



Standard League Robot Soccer EV3 Compass Training Curriculum

Brian Thomas



Standard League Robot Soccer EV3 Compass Training

Overview

Welcome! The aim of this curriculum is to guide you through a sequence of programming steps in order to allow a robot to compete in RoboCup Junior and WRO Football. Every attempt has been made to keep these steps simple and easy to attain. Consequently, modifications may be needed to program a more efficient robot for the competitions. You will find that Standard league is no harder than any other other robot competitions. **It is actually easier for beginners to enter and compete successfully in Simple Simon Soccer!** They can then raise their expectations and skills as they progress.

Incorporating kicking in the correct direction is one of those progressions. It is quite a huge leap up from Simple Simon, so make sure that all steps are followed in sequence. There is no easy answer or quick fix to get a robot kicking in the correct direction, so knowing intricacies of your Simple Simon robot beforehand is important. Students need to understand the differences that may be required for their individual robots.

Students have often gone straight to the more complex programming, have become lost and then disillusioned. Those who work through the curriculum enjoy the success at each step, and carry with them knowledge that will eventually allow them to develop their own algorithms and programs.

The sequence of this Curriculum starts with programming for a Simple Simon Soccer Playing Robot. These instructions can be found at <http://www.robocupjunior.org.au/tutorials> .This will allow any robot to successfully play soccer on any field type. The controlling aspect of any robot soccer game is the ball. If the ball is on the field then so should the robots be, chasing it.

Very few students master Robot Soccer in a single year's preparation. They set their goals at a certain level and return the following year with their goal being the next level. This is usually with a different robot, so they need to reprogram anyway.

Further information can be obtained from our web site www.robocupjunior.org.au . Any questions queries or comments, contact: b.thomas@ccg.vic.edu.au

Please Note

These training notes are the property of RoboCup Junior Australia Inc. All or part of these notes are not to be copied or used for commercial gain without the permission of the RoboCup Junior Australia Inc Committee.

****LEGO, Mindstorms and EV3 are Registered Trademarks of the LEGO Corporation**

Compass 1: Feedback

Overview

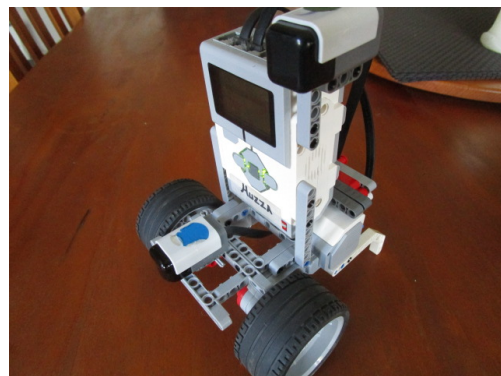
It is assumed that you have worked completely through the Simple Simon Curriculum before starting this unit.

When ever doing a more complex challenge such as this, it is always important to have feedback. This allows you to ensure that the components are working OK and also gives you an immediate message of what they are reading.

Robot Setup

Attach a Hitechnic compass to your robot. Make sure it is as high as possible, but inside the 22cm limit, away from motors and have it aiming forwards. Plug the sensor into Sensor Port 2. The IRV2 should be plugged into Sensor Port 1

Remember, your compass will pick up stray magnetic fields from power cables, motors and metal objects, including table supports.

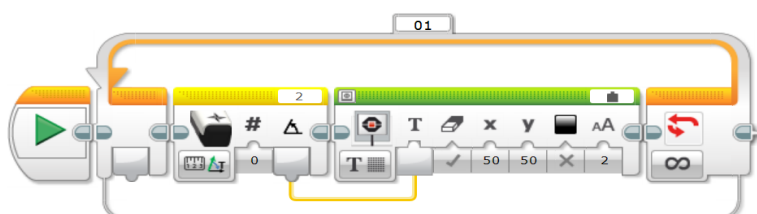


Method

Seeing that the Hitechnic Sensor is not a LEGO product. It sometimes does not do what you expect. The best solution is to unplug and plug in the sensor wire. The Port View function at the bottom right of your screen does not consistently give readings so cannot be relied upon. This program will give your compass reading on the EV3 screen and will use the EV3 indicator lights to let you know when the robot is facing in the correct direction. The compass sensor has the advantage over a Gyro sensor that its direction does not drift over time.

