, so the VEX IQ Clawbot with Sensors can grip and carry an object from one location to another. When the Touch LED is tapped, the robot will grip a Cube, rotate to face the opposite direction, travel a designated distance, and release the object on a target.

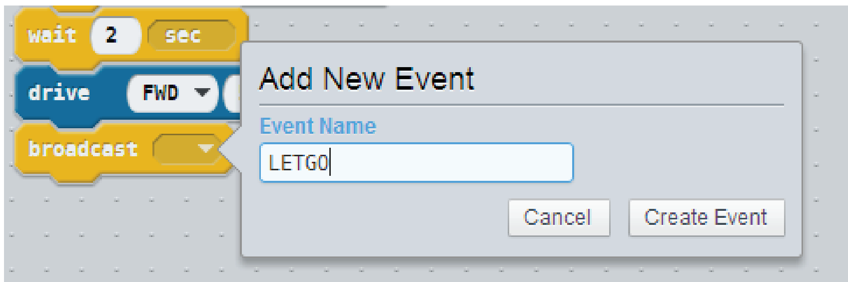


STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.

N.cs3 cont.

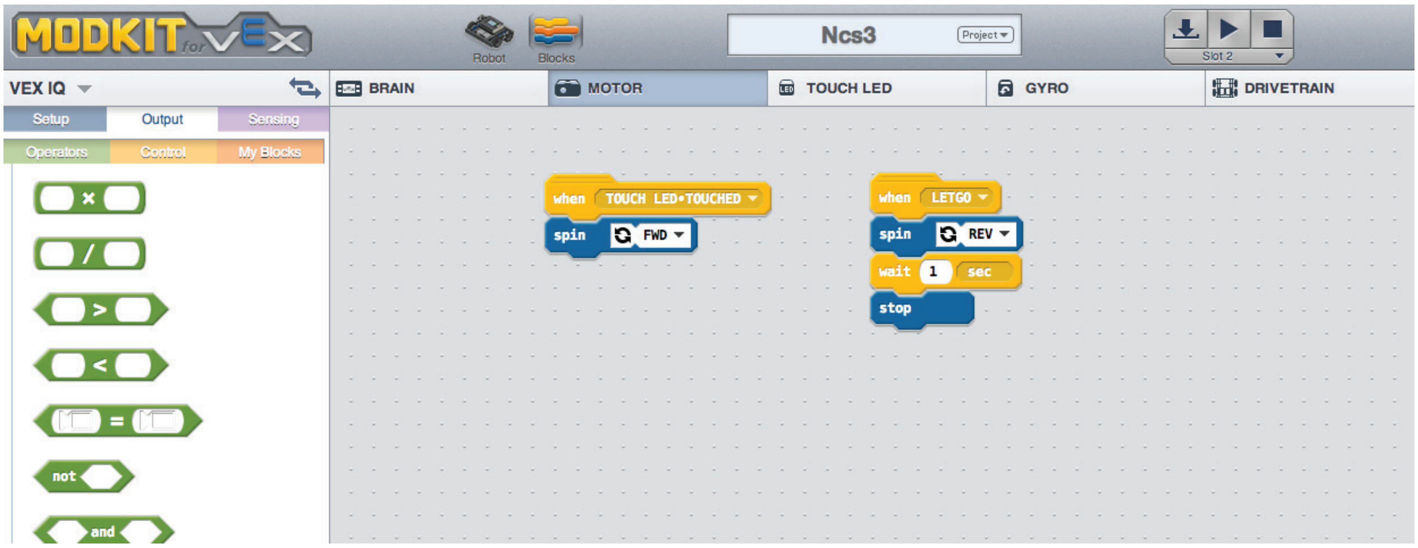


STEP 2: Select BLOCKS view and the DRIVETRAIN tab. Create and save this block of code that controls drivetrain movement using the Gyro Sensor to turn.

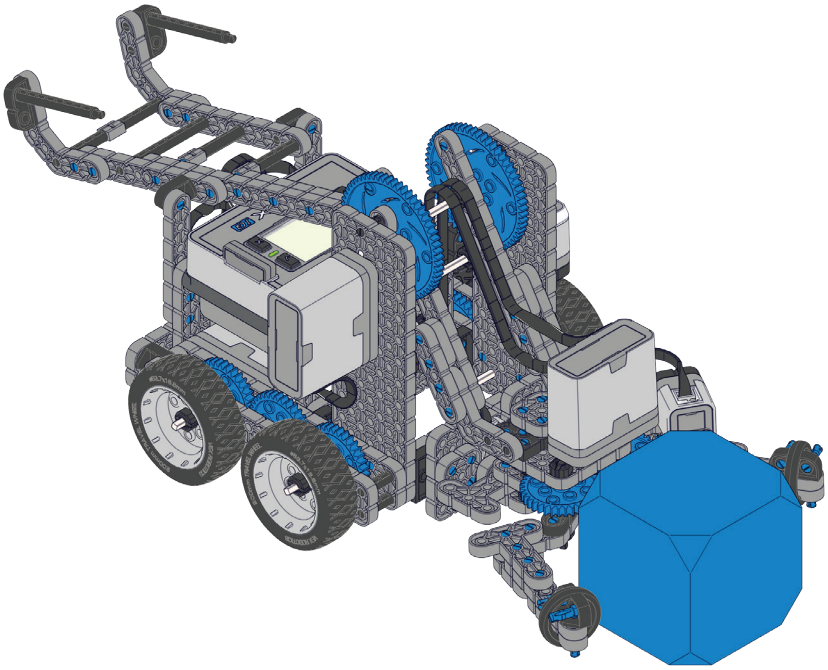


Note: You must create the "LETGO" event by choosing "new" from the drop down of the "BROADCAST" control block, then typing in the new event name.

N.cs3 cont.



STEP 3: Select BLOCKS view and the MOTOR tab. Create and save these blocks of code that control the claw's movement. Then download the program to a chosen slot on the VEX IQ Robot Brain.



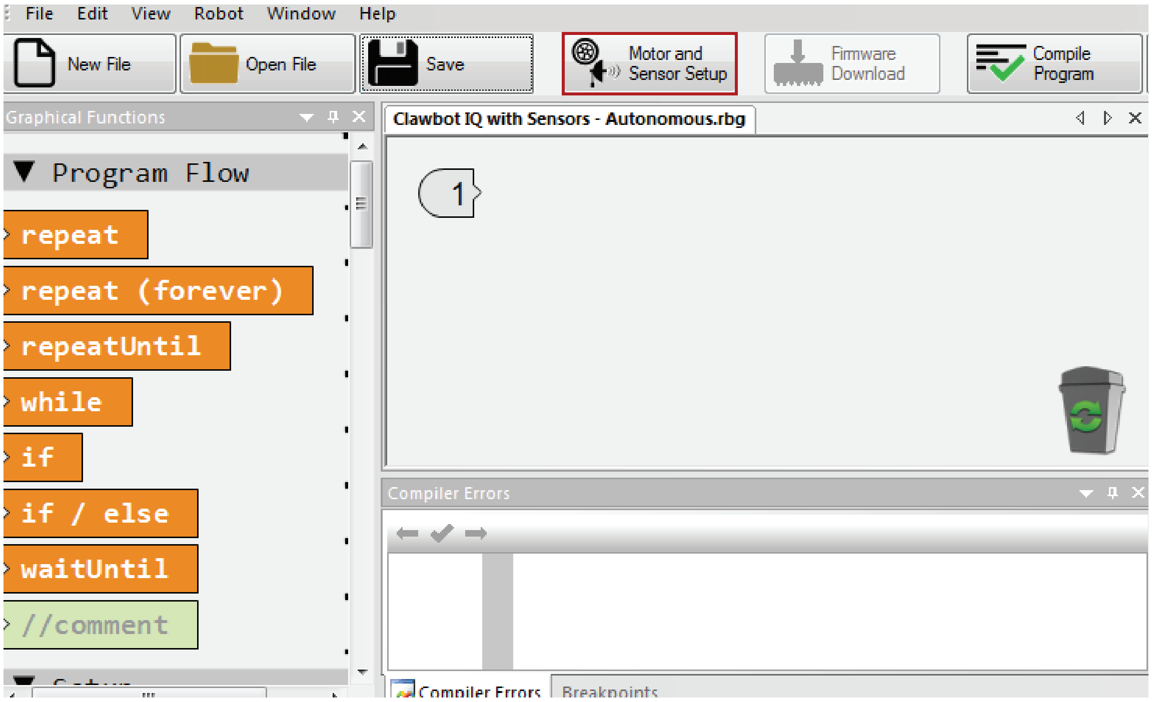
STEP 4: Place the robot down with the claw open and surrounding a Cube, then run this program to turn with the Gyro Sensor and deliver the Cube.

N.cs4

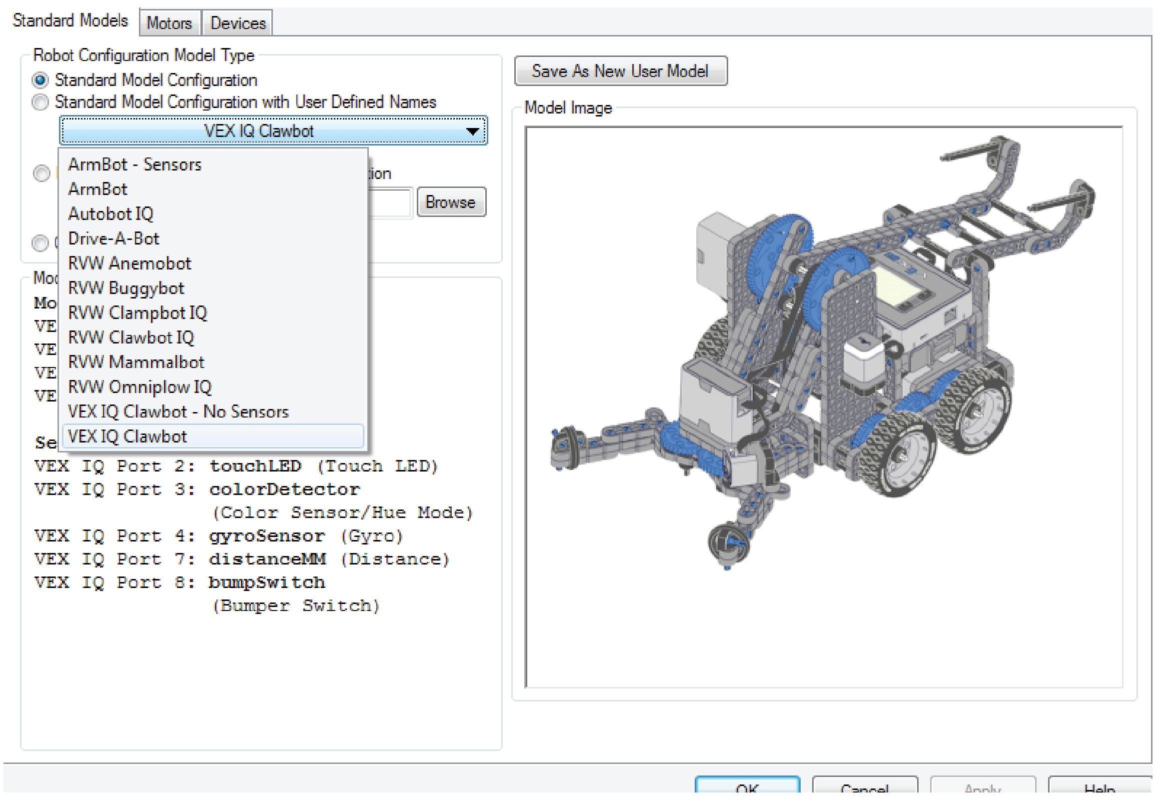
Sample Program Breakdown for ROBOTC Clawbot IQ with Sensors Task - Autonomous

Summary of Program:

This program combines the VEX IQ Touch LED and the VEX IQ Gyro Sensor with Smart Motors and the VEX IQ Robot Brain to grip and carry an object from one location to another. When the VEX IQ Touch LED is tapped, the robot will rotate 180 degrees to face the opposite direction, travel a designated distance, and release the object on a target.



STEP 1: Select the Motor and Sensor Setup button.



STEP 2: Select VEX IQ Clawbot from the Standard Robot Model Configuration.

```
Clawbot IQ with Sensors - Autonomous.rbg*

//  Keeps looping forever
2   repeat (forever ) {
3
4   }
5
```

STEP 3: To make the motor spin so the Claw will grab the object, start by creating an infinite loop (a program that runs forever).

```
Clawbot IQ with Sensors - Autonomous.rbg*

//  Keeps looping forever
2   repeat (forever ) {
//      If the Touch LED is pressed (value of 1)...
4       if ( getTouchLEDValue(touchLED) == 1 ) {
5
6       }
7   }
8
```

STEP 4: To make the motor in the drive train start when the Touch LED is being pressed, set the value of the LED to 1.

If the LED is not being pressed, it will automatically have a value of 0.

```
Clawbot IQ with Sensors - Autonomous.rbg*

//  Keeps looping forever
2   repeat (forever ) {
//      If the Touch LED is pressed (value of 1)...
4       if ( getTouchLEDValue(touchLED) == 1 ) {
//          Turn the claw motor on at power level 25
6           setMotor ( clawMotor , 25 );
7       }
8   }
9
```

STEP 5: To make the Claw on the Clawbot IQ grab onto an object, set the MotorSpeed to any positive number.

