Benefits of a “Scriptable” Mixing Console

The rise of audio over IP technology and introduction of programming/scripting functions enables an expansion of features and capabilities for modern audio consoles

By

Fig. 1: Custom scripts allow you to change controls on surfaces or to create standalone mixing UIs.

The author is a support engineer for Wheatstone Corp. This originally appeared in the ebook “Console Tech 2021.” It is one in a series of articles about how to get the most out of popular radio broadcast products.

Consoles come in all shapes, sizes and forms these days, from legacy hardware surfaces to the newer virtual mixers on a laptop and everything in between.

What worked in your studio yesterday might not work today, and what works today might not work tomorrow. That's where scripting comes in, both in terms of custom scripts for virtual mixers as well as newer hardware consoles with software configured controls.

Creating custom scripts to change controls on console surfaces such as our LXE or GSX models or to create entirely standalone mixing UIs is one practical and affordable way to meet these constantly changing requirements.

Nearly every broadcast mixing surface or console manufactured has a set of standard features that cover 90% of the workflow or use requirements for a studio. Generally, these are: input faders, control room and studio controls, mix-minus sends, and logic I/O for tallies and remote control.

Fortunately, the world of audio over IP enabled us to make several improvements on these features.

[Related: "How to Choose Your Next Console"]

The change from legacy console, where one wires everything to the chassis of the mixer, to a distributed or routed environment with Blades or other I/O units becoming

termination points for routing, replaced miles and miles of cabling in some cases.

But even with all the enhancements the AoIP routed studio brought us, at the end of the day the console (or what is now called a “surface”) is essentially doing the same job. That is, mixing your content together and sending that mix on to the next process in the chain.

The job of the console remains the same, but what has changed is how the job is done — and on what.

Multiplicity of applications

For example, with scripting, you can change the default behavior of any hardware button, fader, encoder or OLED screen on the LXE or GSX console surface.

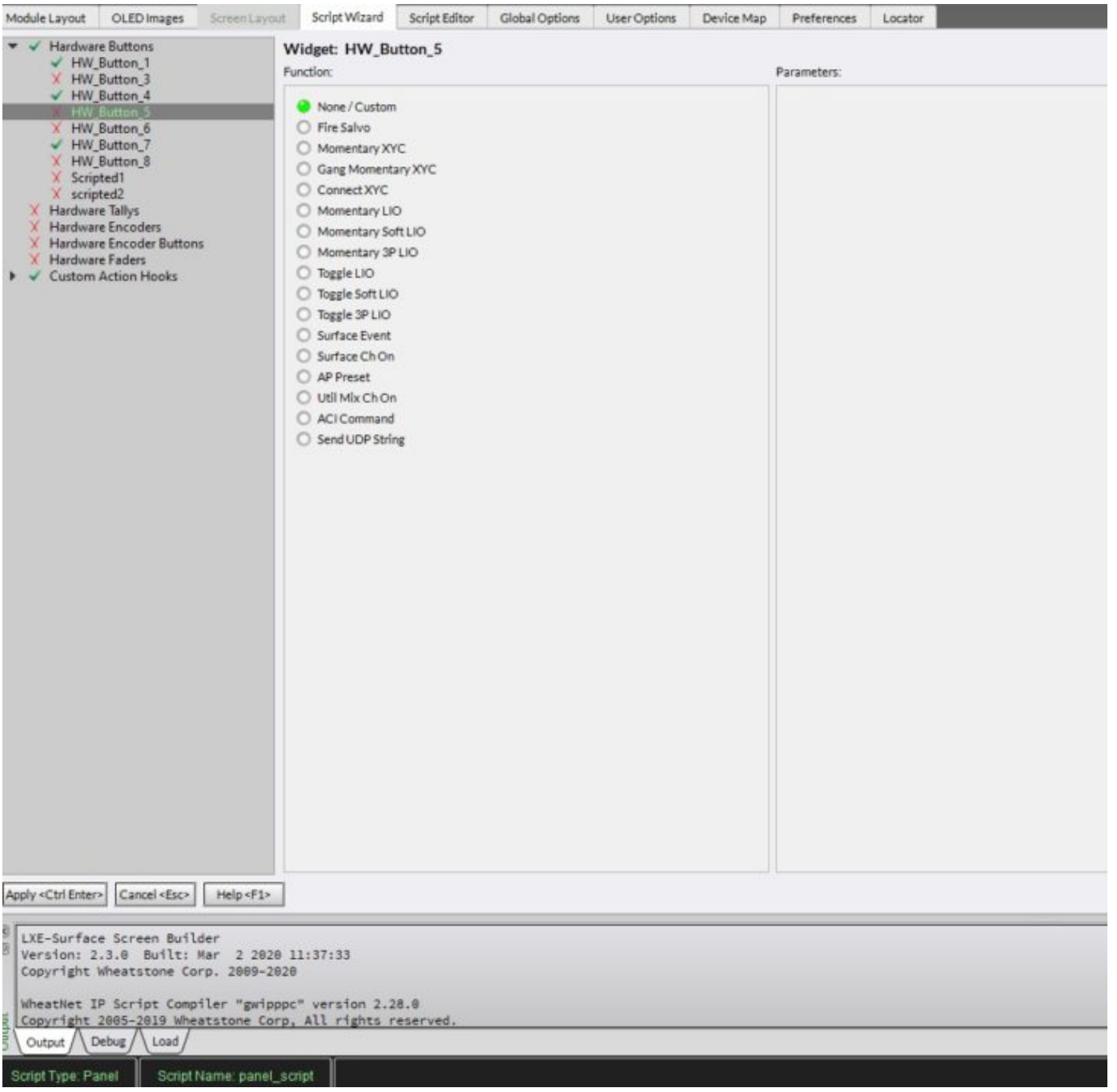
This can be done in easy-to-use setup software, and changes to the surface generally do not require a restart of the surface itself. In addition to a full array of surface standard functions, users also now have control over button colors as well as the behavior of that button.