Programming

- Start a new program
- Add an indefinite loop
- Insert the Yellow Hitechnic Compass Sensor block and set it to Port 2
- Set the Compass Block to Measure-Relative Heading
- Insert a Green Display Tile
- Select Text in the Display Drop Down
- Select Wired from the File Name drop down in the top right corner of the Display Tile
- Change X and Y values to 50 so that the display is in the centre of the screen
- Wire the compass heading to the Text input



If you run the program. The reading on the screen should start at "0" and the give negative readings as it is rotated in a clockwise direction or positive readings for anticlockwise.

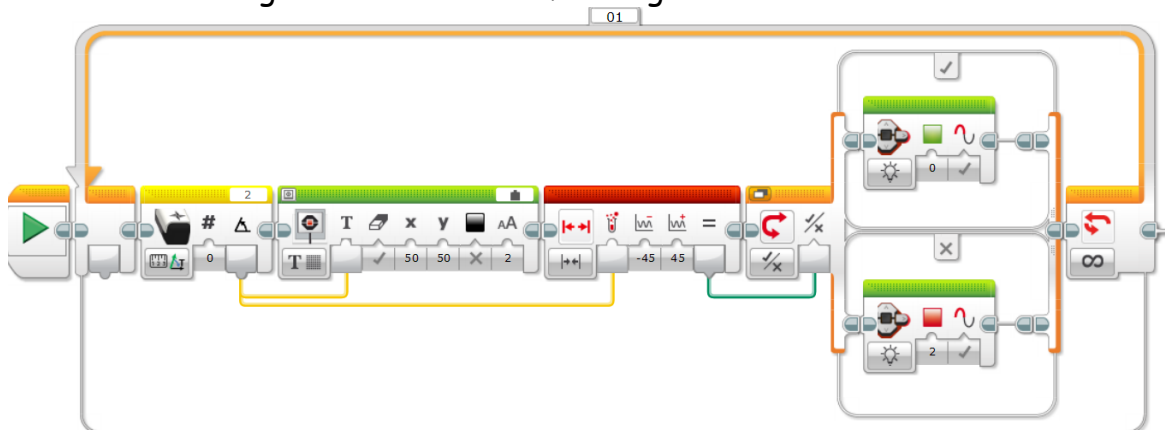
Method

Now to give visual Feedback while the robot is playing a game. Firstly you need to set the direction that the robot wants to aim. Rotate the robot to the desired direction of the goal and take the reading. Insert this reading in the "Compare To" drop down in the sensor block. This needs to be done before each tournament to adjust for the different orientations of the playing fields. Of course this will not change during the day so only has to be done once for each direction that your robots are kicking.

Now, to allow your robot to give on field feedback for correct and incorrect direction. This is done with the robot starting by aiming at the goal it is kicking to and giving a green light on the EV3 when it is aiming towards the goal and red when it is not.

Programming

- Create another program on the same screen as above by adding another "Start" tile
- Insert an Indefinite Loop
- Insert Yellow Compass Sensor Tile
- Set it to Measure-Relative Heading
- Add a Range Tile and Set it to Inside -45 and +45
- Add a Switch set to Logic
- Insert Brick Status Light tiles to the upper and lower branches of the switch
- Set the Upper Light to Green and not flashing
- Set the Lower Light to Red and not flashing



Run this program. Manually turn the robot around with your hand. The robot should flash green when it is aiming towards goal and red when it is aiming away. Turn the robot 360 degrees in both directions. You will see that it continues to flash green when it aims towards its goal. This is the advantage of using a compass over a gyro sensor. The gyro has a "blind spot between +/- 270 degrees and +/- 359 degree rotation