N.cs4 cont.

```
// Keeps looping forever
repeat (forever ) {
    // If the Touch LED is pressed (value of 1)...
    if ( getTouchLEDValue(touchLED) == 1 ) {
        // Turn the claw motor on at power level 25
        setMotor ( clawMotor , 25 );
        // While the robot's rotation is less than 180 degrees
        while ( getGyroDegrees(gyroSensor) < 180 ) {

        }
    }
}
```

STEP 6: To add a condition that will make the drive train turn 180 degrees, create a while loop. The loop will have the VEX IQ Gyro Sensor check if the robot has rotated 180 degrees.

```
// Keeps looping forever
repeat (forever ) {
    // If the Touch LED is pressed (value of 1)...
    if ( getTouchLEDValue(touchLED) == 1 ) {
        // Turn the claw motor on at power level 25
        setMotor ( clawMotor , 25 );
        // Reset the gyro to 0 degrees
        resetGyro ( gyroSensor );
        // While the robot's rotation is less than 180 degrees
        while ( getGyroDegrees(gyroSensor) < 180 ) {

        }
    }
}
```

Note: The 'resetGyro' command should be used before using the gyro in the while loop, to ensure that any drift is accounted for (and the gyro starts from a known value)

Clawbot IQ with Sensors - Autonomous.rbg

```
// Keeps looping forever
repeat (forever ) {
    // If the Touch LED is pressed (value of 1)...
    if ( getTouchLEDValue(touchLED) == 1 ) {
        // Turn the claw motor on at power level 25
        setMotor ( clawMotor , 25 );
        // Reset the gyro to 0 degrees
        resetGyro ( gyroSensor );
        // While the robot's rotation is less than 180 degrees
        while ( getGyroDegrees(gyroSensor) < 180 ) {
            // Slowly turn the robot to the left
            setMotor ( leftMotor , -25 );
            setMotor ( rightMotor , 25 );
        }
    }
}
```

STEP 7: If the robot has not turned 180 degrees, set the MotorSpeed to do so by configuring the leftMotor to rotate any negative number and the rightMotor to rotate any positive number.

```
// Keeps looping forever
repeat (forever ) {
  // If the Touch LED is pressed (value of 1)...
  if ( getTouchLEDValue(touchLED) == 1 ) {
    // Turn the claw motor on at power level 25
    setMotor ( clawMotor , 25 );
    // Reset the gyro to 0 degrees
    resetGyro ( gyroSensor );
    // While the robot's rotation is less than 180 degrees
    while ( getGyroDegrees(gyroSensor) < 180 ) {
      // Slowly turn the robot to the left
      setMotor ( leftMotor , -25 );
      setMotor ( rightMotor , 25 );
    }
    // Forward for 5 seconds at power level 50
    forward ( 5 , seconds , 50 );
  }
}
```

STEP 8: To stop the robot from turning, and make it drive forward, set the leftMotor and rightMotor for 5 seconds. Configure it to sleep (pause) after 5000 milliseconds (5 seconds).

The forward command will automatically stop the robot after 5 seconds have passed.

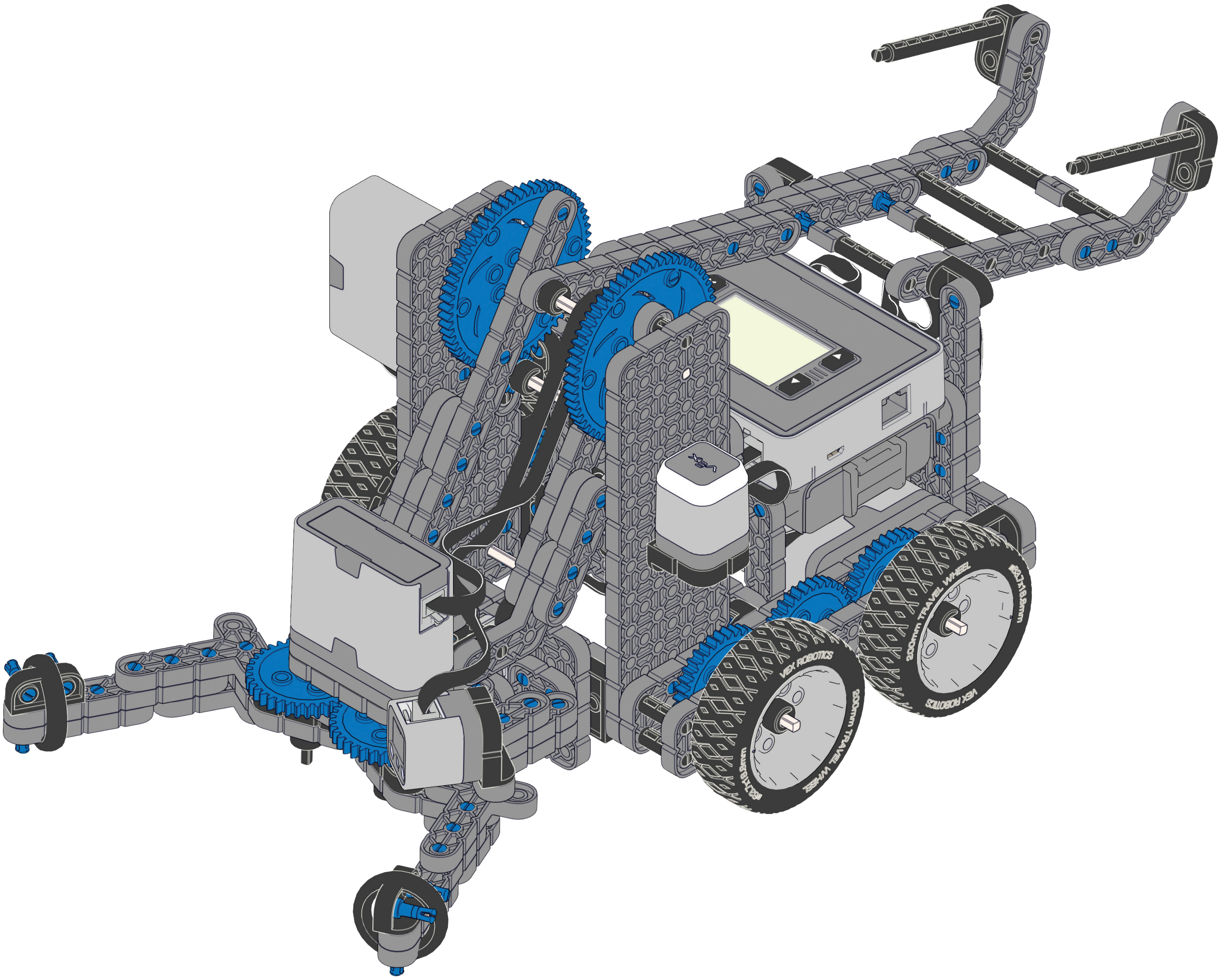
```
// Keeps looping forever
repeat (forever ) {
  // If the Touch LED is pressed (value of 1)...
  if ( getTouchLEDValue(touchLED) == 1 ) {
    // Turn the claw motor on at power level 25
    setMotor ( clawMotor , 25 );
    // Reset the gyro to 0 degrees
    resetGyro ( gyroSensor );
    // While the robot's rotation is less than 180 degrees
    while ( getGyroDegrees(gyroSensor) < 180 ) {
      // Slowly turn the robot to the left
      setMotor ( leftMotor , -25 );
      setMotor ( rightMotor , 25 );
    }
    // Forward for 5 seconds at power level 50
    forward ( 5 , seconds , 50 );
    // Open the claw for 3 seconds at power level 50
    moveMotor ( clawMotor , 3 , seconds , -50 );
  }
}
```

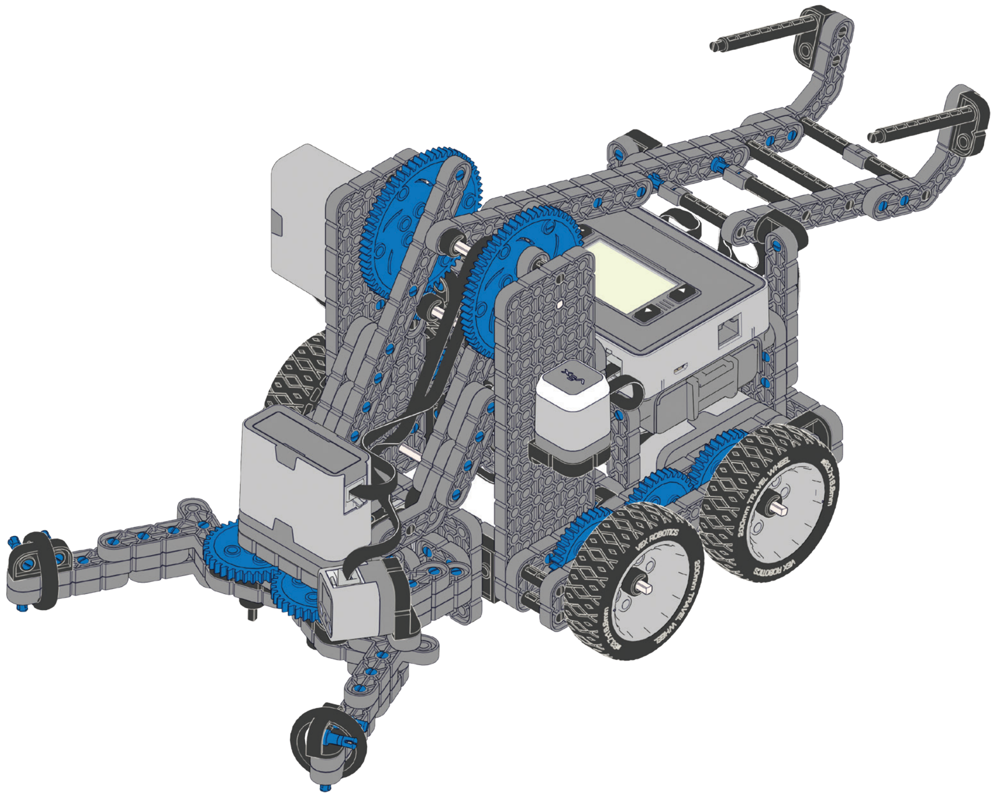
STEP 9: To stop the robot from driving forward, stop the leftMotor and the rightMotor. Set the MotorSpeed to 0 for both.

To release the Claw on the Clawbot IQ and allow some time for it to fully release the object, set the MotorSpeed for the clawMotor to any negative number and then set it to sleep (pause) after 3000 milliseconds (3 seconds).

O

Clawbot IQ with Sensors Task – Hybrid





O.cl1

Overview of Activity:

Campers will select and connect appropriate sensors to the Clawbot IQ to complete one or more of three tasks. In these tasks, the robot will be controlled with the VEX IQ Controller, but certain functions will occur autonomously.

Campers will test their functions one at a time or all together using their Controllers.

Learning Objectives:

- Campers will learn to select appropriate sensors for a particular task.
- Campers will program autonomous functions that integrate with controlled driving of the robot.
- Campers will be able to troubleshoot and solve problems to improve their design.

Suggested Timing:

Up to 2 hours

- This activity will take more time if campers have not completed Learning Activity I – Standard Drive Base with Object Manipulator, Learning Activity L – Gyro Sensor Default and Programming Task, or Learning Activity M – Clawbot IQ.
- This activity will take more time if campers have not already constructed the Clawbot IQ with or without sensors.

O.cl1 cont.

VEX IQ Materials:

- VEX IQ Super Kit
- Completed Clawbot IQ
- Computer for programming
- Modkit for VEX Robotics or ROBOTC for VEX Robotics programming software
- VEX IQ Highrise Cubes
- Clawbot IQ with Sensors 3-Part Hybrid Task Description
- Programming Planning Template

Optional Materials:

- Sample Program Breakdown for Modkit Clawbot IQ with Sensors Task – Hybrid
- Sample Program Breakdown for ROBOTC Clawbot IQ with Sensors Task – Hybrid

Pre-learning Suggestions:

- Learning Activity M – Clawbot IQ
- A variety of manual control and autonomous programming tasks using several different sensors, such as Learning Activities J, K, or L.

Based on VEX IQ Curriculum:

- Build and Test Clawbot IQ – Option 2 (C.2)
- Smarter Machines Unit Challenges (K.5)

O.cl2

Detailed Directions

1. Have each pair of campers think carefully about the Clawbot IQ they have constructed.

   If campers have not yet built the robot, provide them with time to do so. Instructions for its construction can be found in the kit documentation, VEX IQ Build Instructions – Clawbot IQ.

   Questions to ask campers:

   - What can this robot do?
   - What are its limitations?
   - How can you control it?

2. Have campers create a list of tasks or functions that they have successfully completed with their robot throughout their camp experience.

   Questions to ask campers:

   - What tasks have you already completed using teleoperated mode (using the VEX IQ Controller) to control your robot?
   - What tasks have you already completed using autonomous mode (programming) to control your robot?
   - Are there certain functions that seem to be easier or more practical to complete with one mode or the other?
   - Encourage campers to make notes as they brainstorm and tell them this information will help them with the task they are about to receive.

3. Provide campers with the Clawbot IQ with Sensors 3-Part Hybrid Task Description and tell them that they have the option of using teleoperated control (remote), autonomous control (programmed), or a hybrid solution (both teleoperated and autonomous) to complete as many of the tasks as possible in the time provided.

