

Falcon Programmer

Manual

Simon O'Rorke

30th June 2024

Software Version 1.0

## Introduction

Falcon Programmer is an open source batch configuration application for the UVI Falcon software synthesizer. Multiple types of configuration change can be implemented in thousands of Falcon programs with a single batch run, taking seconds to minutes.

There is currently an installer only for Windows. However, the source code should run on macOS; and a macOS installer will be provided as soon as a collaborator can be found to create the installer and test the application on macOS.

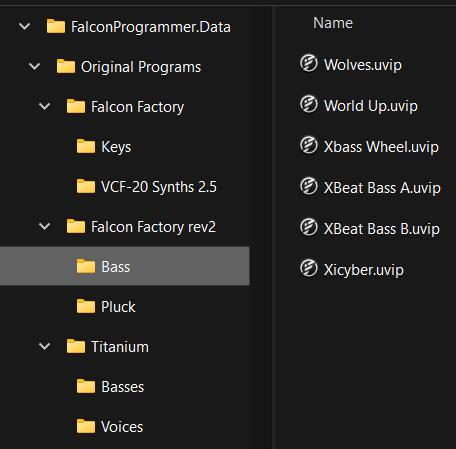
The following configuration tasks are available:

* Restore the program to an original version, ready for the configuration changes to be made.
* Initialise the program's Info page layout with many options, including converting a script-based layout to the standard layout.
* Assign MIDI CC numbers to macros and, provided the program uses the standard Info page layout, optionally append each macro's MIDI CC number to its display name.
* Bypass (disable) all known delay effects and then, provided the program uses the standard Info page layout, remove any macro that no longer modulates any enabled effects.
* If a Release macro is not part of a set of four ADSR macros and the macro is not modulated by the mod wheel, set its initial value to zero.
* Set the values of known reverb macros, with some exceptions, to zero.
* Move release and reverb macros that have zero values to the end of the standard Info page layout.
* Where feasible, replace all modulations by the modulation wheel with modulations by a new 'Wheel' macro on the standard Info page layout.
* If the modulation wheel's modulations have been reassigned to a Wheel macro (the previous task), reuse MIDI CC 1 (the mod wheel) for a subsequent macro, where feasible.
* Prepends a line indicating the program's path (sound bank\category\program name) to the program's description, which is viewable in Falcon when the Info page's ***i*** button is clicked.

Of these configuration tasks, assigning MIDI CC numbers to macros will be of use to many Falcon players. And restoring the program to an original version is just a safety feature to facilitate subsequent transformation. The remainder are merely what the developer has found useful as a Falcon player. Many more configuration tasks are surely possible. Users of the application are welcome to suggest some!

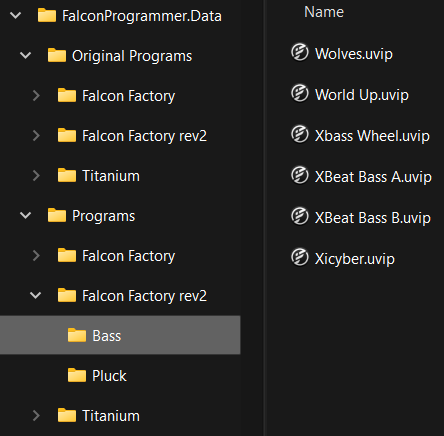
## Getting Started

Save the Falcon programs you want to transform from Falcon sound banks into a folder hierarchy that will hold the original versions of the program. This tedious procedure is unfortunately necessary because Falcon Programmer cannot access the Falcon sound banks. The folder hierarchy must reflect the Falcon sound bank\category hierarchy, like this:



The names of the folders do not have to be exactly the same as the corresponding Falcon sound bank and category names, as Falcon Programmer will get the original names from inside a program when sound bank-specific or category-specific rules are to be applied. However, to avoid confusion, the folder names should at least be similar. For example, if you don't have the Falcon Factory (version 1) sound bank but do have the Falcon Factory rev2 sound bank, you could just call the latter's sound bank folder 'Factory'.

Next, copy the Original Programs folder to a Programs folder, like this:



Falcon Programmer's configuration tasks will update the copies of the Falcon program files in the Programs folder.