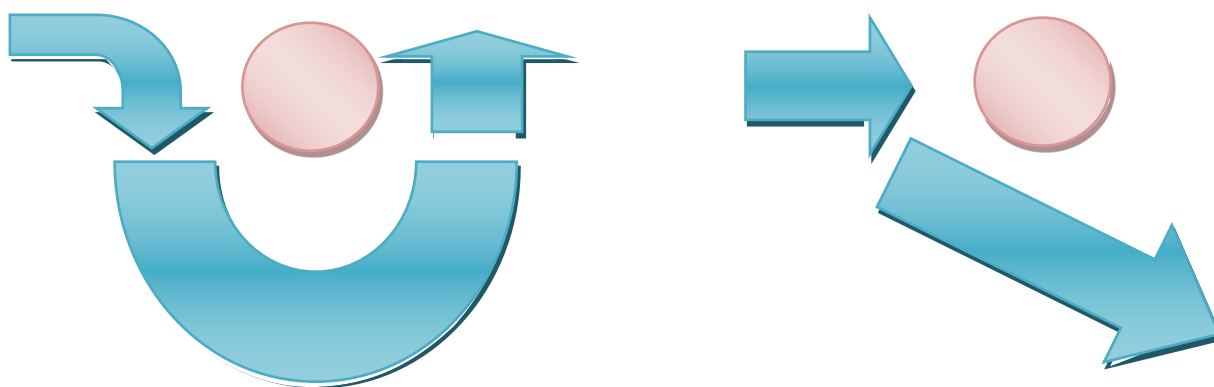
Compass 2: Ball Avoidance

Overview

You are now able to look for the ball and kick it using Simple Simon Programs. The only thing left is to make sure that you will kick it in the right direction. Therefore you must use your Compass sensor to make sure you are kicking in the right direction and not goals for the opposition.

Method

Before using the Compass sensor, you must devise tactics to avoid the ball when it is detected and the robot is heading in the wrong direction.



The diagram above shows possible paths when the robot is kicking from right to left and is stuck on the wrong side of the ball. The circular path takes longer, but allows for greater accuracy on the return. The second one is simpler to program, but relies on a number of moves to get to the correct position. We will focus on the simpler path.

The robot must trace a path that takes it past the ball and quickly begin a scan that will have it approaching the ball from the right direction.

1. The robot senses the ball.
2. The robot approaches the ball to a set distance. This allows the compass reading to settle.
3. The robot checks its Compass to see if it is kicking the right direction.
4. It finds it is on the wrong side of the ball.
5. The robot turns slightly to the right to miss the ball.
6. The robot goes past the ball and starts again.

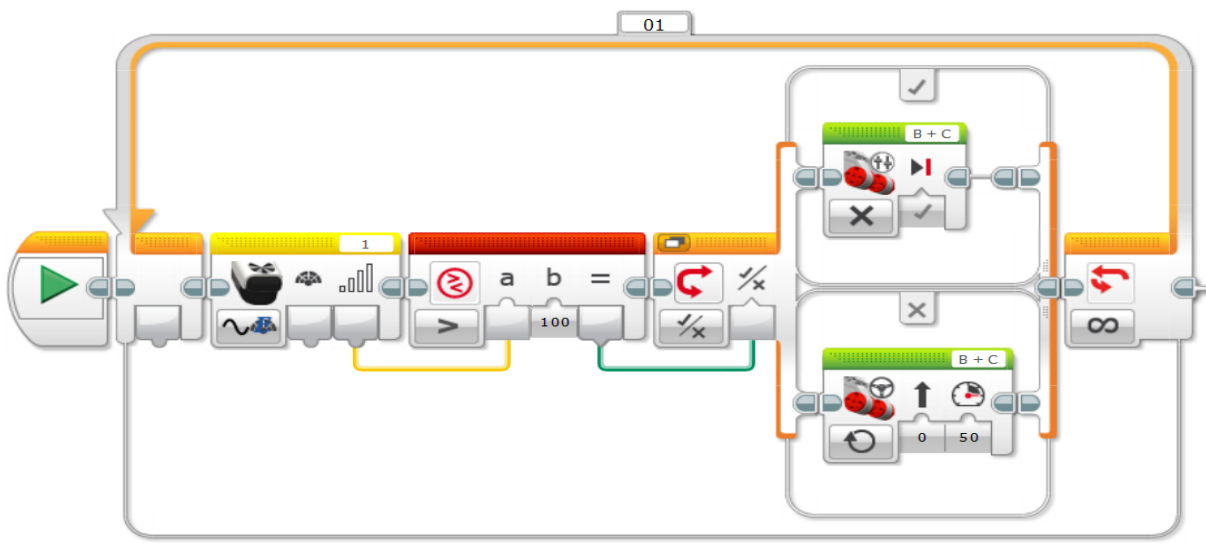
This is not easy, as the path varies from robot to robot and changes as battery levels change.

