Lesson 12: Lesson Plan Logic and Control

Vocabulary

AND Statement - a statement that is true if both of its parts are true
OR Statement - a statement that is true when one or both of its parts are true
Ladder Logic - a relay logic diagram that uses ladder diagrams to represent logic operations
Truth Table - a table with the outputs from all possible combinations of the inputs
Switch - a control mechanism that allows a variable's value to change the flow of a program

Direct Instruction

Engage with Conditional Statements

Open the Engage Slide deck. Show Slide 2 and say, "an AND statement will only provide a true output if both conditions are true. Thus, you can only have ice cream if both A and B are true; you have both money, and the ice cream truck is arriving shortly." Show slide 3 and say, "An OR statement will provide a true output if either one of the input conditions is true."

Ask students what situations may allow a person to have ice cream only if EITHER A or B were true. Some possible answers are that A is true, but the person is at an ice cream shop. Another possibility is that B is true, but the person has a coupon rather than money. Show Slide 4 and reinforce the idea that the AND statement requires A and B to be true and the OR statement only requires A or B to be true for the person to have ice cream.

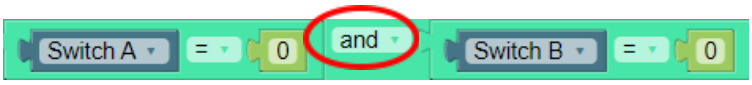
Show slide 5. Say, "This is a truth table; it shows all the possible combinations of A and B being either true or false with the end output. The third column shows the output for the AND statement, and the fourth column shows the output for the OR statement." Slide 6 puts it all together to reinforce the connection between the states and the end result.

Show slides 7 and 8. These slides show the AND and OR statements in a different light. Slide 9 has both circuits on the same page. Direct students to the Student Activity Sheet Engage section to complete the truth table for these two circuits.

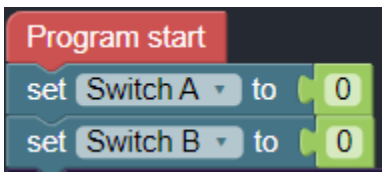
Next, show the Lesson 12 video.

Explore behavior of Switches and Conditional Statements in Ozobot Editor

Open up the **ExploreL12Code** (Share Code: s7ge4wf) and display it so students can view it. Place an Evo in the center of A1. Have students look at the green code blocks and note the AND/OR status of each block.



Direct students to the Explore section of the Student Activity Sheet and have students predict what will happen when the switches at the top are set to different variables.



After students have written down their predictions, run the code (A=0, B=0).

After it is run the first time, give students a moment to fill in their observations. Then, change the values of the set Switch blocks at the top of the code to A=0, B=1, and run the code. Repeat this process with the switch variables of A=1, B=0, A=1, B=1 (all combinations of A and B, 0 to 1). Ensure students have filled out their observation column on their student activity sheets and repeat any A/B switch combination program runs as needed.

Explain Creating Conditional Statements and Effects in Ozobot Editor

Start by opening the **ExplainL12Code** (Share Code: 26vwqjg) and displaying it for the class. Ask students to explain what might be going on in this code. After getting a few responses, explain that the code uses the Switch blocks discussed in the Explore section and a concurrency block that they are already familiar with and pairs these together with a "Repeat While/Until" block. In programming, a While/Until loop is used to allow a program to run constantly until a certain set of conditions are met. These conditions are dictated by the green block.

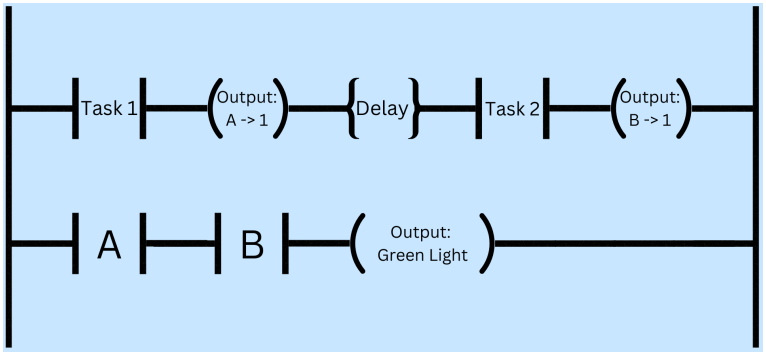
This code, as shown, means that until A and B are both 1, the code in the light blue code block will be repeated (Evo showing a red light, saying the value of A, and saying the value of B). On the left side of the concurrency block, ORA is completing tasks. When the first task is complete, the A switch is 'flipped' to 'On' or a value of 1. ORA pauses for five seconds and then completes the second task which flips B to 'on' or a value of 1.

Place Evo in the center of A1 and run the code as it is. After Evo turns green and says the value of A and B, reset Evo back to A1. On the code, change the 'and' into an 'or' and rerun the program.

