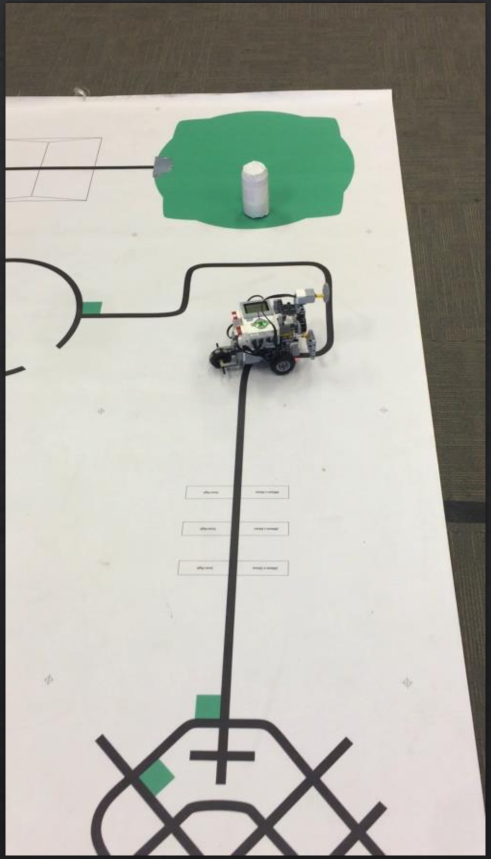
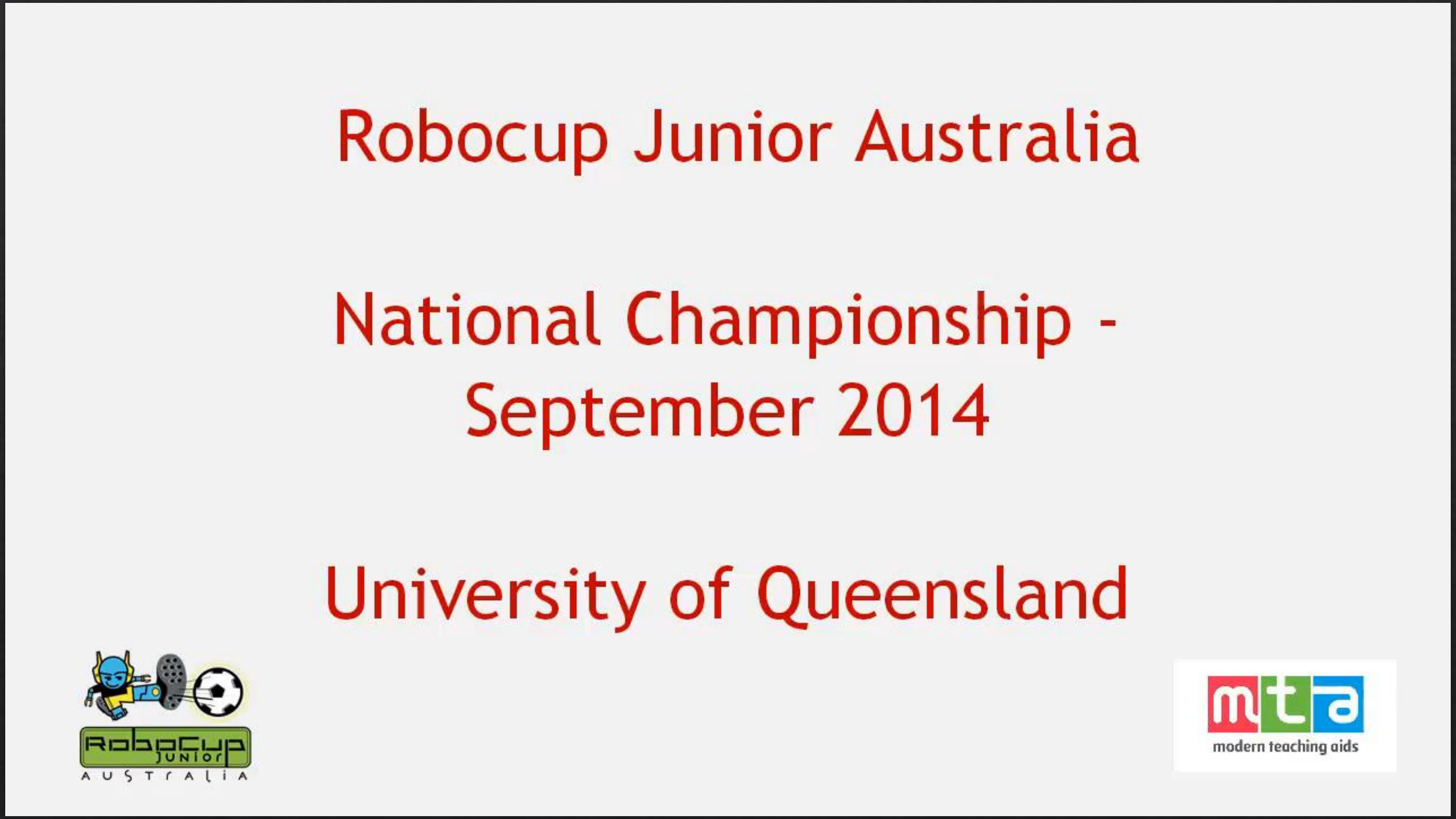
RoboCup Junior Rescue Line