Method

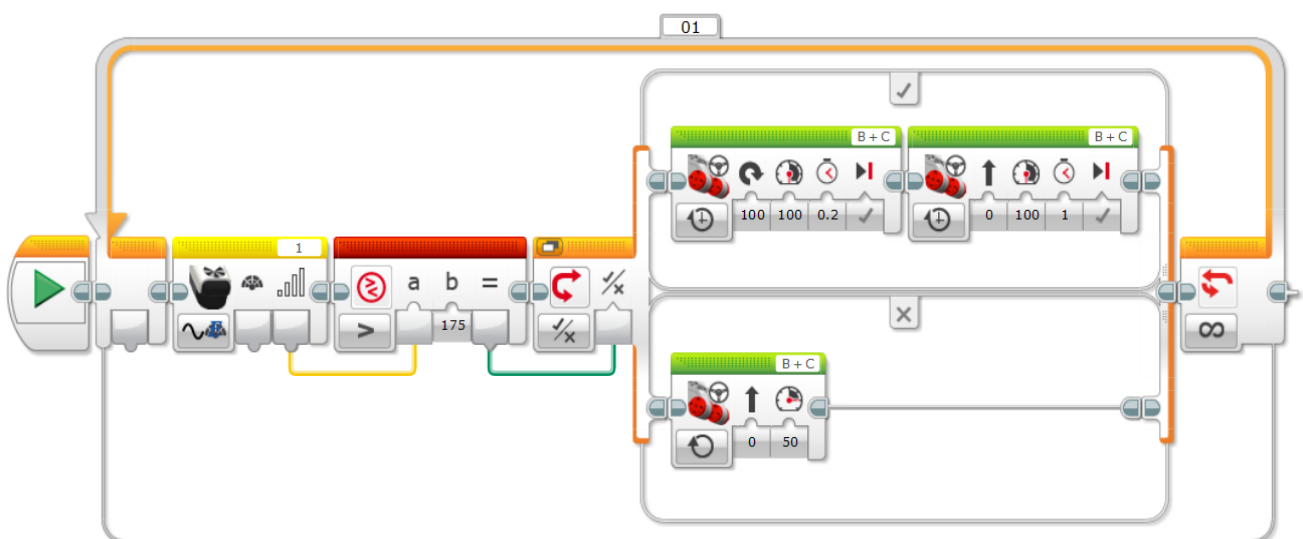
Start with a new program

- Add a loop
- Add in IRV2 sensor tile. Set it to Alternating-Direction/Strength
- Insert a Compare tile and set it to Greater than 100.
- Add a Logic Switch
- Insert Motor Forwards in the "False" branch and Motor Stop in the "True" branch

Run the program with the ball in front of the robot. The robot will approach the ball until the intensity reaches 100 and then stop. The intensity reading needs to be adjusted so that the robot stops at 15 cm from the ball. This will vary with different light conditions at venues and must be regularly adjusted. Practise by running the robot in different light levels and calibrate it to stop at 15 cm.



