

The goal of this tutorial is to introduce the JavaScript interface of the N-Touch controller. Throughout the tutorial, programming terminology is used, but little to no knowledge of JavaScript is assumed. Instead, you are encouraged to copy/paste the examples and modify them to suit your needs.

NOTE: For links to more detailed information outside the scope of this tutorial, see the section [Additional Resources](#) on page 7.

For this tutorial, we are going to build a control panel which allows the following control at the touch of a button:

- Switching multiple sources (a Media Player and a Laptop) to multiple displays (two TVs).
- Turning a TV on and off using serial commands to enable and disable the video port.
- Toggle a display's source back and forth based on feedback.

NOTE: The system example used for this document is shown in FIG. 1 and is built using N2000 Encoders and Decoders. However, the procedure given would be applicable whether the device is controlling an N1000, N2000, or N3000 based system. When using N-Command, ports 80 and 3001 cannot be blocked for the N-Command device's IP addresses (in order for Panel Builder to work correctly).

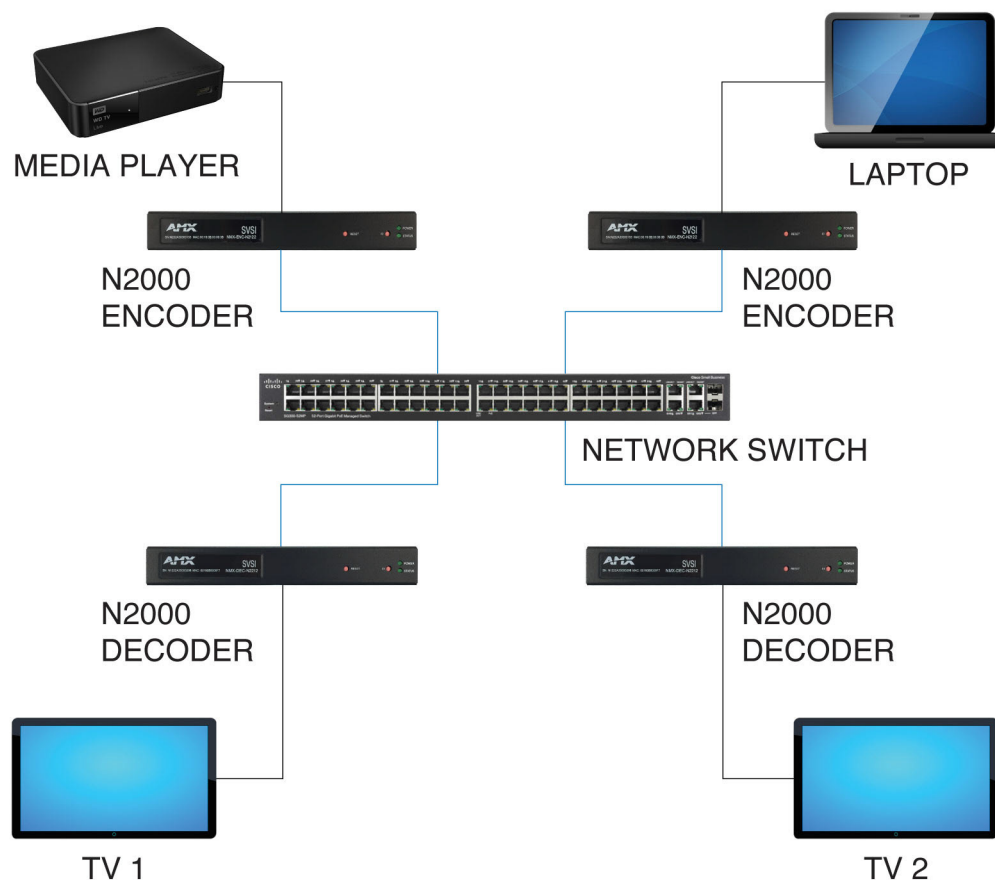
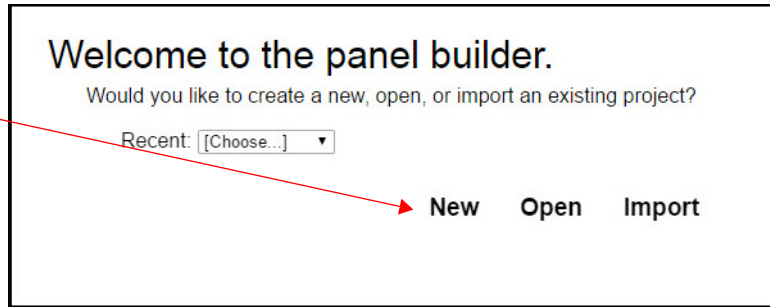