   Tasks include:

   - "Fragile load" – Program the robot Arm joint to stop moving up or down at a given point to prevent the robot Arm from lowering a given object too far. (Suggested VEX IQ Sensors: Touch LED, Bumper Switch)
   - "Sorting safe from hazardous" – Program the robot to recognize a Highrise Cube's color (red, green or blue). Two colors will be designated as "safe" to carry, while one color will be designated as "hazardous." The robot will be programmed to identify the object as safe or hazardous when holding it and print a message about its status on the Robot Brain LCD screen when the object is being held. (Suggested VEX IQ Sensor: Color Sensor)
   - "Easy does it" – After driving to a designated area, program the robot to back straight into a "parking space" and stop 200 mm away from a wall or obstacle to prevent a collision. (Suggested VEX IQ Sensor: Distance Sensor)

4. Prompt campers to think carefully about each part of the hybrid task and decide if they want to investigate one task very thoroughly or if they have time to attempt several tasks. Provide them with some time to discuss and prioritize which tasks to tackle first, using what they have already learned throughout the week. This task can take place on a Highrise Challenge Field or on any flat surface.

   Questions to ask campers:

   - Which of these functions have you done before?
   - Did you complete the tasks in teleoperated mode (using the Controller) or in autonomous mode (using programming)?
   - Did that function work well or do you feel there is a better way to accomplish the task?
   - Have you seen other campers tackle the same problem in another way that seemed effective?
   - How could one or two of these smaller tasks work together to accomplish a more complex task?

5. Provide campers with time to create and test their programs. Encourage them to use the handout, Program Planning Template to plan and document each program.
6. If time allows, encourage campers to make modifications to the physical build of the robot or to create a back story for the hybrid task itself and make it part of the presentation of their robot and program to the group.

Tips and Best Practices:

- If Learning Activity N – Clawbot with Sensors – Autonomous Task has been completed, it may be a good idea to remove the sensors from the robot before beginning so that campers may be more open to trying new sensors and solutions to the challenges presented in this activity.
- Encourage campers who are creating a Hybrid solution to test their programs with the Controller to ensure that programmed functions don't interfere with controlled functions.
- Remind campers that in engineering, you may not be able to do it all. Certain situations call for a choice between quality and quantity of functions. They should think carefully about the time they have to complete a design and decide which functions are worth the time. An idea that doesn't work is not a failure. It may be helpful in a different situation.

Extension:

- Create larger groups of four campers and encourage them to combine their components and programming ideas to create one hybrid robot that completes all three tasks efficiently and effectively.
- Add a function to the program that will stack a Highrise Cube (in preparation for the Highrise Challenges).

O.cs1

Clawbot IQ with Sensors 3-Part Hybrid Task Description

Student Name(s): ____

Instructions:

Think carefully about the robots you have built and how you have made them complete the tasks you wanted them to complete.

Begin with the Clawbot IQ robot or a robot of your own creation. Your job is to read the tasks below and decide if you will complete one, two, or all of them. You can add sensors or assemblies to your robot to accomplish the tasks effectively.

You can also decide if you would like to create solutions that use teleoperated or autonomous control, or a combination of the two modes to complete the tasks.

Remember, in engineering, you may not be able to do it all. Certain situations call for a choice between quality and quantity of functions.

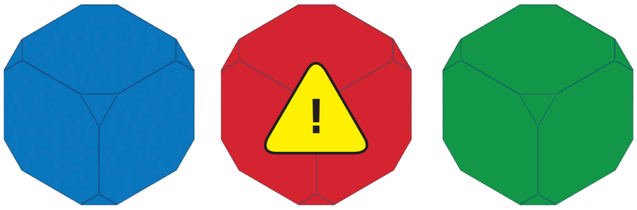
Think carefully about the time you have to complete a design and decide which functions are worth the time. An idea that doesn't work is not a failure. It may be helpful in a different situation

TASK #1: Fragile Load

Program the robot Arm joint to stop moving up or down at a given point to prevent the robot Arm from lowering a given object too far.

TASK #2: Sorting Safe from Hazard

Program the robot to recognize a Highrise Cube's color and identify the object as safe or hazardous. Your robot should print the Cube's designation on the VEX IQ Robot Brain's LCD screen when the object has been detected.



Blue and green Cubes are safe to carry.

Red Cubes are hazardous so leave them where they lay.

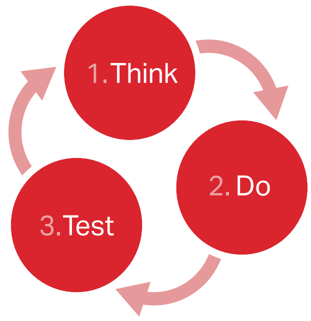
TASK #3: Easy Does It

After driving to a designated area, program the robot to back straight into a "parking space" created from VEX IQ components, stopping 200 mm away from a wall or obstacle to prevent a collision.

0.cs2

Program Planning & Troubleshooting

Student Name(s): ____



Instructions:

Before completing this activity, be sure you are familiar with your programming software and how it works. Follow the steps on this page to document, test, troubleshoot, and refine your program. If your program includes more steps than fit on one page, use multiple copies as necessary.

Describe what you want your program to be able to do here:	
Describe the device/robot you will be programming:	List the sensors that will be used in your program:

"THINK" Write step-by-step program instructions here.	"DO" Write your program using programming software and make notes here as you work.	"TEST" Does this step of your program function as expected? What needs improvement (NI)?
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:
		Yes No NI:

If ANY program steps do NOT function as expected, address your "NI" items using as many copies of this page as necessary until all parts of your program function as expected.

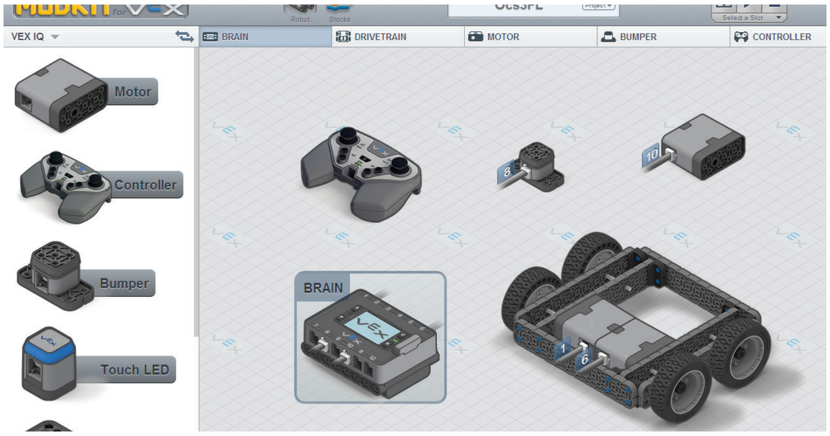
Remember: Problems ARE NOT failures, they are an expected part of the design process!

0.cs3

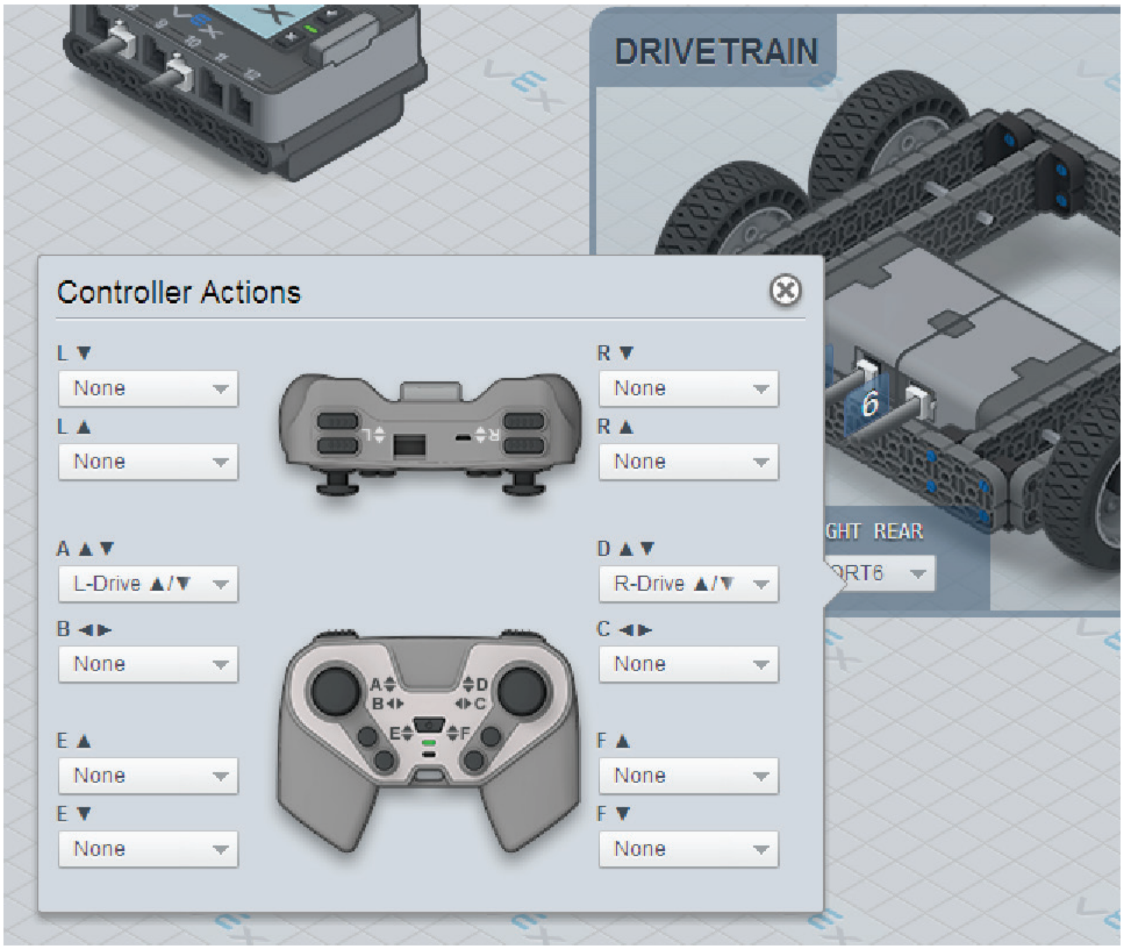
Sample Program Breakdown for Modkit Clawbot IQ with Sensors Task - Hybrid

Summary of Program for Task 1 - "Fragile Load":

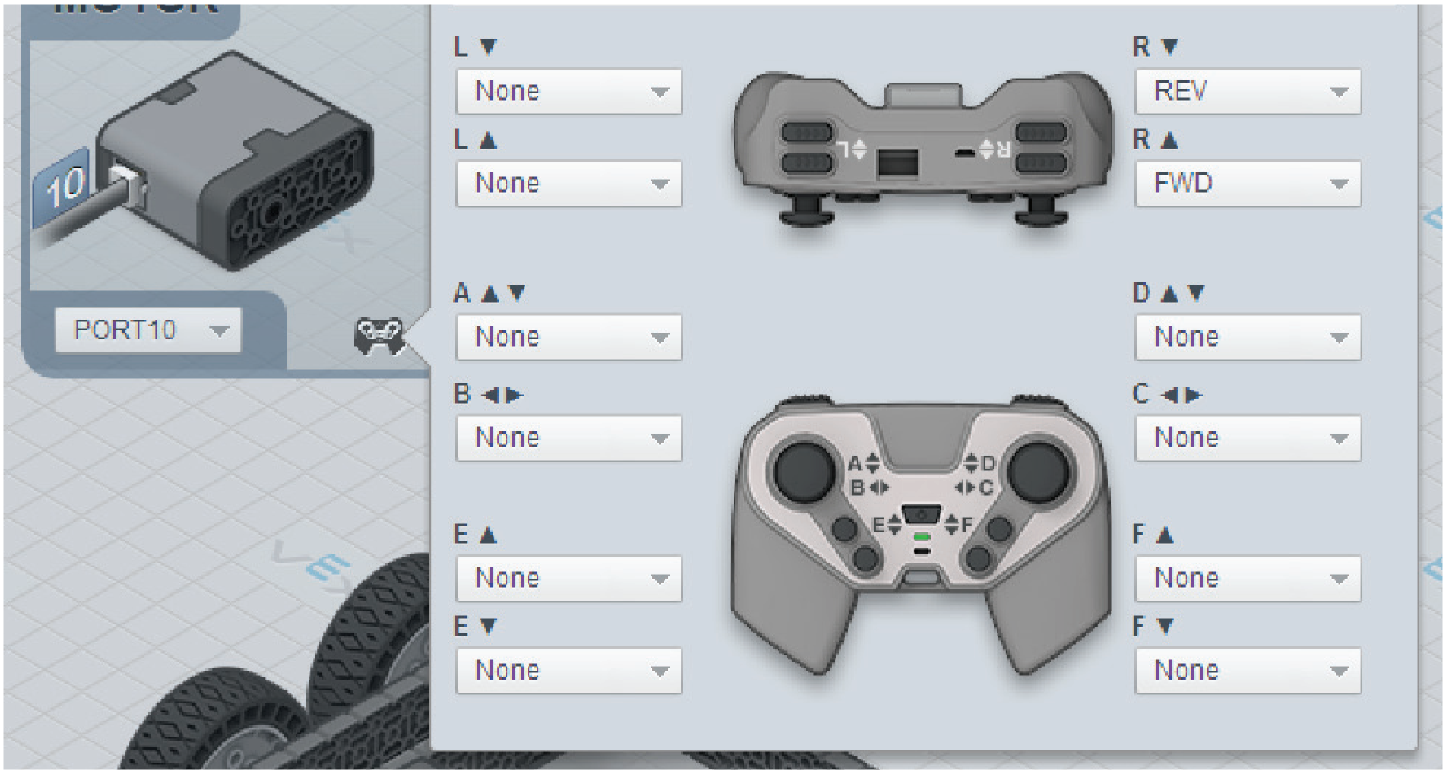
This program uses the VEX IQ Clawbot with Sensors and combines the VEX IQ Touch, LED, Bumper Switch, Smart Motors, Robot Brain and Controller to make the robot arm joint stop moving up or down at a given point and prevent the robot arm from lowering a given object too far.



