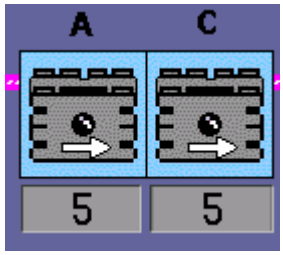
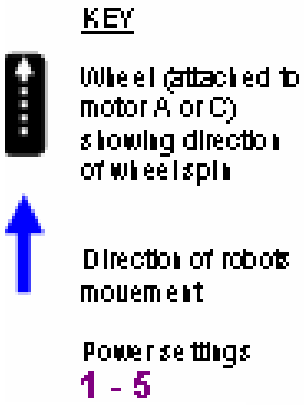
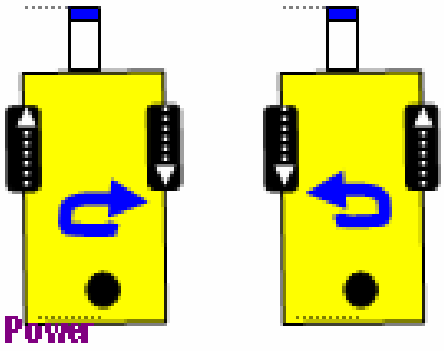
5 STANDARDISE MOTORS- GOING STRAIGHT TEACHERS INFORMATION



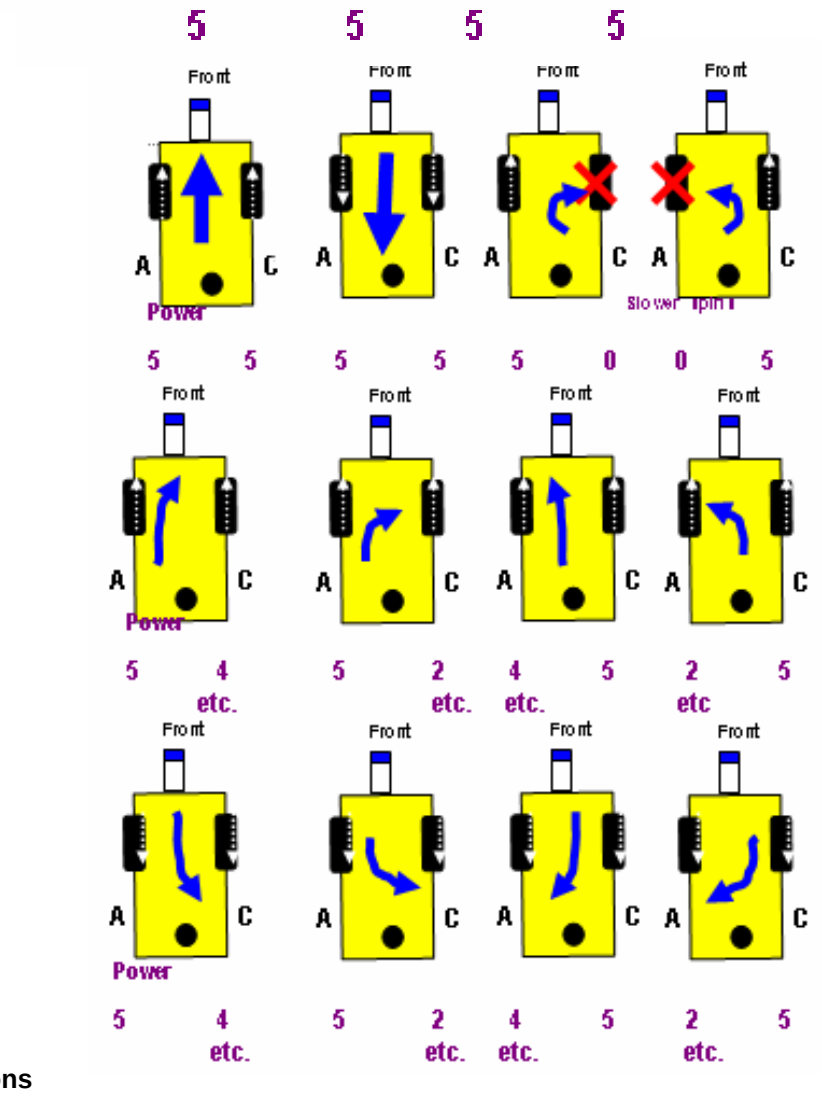
Sometimes, even though your program tells the robot to travel straight, it doesn't.

