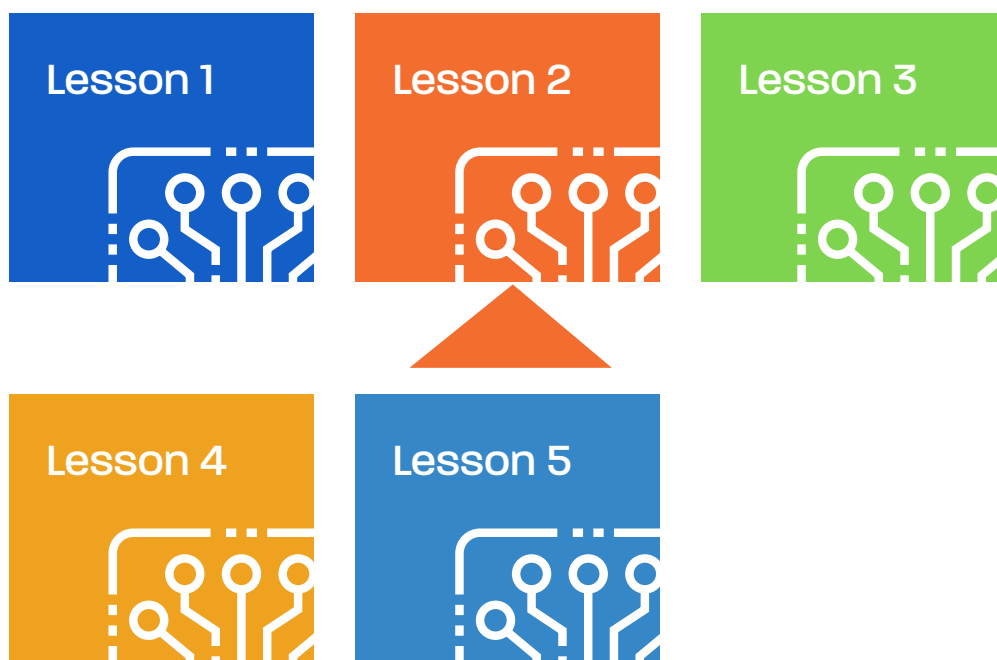
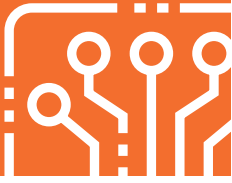


STEAM SNIPS™

Lesson Plans



Lesson 2



STEAM SNIPS™



LED Control Activity

Learning Objectives

Class Activity

Activity Instructions

Understanding Input, Control and Output Logic (1 hour 30 minutes) Number of STEAM SNIPS Recommended (5)

To understand conditional loops and programming logic by using IF, NOT and BRANCH logic blocks

Split the class into groups so each team can complete the 4 unique tasks included in this lesson plan.

Students will then be tasked with applying what they learned to a real-life scenario.

- Hand the teams a set of STEAM SNIPS input, control and output blocks.
- Explain the function of the conduction loop IF, NOT, and BRANCH logic blocks. An explanation of each logic block can be found below.
- Walk the class through each task and what they will need to accomplish.



**What Each Group
Will Need**

- (1) Light Sensor
- (1) Sound Sensor
- (1) Motion Sensor
- (1) IF Logic Block
- (1) AND Logic Block
- (1) NOT Logic Block
- (1) BRANCH Logic Block
- (1) Power Block
- (1) Button Block
- (1) LED (Red)
- (1) LED (Green)

Programming Logic

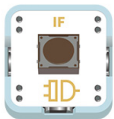
In computer programming, a system can take in two decisions that will lead to specific code blocks being executed using conditional loops.

For a program to perform the desired action, a coder must use programming logic to outline logical operations in order for the program to function correctly.



Class Activity

IF Logic Block



NOT Logic Block



BRANCH Logic Block



Tasks

Task 1

**Understanding the STEAM SNIPS Blocks
(20 minutes)**

An IF statement is a programming conditional statement that, if proved true, the desired action will be performed. When IF is proved false, the action will not be performed.

IF logic blocks are used to record 2 input values.

- Press the button once to register a signal as a “close” signal
- Press the button twice in quick succession to register the signal as a trigger signal

The NOT logic block executes a block of code when an If statement condition is proven false.

