L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

Directions for sharing

- Please change the document name so that it reflects your usernames. For example, *L14:Wheel_Shafts_Encoders_pearcej_wilbornew*
- One person should make a single copy of this worksheet in their Google account.
- Then that person should share it with everyone in the group so they can all edit it. Note that having access will be essential for studying for exams.
- You will be submitting both a downloaded pdf of this document and a program when you are done.

Member Roles

- If you have three people be sure to rotate through all three instead of using odd/even.

<u>**Team Roles**</u>	**Member Name**
First Driver “Drives” at the keyboard on all odd levels, listening to partners. **Facilitator:** Keep track of time and make sure every player contributes and learns equally.	
Second Driver “Drive” at the keyboard on all even levels, listening to partners. **Reflector:** Consider how the answers could be deeper, and how the team could work and learn more effectively so everyone learns	
Third member: Drives every third time.	

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

Actively comments on all ideas and code.	

Lab Activity:

Objectives

- Learn to use an internal sensor: the rotational wheel shaft encoder
- Synthesis concepts learned in previous labs.
- Practice using containers to do a needed computation.
- Practice working with multiple sensors.

Tools and Parts Needed

- Constructed 2-Wheeled NXT Robot
- Appropriate sensors

In this lab, you will be asked to put together concepts discussed in our previous labs and use the new NXT rotational wheel shaft encoders in order to synthesize and apply these concepts in more challenging programming situations.

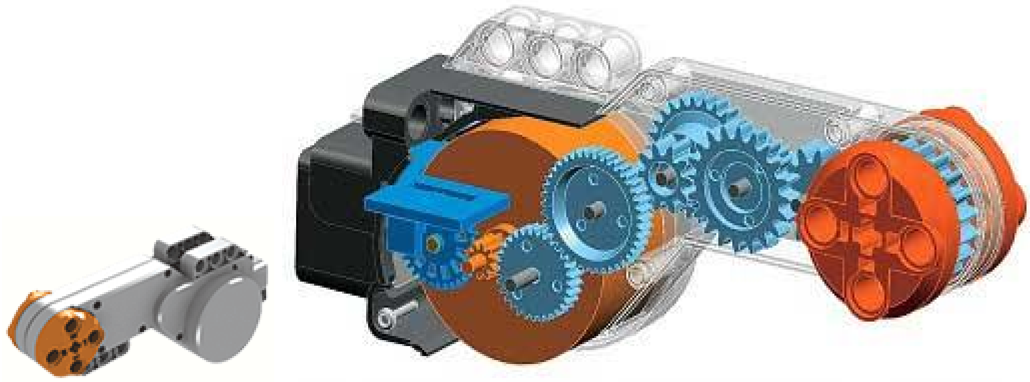
In this lab you will be exploring the improved motors, which unlike the motors for the RCX contain rotational wheel shaft encoders. These help the motor to turn a precise number of degrees of rotation.

1. **(2 min)** Give a couple of real-world examples in which you need a motor shaft to turn a precise number of rotations.

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

The NXT Motors



Contained in the Lego Mindstorms NXT kit are three servo motors. The NXT motor delivers high torque because of its internal speed reduction gear train. Because of that, it also turns slowly and the efficiency is somewhat reduced. Although this motor could physically be connected to RCX with a compatibility cable, it is not recommended for use on a RCX because the high current it consumes is too much for RCX current output.

Even more importantly, these new motors include **rotational wheel shaft encoders**, which tell the NXT the position of the wheel shaft with an amazing 1° accuracy.

This is an important new feature because the RCX motors had no built in wheel shaft encoders, so using "time passed" was one common but inaccurate method for applying current, but using "time" in this way is not an accurate way to determine how much a wheel has turned. The NXT rotational wheel shaft encoder wheel and optical fork provide a very accurate rotation sensor function to the NXT motor, which is very useful for such things as determining how much the wheel has turned, and can be used for computing distances traveled or number of degrees rotated.

2. **(2 min)** Recall a problem you had in a past lab caused by not being able to turn the RCX motor(s) a precise amount**.**

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

Using the NXT Rotational Wheel Shaft Encoders

A wheel shaft encoder sensor detects precisely how many times a motor shaft turns by having an encoder sensor monitored as the wheel shaft turns. It has 12 slits in the encoder, and the motor to encoder gear reduction is 10:32, so for each turn of the wheel hub, the encoders turn 48*10/32=15 turns. Thus, the optical detectors sees 15*12=180 slits. Using both sides of these slits gives 360 ticks per turn resolution or 1 degree resolution.

