

# Button Sequence Program

*FLL*® Programming

# Prerequisites

- Basic computer knowledge.
- A fundamental understanding of EV3 Mindstorms® software.
- Familiar with creating My Blocks in the EV3 Mindstorms® software.

## Recommended

- An EV3 Mindstorms® Brick or robot for testing.



# Master Sequencing programs overview



- A “master sequence program” combines all missions in the desired run order into a single program.
- This reduces time spent in Base by not requiring technicians to navigate, find and select the next program to run.

Most *FLL*<sup>®</sup> teams create separate programs for missions (or trips) out of Base. These take time to find and run.  
***Time in Base is time not scoring.***

# Types of Sequence Programs

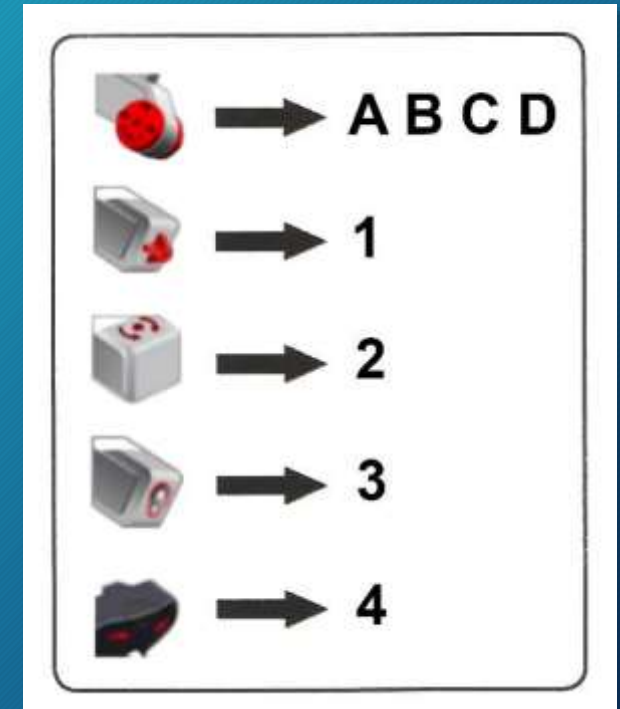
There are many types of sequencers, four are:

- Basic Loop/Switch Sequencer
- Button Sequencer
- Color Scan Sequencer
- Numeric Sequencer



# Programming Assumptions

- Using the Gyro to navigate.
- Using the default port for motors and sensors.
  - If your robot uses other than default ports you must change the port setting on the every programming blocks.



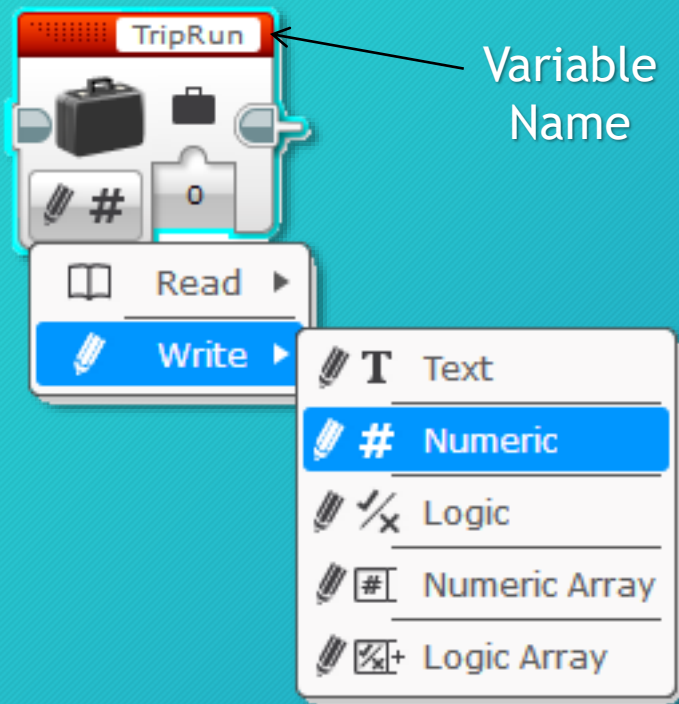
EV3 Default Ports

# Getting Started

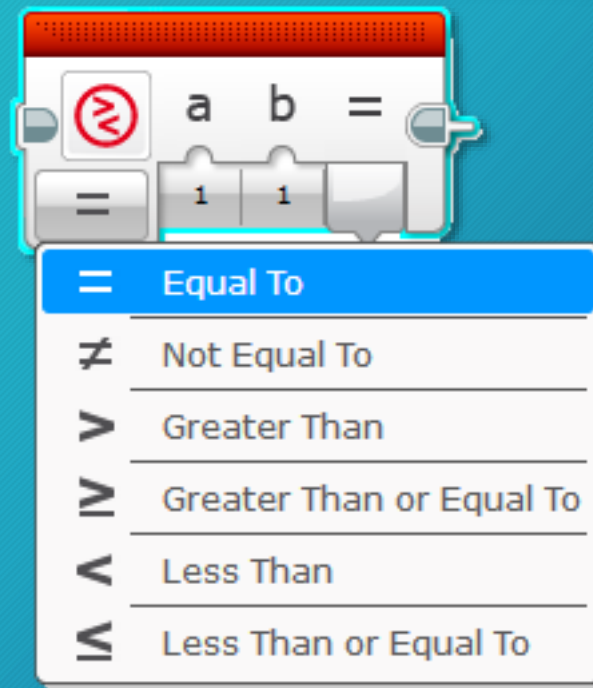
General items



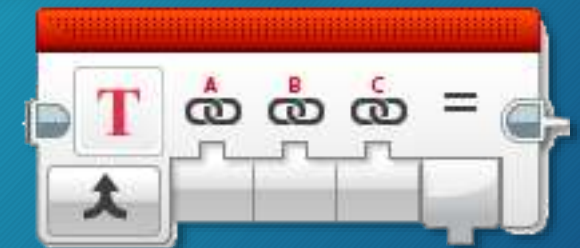
# Variable, Compare and Text blocks overview



Variable blocks write and read data based on the settings



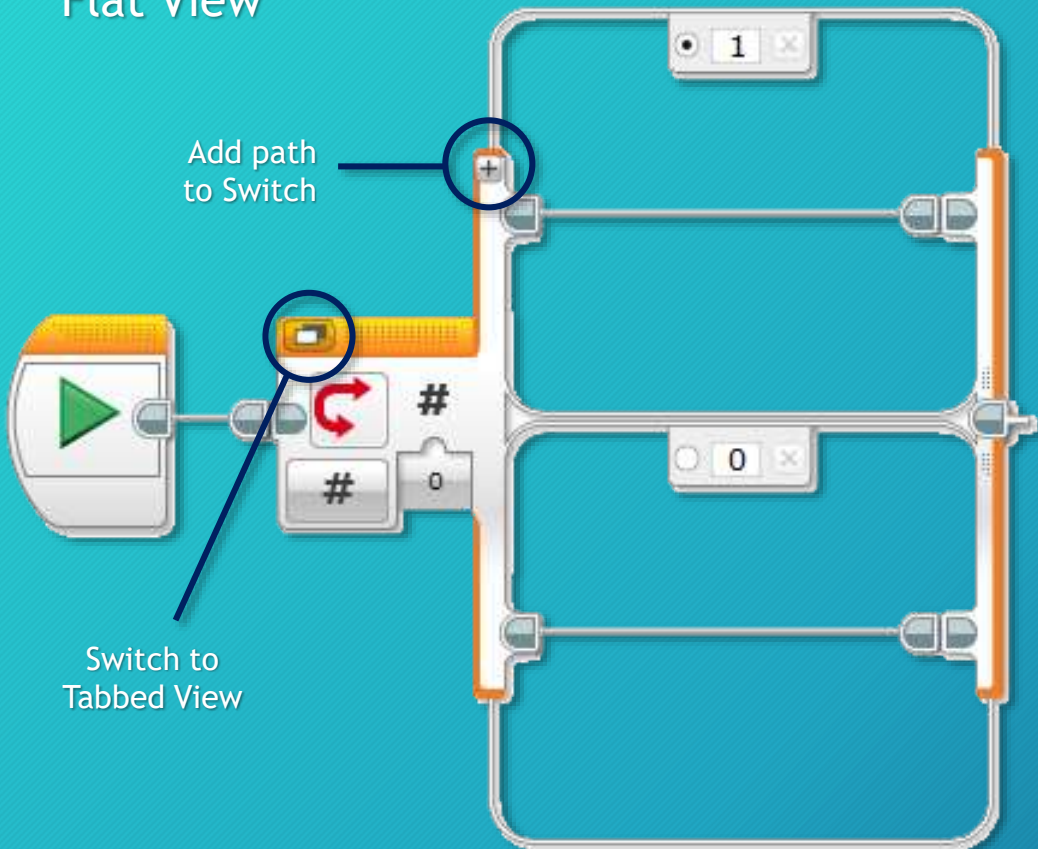
Compare block compares two numbers and returns true or false based on the settings