In the RCX based robots the motors are subject to wear and tear, this causes motors in use to wear unevenly. Your robot's steering is controlled by the two motors turning on and off and you controlling the power levels via the programming to cause the robot to





The diagrams at left give you some idea of the types of movements a tail dragging robot with 2 wheels (2 motors) can do.



The actual dimensions and radii of the curves and spins for each individual robot will be totally dependent on the Performance of each motor installed in the robot. No two RCX robots will perform in exactly the same way

Why do you think this happens?

4 What are some problems that having two motors causes when you are trying to make the robot travel straight?

Wear and tear on motors cause motors to perform differently
No two motors are alike ∴ programming = power & direction
Can result in different outputs from the motors= crooked

5 With what you know about robots and programming list 3 ways that you could adjust a robot with two motors that doesn't travel straight when it is programmed to do so:

Change the power setting
Alter the weight distribution of the robot
Change the motors so the performance of both motors are =

Not straight enough? TESTING ALL THE MOTORS

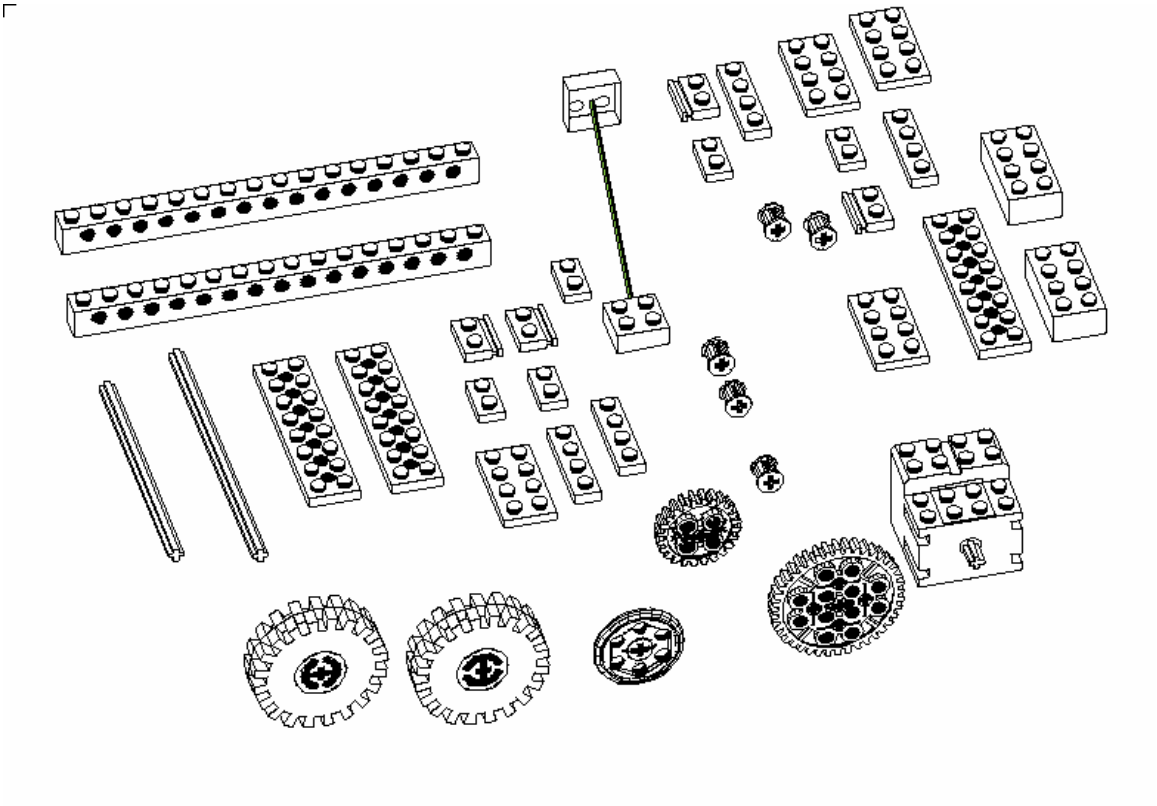
Possibly you have not been able to adjust the programming or physical robot in small enough increments to make your robot travel straight. Probably many of the students in your group will be experiencing this problem too.