A person could also actually write custom code using the Surface Setup GUI and the Wheatstone scripting language to have the hardware button do more than one function, and then change LED state (such as color) based on the status

of whatever it was they intended to control or change.

A simple example of this is to set up a button to fire a salvo.

This is a simple point-and-click procedure using the LXE or GSX Script Wizard in the Surface Setup GUI.

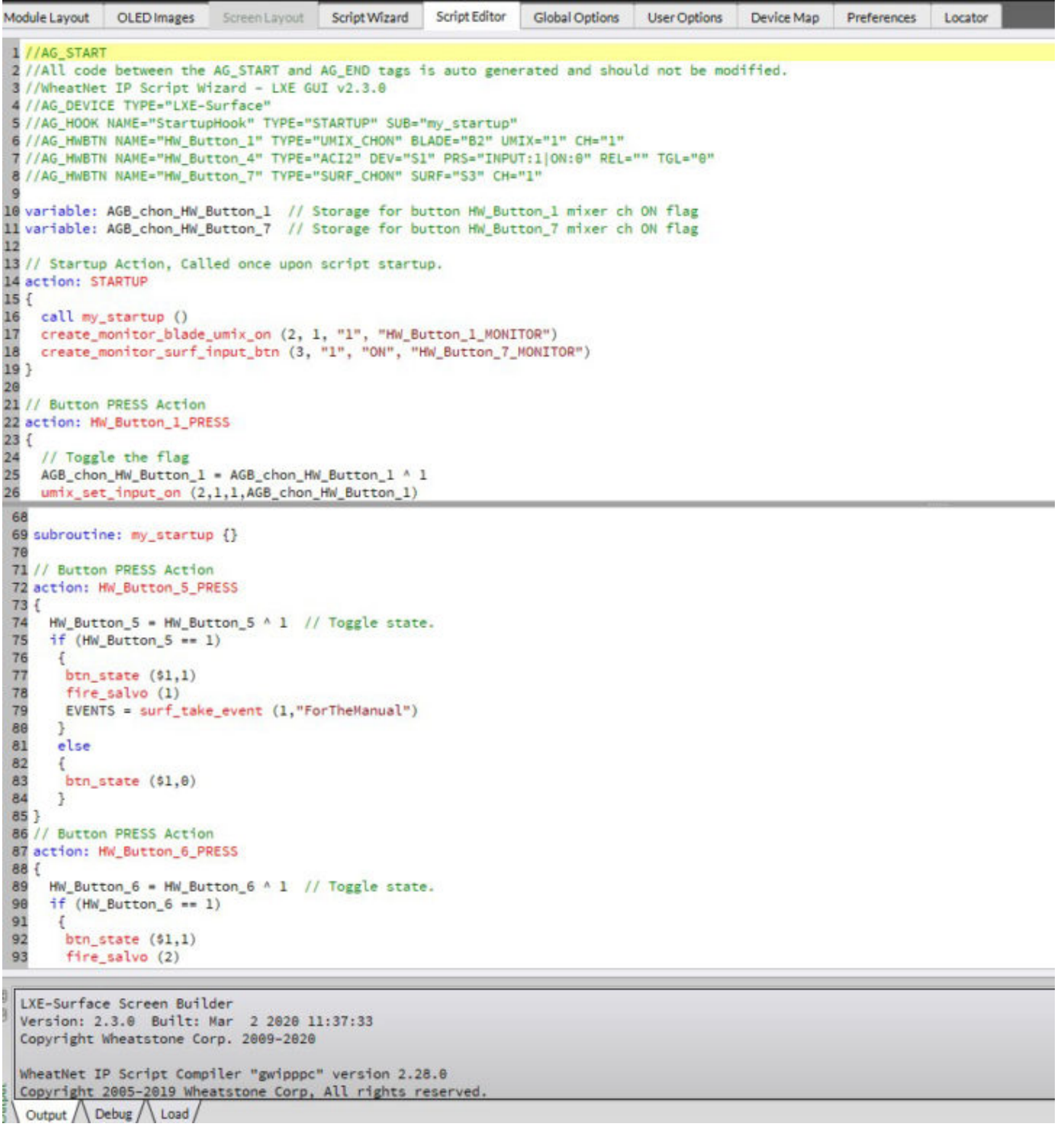


Setting up a button to fire a salvo is a point-and-click procedure using the LXE or GSX Script Wizard in the Surface Setup GUI.

Once the change is sent to the surface, the button becomes a Fire Salvo (macro) button. In addition to firing the salvo to change the audio routing, we can also change the state of logic pins on another Blade in the system and at the same time, change the button to a different color when that logic pin is in the ON state or activated state.

What's more, you can go beyond the Script Wizard and into creating a custom piece of software that executes the salvo and changes the state of the logic pin when the button is pressed, plus change the LED color of the button when that logic is active.

All you'd need is to open the Script Editor and add a few lines of code, as shown below.



You can can go beyond the Script Wizard, to creating custom software that executes the salvo and changes the state of the logic pin when the button is pressed, plus change the LED color of the button when that logic is active. Open the Script Editor and add a few lines of code.

As workflows and requirements change, you can modify salvos and more, and the surface will automatically update without the need to restart it or the mix/DSP engine.

In addition, with scripting tools such as ScreenBuilder, you can add custom screens directly on the console GUI itself. One of the main benefits of being able to build custom screen interfaces directly on the console itself is that these UIs don't have to run on a PC in the studio, which most likely is already doing quadruple duty as an Internet/edit/playback PC.

Screens can be developed using drag-and-drop widgets such as buttons, labels and meters that can be set up with logic controls that modify various aspects of the system for changing audio routing, on/off logic and tallies.

You can set up screens for not only one specific studio, but all of the studios in a WheatNet-IP audio networked system as a whole.