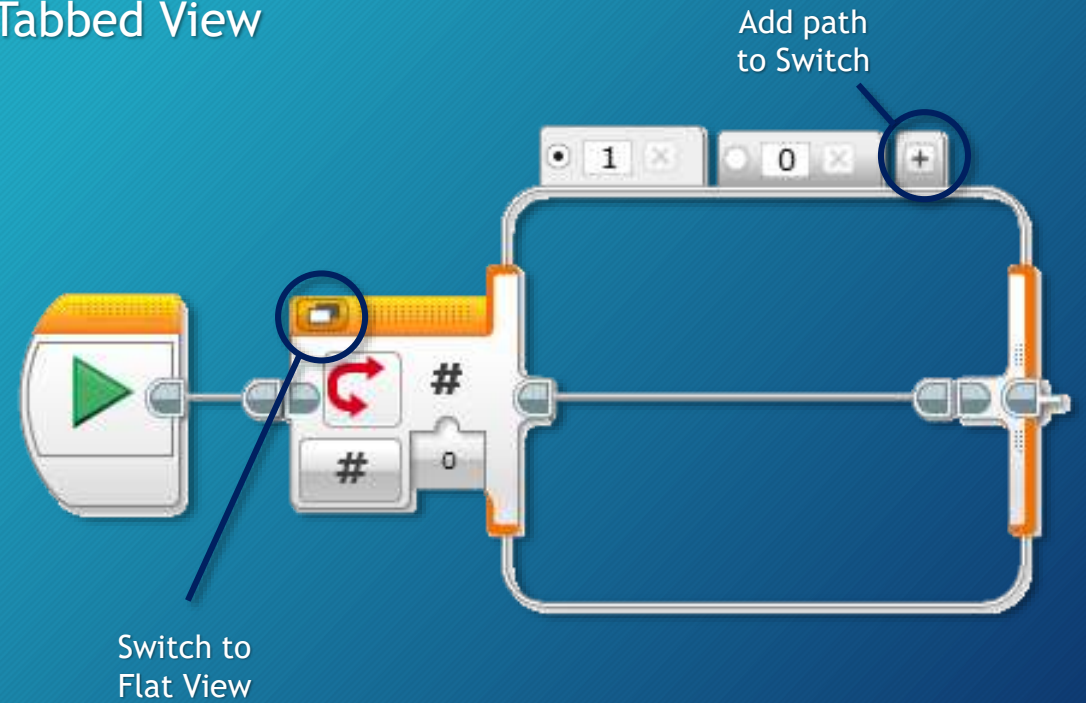
Receives multiple text inputs and combines them

# Switch block basics

Flat View



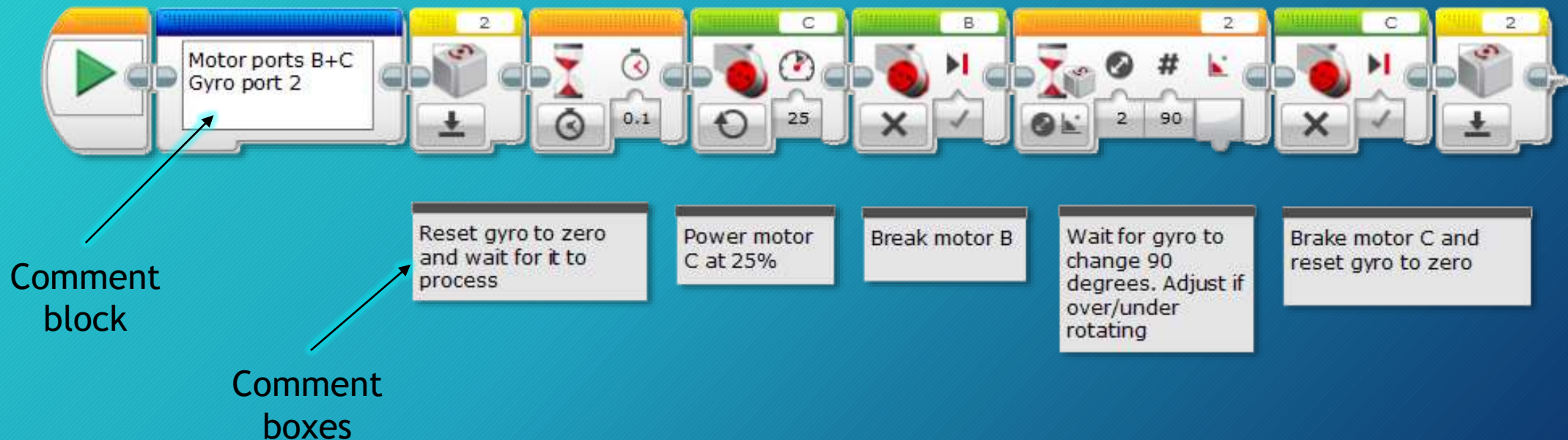
Tabbed View





# Adding comments to program overview

- Properly commenting your programs not only helps others to understand what the program, but helps you to remember what you did and to locate sections of the program quickly.



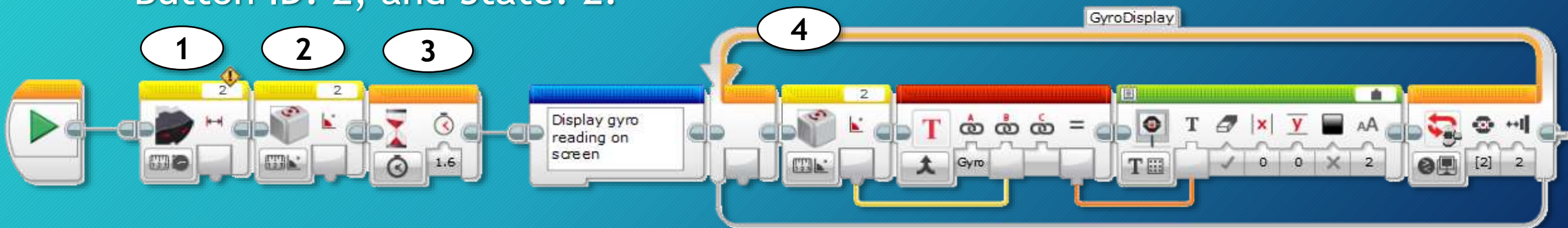
# GyroCal My Block

- The gyro calibrate (GyroCal) My Block helps to reduce gyro drift.
- This version works with new and old models of EV3 gyros.
- The gyro calibrate only needs ran once at the beginning. Robot **must** be sitting on the surface it will be running on and **cannot** be moving during the reset.
- The GyroCal My Block displays the gyro reading on EV3 screen so you can check for drift. If still drifting rerun the program.



# GyroCal My Block

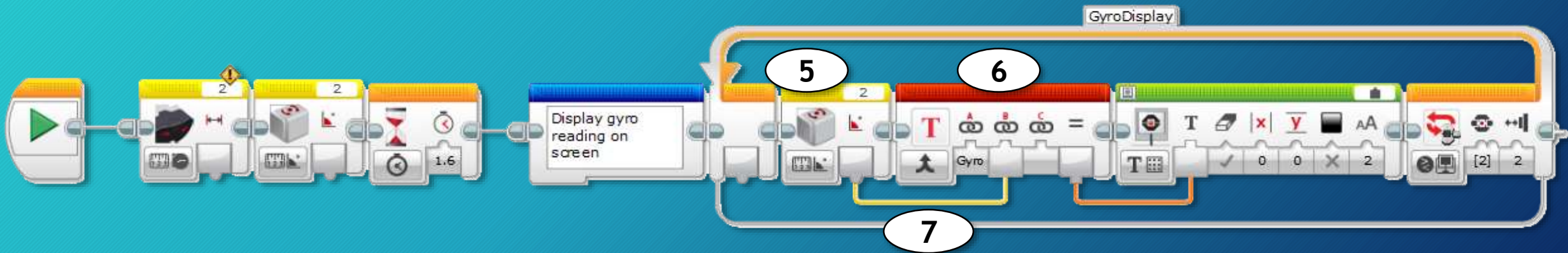
1. Insert an Infrared Sensor block set to Measure → Proximity and the same port (port 2) as your Gyro.
2. Insert a Gyro sensor block set to port 2, and Measure → Angle.
3. Insert a Wait block set to Time and 1 second.
4. Insert a Loop block set to Brick Buttons → Compare, Brick Button ID: 2, and State: 2.



The EV3 Brick indicates an error on the Infrared Sensor block. This is correct and required for the program to work.

