Linn Crestron 4-Series Driver for SIMPL#Pro
Basic Setup Guide

Introduction

This document aims to give guidance in the building and loading of the *Linn Crestron 4-Series Driver for SIMPL#Pro* demo project. This demo project showcases how Linn products can be included in a Crestron system designed in SIMPL#Pro, using the same underlying driver as the one used in our SIMPL Windows support. On completion of this guide, you will have a working demo that fully utilises the available features. More in-depth details of how the driver operates are provided in the installation guide. Basic Crestron knowledge is assumed in this guide.

Requirements:

1. A Crestron 4-Series processor[1]
2. A download of the *LinnCrestron_Pro* project archive
3. Installation of Crestron software (SIMPL Windows, VisionTools Pro-e, etc).
4. A Microsoft Visual Studio[2] installation (any version 2019 onwards) with .NET desktop development
5. At least one "Core4"[3] Linn DSM. This device and any other Linn DS/DSMs should be on the same local network and be associated with the same Linn account. This can be achieved through the Linn App. If no Core4 device is available, then there is an alternative that is detailed in step 1.
6. Crestron Enterprise Development Kit[4] should be installed on the PC that will be used to upload the project to the processor.

Step 1: Setting Up the Linn CI Gateway

The Linn 4-Series driver communicates with the CI[5] Gateway software. This must first be enabled. To achieve this:

- Go to the Linn *Manage Systems*[6] page and sign into the Linn account that your devices are associated with.
- Under *Music Systems* select the device that will be used to run the CI Gateway software. Then go to the *Advanced* tab of the settings and enable the CI Gateway option. You will then be prompted to reboot the device. You need only enable the CI Gateway for one device.
- Once the device is rebooted you can access the gateway configuration by going to the IP address of the host Linn DSM device on port 4100 (e.g., http://192.168.0.142:4100/). Here you can log in to TIDAL, Qobuz, and your Linn Account which will enable additional services in the demo.

---

[1] The demo project is configured for a CP4, but this is changeable through SIMPL Windows. Go to the Configure page on the SIMPL demo program and right click the CP4, then **Replace "CP4"** and choose your device type. Not every 4-Series device has been tested so this may come with unexpected issues, if encountered please report to the Linn help desk.

[2] https://visualstudio.microsoft.com/vs/community/

[3] For a list of viable devices see: https://docs.linn.co.uk/wiki/index.php/Category:CORE4

[4] https://sdkcon78221.crestron.com/sdk/Crestron_EDK_SDK/Content/Topics/Installation.htm

[5] Custom Install

[6] https://www.linn.co.uk/account/music-systems

If no viable device is available, then Kazoo Server software can be used instead which includes the CI Gateway. Simply install it on a PC, Mac, or QNAP device. Go to the configuration for Kazoo Server and you will find an option for the CI Gateway where the same login options can be found.

Step 2: Unpacking the Archive

The demo project archive contains 3 items:

- A folder containing the Visual Studio solution of the demo project (named *Linn4SeriesProDemo*).
- Vision Tools Pro-e files for XPanel and iPad.

The Vision Tools Pro-e files are the same as those included in the driver for SIMPL Windows. To set these up, open the *Linn_Demo_xPanel.vtp* file in Vision Tools Pro-e. The project settings are accessed by pressing *Linn_Demo_xPanel* at the top of the *Project View* window. Under **Connection Settings**, change the host address to that of your 4-Series Processor. Then go to **File -> Compile Project** to build the xPanel. The *Linn_Demo_xPanel.c3proj* directory will be added to the project folder containing the xPanel application file. The *Linn_Demo_iPad.vtp* file can be similarly compiled if you wish to try out the demo using an iPad. There is no need to specify the host address here; just compile and then send it to your processor where it will be hosted.

