



© G. Tardiani 2015

# RoboCup Rescue

Workshop

Part 2



**MINDSTORMS**  
EV3



# Completing the Rescue Challenges

- RoboCup rescue requires more than Line Following
- Extra challenges include, Obstacles and rescuing the Victim
- When solving these challenges do them as individual programs and then add them to the main program when they work successfully
- Your robot will need to do multiple things at once in order to follow the course AND be on the look out for Obstacles and Chemical spill etc
- Your robot will need to change modes when it detects the Obstacle
- This process is called Task Swapping or Multitasking



# RoboCup Rescue Problems

- Line Following

- Follow the line
- Detect Intersections
- Navigate Intersections correctly
- Navigate Gridlock
- Detect Chemical Spill

- Obstacle

- Detect the obstacle
- Navigate around the obstacle

- Chemical Spill

- Find the can
- Don't find the contaminated can
- Control/Pick up the can
- Push the can out and release it, or
- Find the Block
- Put the can on the Block
- Find the Exit and then the line

- Possible complications

- Find the can on an extended spill
- Find the exit on an extended spill
- Challenge tile



# Which are **Tasks?** Which are **My Blocks?**

- Main Control Task
- Line Following
  - Follow the line
  - Detect Intersections
  - Navigate Intersections correctly
  - Navigate Gridlock
  - Detect Chemical Spill
- Obstacle
  - Detect the obstacle
  - Navigate around the obstacle

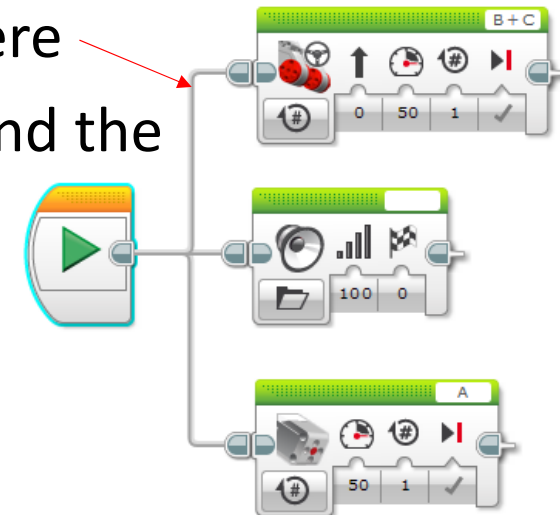
- Chemical Spill
  - Find the can
  - Don't find the contaminated can
  - Control/Pick up the can
  - Push the can out and release it, or
  - Find the Block
  - Put the can on the Block
  - Find the Exit and then the line
- Possible complications
  - Find the can on an extended spill
  - Find the exit on an extended spill
  - Challenge tile





# Doing two things at once???

- RoboCup Rescue requires the robot to be looking for a number of possible events
  - The Touch Sensor to indicate that we have run into the Obstacle, OR
  - The Colour Sensor has detected the Silver at the start of the Chemical Spill, etc.
- We can run multiple tasks simultaneously as shown here
- As the Motors B+C move forward, a sound is playing and the medium motor on port A is waving a flag around?
- Use your mouse to drag new sequence wires from the sequence Start plug to each of the parallel tasks.



# So How do we Start and Stop tasks?

- In RoboCup Rescue we want to Follow the Line and then change the task when we detect the Obstacle. After getting around the Obstacle, we then want the robot to go back to Line Following. How?
- We need to run optional tasks with the main task containing all the programs like, Line Following and getting around the Obstacle etc.
- A Parallel task is constantly looking for the Obstacle etc.
- When it detects the Obstacle, for example, it tells the Main Tasks to stop what they are doing (usually Line Following) and change to the appropriate task, eg Get around the Obstacle.