Across the network

Let's say you have five stations in a location, and there's one person in the facility for overnights who monitors all five stations.

From one control room, the overnight talent could call up a screen to see the status of all five stations at once and swipe through a menu to monitor audio from those stations and to get data from various points in the system.

This can be done directly on any LXE and GSX console surface in any studio, so if the overnight talent is not in his normal position or studio he or she can still see the system from any room there's a GSX or LXE.

Also, these new scriptable consoles have OLED displays for each input fader and two or more for each output module. Each of these displays can be configured independently to display different data sets about sources assigned, program assignments, mix assignments and can be further customized for your own text and graphic displays.You can even add station logos or other images to reinforce station branding, and provide at-a-glance data to the operator.

Another software benefit is the LXE and GSX's ability to have up to 32 inputs and 16 outputs in their mix engines. This means broadcasters have access to ample inputs and outputs yet are able to keep the physical fader or surface size down to a minimum. By carefully deploying layers to the surface, they can page a smaller layout surface of say 4 to 16 faders to get access to those additional inputs and outputs. This allows studio designers to keep a smaller footprint on the furniture and make additional room, or clean up an already crowded space.

When off-site, operators can also remote in to the studio or physical console using apps such as Remote LXE/GSX,

ReMIX or Glass E. These are software extensions of the AoIP network or physical console that can mirror what's happening at the studio. In some cases, remote operation can be done on an entirely standalone virtual console that contains custom scripting, all of which could be the blocks of the all-virtual studio of the future.

Robert Ferguson has been in radio for more than 25 years, with experience both behind the board and in front of it as a broadcast engineer and on-air personality.

Subscribe

For more stories like this, and to keep up to date with all our market leading news, features and analysis, sign up to our newsletter here.