Now add a simple ball avoid movement in the "True" branch to get the robot past the ball without touching it. Once again test-improve, test-improve, test-improve.....



Compass 3: Compass Kick and Ball Avoid

Overview

This is the final step into preventing your robot from kicking own goals. This will be done by combining the Compass Program with the Ball Avoid Program.

Robot Setup

Attach a Compass Sensor to your robot. Make sure it is as high as possible within the rules and have it aiming forwards. Plug this sensor into Sensor Port 2. The IRV2 should be plugged into Sensor Port 1.

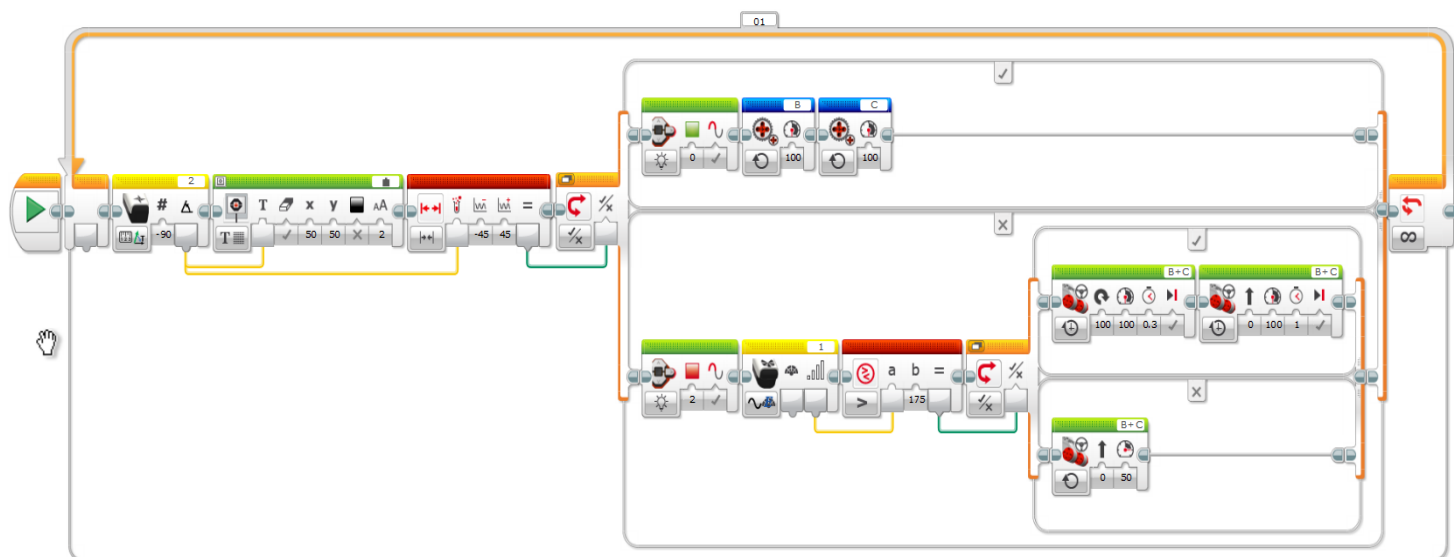
Programming

Open the Feedback program and copy the blocks inside of the loop in the Ball Avoid Program. Paste this into the Feedback program, just after the Red Light. Insert Motor Forward blocks after the green light. This is when the robot is facing goal, so needs to be going as fast as possible. You can use Unregulated Motor for this to get maximum power.

Now test-improve, test-improve, test-improve.....

If the Red Light comes on too often, you may have to change the -45 to +45 range to + 60 or- 60, or maybe even more.

If the robot keeps hitting the ball you it may be getting too close, so you will need to reduce the IR Intensity threshold or turn more during the ball avoid.