Alexander Bush

Getting Started



And also…



Robocup Junior Australia

National Championship - September 2014

University of Queensland

RoboCup
JUNIOR
AUSTRALIA
mta
modern teaching aids

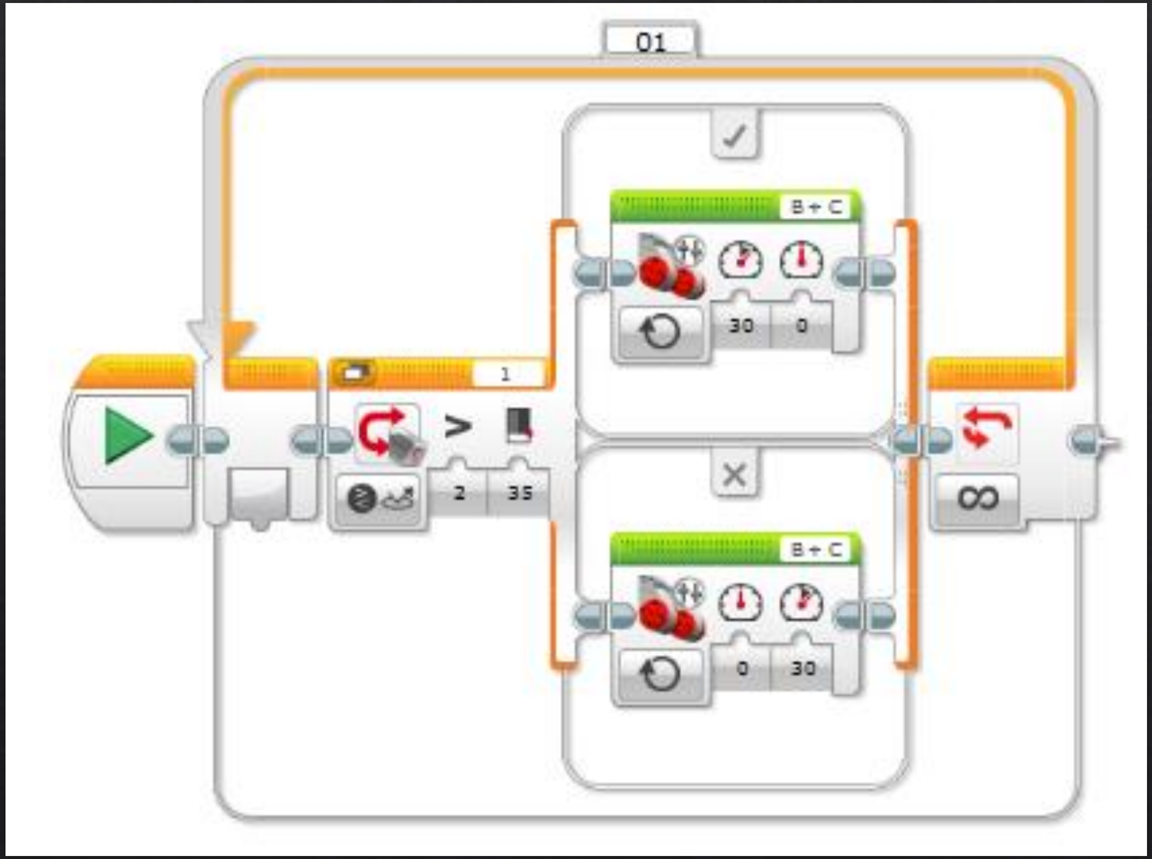
Some Key Rules

- Robot Size
- Course Components
- Ramp angles
- Rescue Canister
- Primary, Secondary and Open Rescue Differences
- Scoring

Breaking the Problem Down

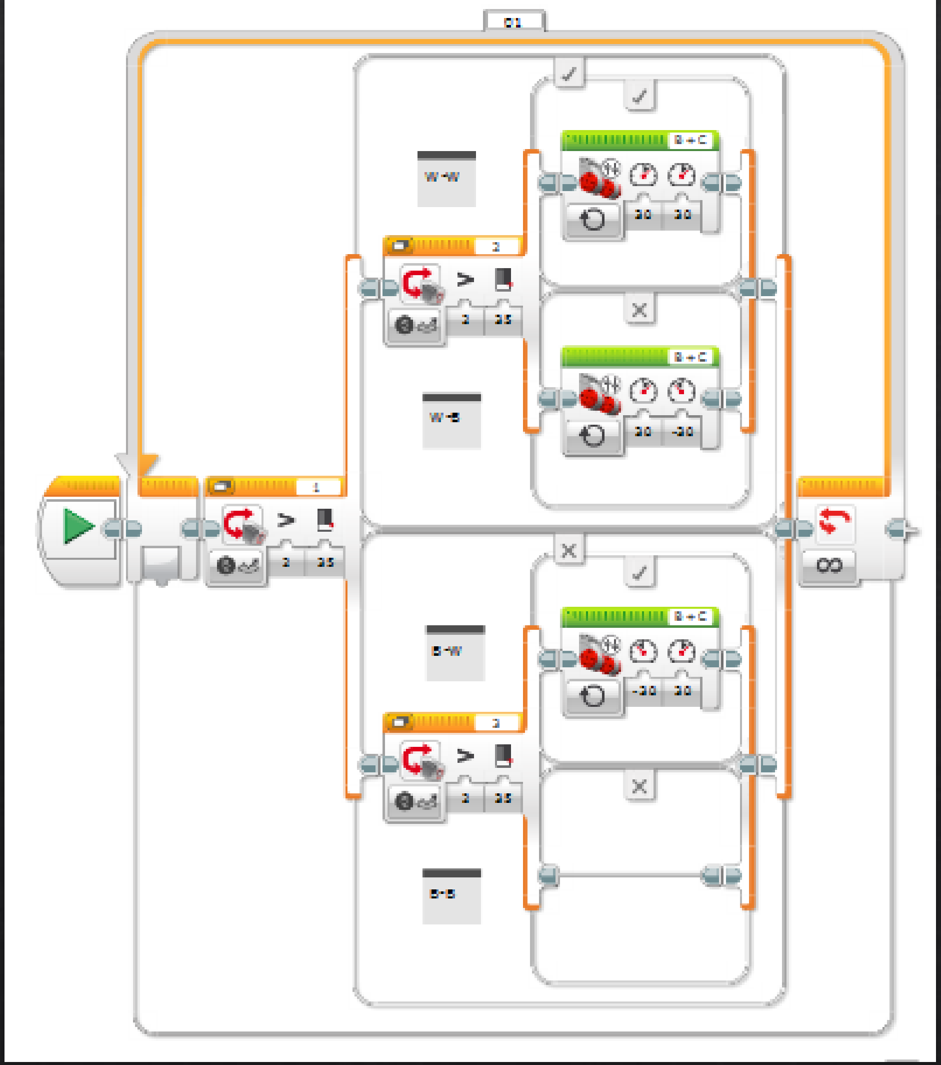
1. Follow a Line
2. Turn on Green Markers
3. Avoid Obstacles
4. Go over Ramps, Bridges, Speed Bumps
5. Detect the Chemical Spill
6. Rescue the Canister