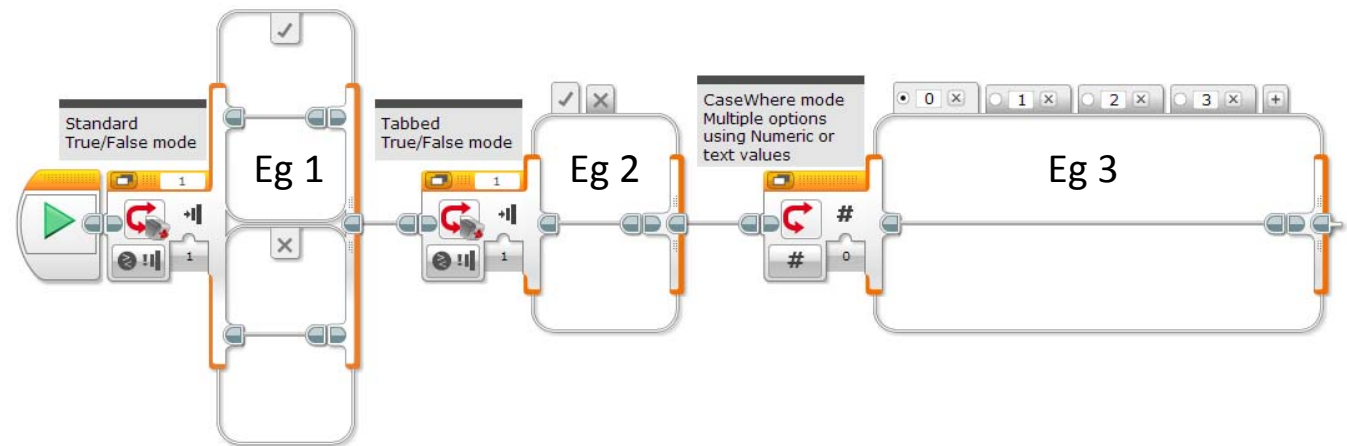




# Switch Block in CaseWhere (tabbed mode)

- The Switch Blocks default mode is True/False (Eg 1)
- It can be changed to a Tabbed view to save space (Eg 2)
- When using Numeric or Text values to control which switch option to use, the Switch Block allows multiple options (Eg 3)
- We can use Variables to control which Switch option is chosen by the robot.

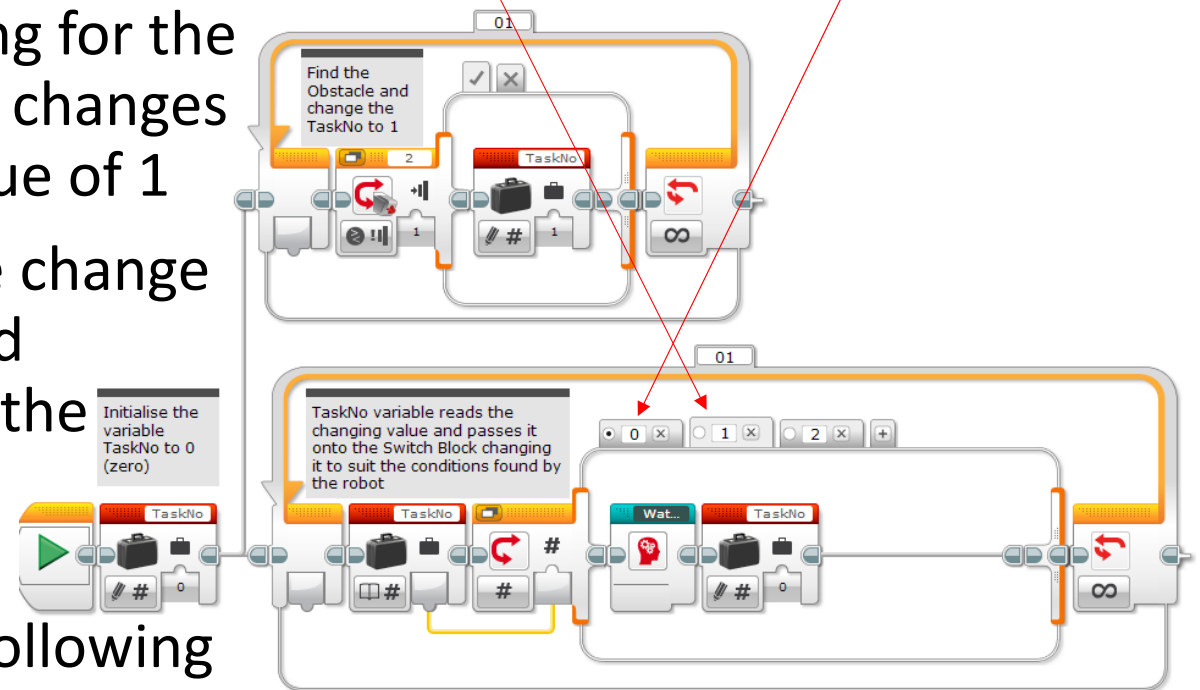
Eg 3 is the key to task swapping in Rescue