FIG. 1 Example Setup

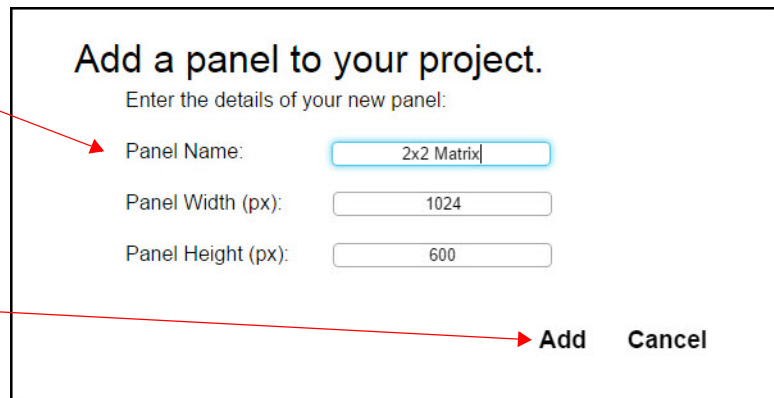
Step 1: Create a New Project

1. Launch **Panel Builder**. The **Welcome** screen appears.

2. Click the **New** button.



3. Enter a **Panel Name, Width, and Height**.



4. Click **Add**.

FIG. 2 Creating a Project and Adding a Panel

5. The system launches the **Project** editor (as shown in FIG. 3).

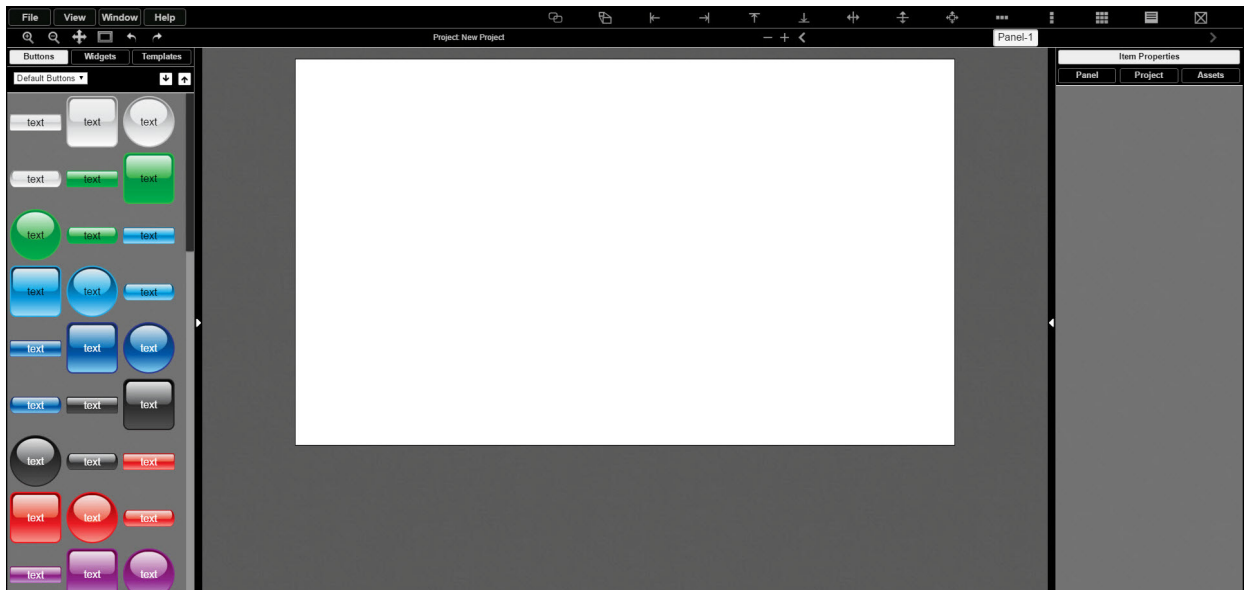


FIG. 3 Project Editor

- For this tutorial, we begin by adding eight buttons. Drag and drop buttons from the left into the project space as shown in FIG. 4. Click in the **text** field to rename your buttons.