1. Follow a Line



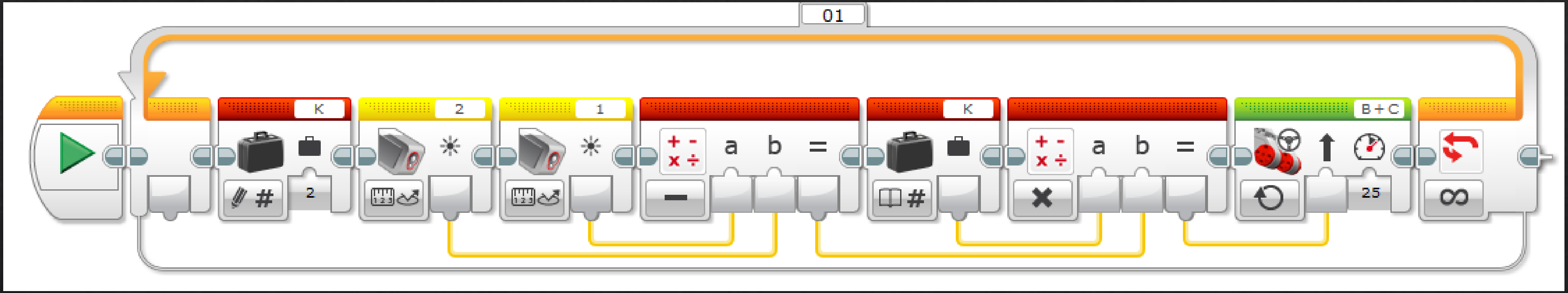
- Using 1 Colour (Light) Sensor
- Loop block so the robot keeps trying to complete the same task forever
- Switch block for the TWO choices that the colour sensor can pick up
- Using REFLECTED LIGHT to see the difference between BLACK and WHITE
- Using Tank Steering block to move the robot – clockwise when the sensor sees WHITE and counter-clockwise when the sensor sees BLACK

1. Follow a Line



- Using two Colour Sensors, keep the line between the sensors
- Loop block – repeat forever
- Switch block for Colour Sensor 1 – Does the left side of the robot see Black or White?
- Switch blocks for Colour Sensor 2 – Does the right side of the robot see Black or White?
- Special Case: What should we do if BOTH sides of the robot see BLACK??