# A possible Solution

- The bottom task contains all our programs to LineFollow (task 0) as well as navigate around the Obstacle (task 1) we will need more than 2
- The top task is always looking for the Obstacle. When it finds it, it changes the TaskNo variable to a value of 1
- The bottom task notices the change and stops Line Following and swaps to navigating around the Obstacle.
- Then is sets TaskNo back to 0 (zero) to go back to Line Following





# Variables – Creating and Using them

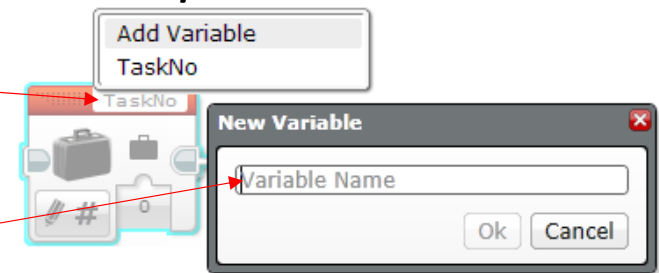


- A Variable is a location on the EV3 Brick that can store a data value
- The Variable block lets you read or write a Variable in your program
- Using the possible solution on the previous slide!
- We will create a Variable and name it TaskNo (Task Number)
- The default TaskNo will be 0 (zero) and will be the Line Follower
- Each detection task eg Obstacle Task will assign their own number
- So when the Obstacle Task detects the 'Bottle', the variable TaskNo will be changed to a value of 1 (one)



# Variables – Creating a new one

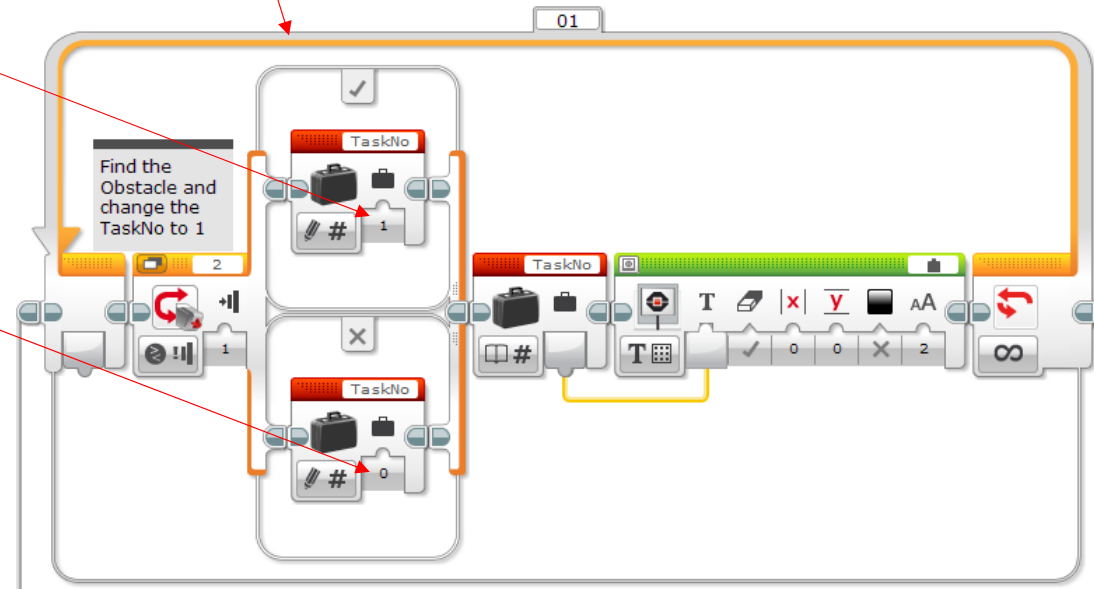
- To create a new Variable firstly, drag and drop a new Variable Block onto your program page. *Red, Data Operations tab*
- Now click in the Variable Selector area
- Select Add Variable and you will then be able to type in the name of your new variable
- We are naming the variable TaskNo
- In Write Mode we can change the value of the TaskNo variable
- In Read Mode the program can use it to make decisions