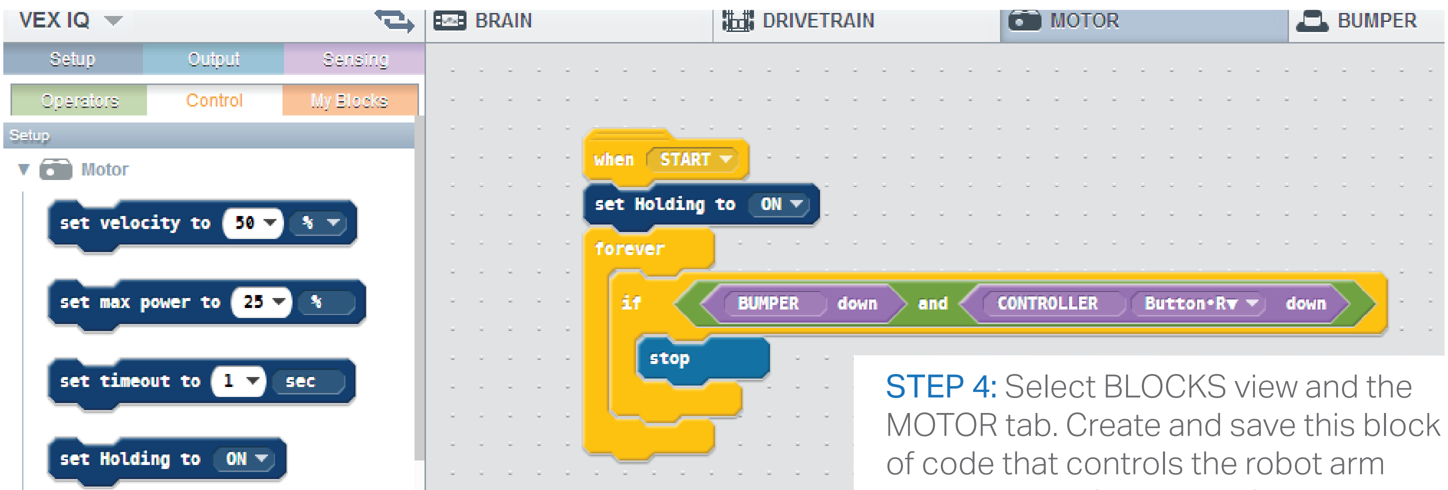
STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.



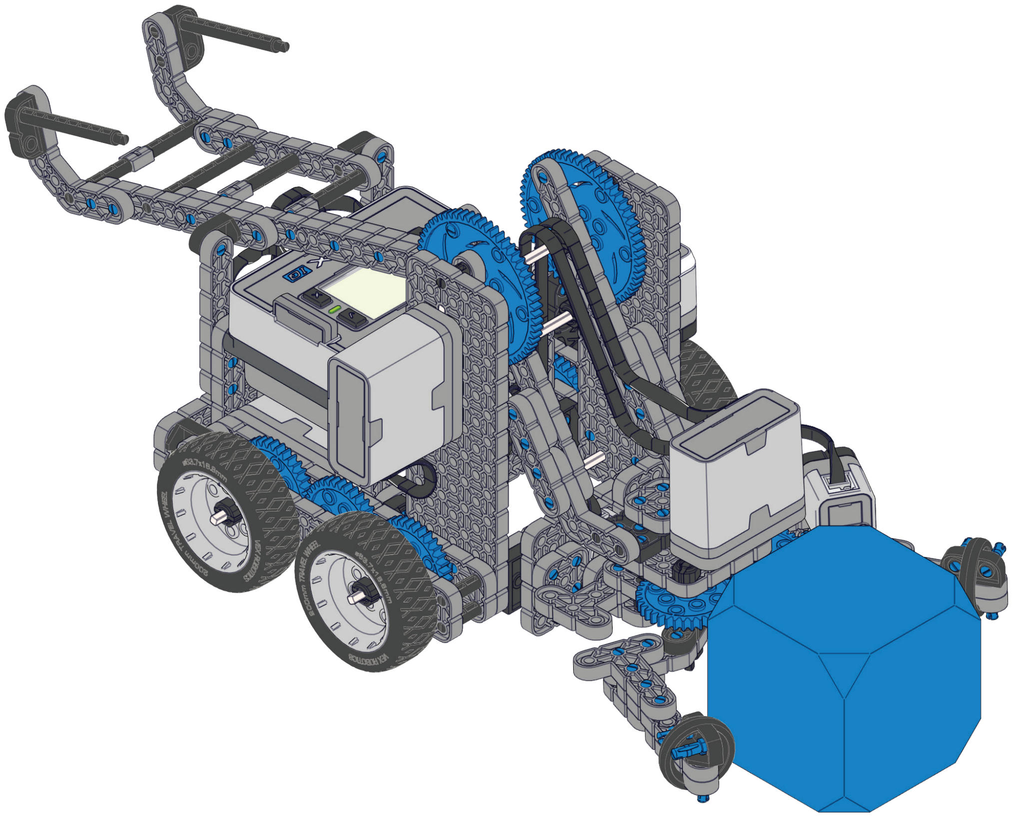
STEP 2: Still in Robot View, configure your controller to control the drivetrain. Select the picture of a controller, then choose "L-Drive ^/v" for Channel A and "R-Drive ^/v" for Channel D. This allows the joysticks to control the robot via tank drive.



STEP 3: Still in Robot View, configure your controller to control the arm motor. Select the picture of a controller, then choose "REV" for Channel Rv and "FWD" for Channel R^. This allows the Channel R buttons to control the arm motion.



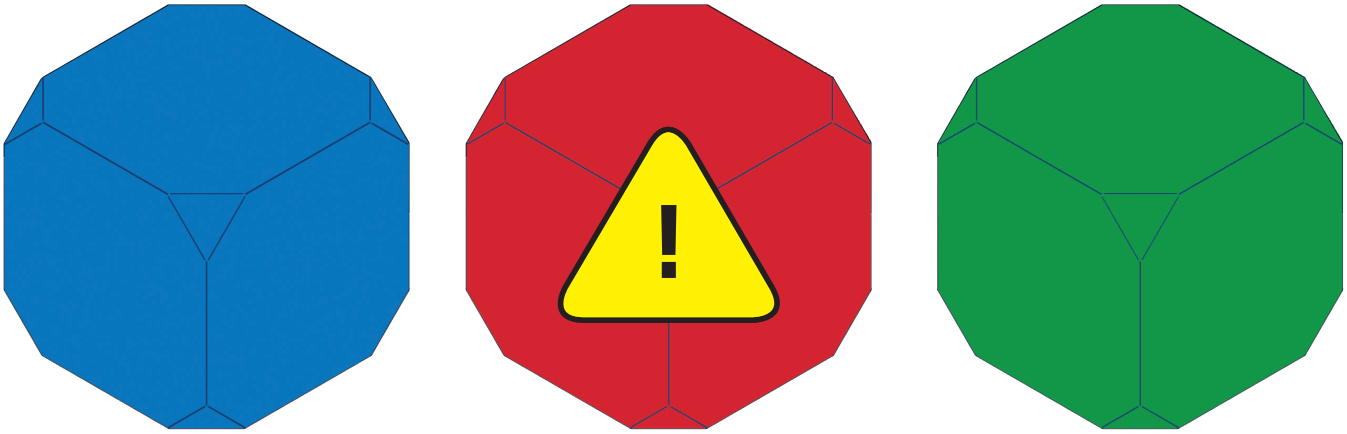
STEP 4: Select BLOCKS view and the MOTOR tab. Create and save this block of code that controls the robot arm movement and prevents the motor from moving the arm down when the Bumper Switch is pressed. Then download the program to a chosen slot on the VEX IQ Robot Brain.

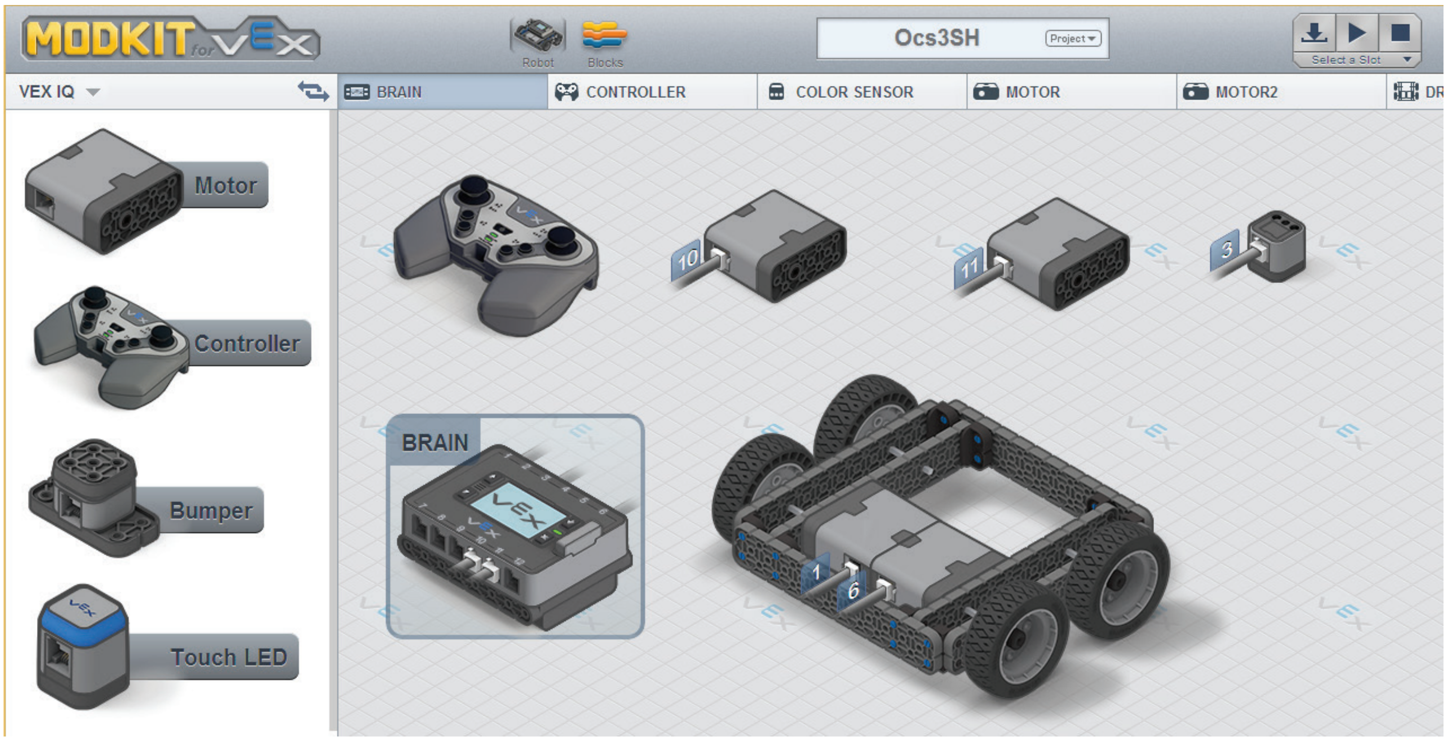


STEP 5: Place the robot down and run this program. Move the arm up and down. Allow the arm to press the Bumper Switch which stops the motor from turning the arm downward.

Summary of Program for Task 2 - "Sorting Safe from Hazardous":

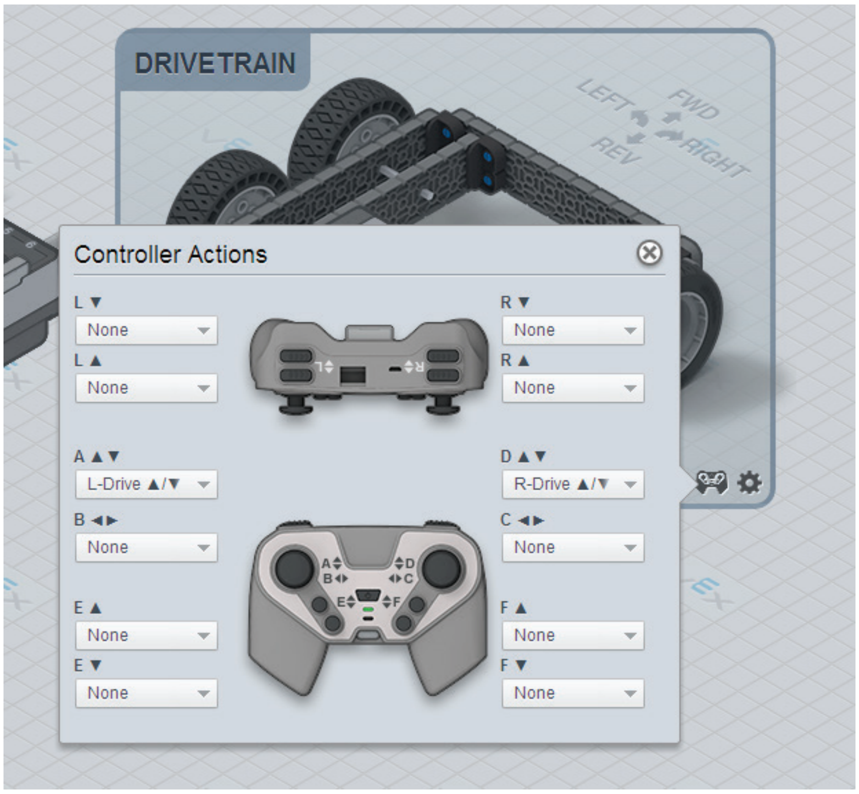
This program combines the VEX IQ Color Sensor with Smart Motors, Robot Brain and Controller to allow the robot to recognize Cube color (red, green, blue) and designate it as safe or hazardous to carry, display its designation on the Robot Brain LCD screen. In this scenario, the robot is driven around with its Claw open with one of each color Cube in the area. As the Claw approaches each Cube, the Robot Brain LCD screen reports if a Cube is "Safe" or "Dangerous" to pick up.