3. **(2 min)** Describe how you might use this same idea to determine how far you have travelled on a bicycle using only the spokes of one of the wheels, a playing card, a way to attach the playing card such as tape or a clothes pin, and your ears and brain.

Using the Wheel Encoders

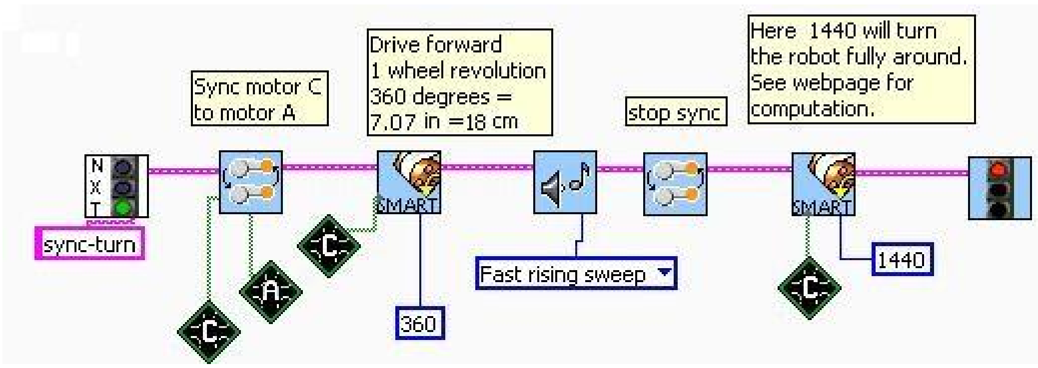
To drive a robot with two motors straight or to turn a precise amount, use the motor sync icon  (located in the NXT menu) and the NXT SMART rotational shaft encoder 

icon (located in the Wait For menu).

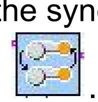
The motor sync icon tells the "follower motor" to follow whatever "the controller motor" does. The following program will drive straight forward 7.0688 inches, and turn 360 degrees and then stop:

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.



In this example, the motor power icon turns on Motor A (and thus Motor C, because it is following Motor A). The SMART rotational wheel shaft encoder controls Motor A's movement to the specified angular position. Motor C follows along, exactly in sync with Motor B.

Once you have synced two motors, they will only respond correctly to commands for the controller motor. If you want to control the follower motor on its own again, you must turn off the synchronization. To do so, use the motor sync icon without any wires attached to it:  .

4. **(5 min)** Write the pseudocode for the above RoboLab code.

Driving the NXT Straight

To drive the NXT straight, we should synchronize the wheels and then drive forward. The formula for the circumference of a circle is Diameter x PI. The NXT wheels are 2.25 inches in diameter, so we can find the distance traveled with two wheels synchronized in 1 rotation of the wheel as 2.25 inches x PI = 2.25 x 3.14159... = 7.0688 inches, which you need to round to 7 in RoboLab. You may use this number to compute distances traveled as long as you understand that each revolution of the wheel is 360 degrees.

5. (5 min) How would you use this information to drive the NXT 2 feet forward?

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

Turning the NXT

To turn the NXT, we can drive just one of the wheels forward. Note that the corners of your turns should be pointed, not curvy--this is most easily accomplished by stopping one wheel while the other wheel drives forward.

The wheelbase of the NXT is 4 inches, so if only one wheel is turning, the NXT will drive in a circle which is 8 inches in diameter. We must convert this full circle to degrees, so we use the fact that there are 360 degrees in a full circle, which is 2 x PI. (8 inches x PI) x (360 / (2 x PI)) = 1440 degrees on one wheel for a full 360-degree turn of the robot.

6. **(1 min)** How many degrees of the motor would you use to make the robot turn 1/2 way around?

7. **(1 min)** How many degrees of the motor would you use to make the robot turn 1/3 way around?

8. **(1 min)** How many degrees of the motor would you use to make the robot turn 1/4 way around?

9. **(3 min)** If the red container contains an integer, which we will call n just to give it a name, how many degrees of the motor would you use to make the robot turn 1/n way around?