Compass 4: Ball Chase and Ball Avoid

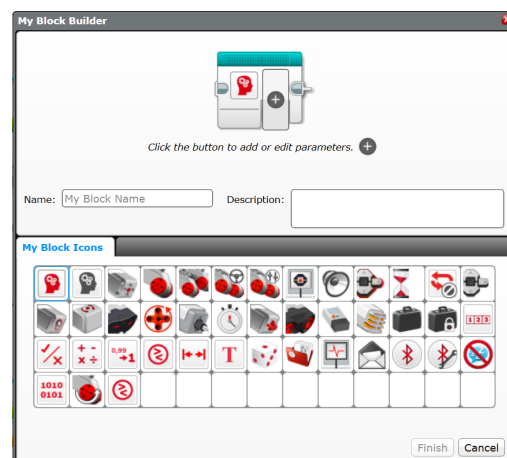
Overview

Only when you are totally happy with your ball avoid program, you can create a "My Block" and then insert this program into the Simple Simon program just as a simple block

Programming

If you place the Ball Avoid into your Simple Simon program, it will become very crowded. A great way to avoid this is to create your own block.

- Select all of the tiles inside of the Ball Avoid Loop
Do not select the loop.
- Go to Tools on the top tab
- Select My Block Builder. The panel to the right will appear
- Type in something like Ball Avoid in the block name
- Select an icon for your block that fits the action of your robot
- Click Finish and your large program will become a single block
- This block can now be found in the teal coloured section of your control panel
- This block can be changed by double clicking on it.



Simply paste this Block into the number 5 tab of the modified Simple Simon program and you are ready to go in Standard League.

You can also experiment by placing the block in 4 and 6. This works well with some robots.

Compass 5: Advanced Ball Avoid

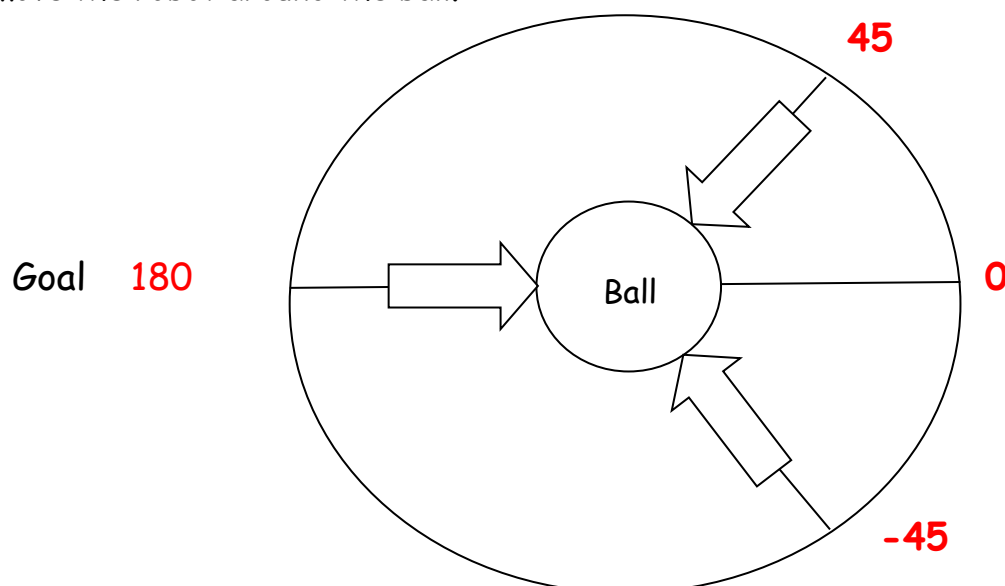
Overview

When your robot is executing a Ball Avoid, it will often go the long way around the ball. For example, it will travel 270 degrees around the ball rather than 90 degree. This program will allow your robot to select the short way around the ball in order to be facing the goal it is kicking to.

This is not easy, it will take time, but the result will be a highly competitive robot if you get it right.

Method

If the robot is kicking towards the left of page on the diagram below. It will register the following readings when facing the ball in the centre. This will take a bit to get your head around. Load up the Feedback Program onto your robot, get a ball put it in front of goal and move the robot around the ball.