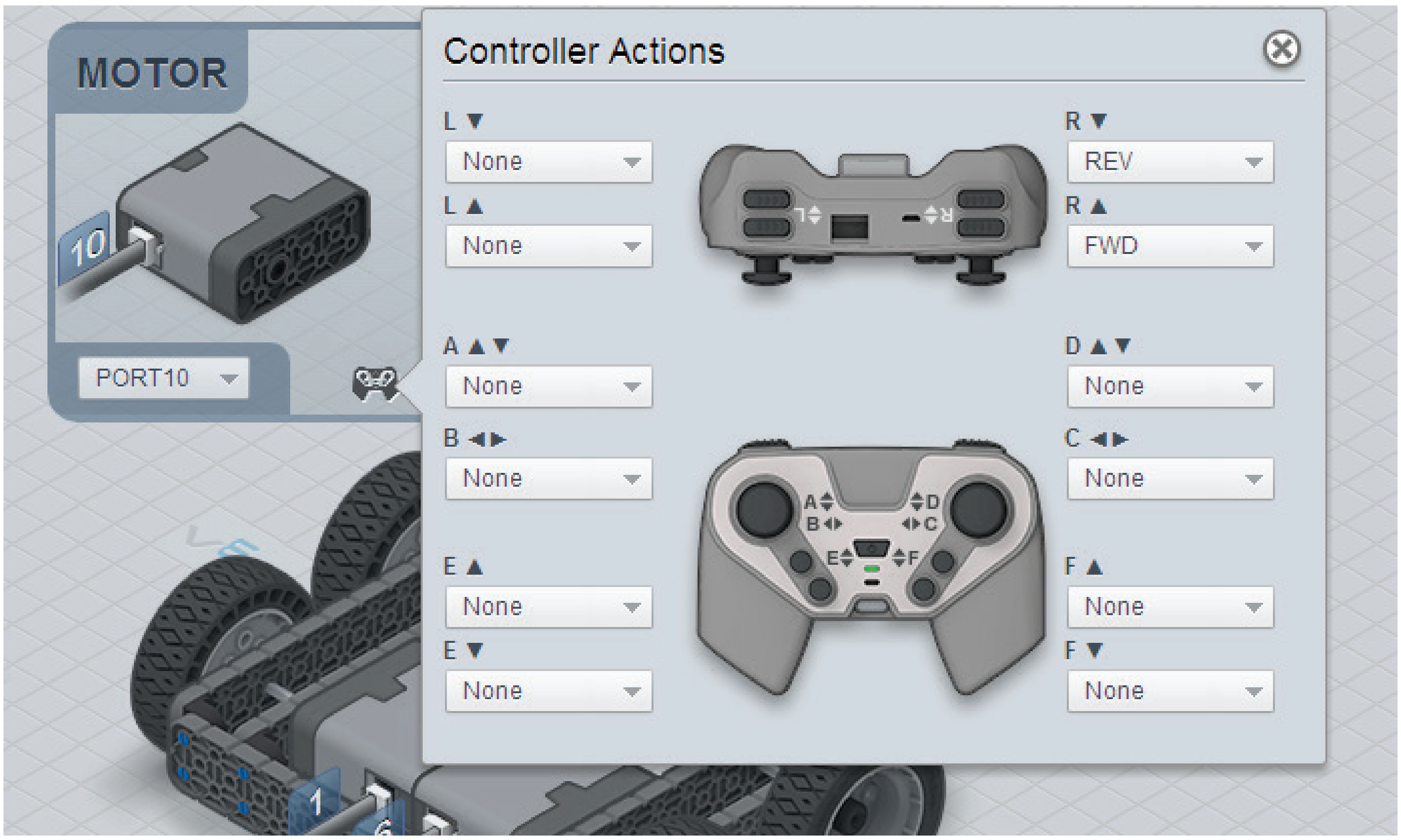


STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.

O.cs3 cont.

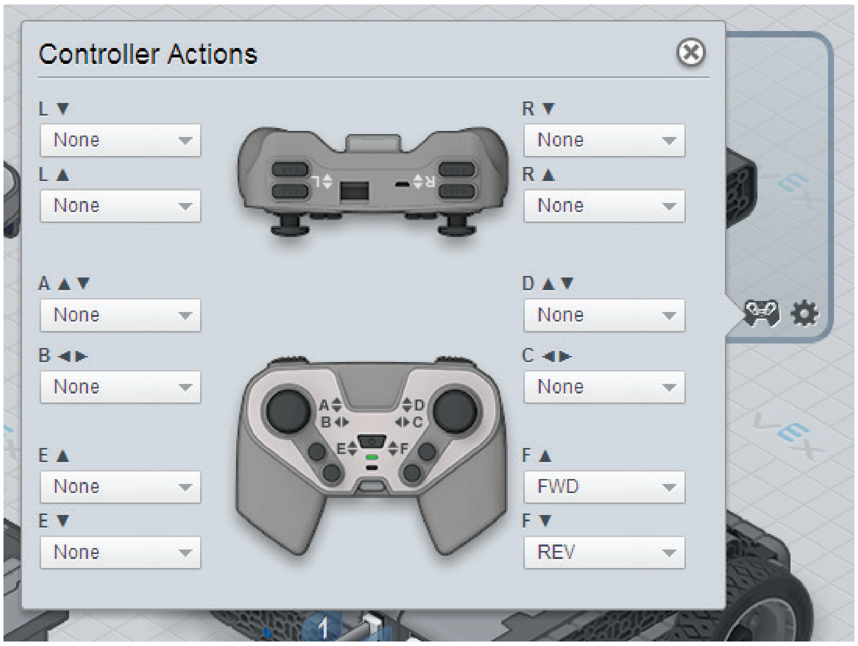


STEP 2: Still in Robot View, configure your controller to control the drivetrain. Select the picture of a controller, then choose "L-Drive ^/v" for Channel A and "R-Drive ^/v" for Channel D. This allows the joysticks to control the robot via tank drive.

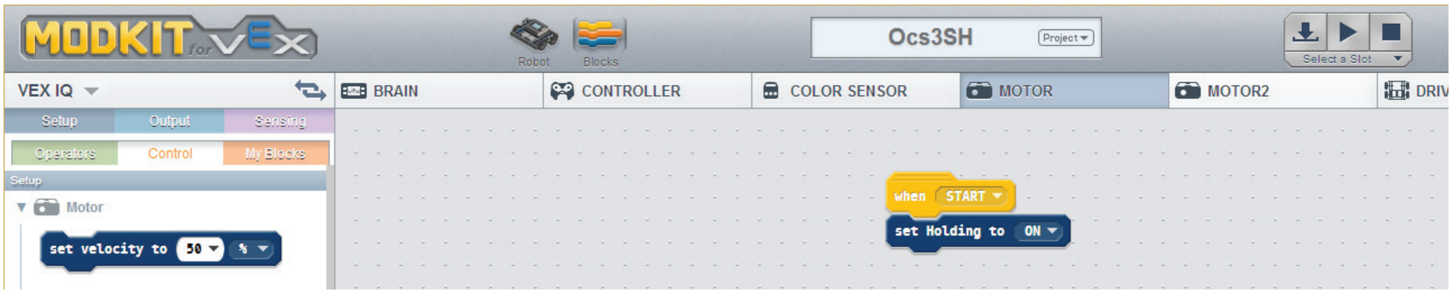


STEP 3: Still in Robot View, configure your controller to control the arm motor. Select the picture of a controller, then choose "REV" for Channel Rv and "FWD" for Channel R^. This allows the Channel R Buttons to control the arm motion.

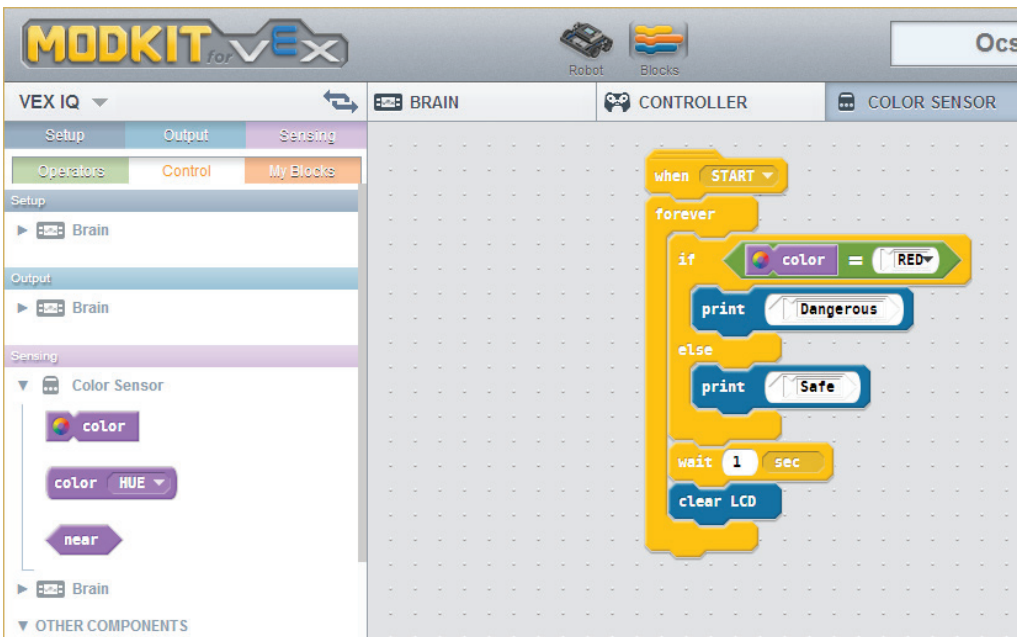
O.cs3 cont.



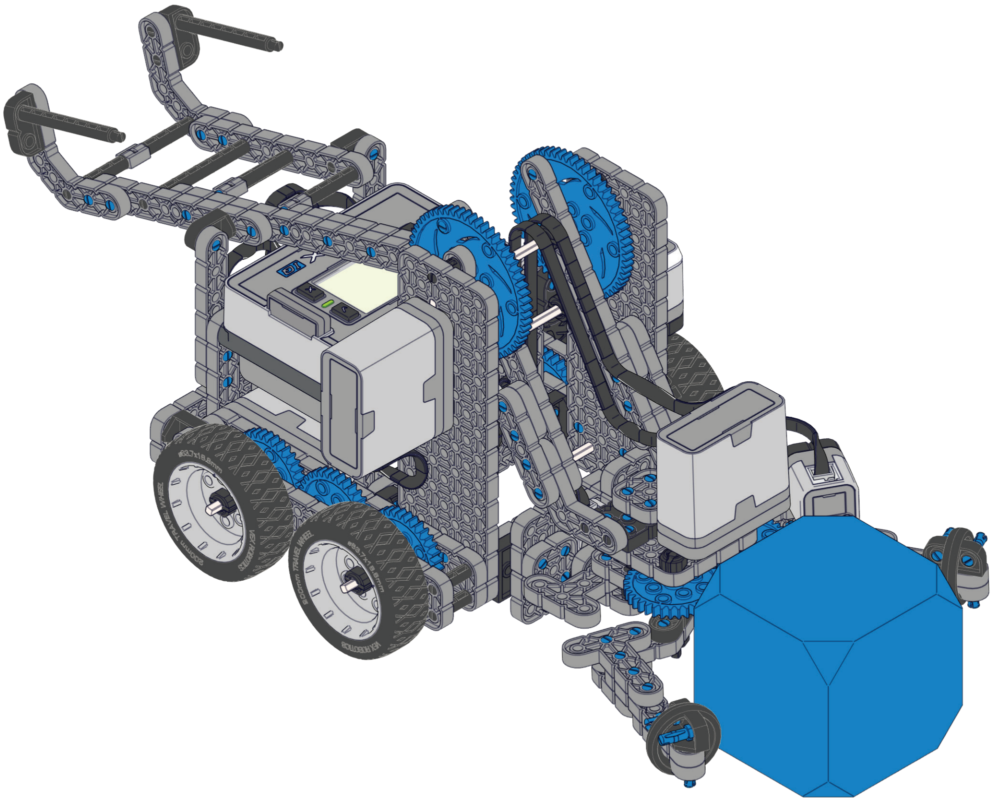
STEP 4: Still in Robot View, configure your controller to control the claw motor. Select the picture of a controller, then choose "FWD" for Channel F^ and "REV" for Channel Fv. This allows the Channel F buttons to control the claw motion.



STEP 5: Select BLOCKS view and the MOTOR tab. Create and save block of code which controls arm movement and keeps it in position at rest.



STEP 6: Select BLOCKS view and the COLOR SENSOR tab. Create and save this block of code which allows the Color Sensor to detect Red Cubes from all other objects and report "Safe" or "Dangerous" on the Robot Brain's LCD screen. Then download the program to a chosen slot on the VEX IQ Robot Brain.