# GyroCal My Block

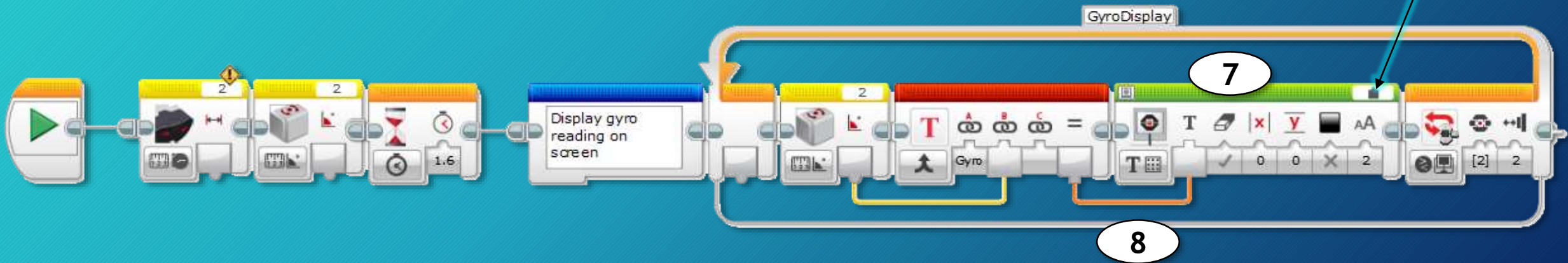
5. Inside the Loop block, insert a Gyro sensor block set to port 2 and Measure → Angle.
6. Insert a Text block. In A, type “Gyro at ” include a space.
7. Drag a wire from the Gyro sensor block output to the B input of the Text block.



# GyroCal My Block

7. Insert a Display block set to Wired, Clear Screen: true,  $|x|$  to 1,  $\overline{y}$  to 1 and Font: 2.
8. Connect a wire from the Text block output (=) to the Display Text input.
9. Convert to My Block with no parameters.

Select here to set to Wired



# Button Sequencer

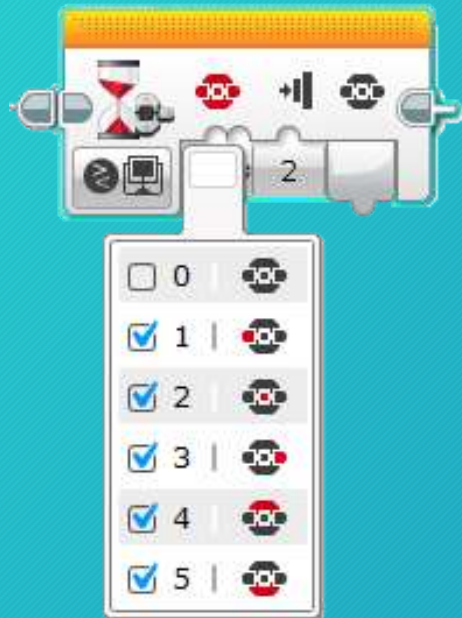


# Button sequencer

- A Button Switch sequencer uses the EV3 Brick buttons to select and run a program.
- Each program (or missions run) is assigned to a EV3 Brick button.
- Programs can be selected and ran in any order by selecting the button it is assigned to.



# EV3 Button Assignments



The button is highlighted in red.



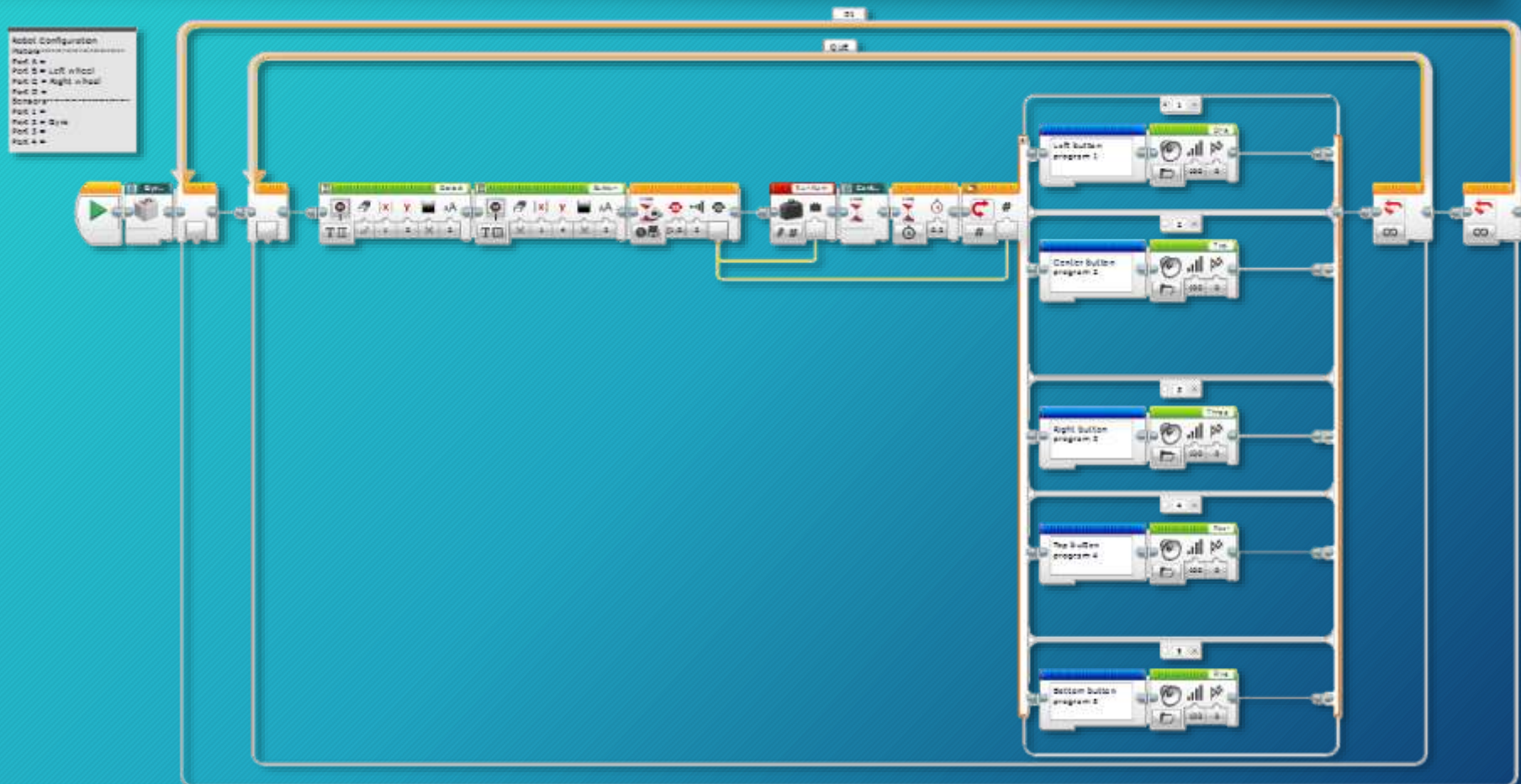
Button output number:

- Left button = [1]
- Center button = [2]
- Right button = [3]
- Top button = [4]
- Bottom button = [5]

The button output number equals the tab (path) number.