The NOT logic block reverses the signal received on the left. It can be used with other logic blocks to reverse the signal from the logic block before it.

Note: The NOT logic block can only accept 1 input.

BRANCH, otherwise known as branch logic, allows the program to take a different path from its original default behavior of executing instructions in order.

This block can be connected in the input or output port and used commonly with AND and OR blocks. When the BRANCH block is not connected to any logic block, its input will be false or 0.

STEAM SNIPS Lesson Plan 2 - Teacher Task Guide Video Instruction
**Have the Teams Experiment with the Blocks
(40 minutes)**

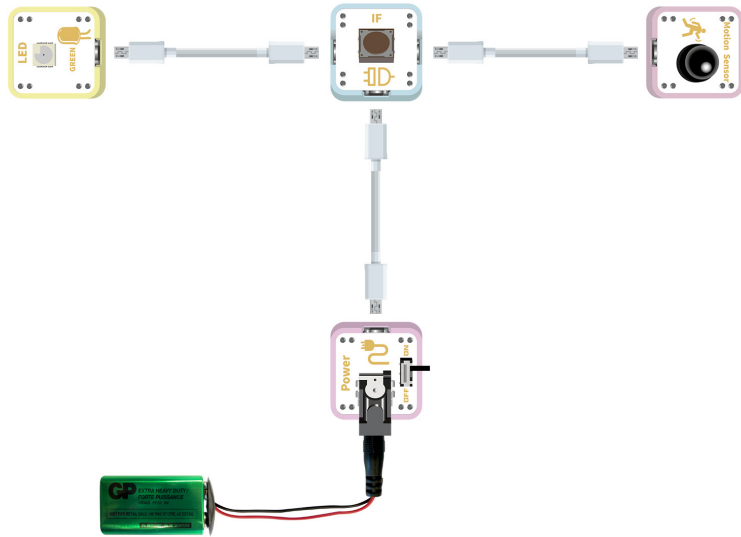
Have the groups complete the tasks outlined below. Each group will use their STEAM SNIPS input, control and output blocks to achieve the objectives laid out in an individual task. They will be able to make a connection between what they are learning and the smart technology they use in their daily lives.

Make an LED turn on when a person walks near it, otherwise it will not light up.

Requirement: Use the IF logic block.



Task 1
Completed Example

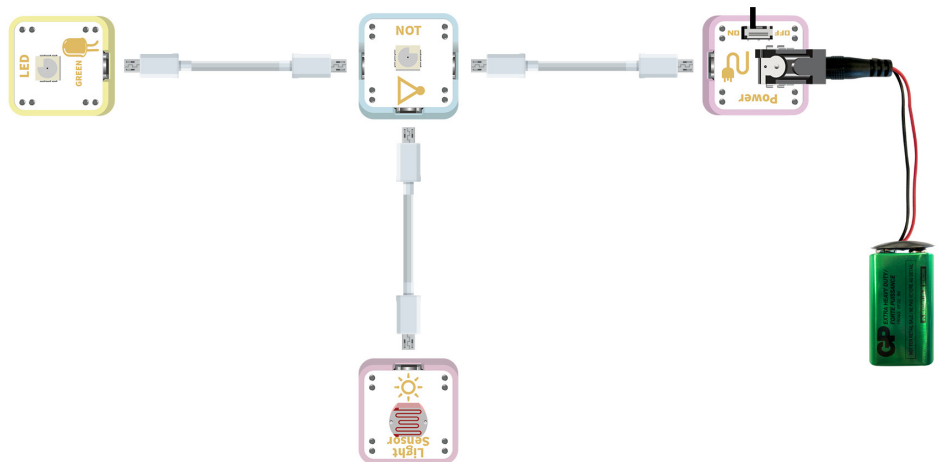


Task 2

When the ambient light is low in a room, LED turns on, otherwise it will not light up.

Requirement: Use the NOT logic block.