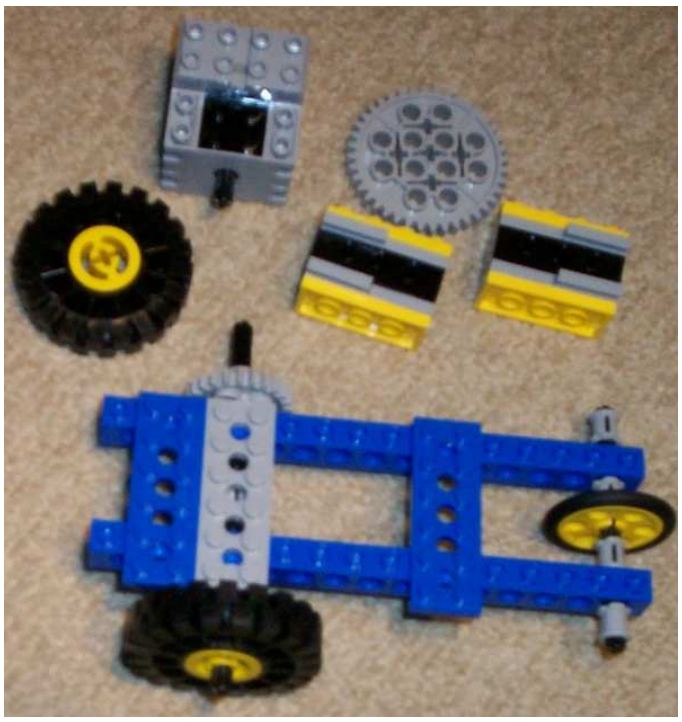
Sometimes the answer to this problem is simply to test all the LEGO motors under consistent conditions with a limited variables and pair motors whose performance is the most similar.

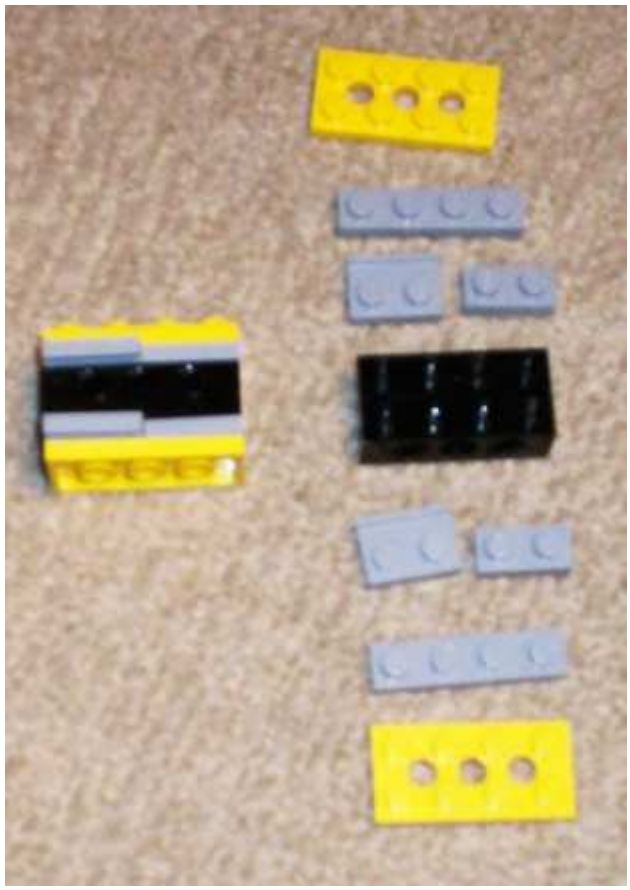
Lets try creating a simple test buggy with only one driving motor, this is our motor test buggy.. We don't need the RCX brick incorporated into this buggy as it will increase the variables. We only want to know about the motors. The motor must have a gear on it that meshes with a single gear on the axle driving the two front wheels. A rubber band over the tested motor and chassis may help secure the two gears together temporarily so that you can test.