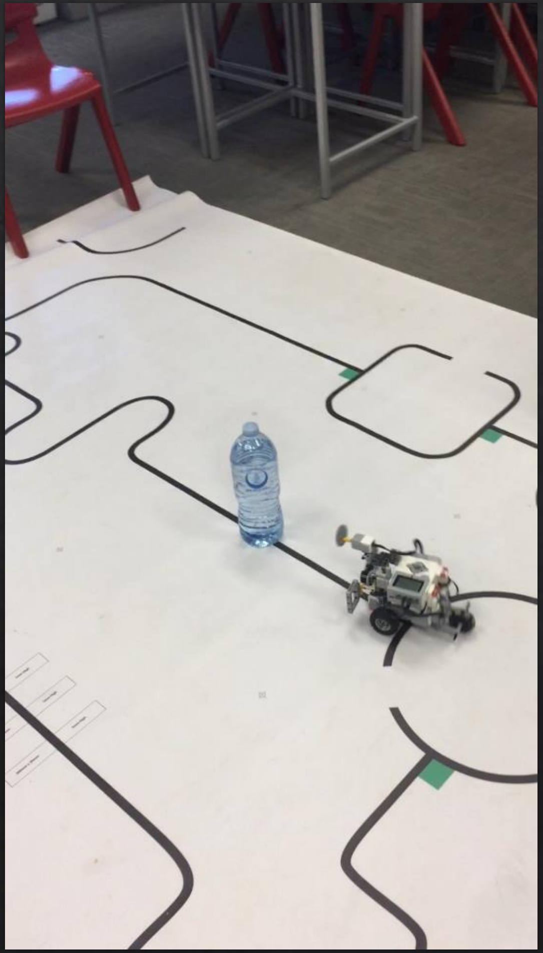
1. Follow a Line



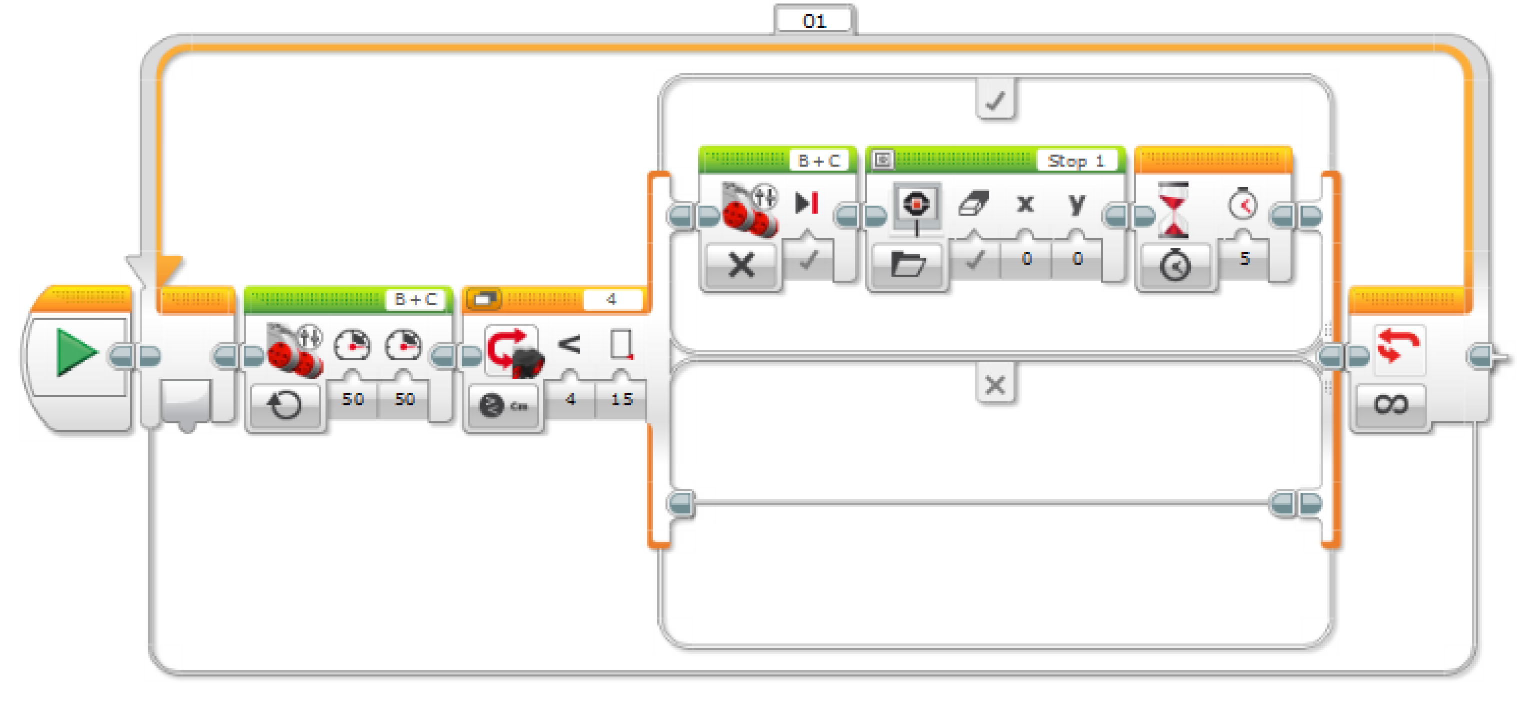
2. Turn on Green Markers

- This is one of the most difficult tasks in EV3!
- Need to understand how your robot “sees” the green, and the limitations of your sensors
- Some strategies are:
  - Treat it like black (have the threshold above green)
  - Switch sensor between light and colour and look for green
  - Perform some action when your sensor sees green (to confirm it is green)
  - Use a subroutine that is “looking out” for green
- This will NOT be covered today unless we have time