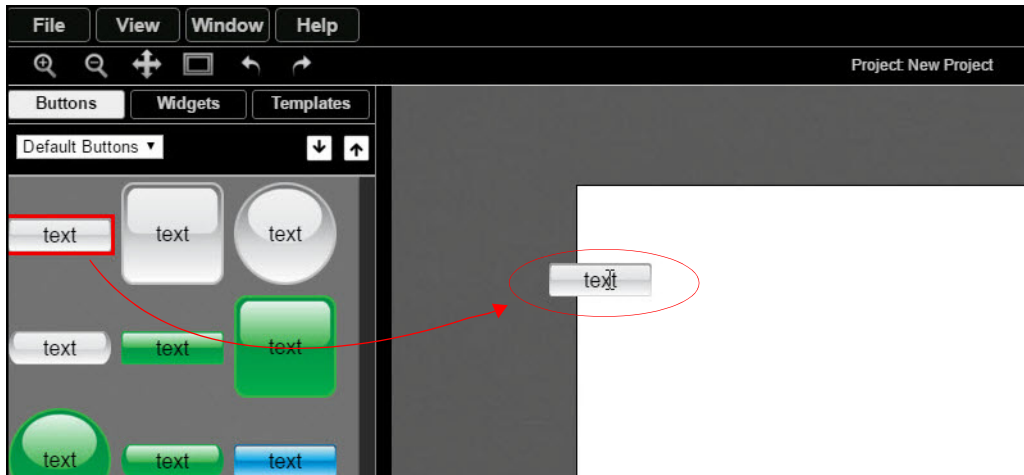


FIG. 4 Adding Buttons

- Continue dragging and naming buttons until you have the setup displayed in FIG. 5. Once this tutorial is complete, the buttons will perform as described in FIG. 5's callouts.

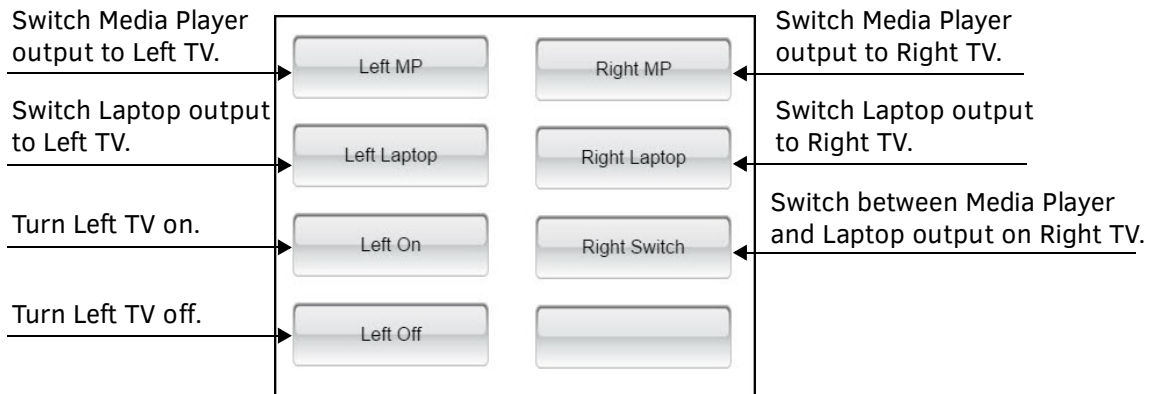


FIG. 5 Button Definitions

Step 2: Discover Encoder/Decoder IP Addresses

If you do not already know the IP addresses of your Encoders and Decoders, use N-Able to discover them.