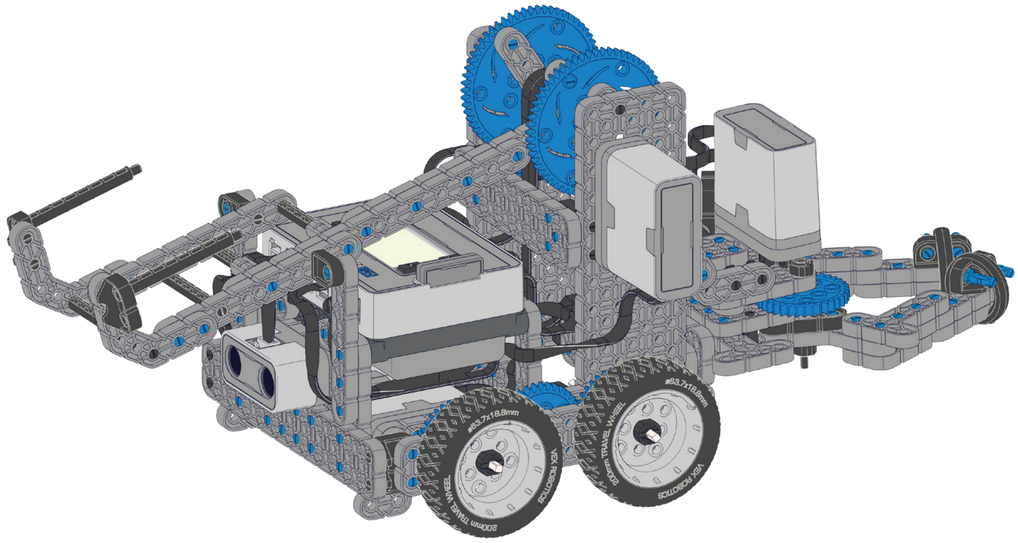


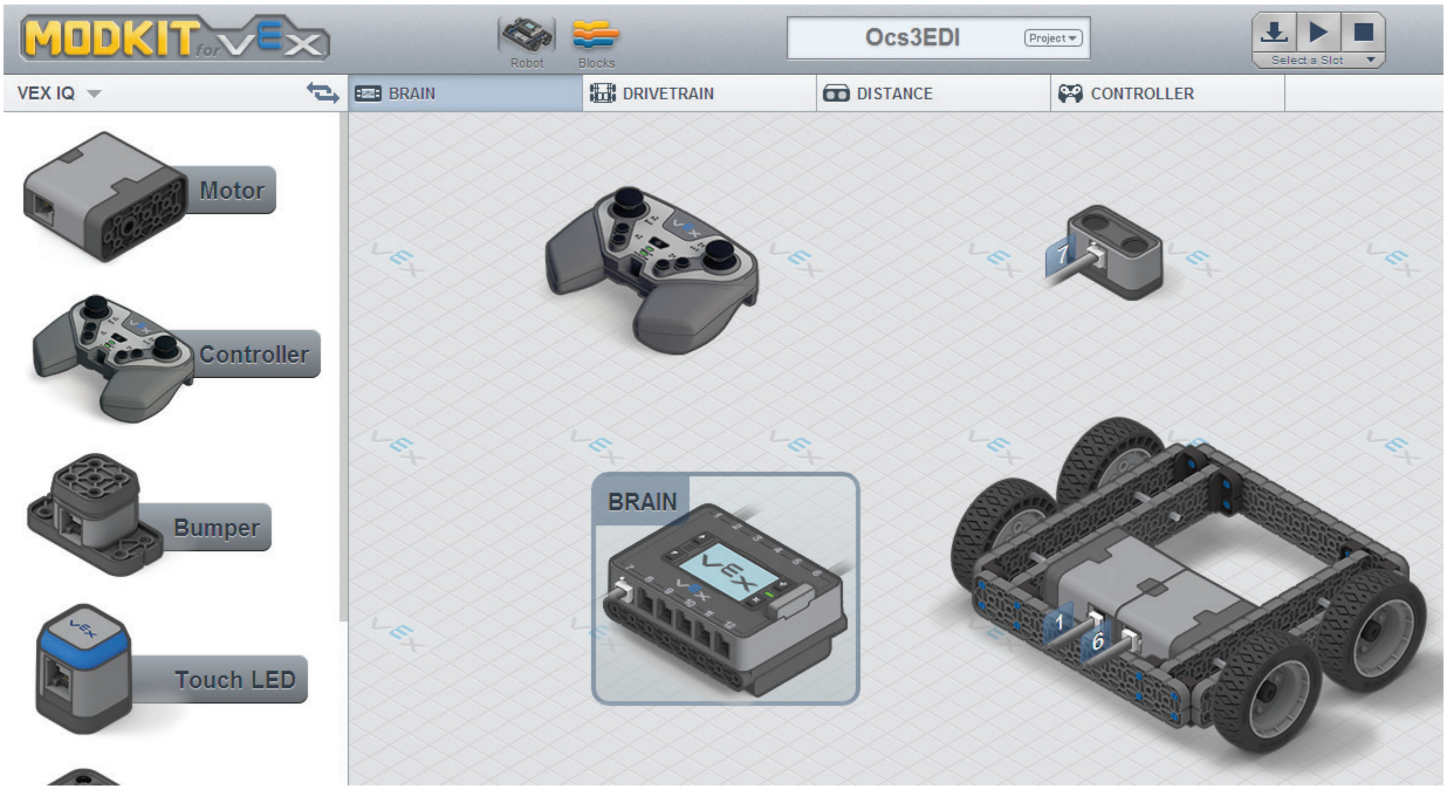
STEP 7: Place the robot down and run this program. Drive the robot around with the claw open to pick up different colored Cubes and observe the LCD screen report what Cubes are "safe" to pick up and what Cubes are "dangerous" to pick up.

O.cs3 cont.

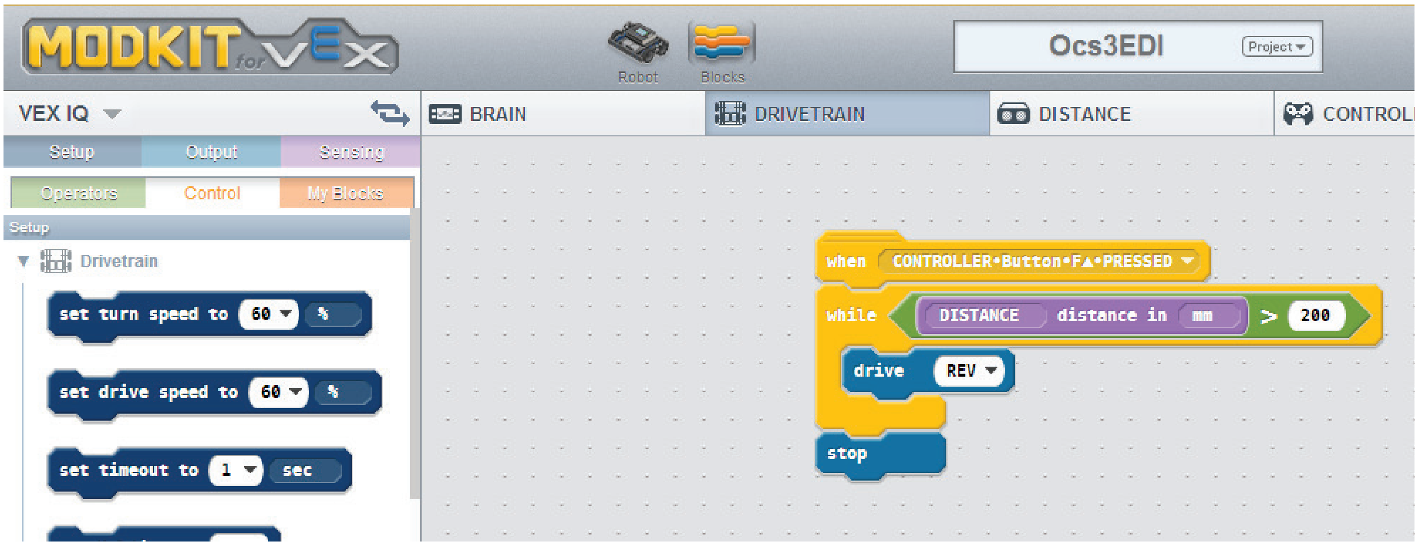
Summary of Program for Task 3 - "Easy Does It":

This program combines the VEX IQ Distance Sensor, Smart Motors, Robot Brain and Controller to make the robot back straight into a "parking space", stopping 200 mm away from a wall or obstacle to prevent a collision. In this scenario you will use the Channel F "up" button to back up the robot toward a wall or obstacle and the Distance Sensor will stop the robot 200 mm from the obstacle.

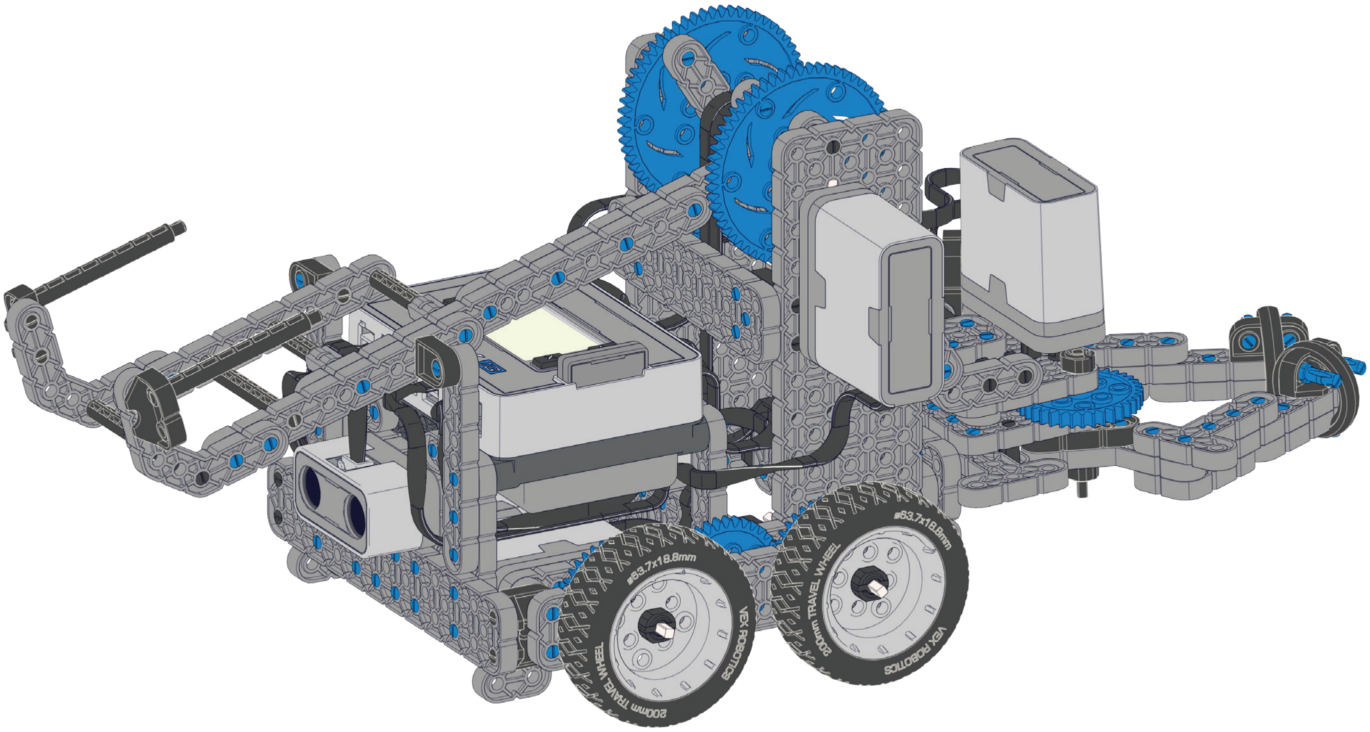




STEP 1: Configure Modkit's Robot View to match your physical robot and save your project.



STEP 2: Select BLOCKS view and the DRIVETRAIN tab. Create and save this block of code that controls the robot drivetrain movement and prevents the robot from backing into a wall or obstacle. Then download the program to a chosen slot on the VEX IQ Robot Brain.



STEP 3: Place the robot down and run this program. Press the F^ button to back the robot up. It will stop 200mm from any obstacle behind it.