# Changing the Value of a Variable

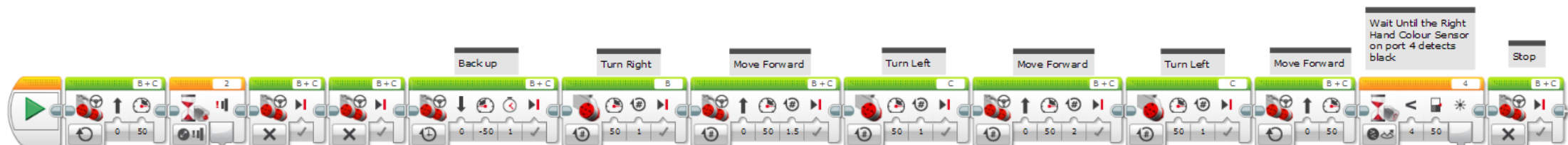
- This is a possible Obstacle Detection task running parallel to the main task
- The Detection task resides inside a Loop so that the robot is constantly looking for the Obstacle (using a Touch Sensor)
- When the Touch Sensor on port 2 is pressed the switch takes the Upper path and changes the value of the TaskNo variable to 1
- IF the Touch Sensor is NOT pressed TaskNo remains on 0
- The extra blocks are so you can see what TaskNo is on the EV3 screen





# Detecting the Obstacle (Water Bottle)

- Ultrasonic sensor is the most popular
  - However, it can detect tile blocks in tunnels, See Saw etc.
  - Mount vertically to narrow the vision of the sensor
- Touch Sensor
  - Reliable as it does not sense alternative obstacles
  - Can also be used to detect the Rescue Capsule (but not find it)

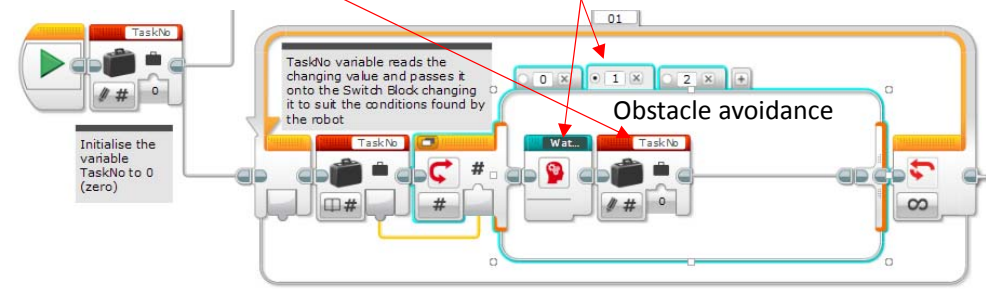
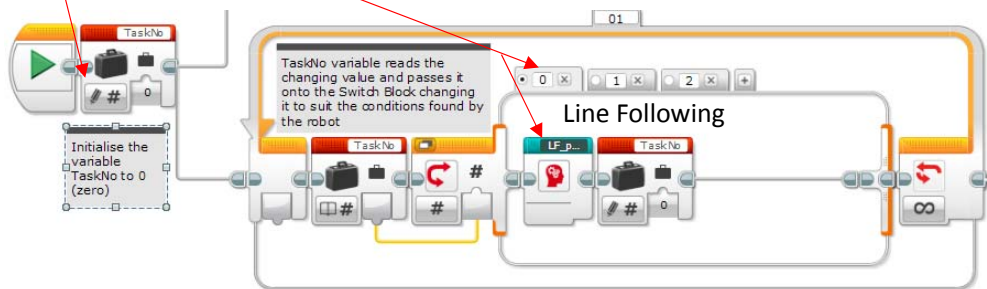


- This possible solution to the Obstacle is a test program. Adjust everything until it works faultlessly, then My Block it and add it to the Main Task