3. Avoid Obstacles

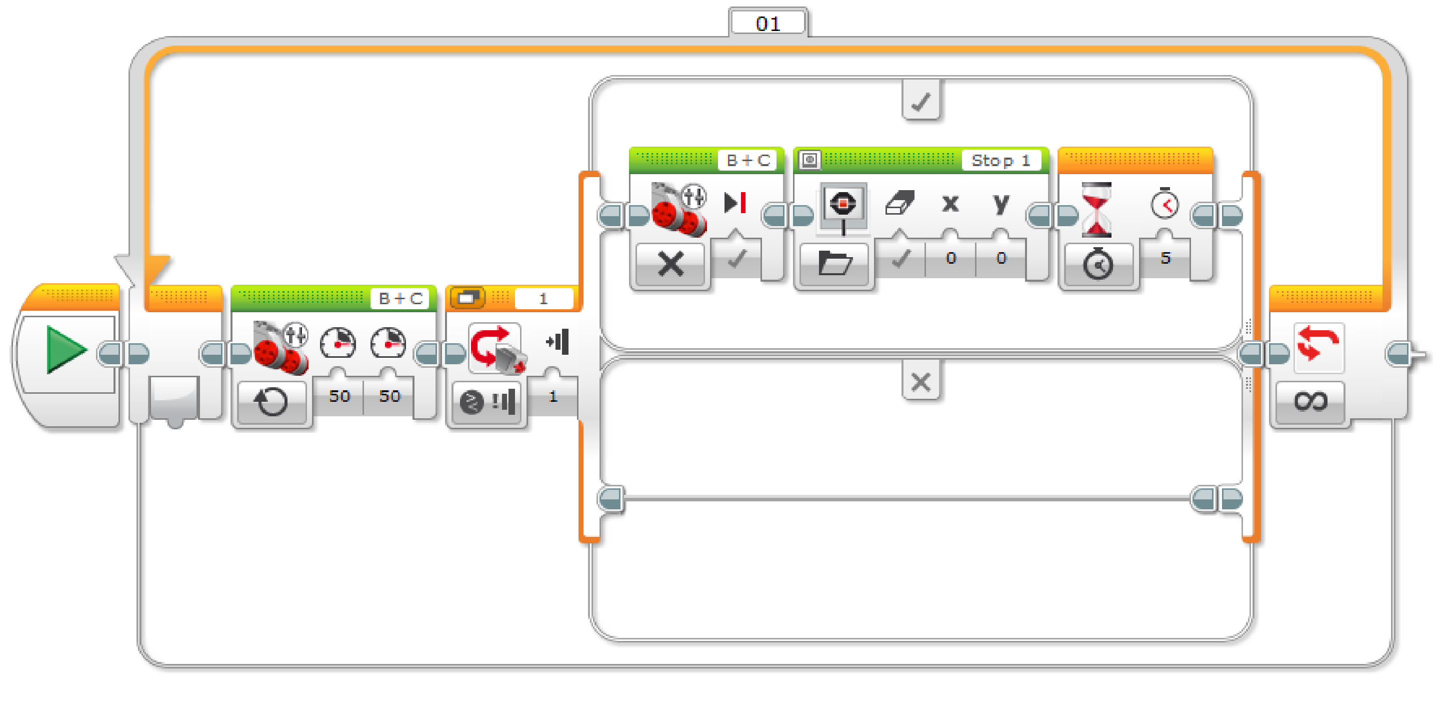


- How do we “See” the obstacle?
- Can use the ultrasonic sensor



3. Avoid Obstacles

- Could use a touch sensor:
- Want it high enough to miss the rescue canister
- Want it low enough to fit through the gate
- Want it wide enough to find the obstacle even when the robot is turning a little

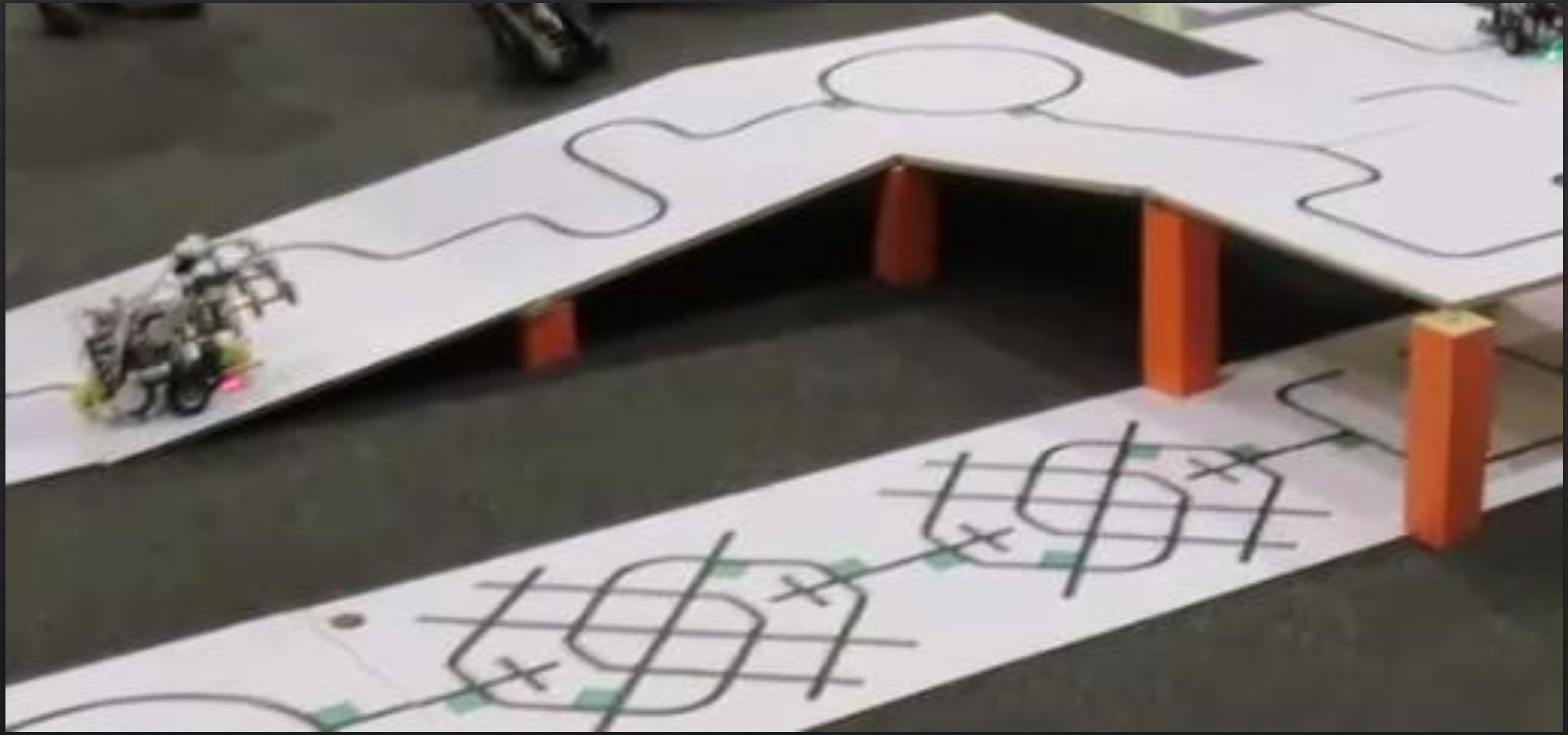


3. Avoid Obstacles

- Going Around – Dead Reckoning
- 90L, FWD-1, 90R, FWD-2, 90R, FWD-1, 90L
- What happens with small errors? In the 90 turn, or at the start?
- Can we fix it?
- (Yes we can!)