NOTE: N-Able is SVSI's free device management software. Follow these links to download:

- PC version - <http://www.amx.com/products/N-ABLE-PC.asp>
- Mac version - <http://www.amx.com/products/N-ABLE-MAC.asp>

For this tutorial, we are using two N2000 Decoders and two N2000 Encoders to connect to our displays (i.e., Left and Right TVs) and our sources (i.e., Media Player and Laptop). The Decoder and Encoder IP addresses are as follows:

Display/Source Device Type	Connected to...	Decoder/Encoder IP Address
Left TV	N2-DECODER-1	169.254.10.25
Right TV	N2-DECODER-2	169.254.10.26
Media Player	N2-ENCODER-1	169.254.10.27
Laptop	N2-ENCODER-2	169.254.10.28

Step 3: Create Startup Script Using JavaScript Objects

Follow these steps to create a startup script using JavaScript objects.

1. Under **Project>Setup Script**, click the **view** button.

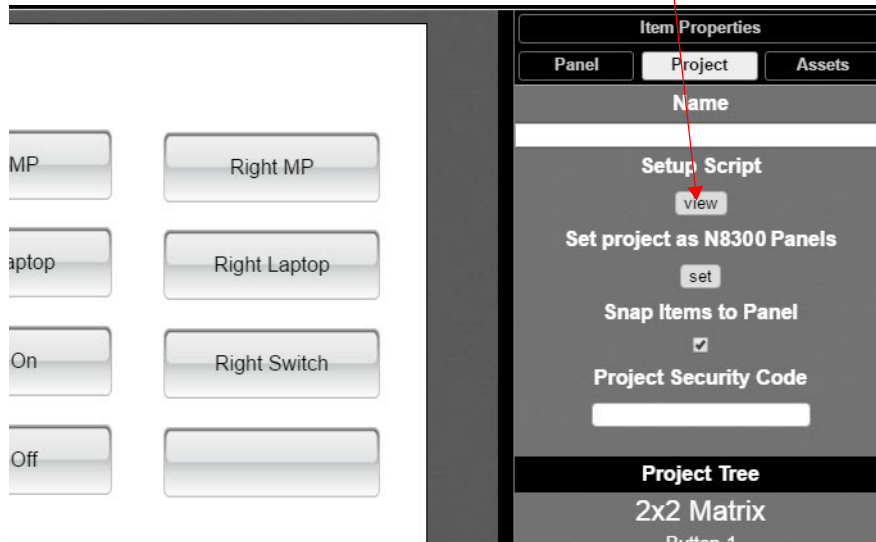


FIG. 6 Creating Startup Script

3. The Project Setup Script window is displayed. Enter the script below:

```
//Create N_Series objects of all our units
LeftTV = N_Series("169.254.10.25");
RightTV = N_Series("169.254.10.26");
MediaPlayer = N_Series("169.254.10.27");
Laptop = N_Series("169.254.10.28");
```

This script defines LeftTV, RightTV, MediaPlayer, and Laptop as N_Series objects that can be used throughout our scripts. Defining an N_Series also calls .getStatus() to do a one-time update of the current status of that unit.

NOTE: Copy/paste scripts from this document and edit to fit your system requirements.

4. Now enter the script below to define your serial on and serial off string.

```
//Turn LG TV on
      LG_TV_ON = "6b 61 20 30 20 30 31 0d";
//Turn LG TV off
      LG_TV_OFF = "6b 61 20 30 20 30 30 0d";
```

NOTE: For this example, we are using the HEX commands to turn an LG TV off and on. Your TV may be different. If you only know your TV's commands in ASCII, do one of the following:

- *Change the script to read:*
`LG_TV_ON = asciiToHex("ka 0 01") + " 0d"; LG_TV_OFF = asciiToHex("ka 0 00") + " 0d";`
- *Convert the ASCII command to HEX. A convenient way of doing this is to use N-Able (SVSI's device management software discussed earlier on page 3). From N-Able's main menu, select Tools > Serial Commands. Enter the ASCII command, and copy/paste the converted HEX command.*

5. Once the script is edited to match your needs, click the **Save** button. See FIG. 7.



FIG. 7 Enter and Save Script

NOTE: This is far more convenient than using raw IP addresses with the tcpclient function. Also, the objects have feedback and methods to make our code easier which will be shown next.