# Swapping Tasks

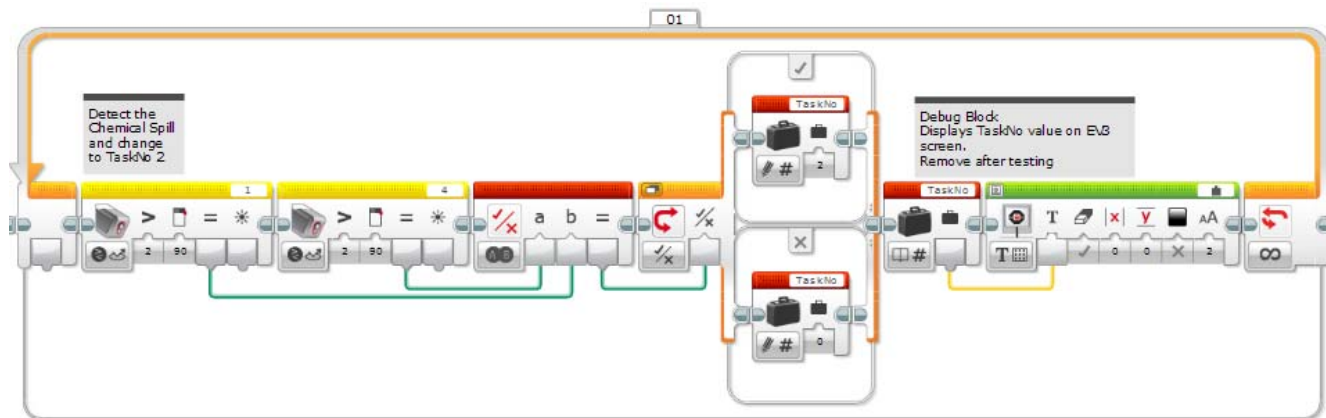
- The MainTask examples below show how the program Initialises the variable TaskNo to 0 (zero)
- The MainTask will then run the LineFollowing program
- The DetectionTask is running above looking for the Obstacle. When the Touch Sensor is pressed it changes TaskNo to 1 (one)
- The MainTask will now swap to task 1 and run the WaterTower program. When the program completes, it sets the task back to 0 (zero)





# Detect the Chemical Spill (Silver foil)

- Need to determine the Light Sensor values for Silver
- Break out of the Line Following Loop by using one of the Light Sensors detecting the silver on the main loop. However, it's difficult to go back to Line Following if needed, OR
- Use the Switch Block in CaseWhere mode to change modes in the Main Task. This method allows for full control at all times



# Find the Rescue Capsule

- Use Ultra Sonic Sensor
  - Most popular amongst teams
  - Try mounting it vertically to narrow the beam
  - Can get false readings
  - Cannot tell the difference between Silver and Black Rescue Capsules
- Light Sensor and small Torch or Light
  - Needs extra batteries for Torch or Light
  - More things to go wrong or forget to turn on
  - Can tell the difference between the silver and black Rescue Capsules.
  - Accurate if programmed well





# Control the Rescue Capsule

- Primary or Secondary
  - Control means the robot could move in any direction and the Rescue Capsule will move with the robot. Note: does not need to pick up.
  - Push or pull the Rescue Capsule out of the Chemical Spill and release it. Secondary Rescue now need to exit the spill and regain the line
- Open
  - Control means a lift to a height that could place the Rescue Capsule onto the platform
  - After placing the capsule on the platform, exit the spill