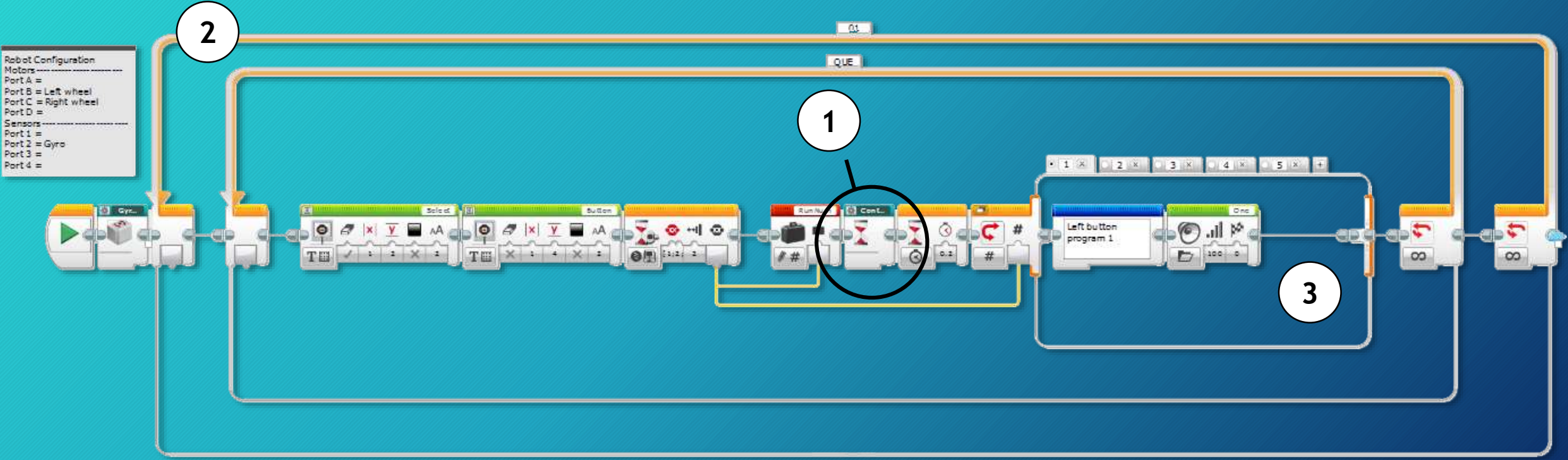


# Button sequencer overview



Flat View

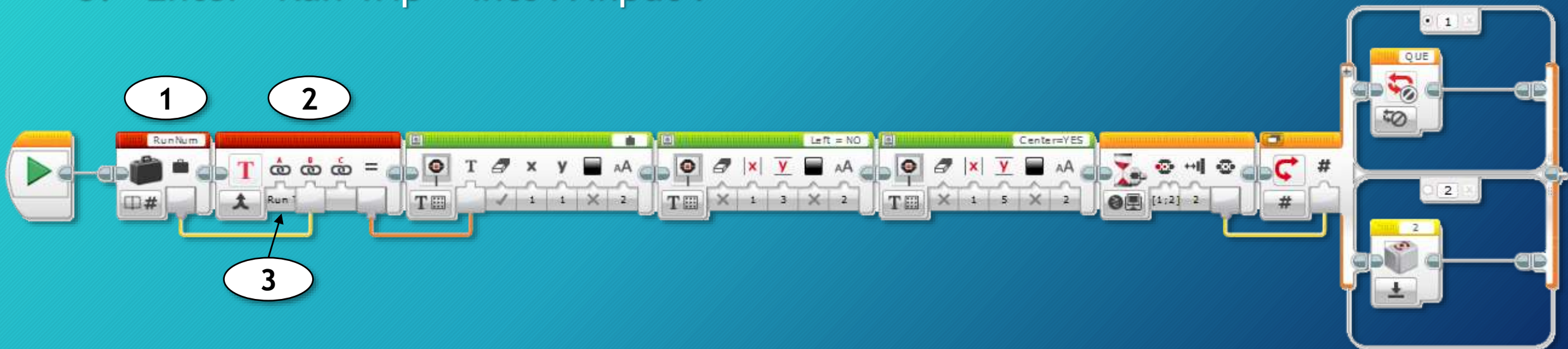
# Button sequencer build sections



Tabbed View

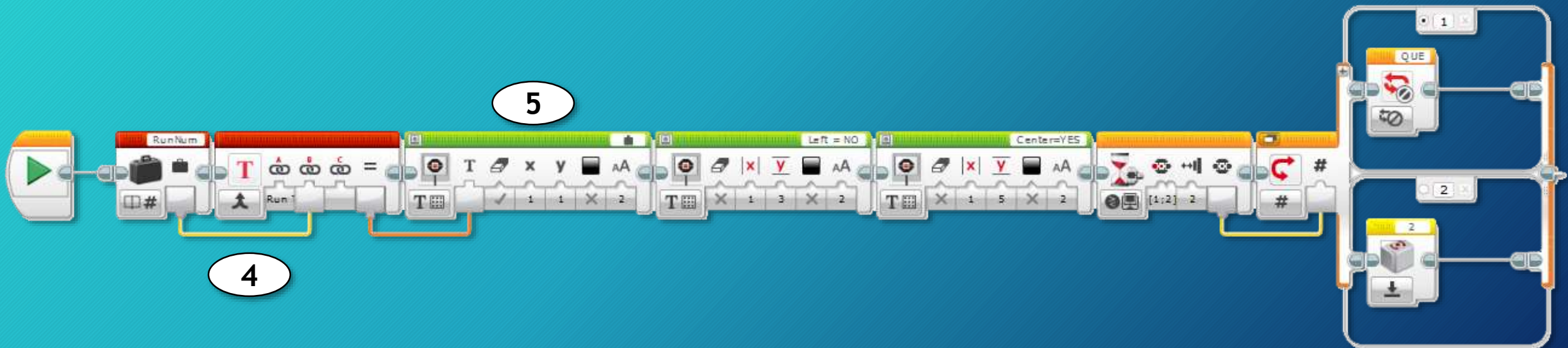
# Build ①: Continue My Block

1. Insert a Variable block named RunNum set to Read - Numeric.
2. Insert Text block set to Merge.
3. Enter “Run Trip ” into A input .



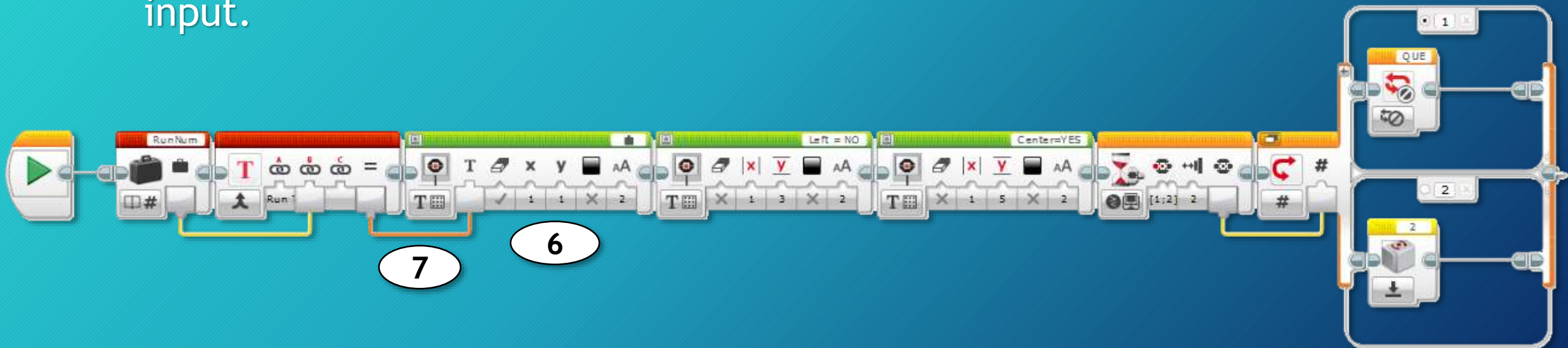
# Build ①: Continue My Block

4. Drag a wire from the Variable block output to the Text block input.
5. Insert a Display block set to Text - Pixels and Wired.



# Build 1: Continue My Block

6. Set Clear Screen to True, x to 1, y to 1, Color to False, and Font to 2.
7. Drag a wire from the Text block output to the Display block input.



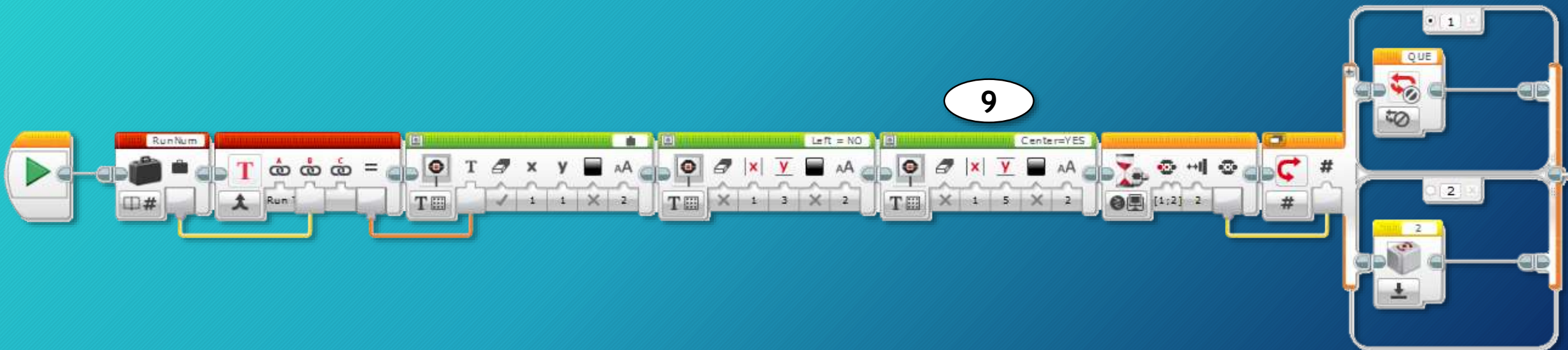
# Build 1: Continue My Block

8. Insert a second Display block set to Text - Grid, Clear screen to False, x to 1, y to 3, Color to False, and Font to 2. Enter "Left = NO" in the text box.



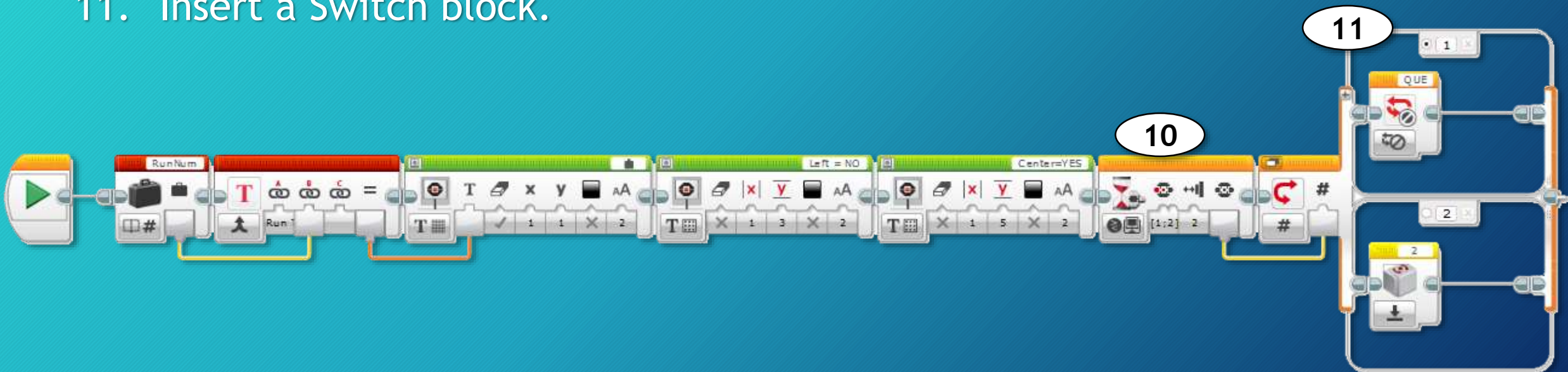
# Build 1: Continue My Block

9. Insert a third Display block set to Text - Grid, Clear screen to False, x to 1, y to 5, Color to False, and Font to 2. Enter "Center = YES" in the text box.