- Going Around – Constant Turn
- Some lft turn amount…
- Work out the radius of turn needed and the arc length to get around the obstacle
- Start this and look for the black line – the inside (right side) sensor should see it first!

4. Going over Stuff



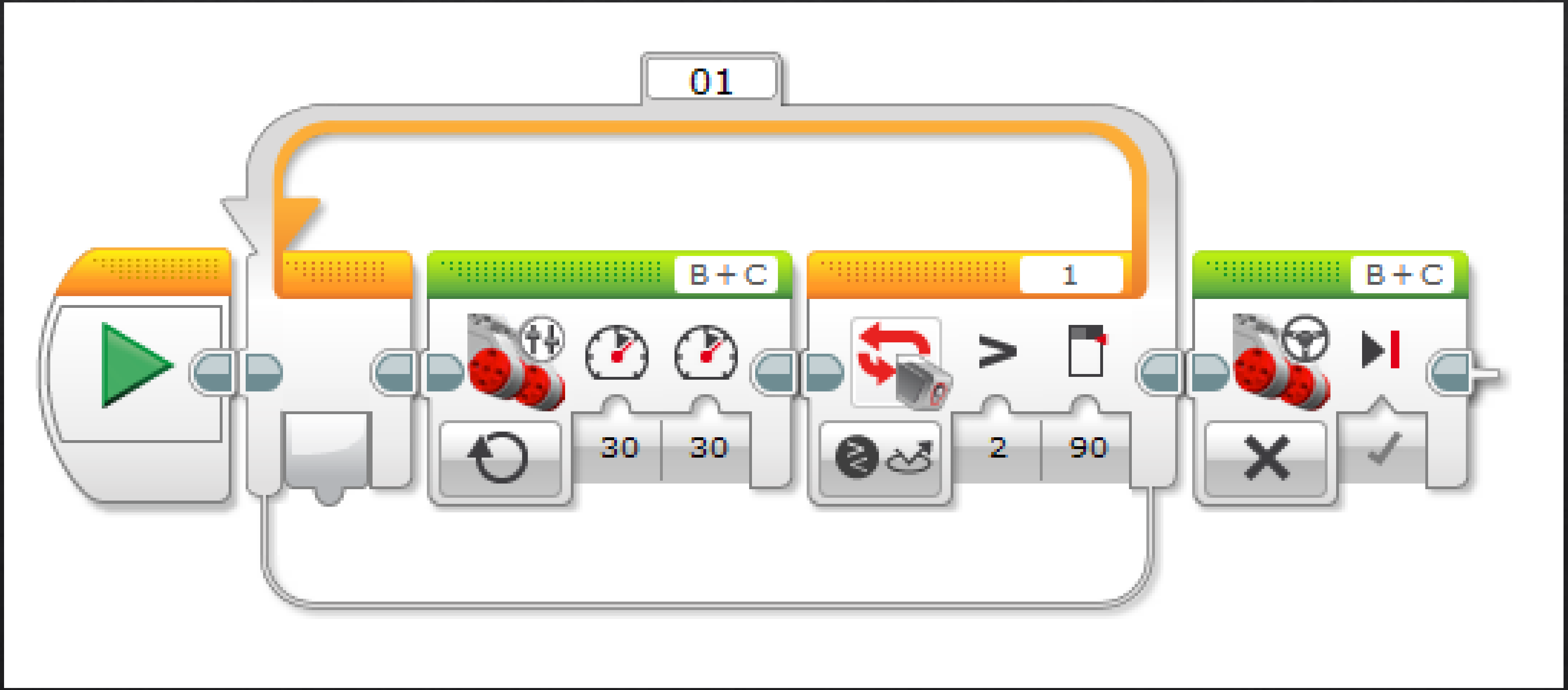
- Engineering Skills
- Weight Distribution
- Wheel Base
- Sensor Height
- Sensor Distance from Axles
- Rear Wheel Type

5. Detect the Chemical Spill

- Once you detect the spill, line following ends
- How can you tell you have entered the chemical spill?
  - Light Sensor detects SILVER?
  - Both Light Sensors detect SILVER?
  - Both Light Sensors detect GREEN?