Extract the *Linn4SeriesProDemo* directory to a suitable location. Open the solution, *Linn4SeriesProDemo.sln*. This solution contains the SIMPL#Pro equivalent of the Linn 4-Series Driver for SIMPL Windows. Some details within this project will have to be customized for your setup:

- The source URIs for the image files used to represent sources are settable through strings in the project. The image path URIs are images that will show in the now-playing GUI space when an external source type is active. In a customer's project, these would be changed to image files that fit the theme of their project. The assets for the demo project[7] can be uploaded using the File Manager in Crestron Toolbox or another FTP client. Ensure they are in the same locations as specified in the image path strings: *defaultSourceImagePath*, *analogSourceImagePath*, etc.
- If you would like to test out the *LinnLocalServer* module of the driver, which gives the ability to browse and search a UPnP media server, some further setup will be required. The Local Server Module in the demo project has an example server name that will need to be customized to a valid server name on your network to work. This is set by the string: *localServerName* in the project.

Step 3: Building

The necessary packages all come included in the project archive but it's worth ensuring that Visual Studio can download any if they are missing. Go to **Tools -> Options -> NuGet Package Manager** and ensure that both *Package Restore* options are enabled as shown in the image below:

---

[7] https://linn-artifacts-public.s3-eu-west-1.amazonaws.com/custom/LinnCrestron_Demo_Assets.zip
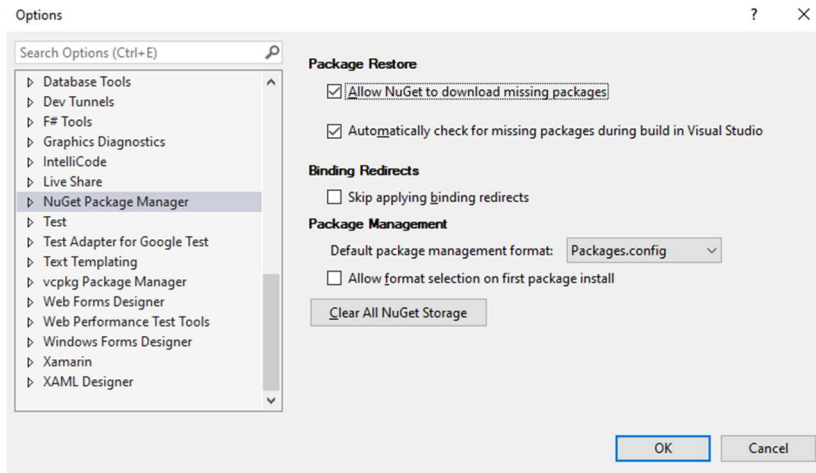
To build, in the top bar go to **Build -> Build Solution**. The build will then start. If successful, the output window should look like the image below:
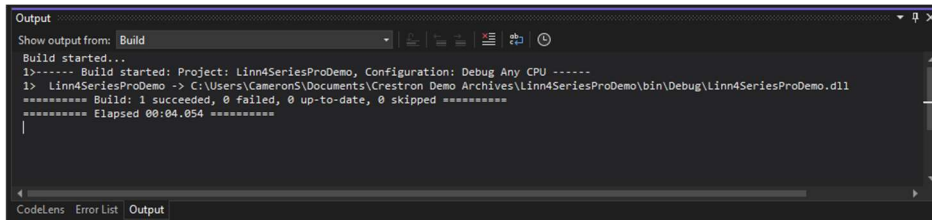
The build should output a compiled file that can be run on the 4-Series Crestron processor. In the solution folder, check the /bin/Debug directory for the file *Linn4SeriesProDemo.cpz* (Note the .cpz extension, if there isn't a file with this extension then something went wrong with the build process).