# Build 1: Continue My Block

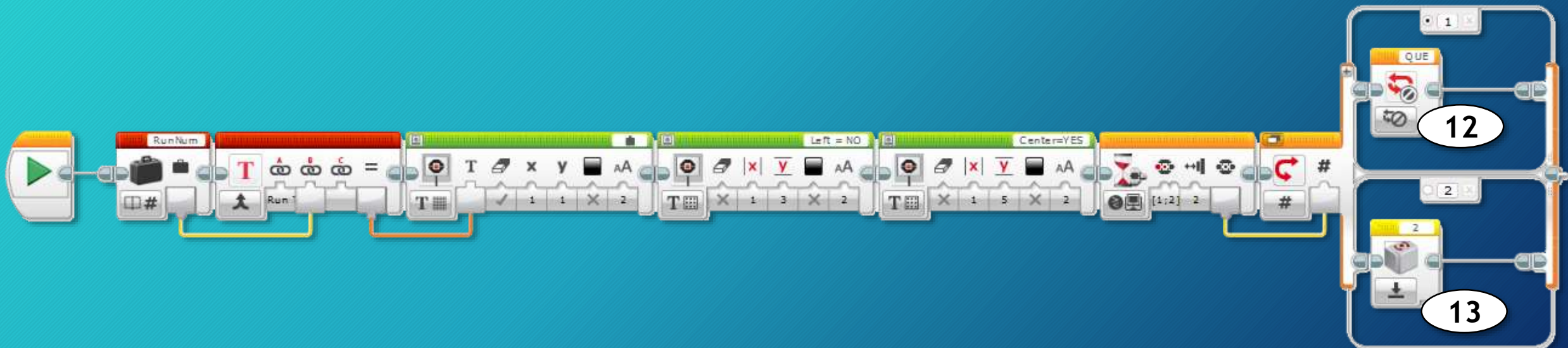
10. Insert a Wait block set to Brick Buttons - Compare, the Set of Brick button IDs to [2] and State to 2.
11. Insert a Switch block.





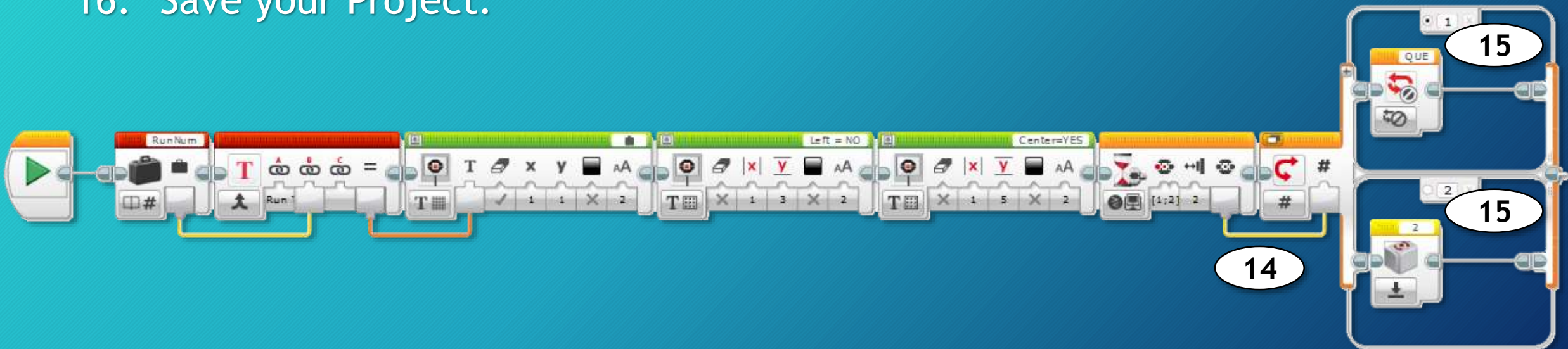
# Build 1: Continue My Block

12. In path 1 insert a Loop Interrupt block set to numeric and set to QUE.
13. In path 2 insert a Gyro block set to Reset and port 2.



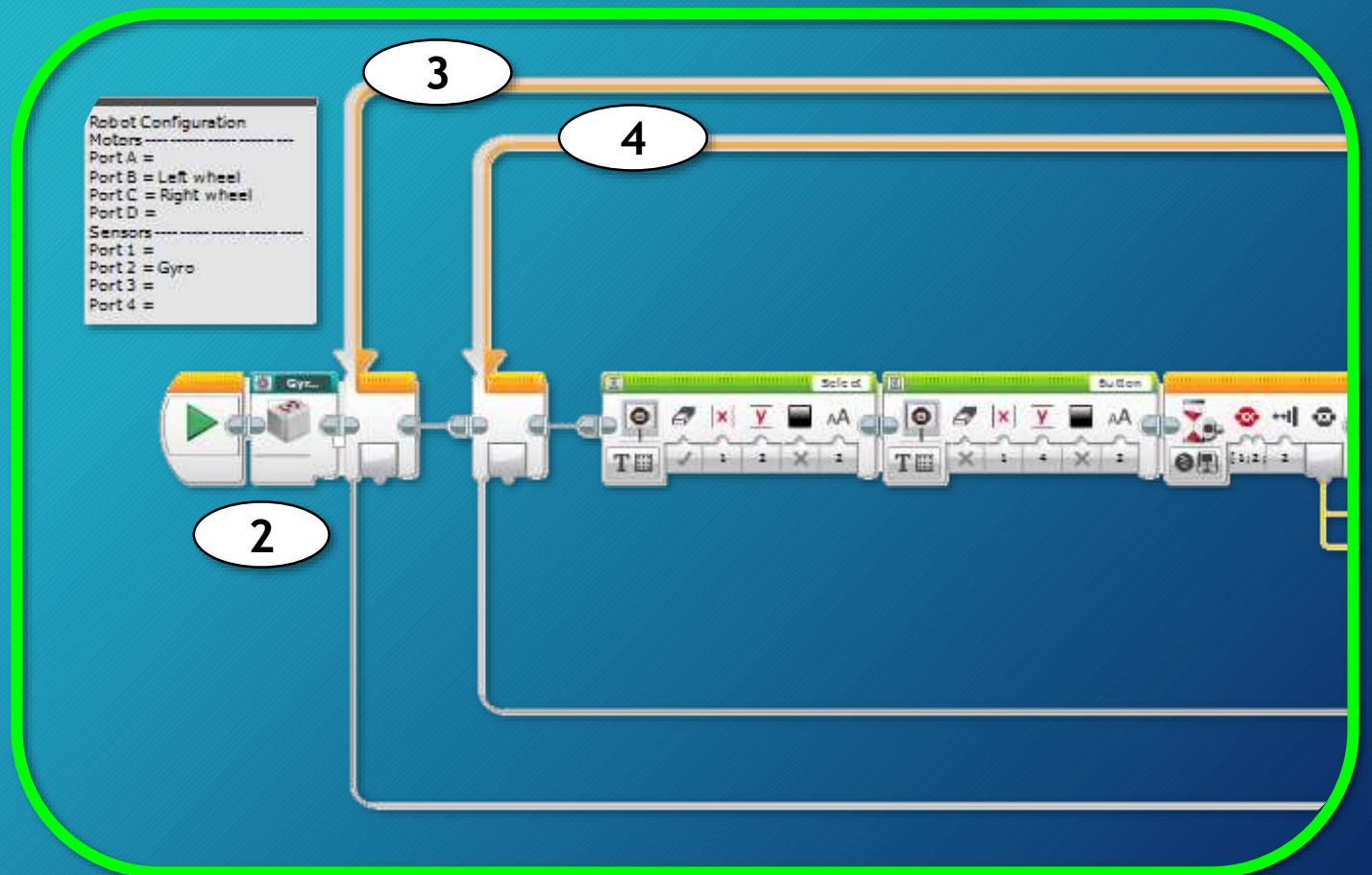
# Build 1: Continue My Block

14. Drag a wire from the Wait block output to the Loop input.
15. Change the Switch path numbers to 1 and 2.
16. Save your Project.