Here's a simple design for the test buggy. You may use this design or modify it a little so that it is a little stronger.

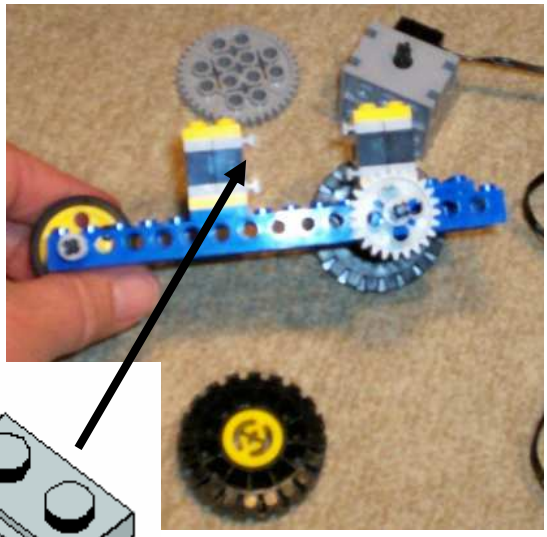
Make simple test trolley like the one pictured below, use this to test one motor at a time





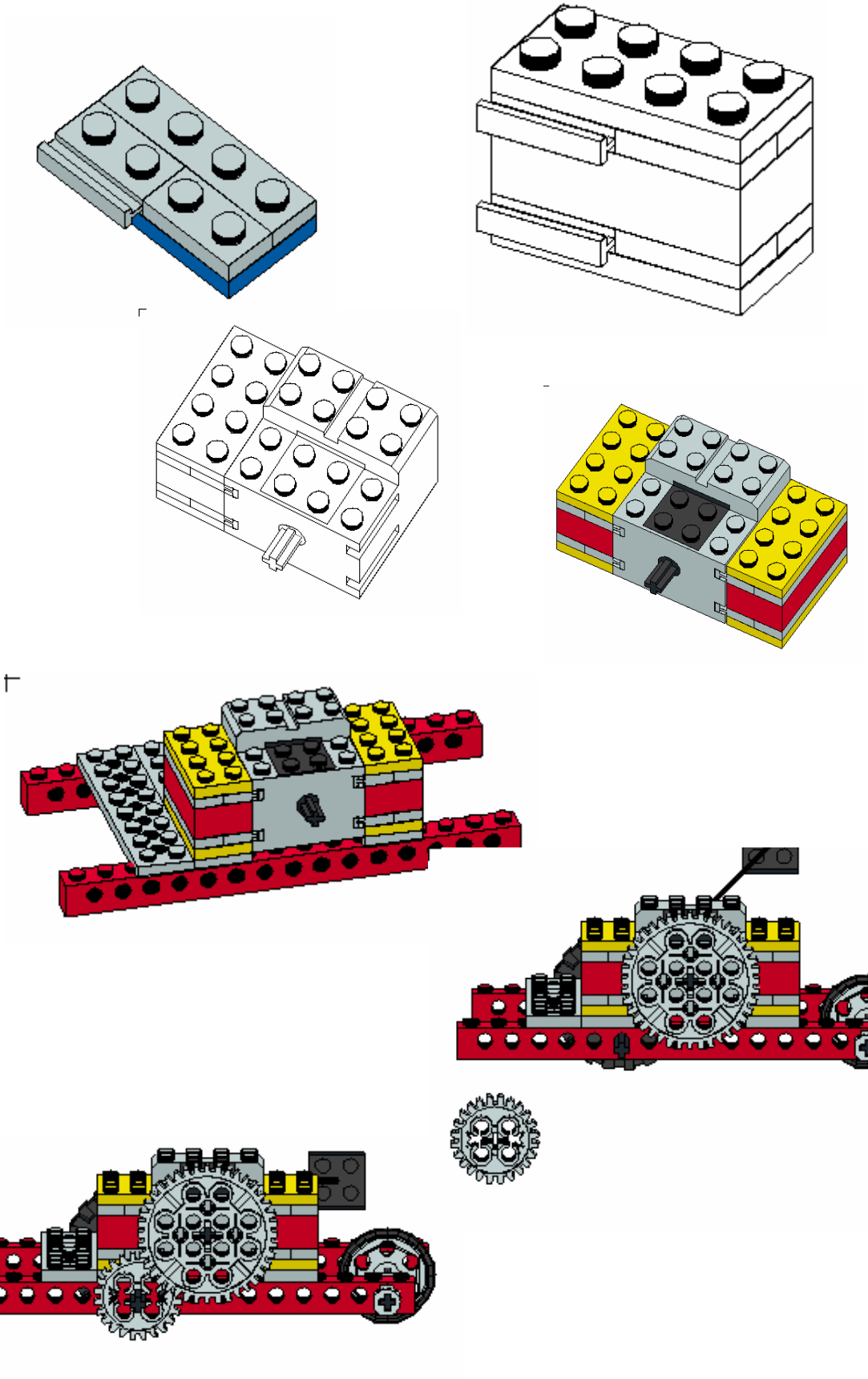


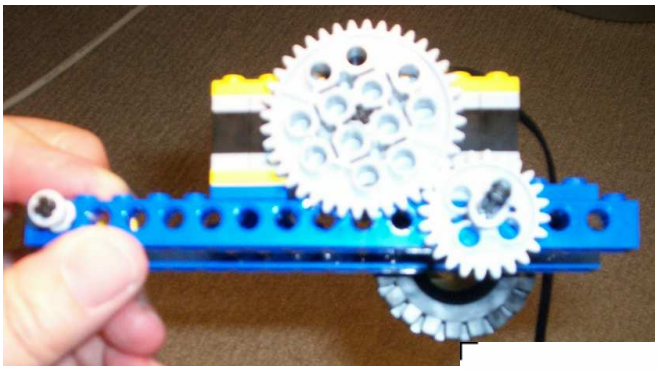
The motor you are testing slides into
The space created. See right

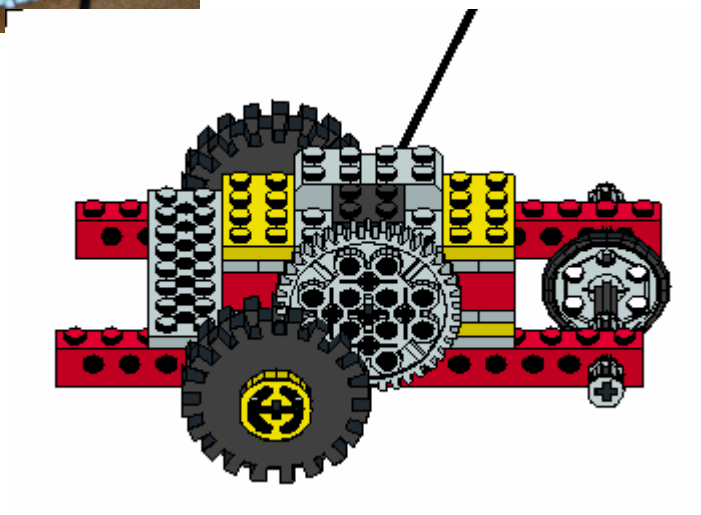


Attach the 40 tooth gear wheel so that it meshes with the other gear (photo below) and push the wheel on. Clip a long electrical lead onto the motor and the other end to an output port on the RCX

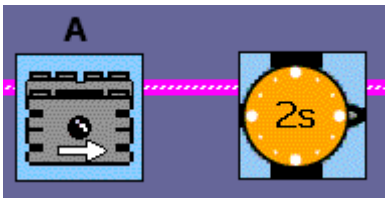
Building hint : use the 2 stud slider plates to hold the motor in place. The motor slides onto these plates and is held securely, it's also easy to change the motor over,







Program the RCX so that motor A runs forward for 2 seconds. Measure accurately exactly how far each motor in the buggy travels in 2 seconds.

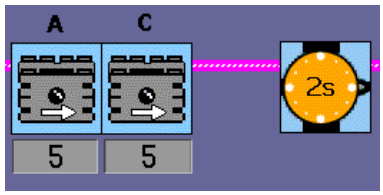


As you will be testing so many motors make sure you number each motor with a permanent marker so that you can identify them.