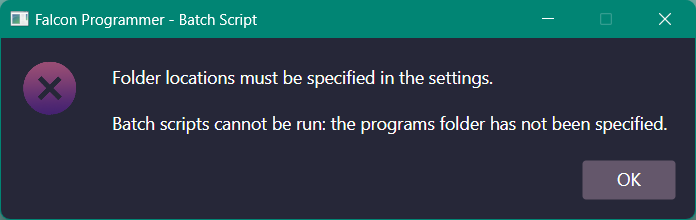
The last addition to the file system required before you run Falcon Programmer is a folder, initially empty, to contain Falcon Programmer's settings:

A screenshot of a computer screen

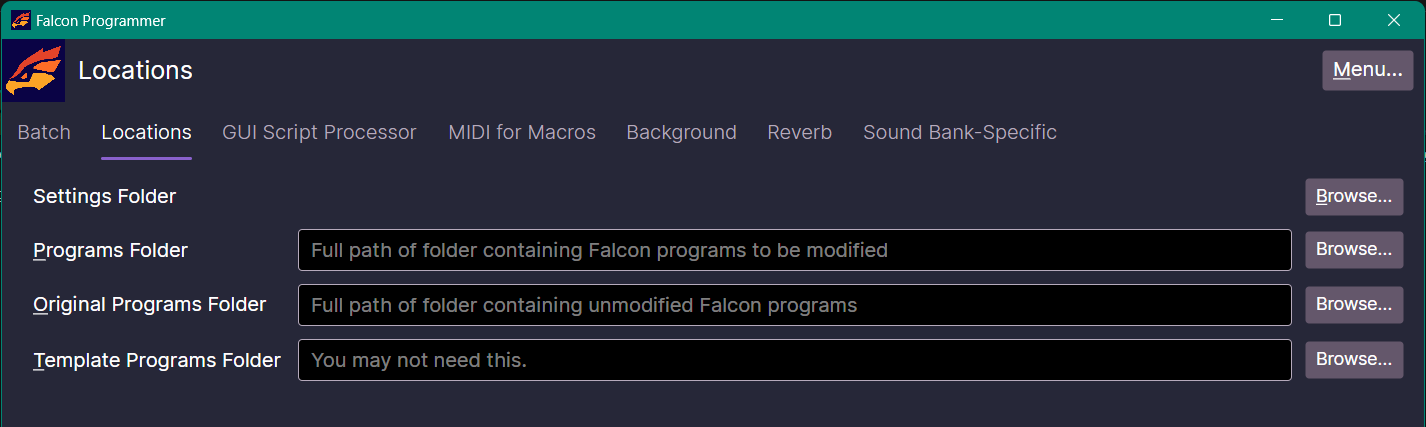
Description automatically generated

Falcon Programmer will save its settings to file Settings.xml, which it will create in the Settings folder.

Now load Falcon Programmer. As you have not yet told Falcon Programmer where your data is, you should see an error message like this:



Click the OK button. The Locations page will be shown:



Leave the Template Programs Folder field empty for now: as the tip in its text box says, you may not need it. Specify the other three folder locations:

A screenshot of a computer

Description automatically generated

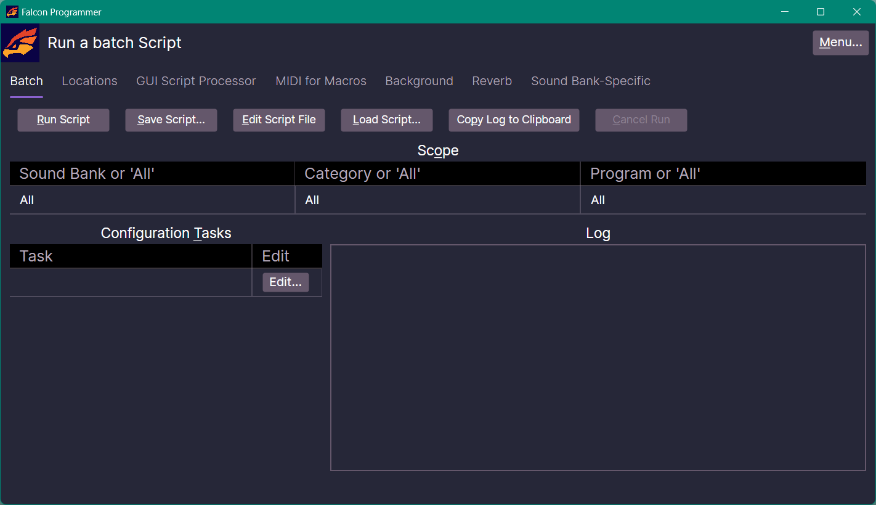
**Note:** Falcon Programmer will save its settings whenever you go to a different tabbed page or close the application.