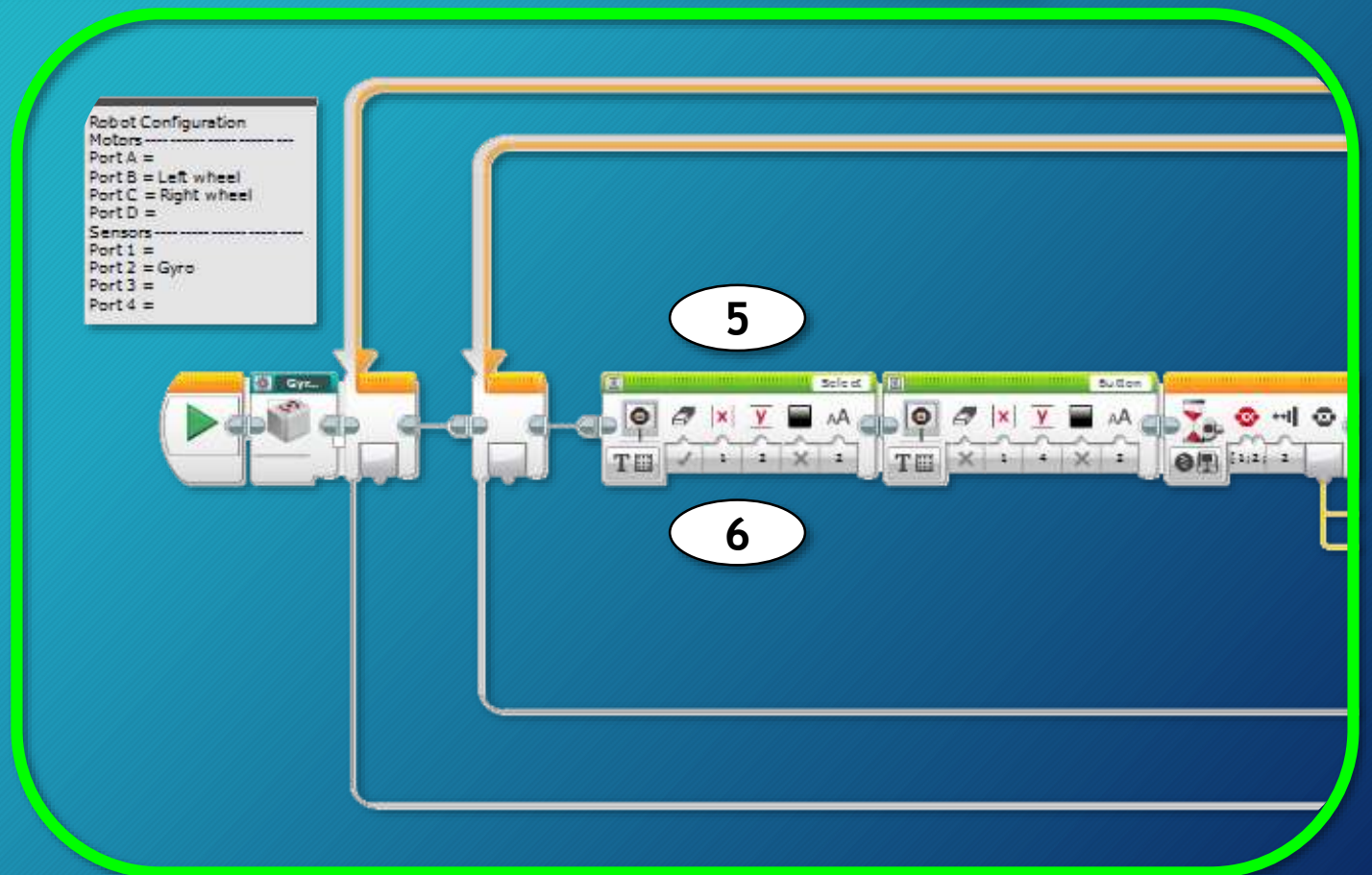
# Build ②: Sequencer

1. Open, name and save a New Program.
2. Insert a GyroCal My Block.
3. Insert a Loop block set to Infinite.
4. Inside the first, insert a second Loop block set to Infinite named QUE.



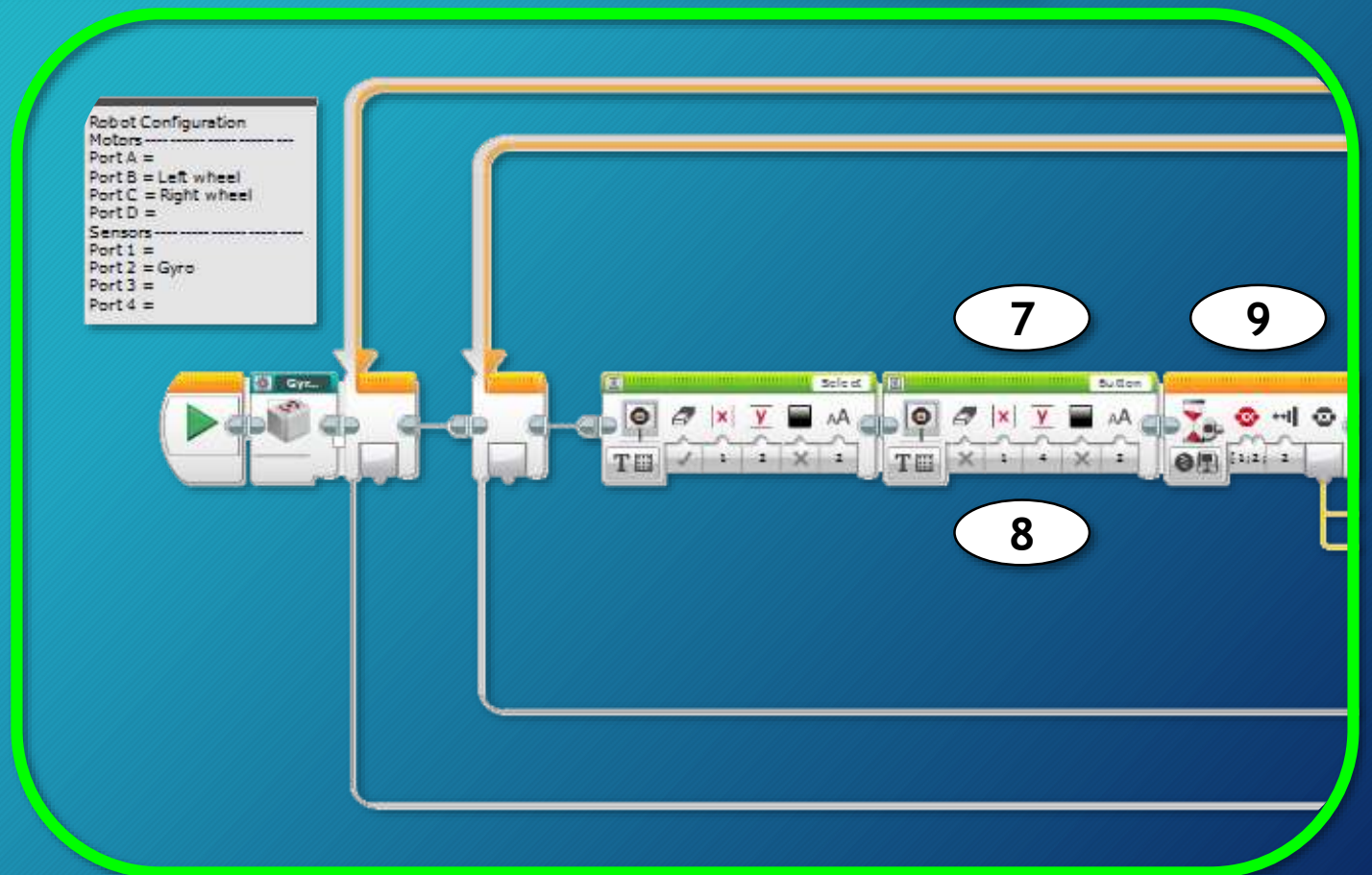
## Build ②: Sequencer

5. Inside the second Loop, insert a Display block. Set Clear Screen to True,  $|x|$  to 1,  $\overline{y}$  to 2, color to false,  $\overline{}$  and text size to 1.
6. Enter “Select” in the Display block text block.