Step 4: Edit Button Scripts for Switching Displays and Sources

Follow these steps to add functionality which will allow you to:

- Switch Media Player output to the Left and Right TVs (using the **Left MP** and **Right MP** buttons).
- Switch Laptop output to the Left and Right TVs (using the **Left Laptop** and **Right Laptop** buttons).

1. In the **Project** editor main page, click the **Left MP** button.

2. Click the **Edit Scripts/Conditions** button.

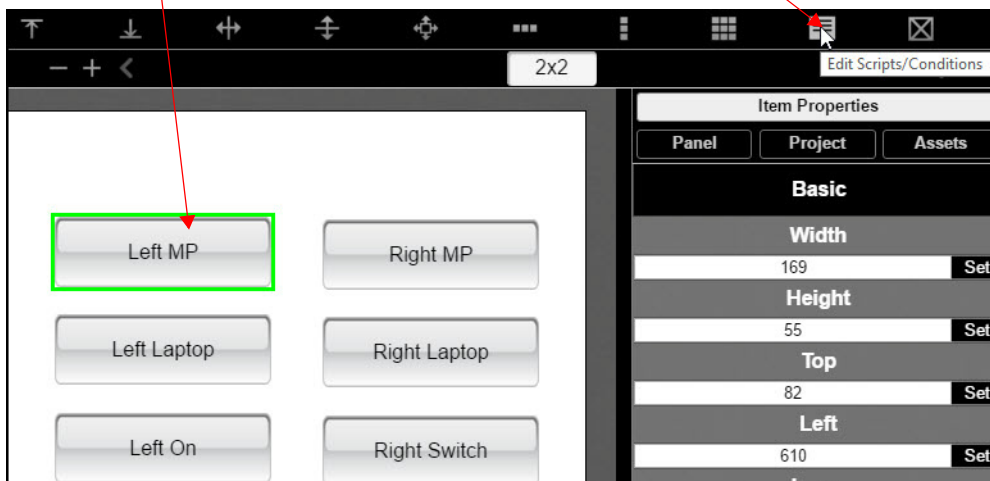


FIG. 8 Launch Left MP Button's Script Editor

3. Enter (copy/paste) the following script into the script editor. This script causes the Left TV to display the Media Player source (see FIG. 9).

```
LeftTV.switch(MediaPlayer);
```



FIG. 9 Left TV - Media Player Script

4. Once the script is edited to match your needs, click the **Save** button.
5. Launch the **Left Laptop** button's script editor. See FIG. 10.

Click the **Left Laptop** button and click the **Edit Scripts/Conditions** button.

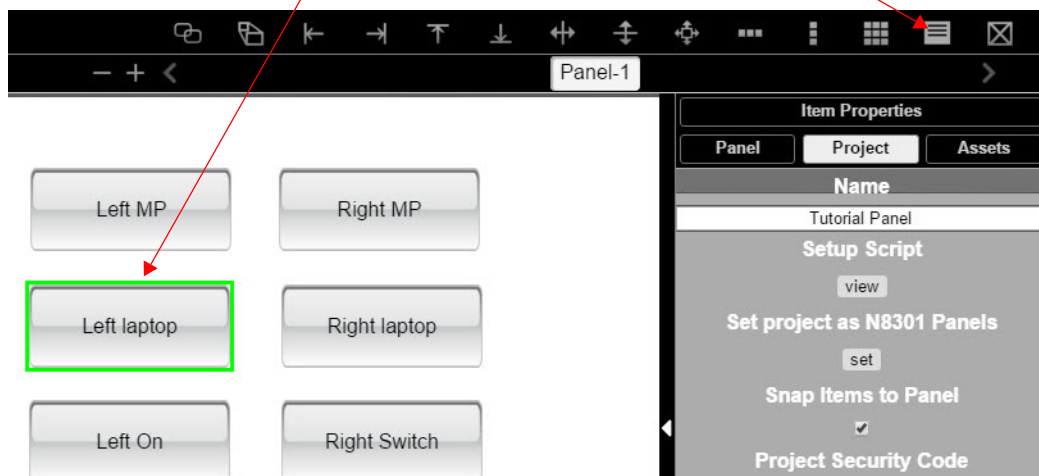
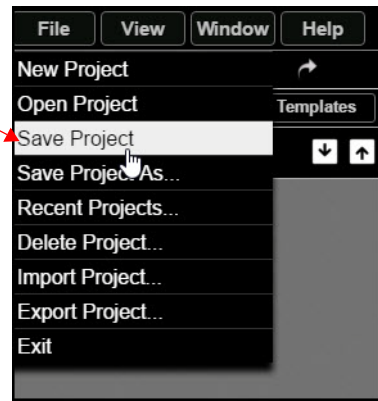