Real-world Error

Are these distances and turns going to be perfect? No. Because of real-world physical issues, we may not see a perfect response. Even so, this is an improved methodology over the RCXs.

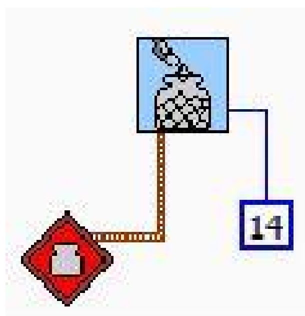
10. **(5 min)** What physical issues might make the robot's turn imperfect?

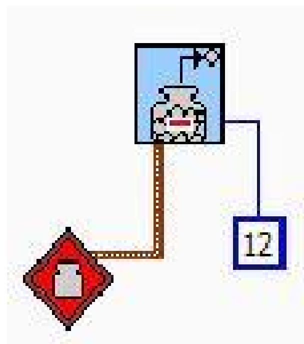
L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

Reminder: Working with Values in Containers

We have learned that a programmer use the variable to store data or any information by assigning values. As you know, in RoboLab, we store data in a container by filling it with the data. In the diagram below, the red container is set to the value of 14. Simple math can be used to store data and change the values that are already stored. As you know, there are other mathematical operations like subtraction, division, and addition. So one can use any math and also any value to use with those operations. In the diagram on the right, the number 12 is being subtracted from the red container. Therefore, if this is run after the above command, the value of the red container is now 2. (14 - 12 = 2)





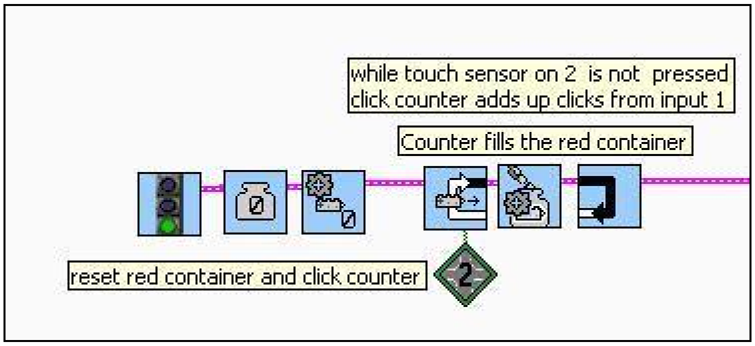
In this lab, you will need to some kind of arithmetic technique with containers in order to determine how far the robot will turn....

Reminder: Using the Click's Container

In addition, remember that you can use the Clicks Container,  which allows the robot to count and store the number of clicks since the Clicks Container was last reset. By using this, the robot can count the number of "clicks" made on one touch sensor. One can use this, for example, to count clicks made on one touch sensor while another touch sensor is not pressed. Just like any other container, it is important to zero it before using it. Remember that we used the following code segment to allow the engineer to enter a count:

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

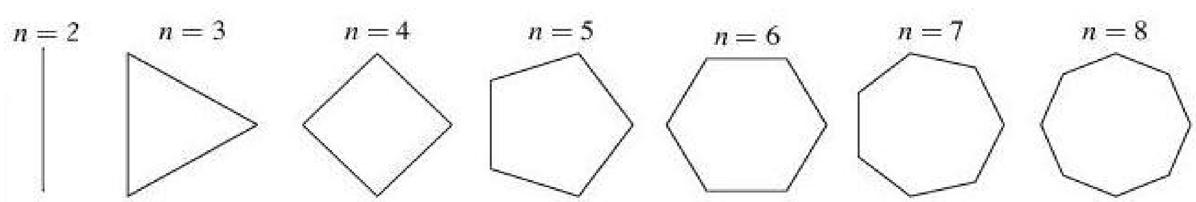


One really nice feature of the NXT is that the display of the sensors will happen automatically when they are used.

Your Primary Programming Task

In this lab, you will create a program that allows the engineer to enter a number that is 2 or higher. (You may do this with two touch sensors set up with one as a "counter", and the other as a "stop" to record the number, as we did in an earlier lab.)

After this number has been entered and recorded, your robot should drive around and around and around following a polygonal pattern on the floor given by the number entered by the engineer. For, example, if the engineer enters 3 clicks, the robot should drive around and around in a triangular pattern. As another example, if the engineer enters 4 clicks, and then the robot should drive around and around in a square. In general, the pattern driven by the robot should be the appropriate one of the following patterns:



.