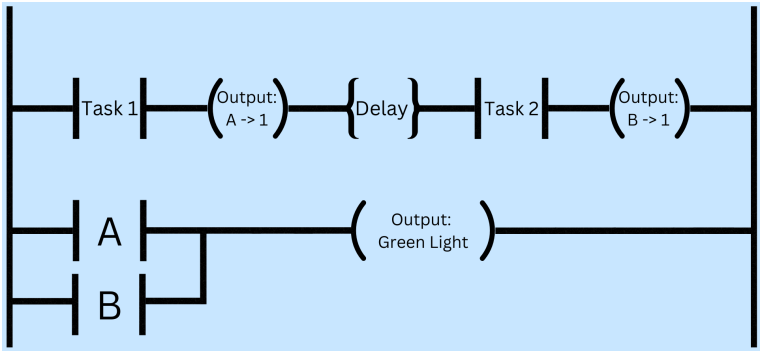
The students should pick up on the new behavior of the code. Now, since there is an 'or' in the logic block, the Evo changes to green before the second task is completed. This is because flipping the A switch satisfied the 'or' condition that controlled the until loop.

Ask students to recall the concept of ladder logic diagrams from the video. If they were to create ladder logic diagrams for the Explain code that was just demonstrated, what would it look like? After students have mulled it over, show the diagrams below. On the top is the AND code. On the bottom is the OR code. These are abbreviated and do not show every action of the ORA or Evo.

In the AND ladder logic diagram, the top 'rung' gets run. The bottom 'rung' has A and B as gatekeepers. This means that A and B both need to be true before the output can occur.



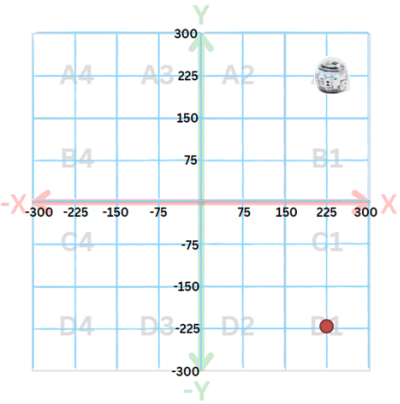
In the OR diagram, the top 'rung' is the same as the AND Diagram, but the bottom 'rung' is not a continuous bar across. The path shows that either A OR B can be true, and the output can occur.



Elaborate: Filling Functions to Satisfy Ladder Logic

In this challenge, students will receive a pre-constructed ladder logic diagram with functions in place for the missing code blocks. Students will fill out the four functions to satisfy the ladder logic diagrams for both an AND and OR statement.

Setup:
Place Evo in the center of A1 and a wooden cylinder, on end, in the center of D1.



Task Requirements:
Students must:

1. make changes to the ElaborateL12Code (Share Code: xe3m394).
2. ONLY change the functions, not the above code.
3. create functions to satisfy the ladder logic diagrams shown above.
4. run the program with the AND logic and the OR logic statements.
5. ORA must finish in the Initial Position.

Direct students to plan this code in the Elaborate section of their Student Activity Sheet. After creating the pseudocode and drawing the code, direct them to

create this code on their devices in the Ozobot Editor. Ensure students are connected to your shared workspace.

As students complete their code, they will test it. Click on a given student's or student group's tab on the Ozobot Editor. Click "Run" to execute their code. Reset the Evo between each code run (if needed). Remember to follow the Safety Protocol established in Lesson 1.

If students are waiting for the ORA, they could:

- Answer Questions on the Student Activity Sheet
- Complete the Evaluate section of the Student Activity Sheets.

Evaluate: Exit Ticket

Students will fill out and submit the Evaluate section of the Student Activity Sheets. (The last page of the Student Activity Sheets).

Optional Extension: Combine AND and OR statements

In this activity, students will assume the role of a Control Systems Engineer in coding a Programmable Logic Controller. The PLC must be coded for the ORA to complete both an AND and OR statement to complete a task. Students will also create a ladder logic diagram of their code to show the statements in action.

Source: Wikicommons

Setup: Place Evo in the center of A1 and a wooden cylinder, on end, in the center of D1 (Same as Elaborate).

Task Requirements:

1. ORA must start in the Initial Position.
2. ORA's end tool must be set to Finger Gripper and set to Open.
3. ORA and Evo must both be used in this challenge.
4. Students must use concurrency and while/until loops to create their program.
5. Students must plan out their code and ladder logic diagrams on Engineering Paper.
6. Evo must have a unique sound or activity for each different output.
7. ORA must move Evo or the wooden cylinder in each task.
8. One AND statement and one OR logic statement must be used in the code.
9. ORA must return to the Initial Position when complete.

Sample Solution Code: https://editor.ozobot.com/blockly?programId=53ddu79

Sample Ladder Logic Diagram:
https://www.google.com/url?q=https://drive.google.com/file/d/1FO0HOa0uF3WjyZG3JHu53NpPm0_yv-cJ/view?usp%3Ddrive_link&sa=D&source=docs&ust=1722368386236116&usg=AOvVaw28VhrSZQvtfOdAEihXwgG5