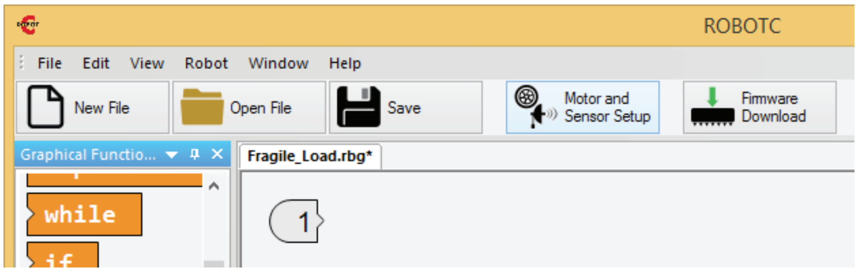
0.cs4

Sample Program Breakdown for ROBOTC Clawbot IQ with Sensors Task - Hybrid

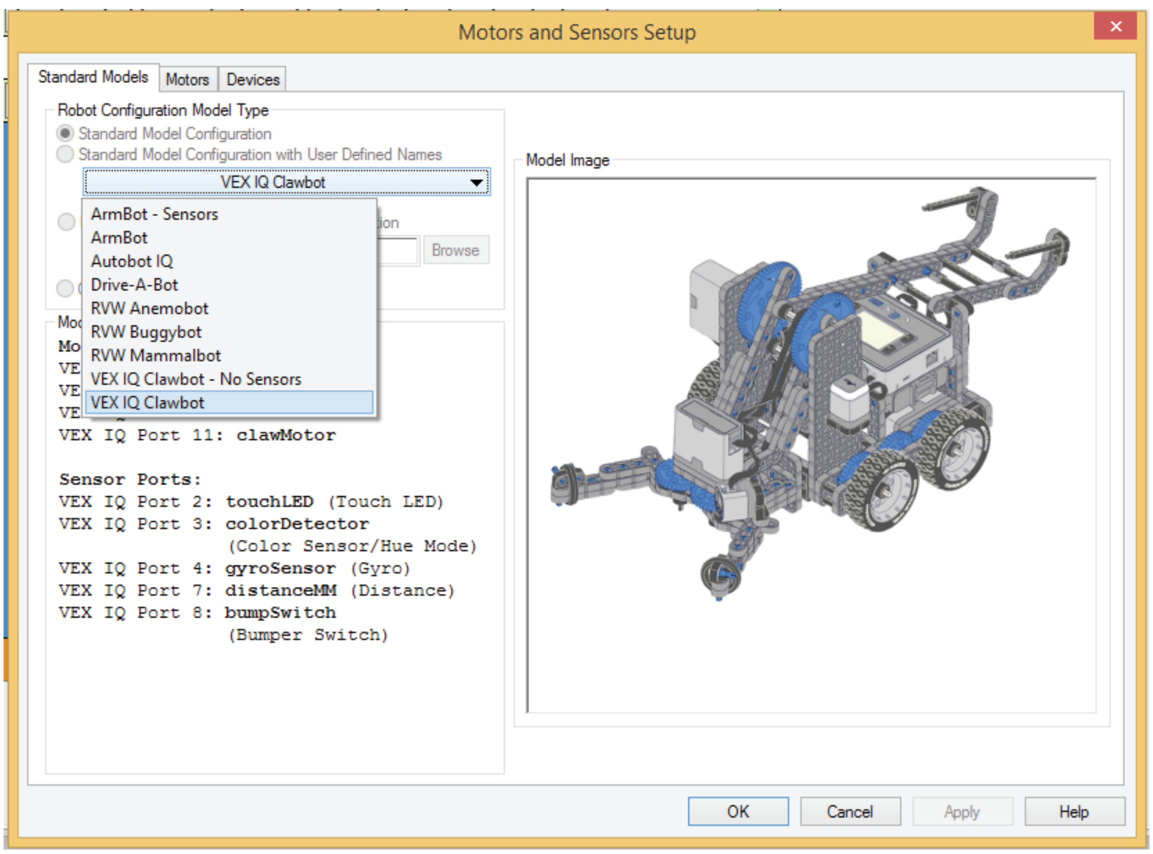
Summary of Program for Task 1 - "Fragile Load":

This program combines the VEX IQ Touch LED and the VEX IQ Bumper Switch with Smart Motors and the VEX IQ Robot Brain and Controller to make the robot Arm joint stop moving up or down at a given point and prevent the robot Arm from lowering a given object too far.

Note: Alternatively, campers can complete their programming in ROBOTC for VEX Robotics – Graphical programming software. This can then be exported to text-based programming if needed.



STEP 1: Select the Motor and Sensor Setup button



STEP 2: Select VEX IQ Clawbot from the Standard Robot Model Configuration

Fragile_Load.rbg*

```
//  Keep looping forever
2   repeat (forever ) {
3
4   }
5
```

STEP 3: To configure the Controller actions for the Arm and the Bumper Switch, and control the wheels on the drivetrain, start by creating an infinite loop (a program that runs forever).

Fragile_Load.rbg*

```
//  Keep looping forever
2   repeat (forever ) {
//      Control the robot with Tank Control style driving
4       tankControl ( ChD , ChA , 10 );
5   }
6
```

STEP 4: To control the wheels on the Drivetrain, add a tankControl command inside of the loop.

Fragile_Load.rbg

```
//  Keep looping forever
2   repeat (forever ) {
//      Control the robot with Tank Control style driving
4       tankControl ( ChD , ChA , 10 );
//      If the Bumper Switch is not pressed (value of 0)
6       if ( getBumperValue(bumpSwitch) == 0 ) {
//          Control the Arm Motor with buttons RUp and RDown (power level 75)
8           armControl ( armMotor , BtnRUp , BtnRDown , 75 );
9       } else {
```

STEP 5: To check if the Bump button is released or not, add an if/else block below the tankControl command. If the Bump Button is released (value of 0), the robot's arm will be controlled with an armControl command.

0.cs4 cont.

Fragile_Load.rbg

```
// Keep looping forever
repeat (forever ) {
    // Control the robot with Tank Control style driving
    tankControl ( ChD , ChA , 10 );
    // If the Bumper Switch is not pressed (value of 0)
    if ( getBumperValue(bumpSwitch) == 0 ) {
        // Control the Arm Motor with buttons RUp and RDown (power level 75)
        armControl ( armMotor , BtnRUp , BtnRDown , 75 );
    } else {
        // Else, the Bumper Switch is pressed
        // If the Joystick's RUp button is pressed
        if ( getJoystickValue(BtnRUp) == 1 ) {

        } else {

        }
    }
}
```

STEP 6: Add another if/else statement that checks to see if the Joystick Controller's RUp button is pressed or not. This new if/else block should be inside of the first if/else block's "else" statement.

Fragile_Load.rbg*

```
// Keep looping forever
repeat (forever ) {
    // Control the robot with Tank Control style driving
    tankControl ( ChD , ChA , 10 );
    // If the Bumper Switch is not pressed (value of 0)
    if ( getBumperValue(bumpSwitch) == 0 ) {
        // Control the Arm Motor with buttons RUp and RDown (power level 75)
        armControl ( armMotor , BtnRUp , BtnRDown , 75 );
    } else {
        // Else, the Bumper Switch is pressed
        // If the Joystick's RUp button is pressed
        if ( getJoystickValue(BtnRUp) == 1 ) {
            setMotor ( armMotor , 75 );
        } else {

        }
    }
}
```

STEP 7: If the RUp button is pressed, raise the arm with a positive motor power level. To do this, set the motor's power level inside of the if statement.

0.cs4 cont.

Fragile_Load.rbg*

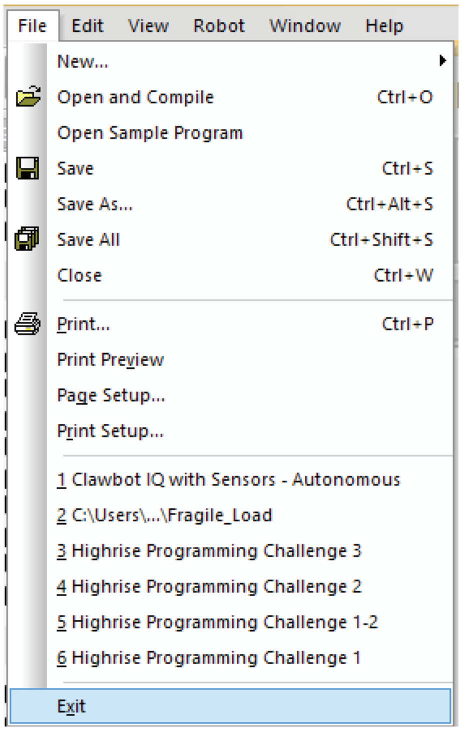
```
// Keep looping forever
repeat (forever ) {
    // Control the robot with Tank Control style driving
    tankControl ( ChD , ChA , 10 );
    // If the Bumper Switch is not pressed (value of 0)
    if ( getBumperValue(bumpSwitch) == 0 ) {
        // Control the Arm Motor with buttons RUp and RDown (power level 75)
        armControl ( armMotor , BtnRUp , BtnRDown , 75 );
    } else {
        // Else, the Bumper Switch is pressed
        // If the Joystick's RUp button is pressed
        if ( getJoystickValue(BtnRUp) == 1 ) {
            setMotor ( armMotor , 75 );
        } else {
            // Else, stop the arm motor
            stopMotor ( armMotor );
        }
    }
}
```

STEP 8: If the RUp button is not pressed, the motor should instead be stopped. To do this, stop the motor inside of the else statement.

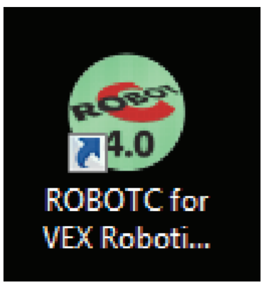
0.cs4 cont.

Summary of Program for Task 2 - "Sorting Safe from Hazardous":

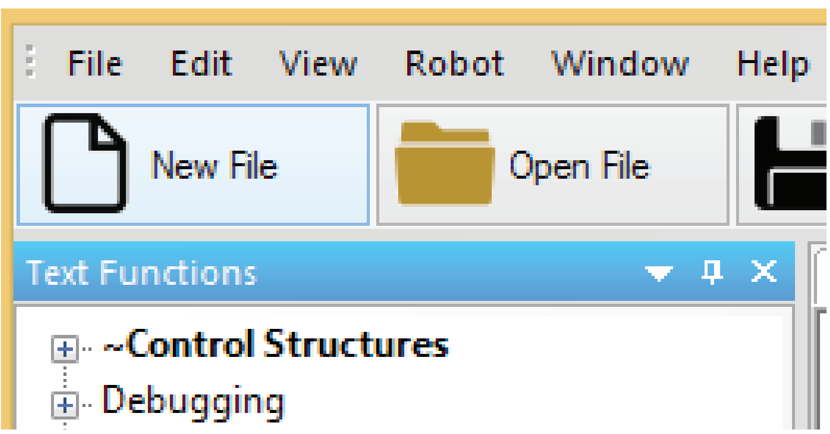
This program combines VEX IQ Color Sensor with VEX IQ Smart Motors and the VEX IQ Robot Brain and Controller to allow the robot to recognize a Highrise Cube's color (red, green or blue) and designate it as safe or hazardous to carry and display its designation on the Robot Brain LCD screen.



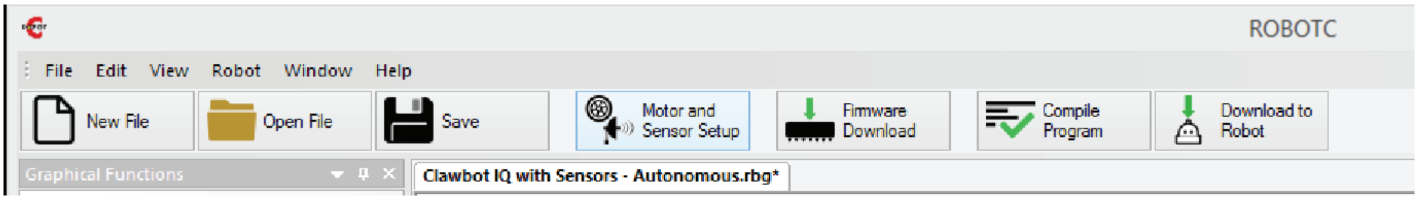
STEP 1: This advanced Color Sensor task is best programmed with RobotC's Natural Language. To switch to Natural Language from Graphical, first close the ROBOTC program. This can be done through the 'File -> Exit' menu option, or by simply clicking the red X in the top-right corner of ROBOTC.



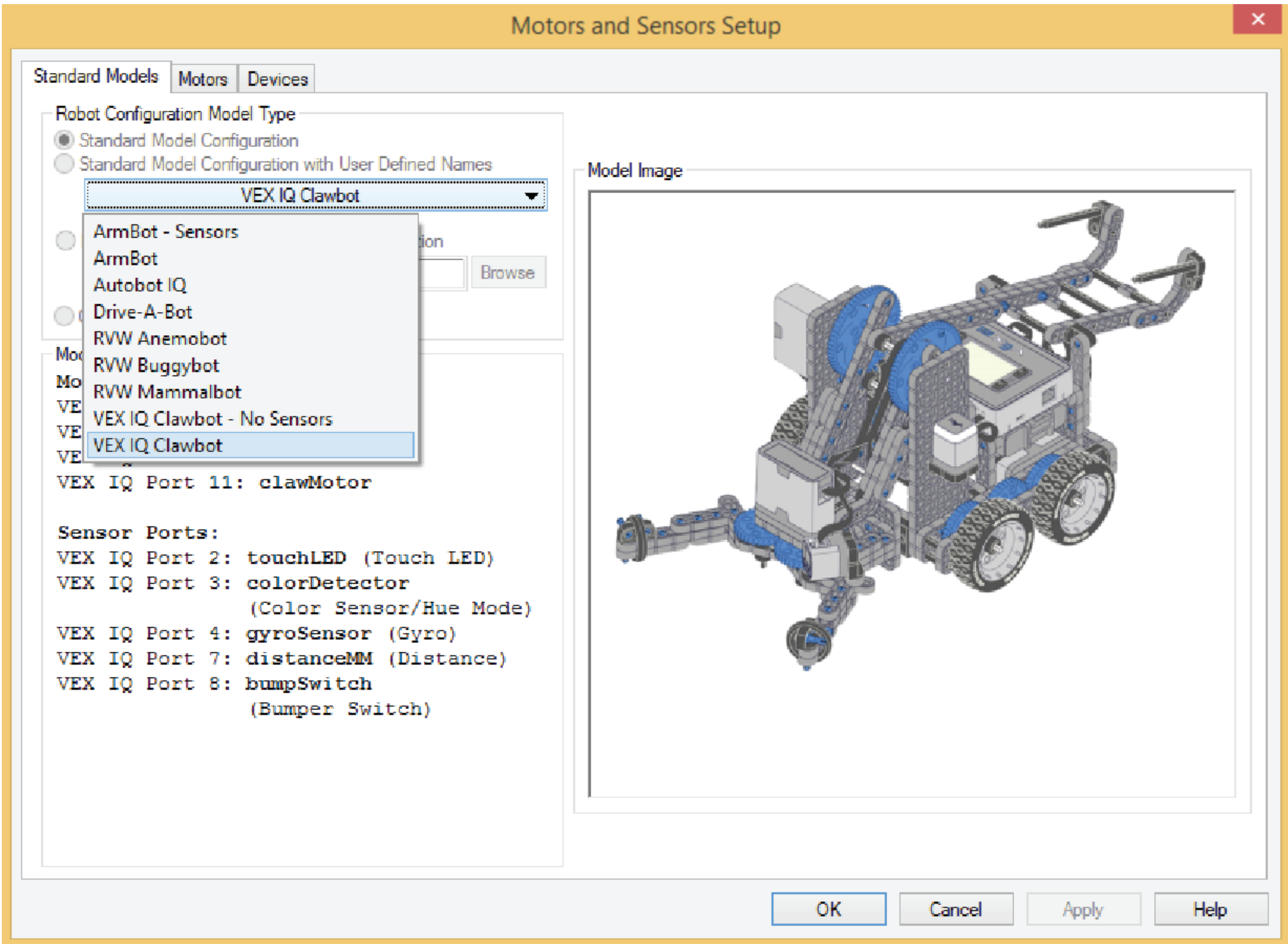
STEP 2: Once ROBOTC is closed, it will need to be reopened in 'text' mode. This can be done by clicking on the circular ROBOTC icon on the desktop.