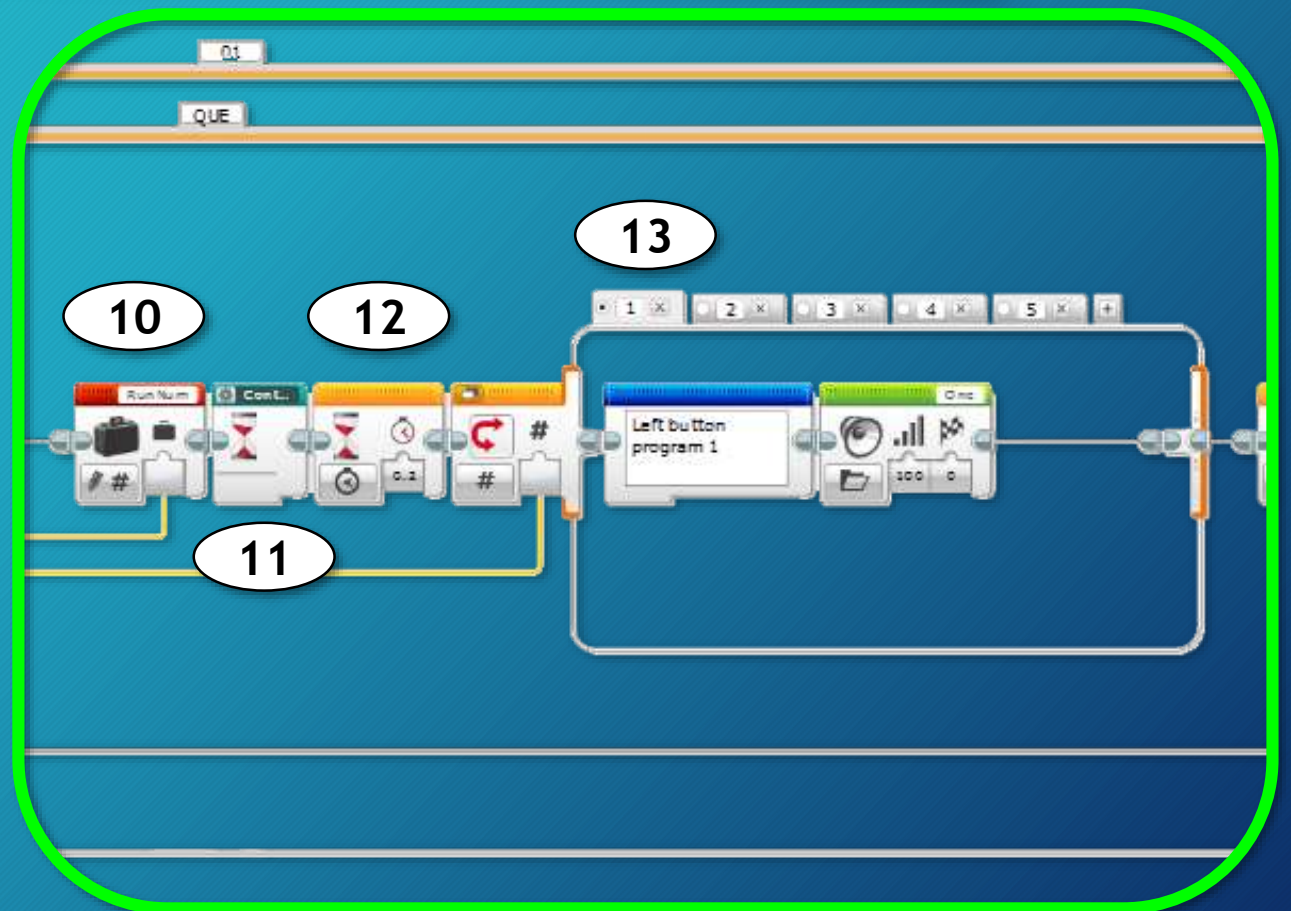
## Build ②: Sequencer

7. Insert a second Display block. Set Clear Screen to True, |x| to 1,  $\bar{y}$  to 4, color to false,  $\bar{a}$  and text size to 1.
8. Enter “Button” in the Display block text block.
9. Insert a Wait block set to Button - Compare. Select all buttons [1, 2, 3, 4, 5].



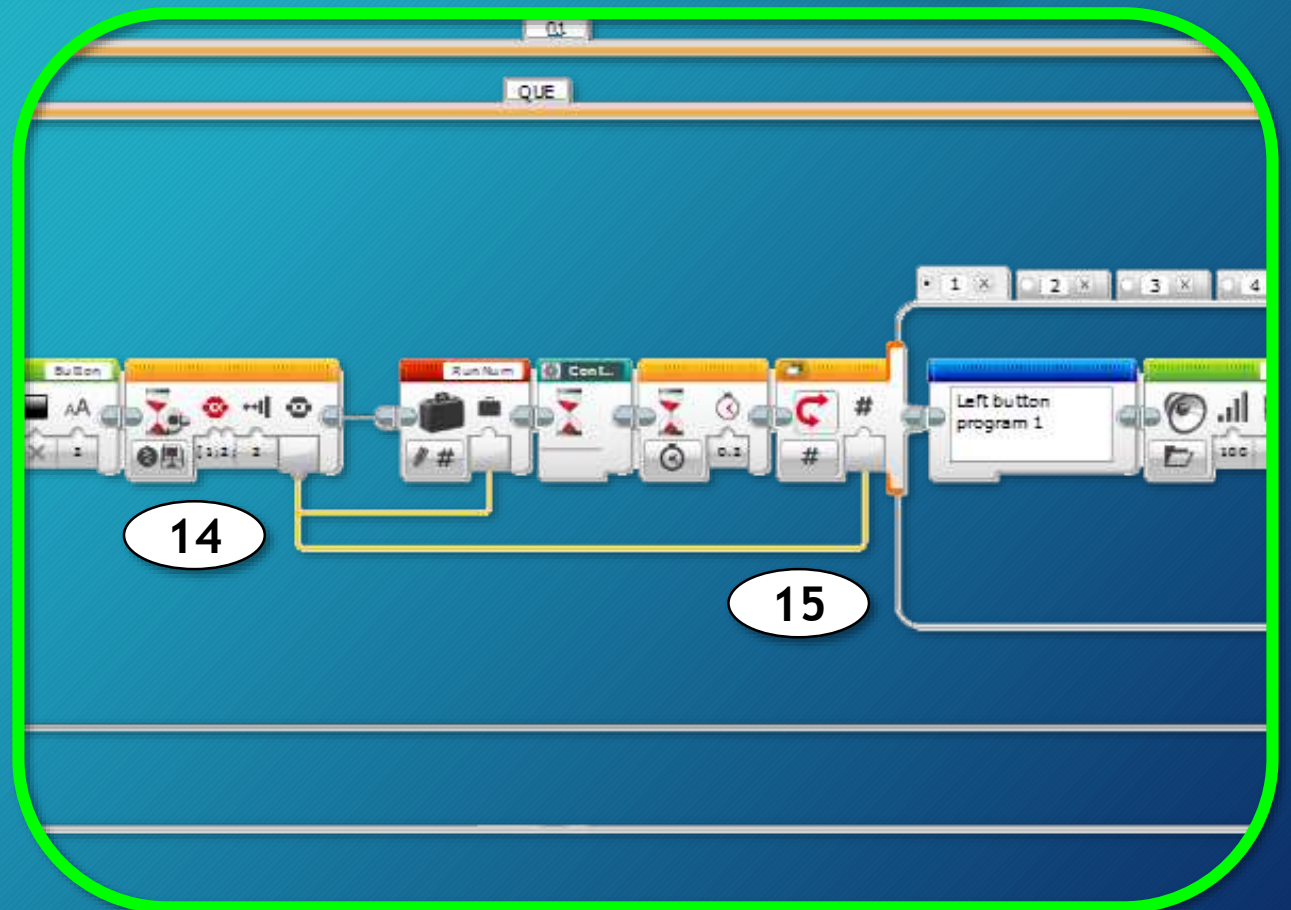
## Build ②: Sequencer

10. Insert a Variable block set to Write - Numeric.
11. Insert a Continue My Block.
12. Insert a Wait block set to 0.2 seconds.
13. Insert a Switch block set to numeric, add three more paths (5 total) and number the paths 1 -5.



## Build ②: Sequencer

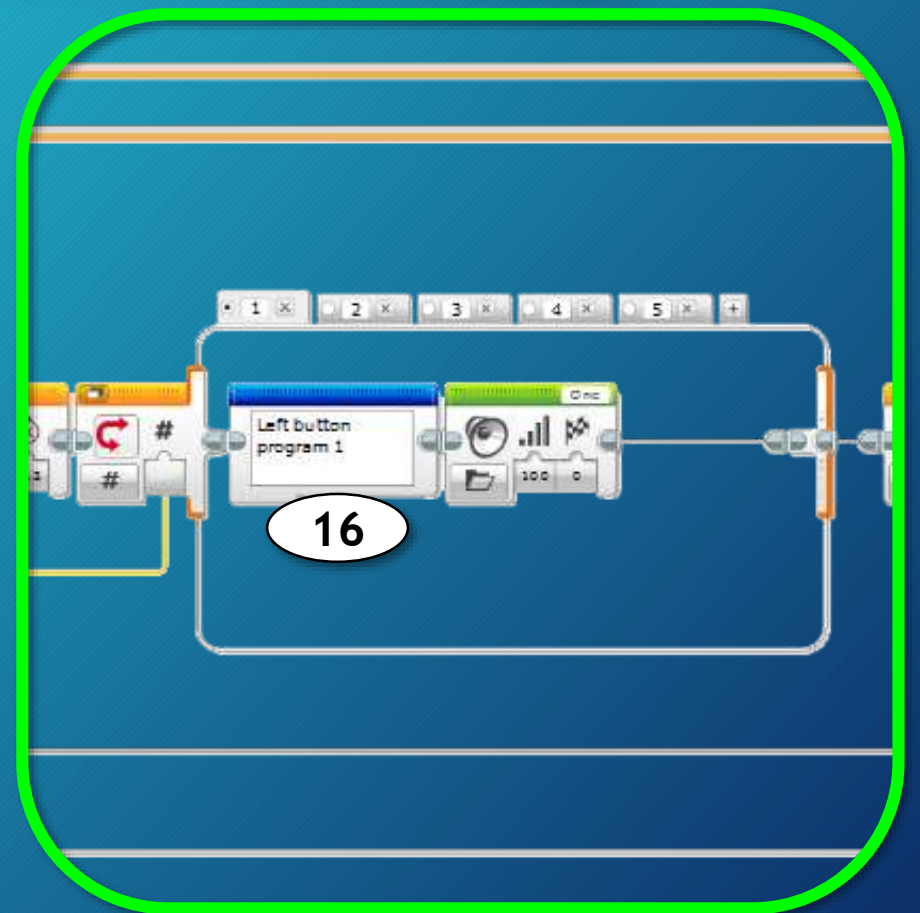
14. Drag a wire from the Wait block output to the Variable block input.
15. Drag a second wire from the Wait block output to the Loop input.