While we are on the Locations page, have a look at the page tabs:

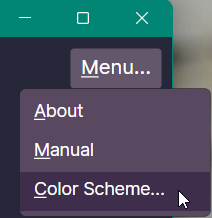
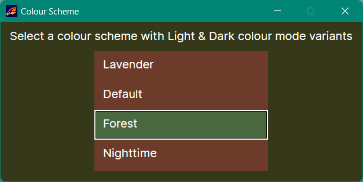


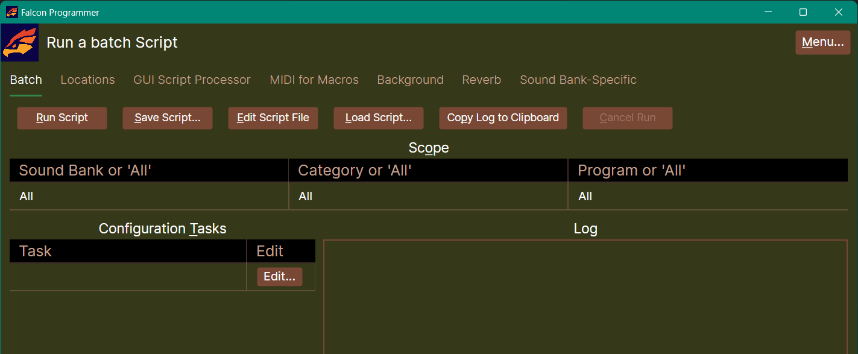
To the right of the Locations page are several more pages of settings. Many configuration tasks require some of these additional settings to be specified.

But let's try running a batch script that requires no more settings. Go to the Batch page:

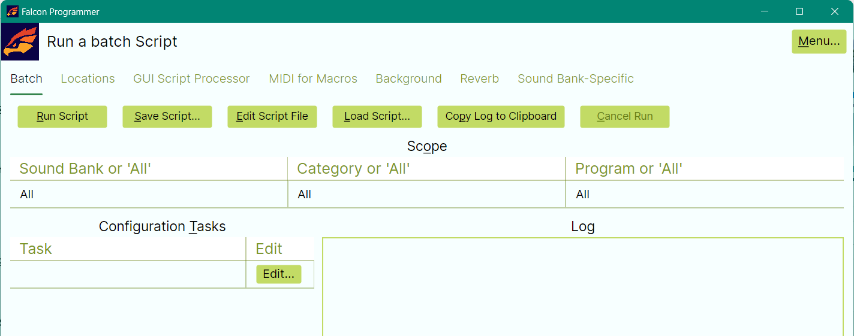


**Tip:** If you would prefer a different colour scheme, you can pick one from the menu at the top right of the main window.

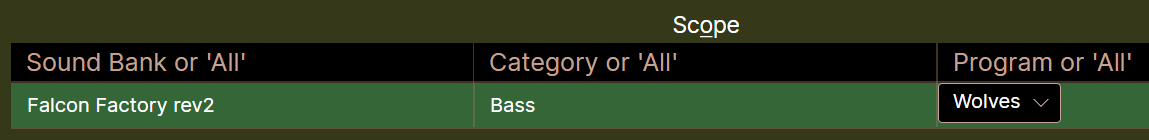
 



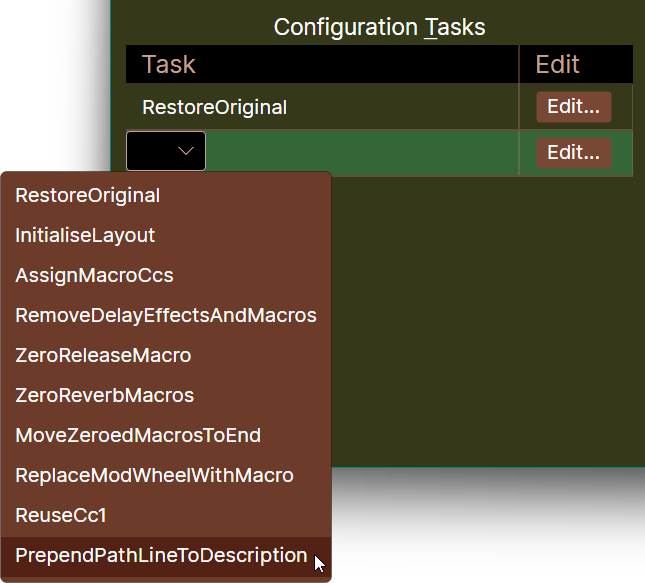
Colour schemes adapt to the operating system's light/dark colour mode. This is the Light variant of the Forest colour scheme, whose Dark variant is shown above:



Select the scope for the batch run. For this tutorial, pick one program.