STEP 3: Just like in the Graphical program, a new program can be opened by clicking the 'New File' button at the top of the screen. This will open a blank ROBOTC program file in the text format.



STEP 1: Select the Motor and Sensor Setup button.



STEP 2: Select VEX Clawbot IQ from the Standard Robot Model Configuration

O.cs4 cont.

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {

  }
}
```

STEP 3: To configure the Controller actions for the Color Sensor and control the wheels on the drivetrain, start by creating an infinite loop (a program that runs forever).

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);


  }
}
```

STEP 4: To program the motors that control the wheels on the drivetrain, the robot Arm, and the Claw, set up basic controls in the loop.

O.cs4 cont.

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);

    //If the Color Sensors sees Blue or Green
    if(getColorName(colorDetector) == colorBlue || getColorName(colorDetector) == colorGreen)
    {

    }
  }
}
```

STEP 5: Add a condition that will check the color of the Highrise Cube to see if it is green or blue.

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);

    //If the Color Sensors sees Blue or Green
    if(getColorName(colorDetector) == colorBlue || getColorName(colorDetector) == colorGreen)
    {
      //Display "Safe to Grab" on line 1 of the LCD screen
      displayTextLine(1, "Safe to Grab");
    }
  }
}
```

STEP 6: To tell the robot which message to display on its LCD screen, add a command to display text and type the text you would like the robot to display. In this case, if the sensor sees green or blue, it will display “safe to grab.”

O.cs4 cont.

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);

    //If the Color Sensors sees Blue or Green
    if(getColorName(colorDetector) == colorBlue || getColorName(colorDetector) == colorGreen)
    {
      //Display "Safe to Grab" on line 1 of the LCD screen
      displayTextLine(1, "Safe to Grab");
    }

    //Else (the Color Sensor sees a color that is NOT Blue or Green)
    else
    {

    }
  }
}
```

STEP 7: Add a second condition to check for colors that are not green or blue.

0.cs4 cont.

Sorting_Safe_From_Hazardous.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);

    //If the Color Sensors sees Blue or Green
    if(getColorName(colorDetector) == colorBlue || getColorName(colorDetector) == colorGreen)
    {
      //Display "Safe to Grab" on line 1 of the LCD screen
      displayTextLine(1, "Safe to Grab");
    }

    //Else (the Color Sensor sees a color that is NOT Blue or Green)
    else
    {
      //Display "Dangerous" on line 1 of the LCD screen
      displayTextLine(1, "Dangerous");
    }
  }
}
```

STEP 8: To tell the robot which message to display on its LCD screen, add a command to display text and type the text you would like the robot to display. In this case, if the sensor sees a color that is not green or blue, it will display "dangerous."

O.cs4 cont.

Sorting_Safe_From_Hazardous.c

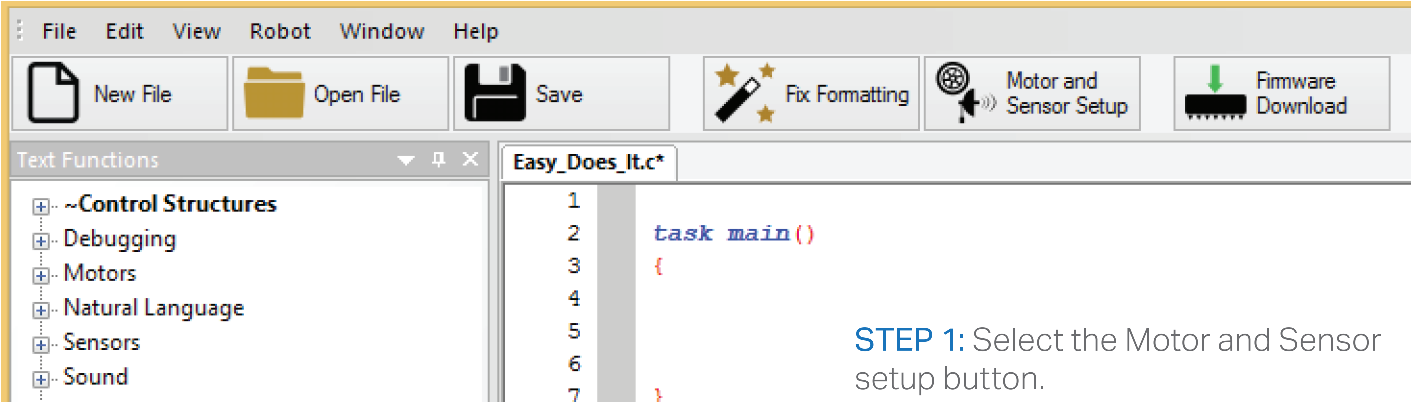
```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
    //Control the arm motor with buttons RUp and RDown
    armControl(armMotor, BtnRUp, BtnRDown, 50);
    //Control the claw motor with buttons EUp and EDown
    armControl(clawMotor, BtnEUp, BtnEDown, 50);

    //If the Color Sensors sees Blue or Green
    if(getColorName(colorDetector) == colorBlue || getColorName(colorDetector) == colorGreen)
    {
      //Display "Safe to Grab" on line 1 of the LCD screen
      displayTextLine(1, "Safe to Grab");
    }

    //Else (the Color Sensor sees a color that is NOT Blue or Green)
    else
    {
      //Display "Dangerous" on line 1 of the LCD screen
      displayTextLine(1, "Dangerous");
    }
    //Display the text for 20 milliseconds and clear the screen
    sleep(20);
    eraseDisplay();
  }
}
```

STEP 9: To ensure that the robot's LCD display is showing the correct message, a small sleep command (20 milliseconds) and a command to erase the display should be added as the last commands in the loop

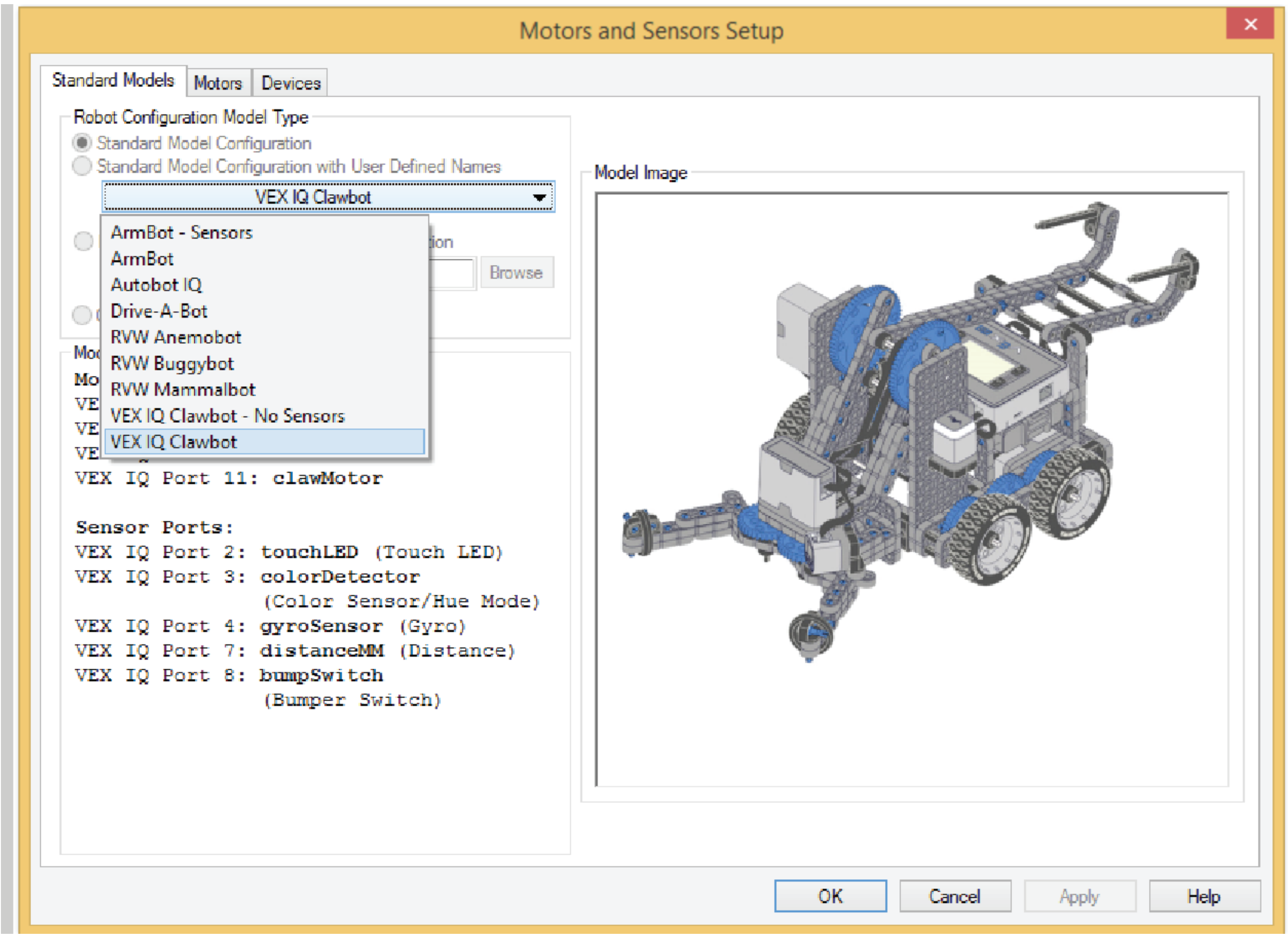
O.cs4 cont.

Summary of Program for Task 3 - "Easy Does It":

This program combines the VEX IQ Distance Sensor with Smart Motors and the VEX IQ Robot Brain and Controller to make the robot back straight into a "parking space", stopping 200 mm away from a wall or obstacle to prevent a collision.



STEP 1: Select the Motor and Sensor setup button.



STEP 2: Select VEX IQ Clawbot from the Standard Robot Model Configuration.

O.cs4 cont.

Easy_Does_It.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {

  }
}
```

STEP 3: To configure the Controller actions for the Distance Sensor and control the wheels on the drivetrain, start by creating an infinite loop (a program that runs forever).

Easy_Does_It.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);
  }
}
```

STEP 4: To control the wheels on the drivetrain, add values for the leftMotor and rightMotor.

0.cs4 cont.

Easy_Does_It.c*

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);

    //While button FUp is pressed AND the Distance Sensor is greater than or equal to 200mm
    while(getJoystickValue(BtnFUp) == 1 && getDistanceValue(distanceMM) >= 200)
    {

    }
  }
}
```

STEP 5: Add an if statement that checks if both a button on the Controller is pressed AND the Distance sensor is returning a large distance. In this case, BtnFUp has been selected for the Controller button and a value of 200mm has been selected for the distance

Easy_Does_It.c

```
#pragma config(StandardModel, "VEX IQ Clawbot")
//*!!Code automatically generated by 'ROBOTC' configuration wizard               !!*//

task main()
{
  //Keep looping forever
  repeat(forever)
  {
    //Control the robot with Tank Control style driving
    tankControl(ChD, ChA, 10);

    //While button FUp is pressed AND the Distance Sensor is greater than or equal to 200mm
    while(getJoystickValue(BtnFUp) == 1 && getDistanceValue(distanceMM) >= 200)
    {
      //Move the robot backward
      setMotor(leftMotor, -50);
      setMotor(rightMotor, -50);
    }
  }
}
```

STEP 6: If both BtnFUp is pressed and the Distance sensor is returning a value greater than 200mm, turn both motors on at a power level of -50.

VEX IQ Robotics Camp Handbook

228-4411

228-4411-750 Rev7

This product is solely intended to be used as part of the VEX IQ system.

4411

8 07648 04411 9

A SUBSIDIARY OF INNOVATION FIRST® INTERNATIONAL