# Rescue the Capsule Primary/Secondary



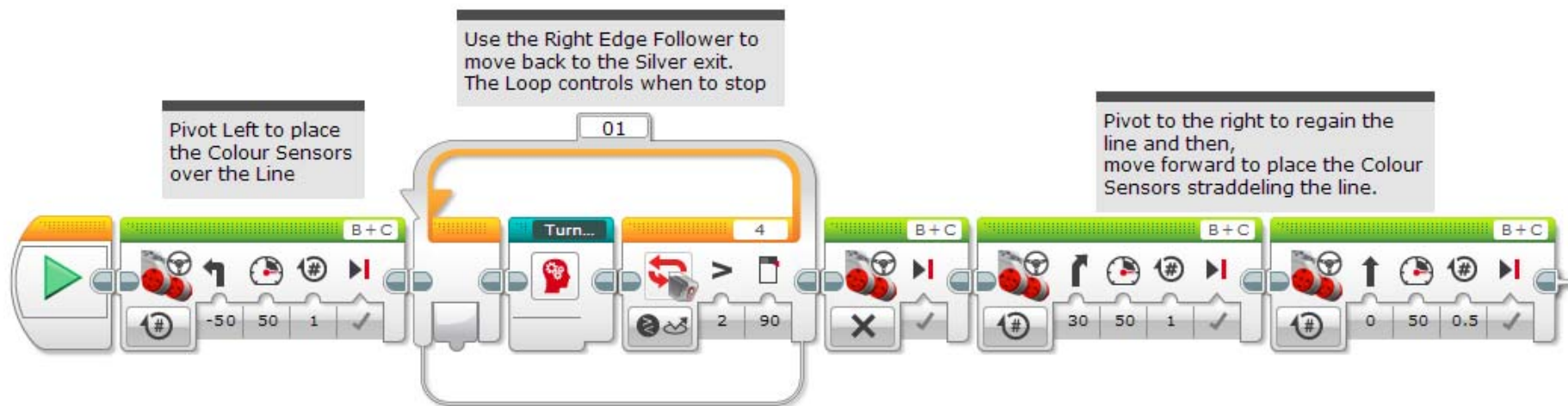
© G. Tardiani 2015

- Once the Capsule is Controlled the robot should only need to move forward to save it from the toxic chemicals.
- Primary robots can simply move off the Chemical Spill tile
- Secondary robots need to use their Colour Sensors to detect the edge of the Chemical Spill and then STOP.
  - Ensure that the design of your robot has the Rescue Capsule completely out of the Chemical Spill when the Colour Sensors detect the edge of the spill
- Once the Rescue Capsule is rescued the robot must release it to gain rescue points
- Note: Primary robots simply need to push the Capsule out



# Exit the Chemical Spill Secondary

- As the Robot is poised on the edge of the Chemical Spill it simply needs to pivot either left or right and then follow the edge of the Chemical Spill back to the Silver spill indicator. Don't fall of the edge.
- Finally, the robot needs to move over the line and start following it to get he additional 10 points for exiting the Chemical Spill



# Strategy Secondary/Open

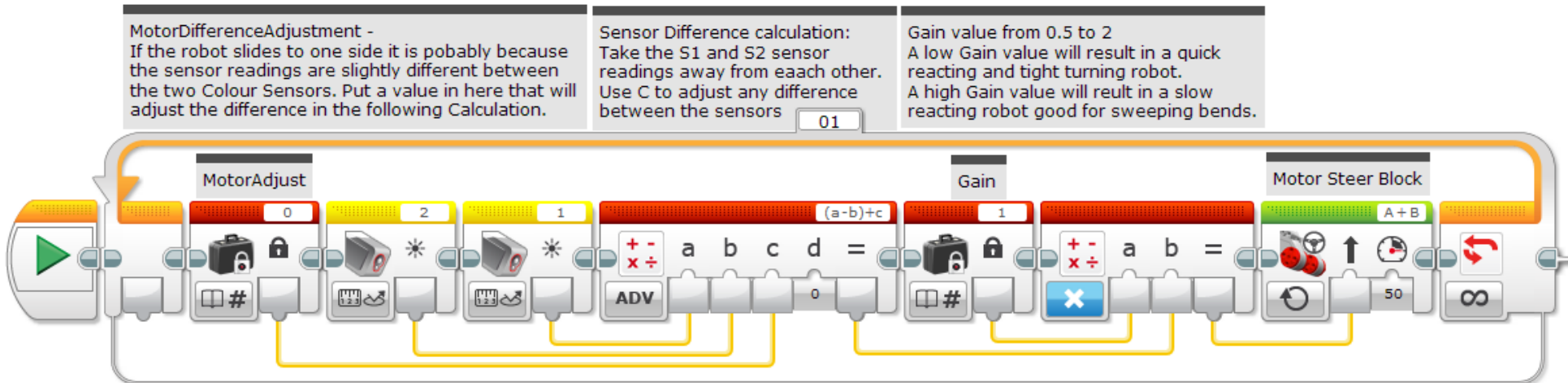
- Consider using your Drop Zone as your starting point
- Rescue the victim and then exit the spill
- Navigate the rest of the course, in reverse order, to the City Limits start tile to gain full points and stop the clock
  - This strategy will maximise a robots points quickly by rescuing the victim early
  - Works well if you know your robot will have trouble with a particular tile
  - Does require the robot to traverse the last two tiles twice





# PID Line Following

- Stands for Proportional, Integral, Differential Control System
- It is used to smooth out the reactions the robot makes when moving
- Used in Line Following – it smooths out the wobble
- Below is a very simple Proportional solution using 2 sensors
- Note: this is just the P bit of PID. *See Part 3 for more details*



# 3<sup>rd</sup> Party Sensors and Actuators

- MTA – Everything Lego Education  
<http://www.teaching.com.au>
- Omni Wheels – RotaCaster designed for Lego  
<http://www.rotacaster.com.au/>
- Linear Actuators – Firgelli have NXT and EV3 models  
<http://www.firgelli.com>
- HiTechnic – Official 3<sup>rd</sup> party Lego Sensors  
<http://www.hitechnic.com/>
- MindSensor – Unofficial 3<sup>rd</sup> party Lego Sensors  
<http://www.mindsensors.com/>
- Dexter Industries – Advanced sensors  
<http://www.dexterindustries.com>



© G. Tardiani 2015