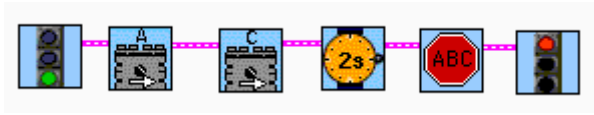
Use the Excel spreadsheet to record your findings. If you know how to use the Excel graph function use the results to create a graph that shows the performance of the various motors..

Once all the motors have been tested, **pair the motors that have the most similar performances and use them in your robot** to make your robot go straight

USING “TIME” TO CONTROL YOUR ROBOTS DISTANCE TRAVEL



The RCX’s motors have no feedback controls, this means that there are no in-built sensors inside the motor that tell the robot exactly how many degrees of rotations the spindle of the motor has made. This poses specific problems for the RCX programmer relating to robot distance control.



Say that the programmer has standardised his robot by testing the motors and installing motors that have exactly the same performance. The programmer then programs the robot to go forward for 2 seconds, he repeats this program several times over the day and gets different results in distance travelled every single time the robot is run!

What are the variables associated with no motor feedback and using the WAIT FOR TIME icon that causes this?

- Battery Power gradually reduced over time and produced less power on power level 5 (100%)
- Friction (you ran on different surfaces)
- Changed the wheel size

How can you program distance travelled more accurately?

If possible use a sensor to provide environmental feedback. Eg use a light sensor to “see” an object /line /colour or a touch sensor.

Keep the batteries fresh, check battery levels.

Always run on the same surface

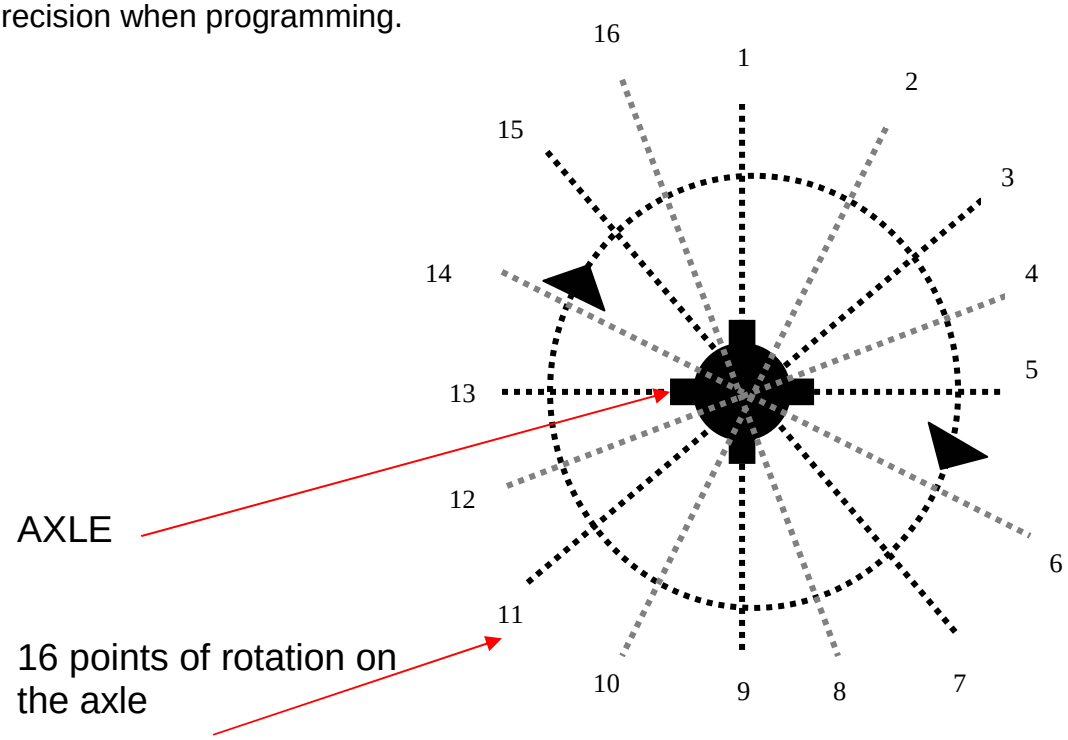
Don’t change the wheel size during tests.