Task 2
Completed Example



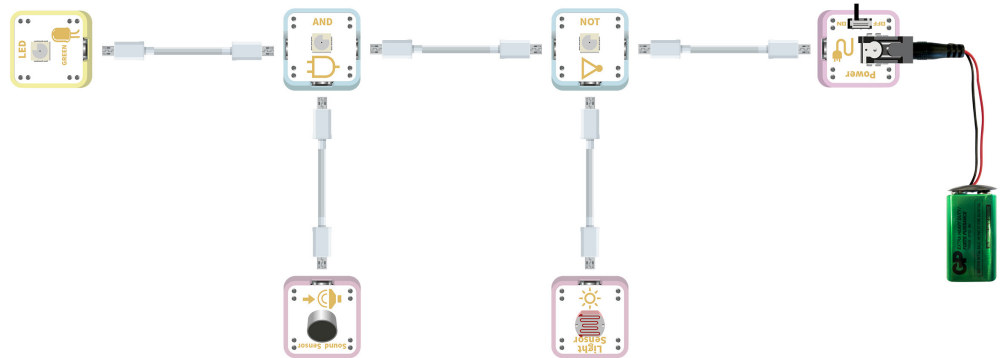


Task 3

When sound volume is high and ambient lighting is dim, LED will light up, otherwise it will not light up.

Requirement: Use AND logic block and NOT logic block

**Task 3
Completed Example**

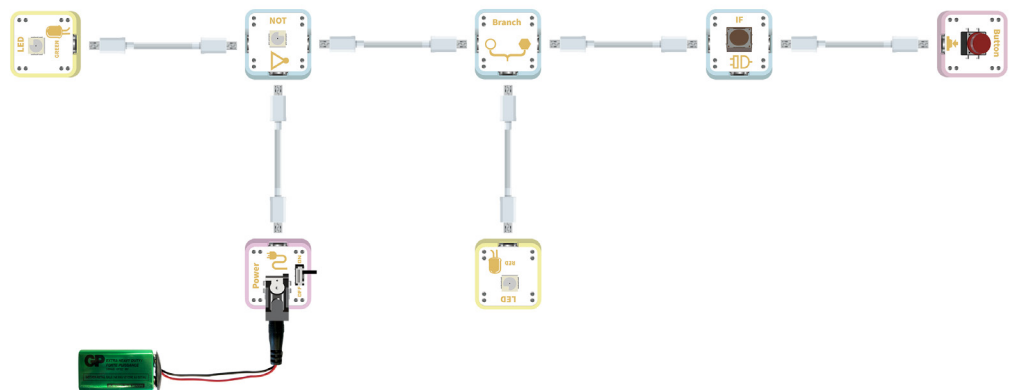


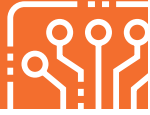
Task 4

Red light – Green light: Make a switch to toggle both LED lights on and off. Make a push switch that controls (2) two separate LED's. When pressed, one of the LED's should light up, once released, the other LED lights up and the previous LED is switched off.

Requirement: Use BRANCH, IF, NOT logic blocks and (1) BUTTON block.

**Task 4
Completed Example**





Have students record the problems they faced when completing the tasks above.

Create Their Own Smart Home Hardware (25 mins)

Have students use different input, control, and output logic to create solutions to a real-life problem. Students can use what they learned to put the input, control, and output logic into action and create their own smart home hardware. Have them list which input, control, and output devices they would use and write a short example of the purpose of their smart home hardware.

Group Name			Date	
Item	Input	Control	Output	Application (state an example)
1				
2				
3				

Lesson 2



Summary

(5 minutes)



Standards-Aligned

CCSS

CCSS:ELA-LITERACY.RST.6-8.3 Key Ideas and Details

CCSS.ELA-LITERACY.RST.6-8.6 Craft and Structure

CSS.ELA-LITERACY.RST.9-10.7 Integration of Knowledge and Ideas

ISTE

ISTE Empowered Learner 1c, 1d

ISTE Knowledge Constructor 3d

ISTE Innovative Designer 4a, 4b, 4c, 4d

ISTE Computational Thinker 5a, 5c, 5d

ISTE Creative Communicator 6a, 6b, 6c

ISTE Global Collaborator 7c

NGSS

NGSS: MS-ETS1-1 Engineering Design

NGSS: MS-ETS1-2 Engineering Design

NGSS: HS-PS3-3 Energy

NGSS: HS-ETS1-2 Engineering Design