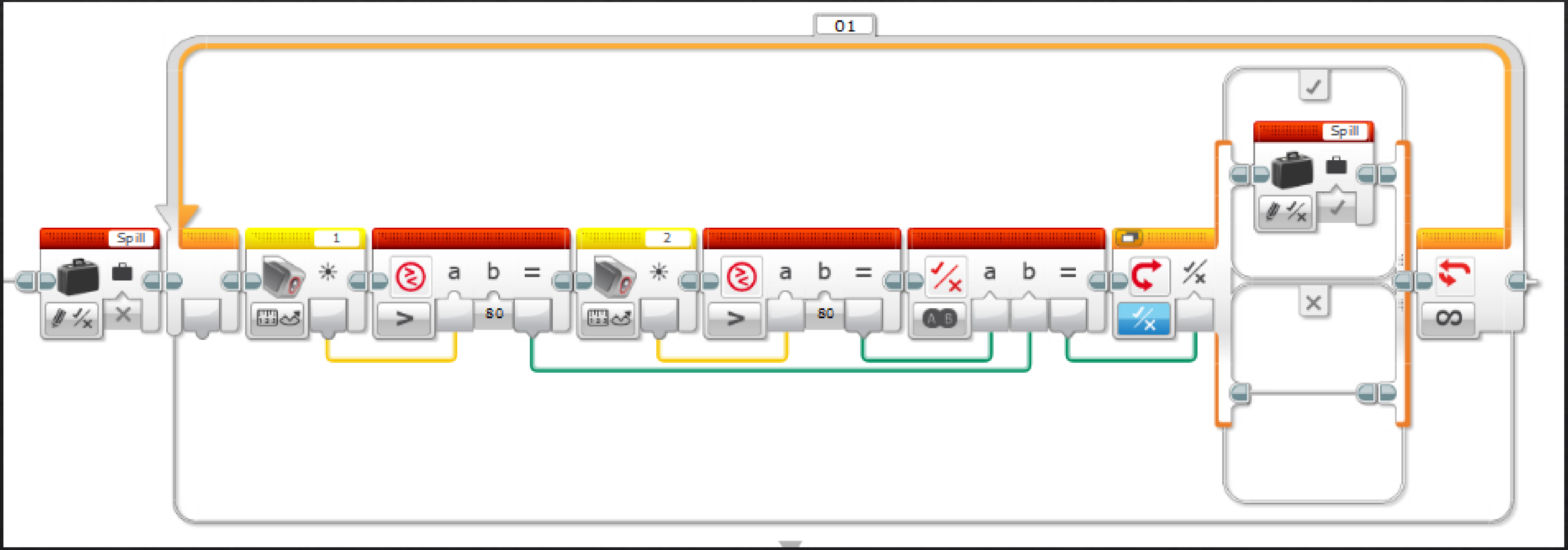
5. Detect the Chemical Spill

- Light Sensor detects SILVER?



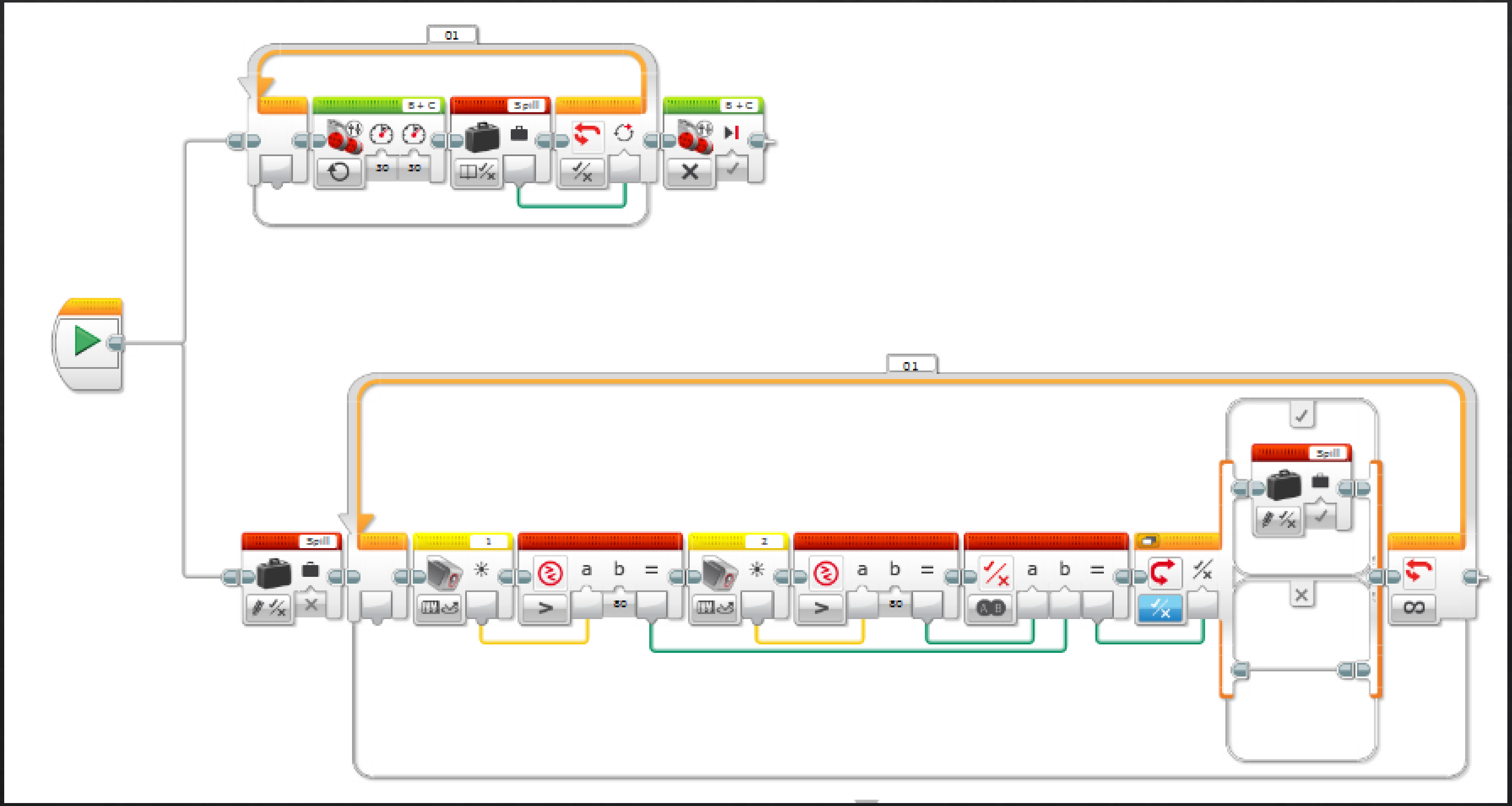
5. Detect the Chemical Spill

- Two Light Sensors detect SILVER? If either sensor over the silver threshold (80ish) then trigger the end of the loop.