OR

Use 2 X rotation sensors

USING ROTATION SENSORS TO CONTROL YOUR ROBOTS MOVEMENT MORE PRECISELY

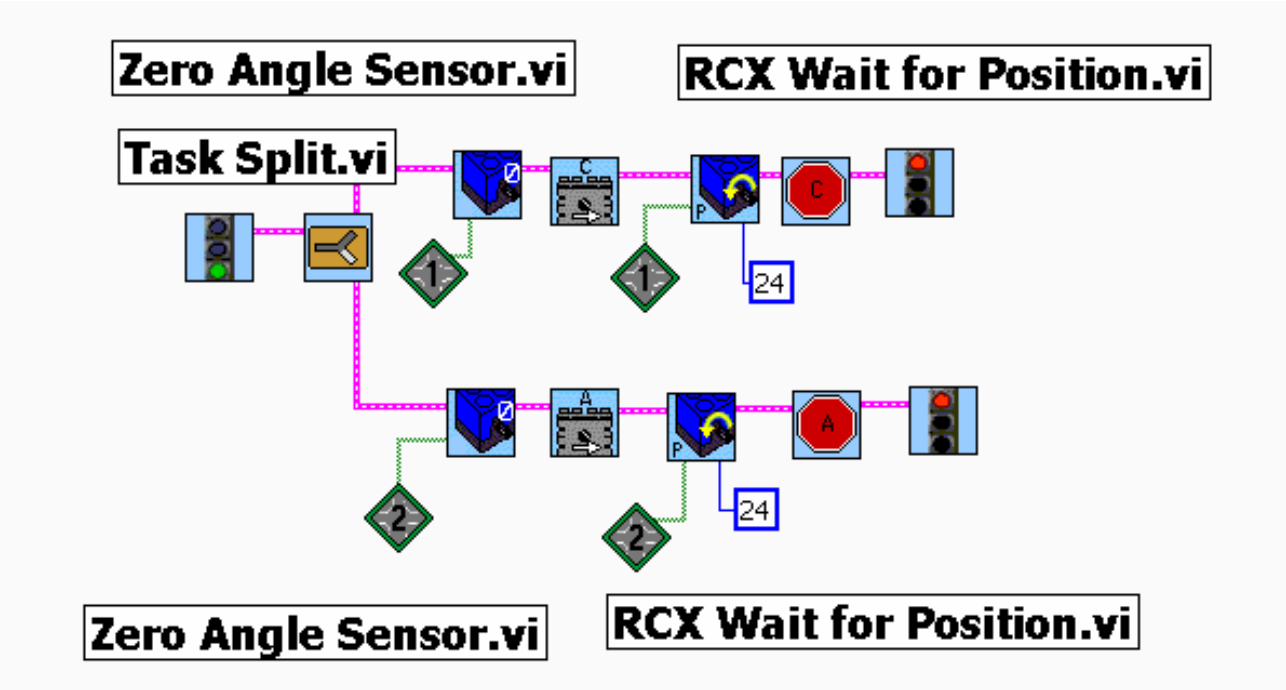
Perhaps add a **LEGO rotation sensor** to the axle so that you have some direct rotational motor feedback that does not rely on the previous variables. This sensor does not offer refined control as it only senses 16 points on a circle (rotation) can offer a little more precision when programming.



The INVENTOR 4 program illustrated below incorporates 2 Angle sensors,
1 connected to motor axle/wheel A and another on axle motor C
Rotation Sensor 1 is monitoring Motor C angles and Rotation Sensor 2 is monitoring Motor A's angles. Students would then need to undertake experiments to determine exactly how far each point of rotation causes their robot to travel.

Note also that this program uses a TASK SPLIT to cause A and C motors to exactly co-ordinate

Also note that I have set the rotation counter to “O” at the start of the program.



This program is included on your CD in the

BASIC PROGRAMMING .LLb as 12 SYNCHRONISE A & C MOTORS
Open this program from INVENTOR level 4

If you have 2 rotation sensors, connect them and experiment with this program.