Range	Robot Aim	Robot Action
-45 to +45	Toward Goal	Kick Goal
-45 to -180	Away from Goal-Left	Ball Avoid Right
45 to 180	Away from Goal-Right	Ball Avoid Left

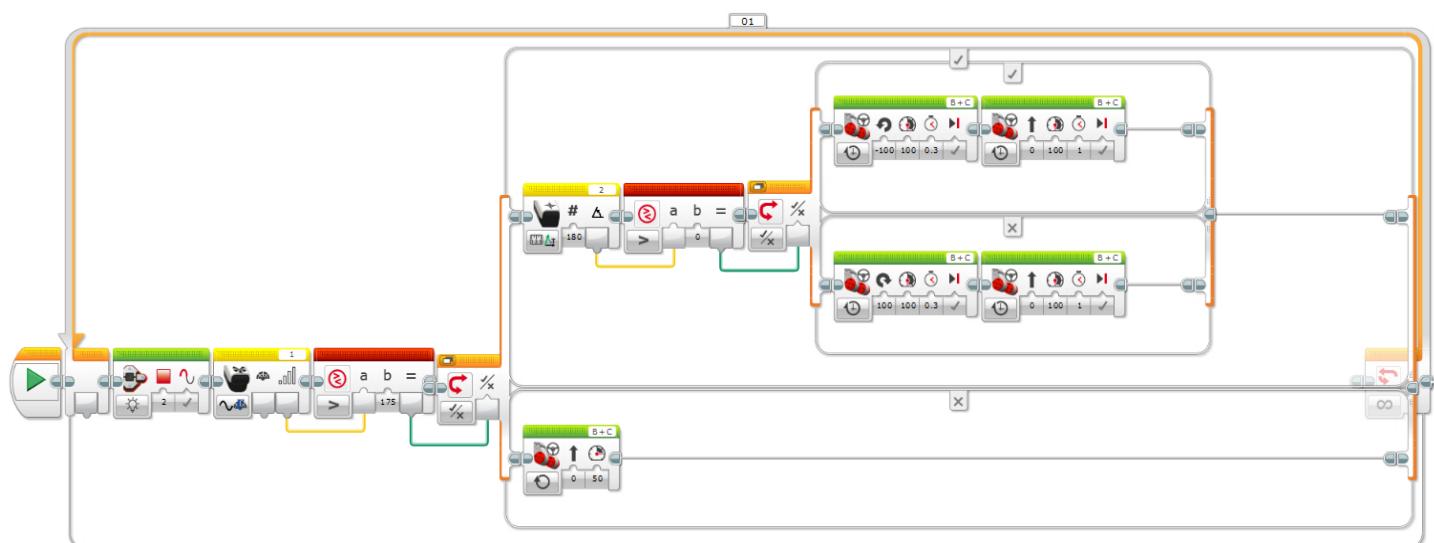
Do not start programming until you have this totally clear in your head

Simply if the Compass is greater than 0, it needs to Ball Avoid Left. If it is less than zero Ball Avoid Right.

Programming

You will now create a Compass range filter, which will have your robot avoiding the ball in the shortest direction to get it to face goal. Use the ball avoid action that you have developed in the previous unit

- Start with the Ball Avoid Program
- Add a Compass Greater Than Zero Tiles and Switch to the Top Branch
- Program the robot to Ball Avoid Left in the True branch
- Program the robot to Ball Avoid Right in the False branch



The programming is actually easier than working out the logic of what is going on. Run the program, place the robot at different sides of the ball, heading right or left from the Compass Zero. The robot should work its way around the ball until the ball is between the robot and the goal every time.

Now test-improve, test-improve, test-improve.....

When you have this working perfectly the contents of the loop can be copied and pasted into the Ball Avoid Section of the My Block you have created.

This section has a Target Compass Heading in it. When calibrating for a competition field, this must be changed as well.