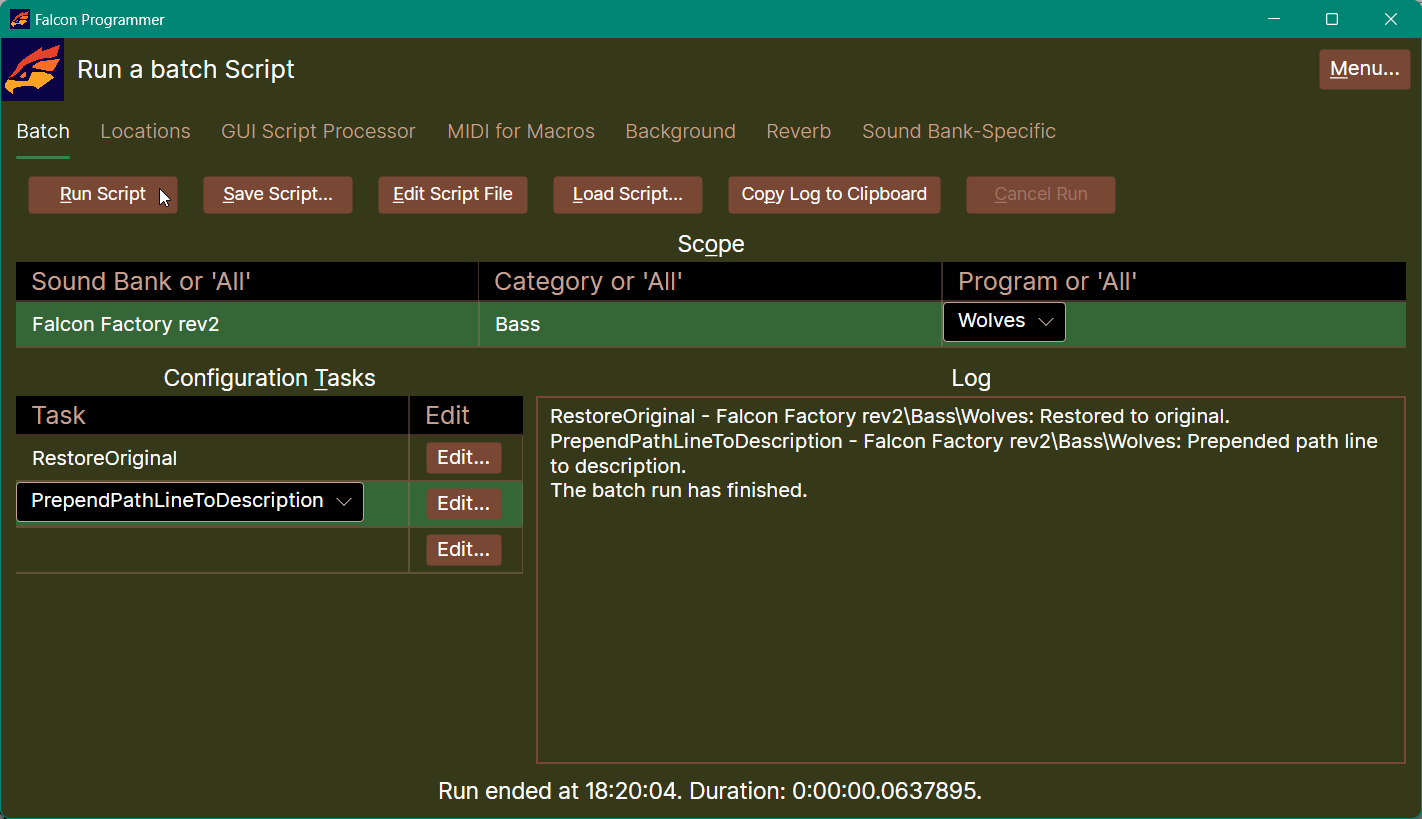
Now select configuration tasks RestoreOriginal and PrependPathLineToDescription, in that order.



**Tip:** The Task drop-down list shows the tasks in a logical order for running: a task is listed after any tasks that need to be run before it. *PrependPathLineToDescription always has to be run last*, due to a technical constraint on how the line breaks in the description are conserved.

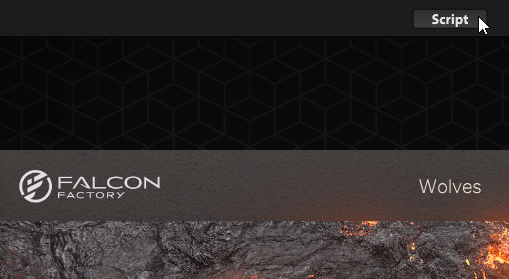
If you have not run any batch scripts yet, RestoreOriginal will make no difference to the content of program file. However, *it is good practice to always start with RestoreOriginal*, and follow it with tasks in a logical order to transform the program to the state you want it in.

You have created a batch script, which consists of a scope and a list of configuration tasks. Click the Run Script button to run it. You should see something like this:



Have a look at the log. It shows the task names, the paths of the programs run and specific actions taken (sometimes more than one per task for some tasks). *The log can also show tasks that could not be run or actions that could not be taken for a program, with reasons why. So it is very useful for problem solving.*

Finally, have a look at the change you made. In their original forms, Falcon Factory rev2 programs all have script-based Info pages. So you first need to click the Script button to show the standard Info page.



And click the ***i*** button …



… to see the path line prepended to the (in this case very short) description.