The robot should then continually drive in this pattern on the floor over and over forever.

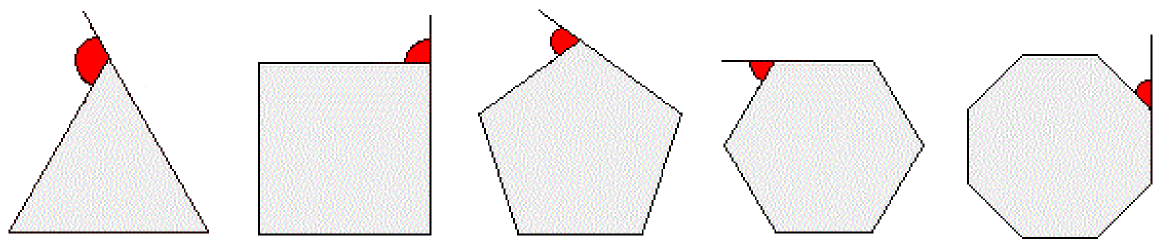
Background Geometry Needed to Compute the Robot's Turn Angle

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

When the robot makes each turn in the polygon, how far does it need to turn? Consider, the following images of a selection of the regular polygons:

. . .



. . .

Mathematicians call angles colored in red in the above images "exterior angles" of the polygon. If you imagine the robot driving straight along the side as it goes counter-clockwise around the polygon, you should see that the "exterior" angle is exactly the angle which the that robot needs to turn through in order to be able to drive along the next side.

So, our next question is how big is the exterior angle of a regular polygon?

Finding the size of each exterior angle of a polygon is actually pretty simple. It is a theorem from geometry that for all polygons, the sum of all of the exterior angles is always equal to 360º. Note that each polygon has the same number of exterior angles as it has sides. Since all the exterior angles in a regular polygon are equal in measure, to find the measure of each exterior angle that our robot must turn, we just need to divide “the full turn” by the number of exterior angles.

Thus, the turn our robot must make is just
“the full turn” / n, where n is the number entered by the engineer.

Isn't that a surprisingly simple computation??

An observation from a student:
"This makes so much sense! If the robot is going to drive around a triangle, it has to turn 1/3 of the way around each time. So, in general, it has to turn 1/n of all of the way around each time. And, you show us how to make it turn all the way around..."

11. (2 min) If the red container contains this n, how many degrees of the motor would you use to make the robot turn through the exterior angle in order to make the n-sided polygon that we want.

L14: CSC 126, Introduction to Robotics

This is a team assignment designed as an in-class activity.

With that Mathematical background, you should be ready to design your main algorithm...

12. (2 min) Which sensors will you use and which ports will they be using?

Use **PseudoCode** to design a program which does the following:

a. Robot detects the number entered by the engineer, and the indication that it should begin driving...
b. Robot computes how to make the appropriate turns.
c. Robot drives around and around and around (forever) in the appropriate polygonal pattern. The polygon's sides should each be 2 feet long. Note that the computations for the sides lengths and the turn angle should be computed as best as you can compute it... just do your best to get the robot to drive in a regular polygonal pattern by using a correct computation. Note that the corners of your polygons should be pointed, not curvy--this is most easily accomplished by stopping one wheel while the other wheel drives forward. Note all this precision requires the new NXT icons be used.
d. *Optional as time permits*: As you construct your algorithm, be sure that your robot will do something appropriate if the engineer enters less than 2.

13. (5 min) Write your **PseudoCode** below:

Next, implement this algorithm in RoboLab:

Be sure to include explanatory comments to your RoboLab code by using the Edit Text icon:  In particular, it is required that you add the following comments:

- Lab L14
- All of your team members' names
- Appropriate comments on the computation of the turn.

- Comments about what the robot will do if the user enters 0 clicks or 1 clicks.

14. (5 min) Describe any challenges you overcame to get your program running correctly.

15. (2 min) Please include any suggestions you have for how this lab could be improved.

To Earn Credit

- **Please note:** Your submission in Moodle means that you were fully involved with all of the work and all of the thinking described above. If you submit work in which you were not wholly a collaborative participant, not only will you not have done the learning needed for the exams, it will be considered academic dishonesty and will be treated as such.
- **Submit the pdf of your completed lab activity in Moodle, AND upload the program *yourusername1-yourusername2*-L14.vi**