FIG. 10 Launch Left Laptop Button's Script Editor

6. Enter (copy/paste) the following script into the script editor. This script causes the Left TV to display the Laptop source:
`LeftTV.switch(Laptop);`
7. Once the script is edited to match your needs, click the **Save** button.
NOTE: The `switch` script switches the stream of a Decoder. It can accept a raw stream number (such as 200) or it can accept another N_Series object.
8. Save the project so that you can test the changes made so far.

Click **File > Save Project**.



Click **Here's a link to your project** to display the panel in your browser:



FIG. 11 Saving Your Project to Test

9. The **Left MP** and **Left Laptop** buttons should now be active. Click the buttons to switch the Left TV display between the two sources.
10. Repeat this process for the Right TV, using the following scripts:
 - **Right MP** button script:
`RightTV.switch(MediaPlayer);`
 - **Right Laptop** button script:
`RightTV.switch(Laptop);`
11. Again, save the project so that you can test the changes made so far. The N-Touch now has a functional 2x2 matrix.

Step 5: Edit Startup Scripts for Turning a TV On/Off

Follow these steps to add functionality which will allow you to turn the Left TV display off and on.

1. In the **Project** editor main page, click the **Left On** button.

2. Click the **Edit Scripts/Conditions** button.

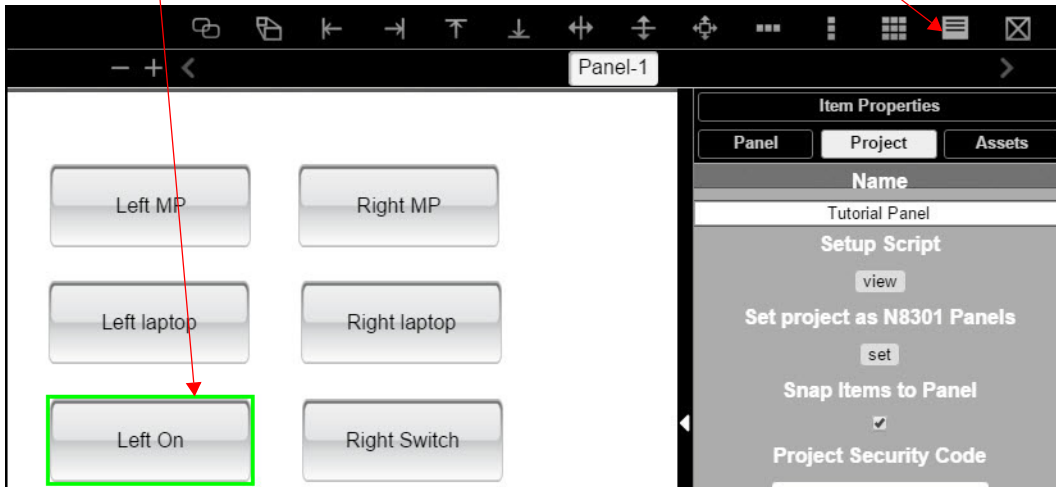


FIG. 12 Launch Left On Button's Script Editor

3. Enter (copy/paste) the following script into the script editor. This script sends the serial HEX string and turns the DVI on.

```
LeftTV.serialHex(LG_TV_ON);  
LeftTV.dviOn();
```