Step 4: Upload to your Crestron 4-Series processor

The easiest way to upload your compiled SIMPL#Pro program is using the Crestron Enterprise Development Kit (EDK)[8]. This is a collection of PowerShell modules that enables users to interface with Crestron devices over ethernet. The specific PowerShell command within this kit that will be used is the **Send-CrestronProgram** command. Documentation for this is below:

https://sdkcon78221.crestron.com/sdk/Crestron_EDK_SDK/Content/Topics/PSCrestron.htm#Send-CrestronProgram

Within the solution folder, there is a PowerShell script named *LoadLinnDemo.ps1*. Open this using a text editor and replace the variables with the information relating to your setup:

- $file: the file path of the newly built .cpz file on your PC
- $ip: the IP address of your Crestron processor
- $slotNumber: the target program slot number on the processor to use
- $username: your processor username
- $password: your processor password

Once complete, save the file and open a PowerShell command prompt. Change the working directory to the project solution using the *CD* command, eg:

   CD C:\Users\Name\Documents\Linn4SeriesProDemo

---

[8] https://sdkcon78221.crestron.com/sdk/Crestron_EDK_SDK/Content/Topics/EDK-Overview.htm

Then run the PowerShell script:

> .\LoadLinnDemo.ps1

The EDK will then start uploading the program to the Crestron processor. If successful, you should see a message like the following:

```
PS C:\Users\camerons\Documents\WindowsPowerShell\CrestronScripts> .\LoadLinnDemo.ps1

DeviceLAN     : 192.168.0.141
DeviceCS      :
StartTime     : 23/11/2023 15:43:48
EndTime       : 23/11/2023 15:44:33
Pass          : True
ErrorMessage  :
ProgramSlot   : 1
ProgramFile   : Linn4SeriesProDemo.dll
```

On loading, the driver will automatically enable and after booting you should be met with the screen below which indicates that the setup has been successful, and the demo is ready for use. Select a room by pressing the box in the top bar and the buttons for accessing various features will be enabled.
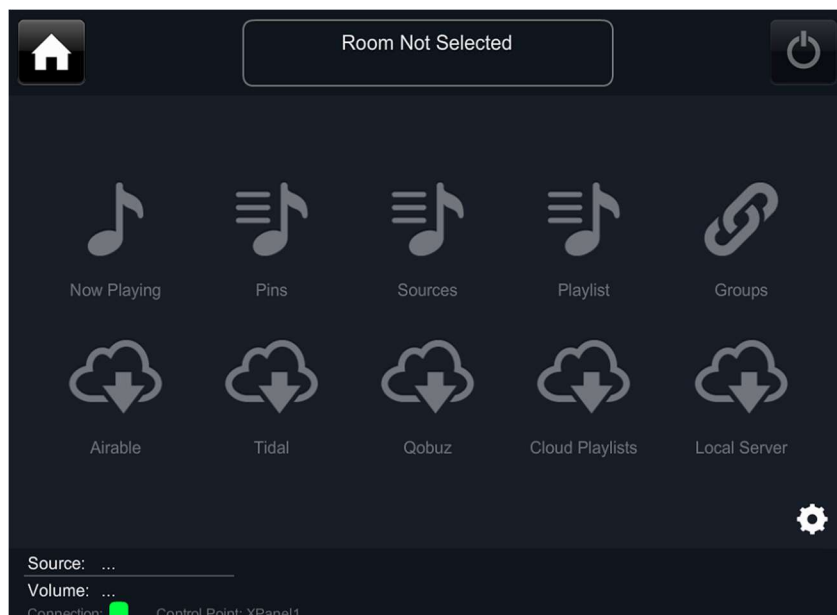
Using the driver in your own SIMPL#Pro projects

The demo project simply showcases one example way of utilising the driver in the project. For a project of your own simply copy across the 'dlls' folder to your solution and add a reference to the *Linn4SeriesProDriver.dll* file. You can then add the using directive for *Linn4SeriesProDriver* and access the module objects. If you take a closer look at the *InitializeSystem* method in the *ControlSystem.cs* file, you can see how modules are added to the program similarly to the symbols in the SIMPL Windows demo project. The module objects are created with a constructor that takes in the usual parameters such as control point names and page sizes. The event handlers within the module objects are then hooked up. Once this is complete, the Initialize method is then called which registers the modules. **The driver module must be initialized last.** It is then up to the programmer to design the rest, set up signal handling, and smart object graphics (or HTML5!). Control for other devices in the project can be added to the SIMPL#Pro program.