## Build ③: Mission area

16. Insert a Comment block to identify each Switch path for the button that runs it.

- Left button [1] = Tab (path) #1
- Center button [2] = Tab (path) #2
- Right button [3] = Tab (path) #3
- Top button [4] = Tab (path) #4
- Bottom button [5] = Tab (path) #5

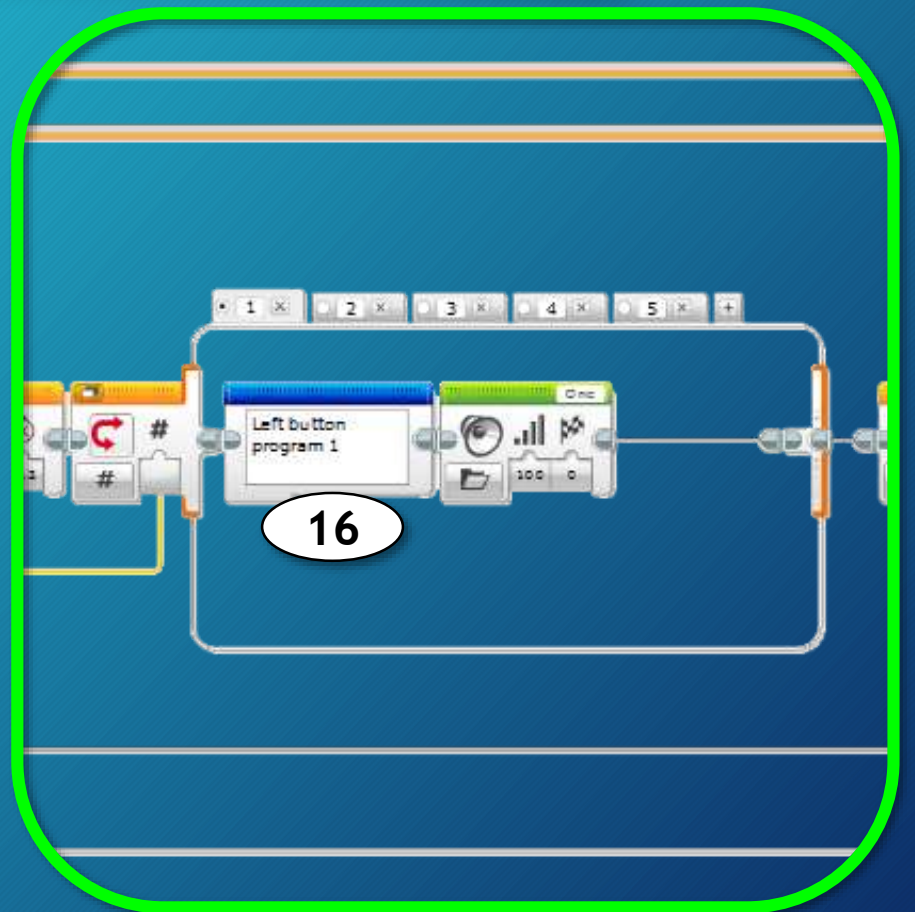




## Build ③: Mission area

17. To test insert a Sound block to identify each Switch path for the button that runs it.
18. Save your Project.
19. Test your Program.

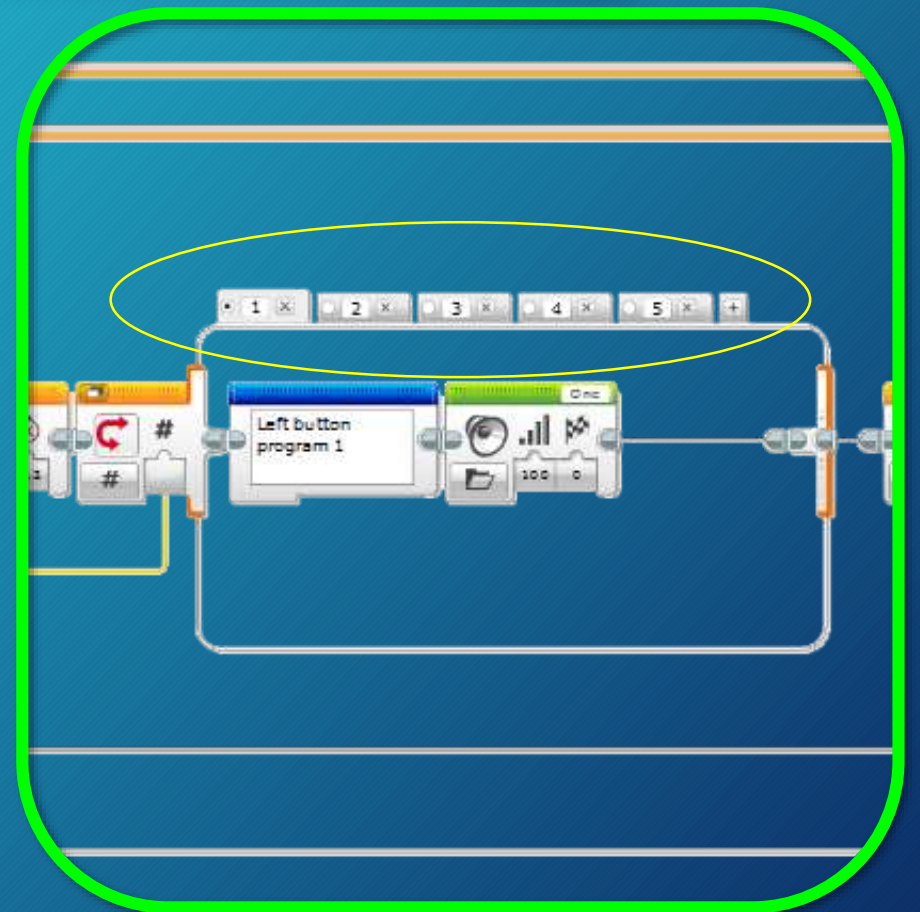
Remove Sounds blocks before programming.



# Sequencer - Tips and Tricks

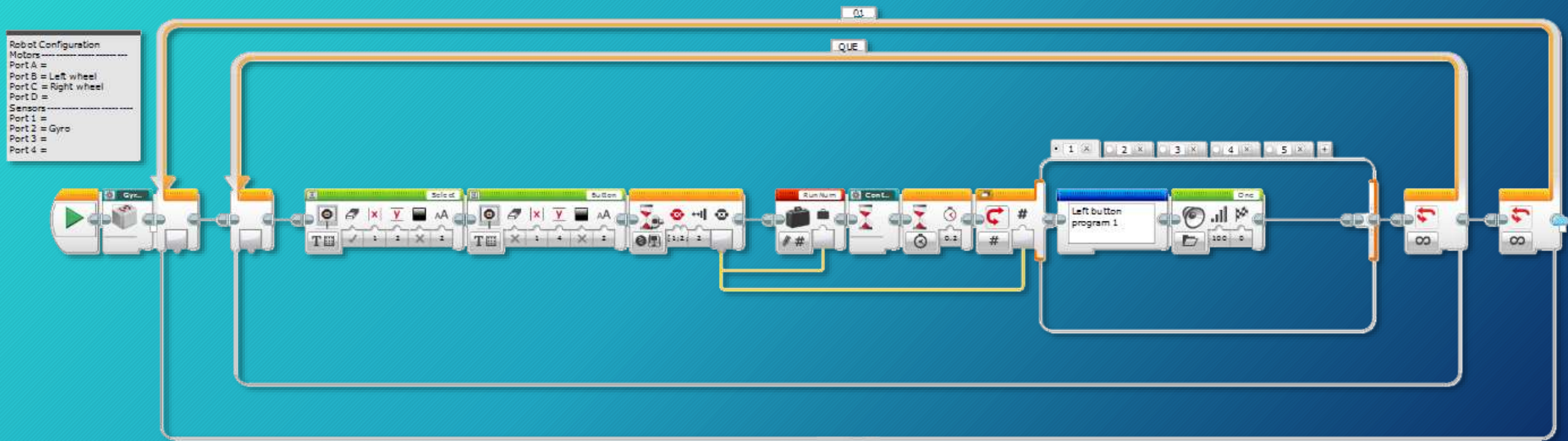
If you would like to change the button a program is assigned to, simply change the path number to that button number.

- Paths do not need to be in order.
- No paths can have the same number.
- All paths must be numbered.



# Button sequencer overview

Use this overview to check your program.



Tabbed View

Everything is  
awesome!

- Emmet Brickowski



Corrections and Comments: [james.brodnick@gmail.com](mailto:james.brodnick@gmail.com)