4. Repeat the process for the **Left Off** button using the following script:

```
LeftTV.serialHex(LG_TV_OFF);  
LeftTV.dviOff();
```

5. Once the script is edited to match your needs, click the **Save** button.

The `.serialHex()` is a function built in to `N_Series` objects to send a space-delimited string of HEX characters from the serial port. Because `LG_TV_ON` was defined in the Startup Script, we can use a variable name instead of retyping a long string of characters. Assuming an LG TV is connected properly to the LeftTV's serial port, those buttons should now turn the TV off and on. The `dviOff()` and `dviOn()` are also built-in `N_Series` commands. Some TVs will automatically turn off and on when disconnected. If yours supports this, using these commands to turn on/off the Decoder's output may be preferred.

Step 6: Toggle Scripts

Follow these steps to add functionality which will allow you to use the **Right Switch** button to toggle the Decoder (connected to the Right TV) between the two Encoders (depending on the stream it is currently on). In other words, if the Right TV is currently displaying output from the Laptop, clicking the **Right Switch** button causes the Right TV to begin displaying Media Player output.

This example demonstrates configuring a button to function based on feedback and will be the most complicated script in this tutorial.

1. Launch the **Right Switch** button's script editor.
2. Copy/paste the following script into the editor:

```
RightTV.getStatus(function(){
  if(RightTV.status.stream == MediaPlayer.status.stream) {
    RightTV.switch(Laptop);
  } else {
    RightTV.switch(MediaPlayer);
  }
});
```

The commands are defined individually below:

```
RightTV.getStatus()
```

This updates the status of RightTV. The `getStatus()` function accepts another function as a parameter. This is called a "callback function." This function gets executed when the get status is finished.

```
if(RightTV.status.stream == MediaPlayer.status.stream) {
```

All the statuses are available in a `.status` object of your `N_Series` object. To look at the current stream number of RightTV, we look at the `RightTV.status.stream`. The if statement is asking, "Does this stream number equal the stream number of the MediaPlayer?"

```
RightTV.switch(Laptop);
```

If the stream equals the MediaPlayer stream, then switch to the Laptop.

```
} else {
```

Otherwise....

```
RightTV.switch(MediaPlayer);
```

...switch to the MediaPlayer.

The panel is now complete. It supports a 2x2 matrix, serial commands, turning the DVI off and on, and a toggle button that looks at the feedback of a Decoder.

Additional Resources

JavaScript

- Mozilla Developer Network (function reference):
<https://developer.mozilla.org/en-US/>
- Codecademy Online Coding Classes:
<https://www.codecademy.com>
- Stack Overflow (programmer Q&A):
<https://stackoverflow.com/>
- W3Schools (tutorial-based help):
<http://www.w3schools.com/js/>
- jQuery (N-Touch has this useful library available for general use):
<https://api.jquery.com/>

Module Help

- Source code and function list are available on the N-Touch:
<http://<unit ip>/panels/modules.php>
- Press **F12** (for Mac users, press **Command + Shift + I**) to launch the built-in Google Chrome debugger on your web browser. The debugger console may show clues to the problem.

If using the Chrome debugger, the method ".print()" will echo the entire object to the console.

NOTE: SVSI devices list their most recent commands on the Logs page. Look at Logs to see if the command you expected appears. You may wish to periodically reset the Logs to clear out older commands.

The screenshot shows the N-Touch Controller interface. At the top, there's a header with 'AMX N-Touch Controller | 00:19:0B:81:01:33'. Below the header, there are navigation tabs: 'Settings', 'Logs' (highlighted with a red circle), 'LLDP', 'Users', and 'Panel Builder'. The main content area shows 'Your IP address is 169.254.55.86' and buttons for 'Refresh Logs' and 'Reset Logs'. Below this is a 'Command Log' table with the following data:

Elapsed Time	IP	Port	Method	Command
3 days, 19 hrs, 54 min, 57 sec	169.254.245.18	50001	TCP	getStatus
3 days, 22 hrs, 1 min, 41 sec	169.254.245.18	50001	TCP	getStatus
3 days, 22 hrs, 4 min, 38 sec	169.254.245.18	50001	TCP	getStatus

FIG. 13 Logs Page