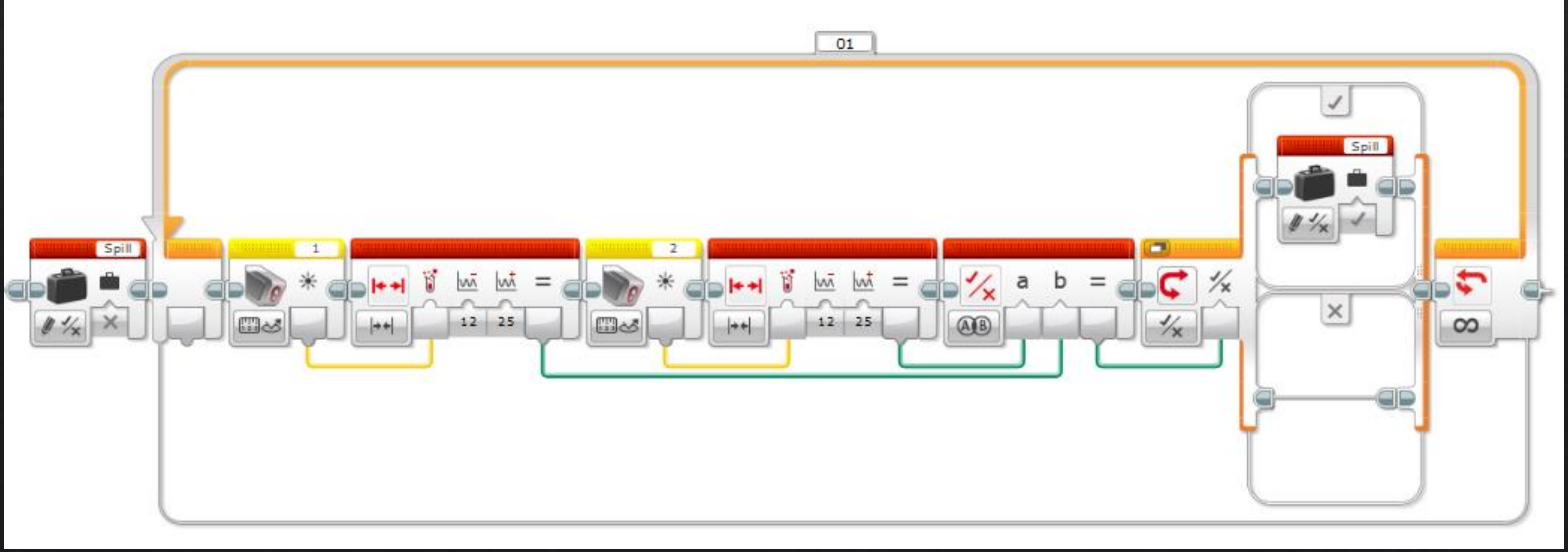
5. Detect the Chemical Spill

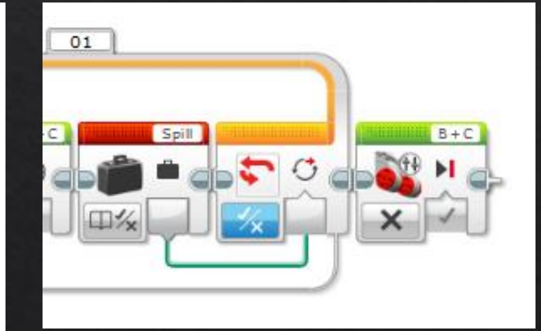
- Whole Program (see previous pages for detailed parts)



5. Detect the Chemical Spill

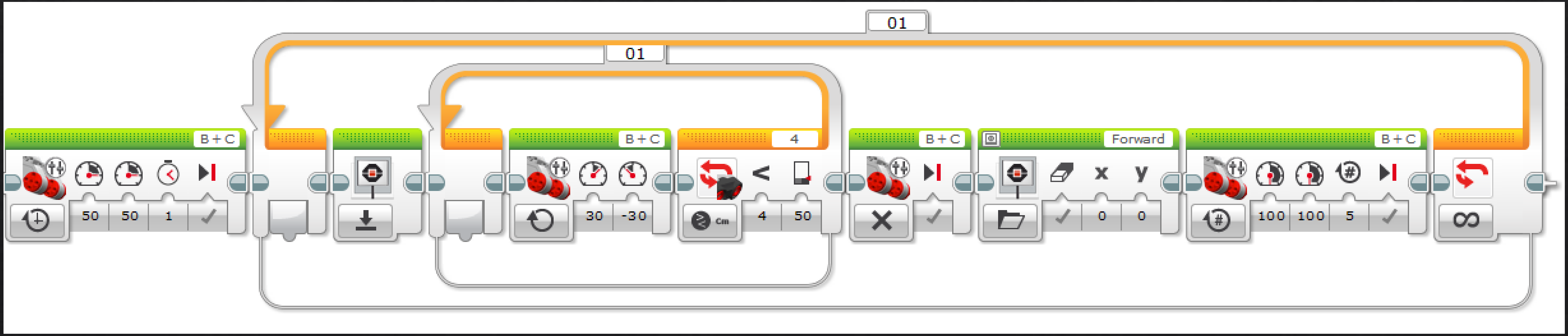
- Two Light Sensors detect GREEN?
- Create a sub-routine to check each sensor is in the green range, use logic AND to return true when BOTH are green, and use this variable result to trigger the end of the loop.





6. Rescue the Canister

- Have a plan to find the canister and rescue it:
- (MY PLAN)
  - Drive to the centre of the spill
  - Turn slowly until my Ultrasonic sensor sees the canister
  - Drive really fast for a long time at the canister



Putting it all Together

- Make sure that you understand the consequences of putting two different programs together:
  - What is happening to the motors?
  - Are the sensors looking or are the motors running for time and/or distance, preventing the sensors from seeing?
  - Are there any unintended consequences?