DigiScan Stack Acquisition tool – Help

credits

The tool was made by Bernhard Schaffer
version: 2015-08-19
GMS: GMS 2.3 or higher is required for this plugin to work.
The script requires a properly installed DigiScan2 hardware.

purpose

This script conveniently builds stacks of rapidly acquired DigiScan images in a similar manner as the general "StackBuilder (on image update) tool" available from the same author.
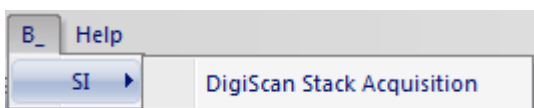
Contents

Setup

Install

Copy the *b_DigiScanStackAcquisition.gtk* file into the plugins folder and restart DigitalMicrograph. There should now be a new menu entry:

    B_ → SI → DigiScan Stack Acquisition

Using the tool

Initialization

The tool uses the regular parameter setups for *Search* , *Preview* and *Record* from the DigiScan palette. However, on DigitalMicrograph start-up or whenever these settings are changed in the dialog, the according acquisitions need to be started manually via the UI buttons once[1].

Launching the tool

Select the menu entry

B_ → SI → DigiScan Stack Acquisition

to launch the tool as a modeless dialog.
The tool stays available until the dialog is closed with the little "x" button in the top-right corner.

Capturing a DigiScan stack

1) Enter the wanted number of frames to be acquired in the **# frames** field.
2) Select the parameter set which should be used from the radio button list in the **Basic parameter set** box.
3) Press the **Capture Stack** button.

This will then perform the following actions:
- Any running DigiScan acquisition is stopped.
- The DigiScan dialog gets disabled
- An acquisition of the according parameter set is started with *continuous* acquisition. All acquired signals are displayed in the upper half of the screen.
- Stacks for each signal are displayed in the bottom half of the screen.
- Whenever a frame is completely scanned, the data gets copied into the stack (and the according slice is shown in the stack)
- When all frames have been acquired – or when any of the top-row windows is closed manually – the acquisition stops.

Overriding parameters

The second box becomes enabled when the **Override** checkbox is selected. Parameters are then still taken from the selected **Basic parameter set** (image size & signal selection) but **Pixel Time**, **Flyback** setting, and **LineSync** setting are used as specified in the according fields instead.

---
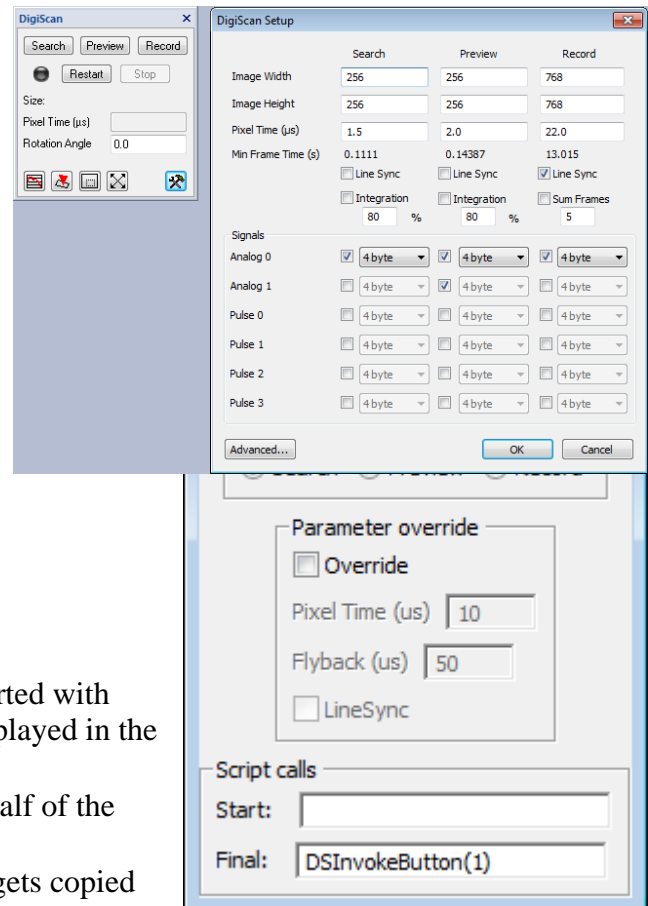
[1] This copies the parameters into a location which is accessible for the script. If this step is not done, pre-cached parameter sets are used.

Script calls

The last box in the dialog offers two text fields into which scripting commands can be typed. These scripts are called when the **Capture stack** button is pressed. The first line specifies a script which is executed immediately after pressing the button before any acquisition starts, while the second line specifies a script which is executed after the stack-acquisition has been finished.

The default text of the **Final** field is *DSInvokeButton(1)*. This script call is equivalent to pressing the *Search* button on the regular DigiScan dialog. Hence, a search will be restarted once the acquisition is finished.

To 'attach' more sophisticated scripts, put all functionality into a single script function
(f.e. of type *void FinalCall( )*) and install it as a library.
You can then put the function-call into the field.
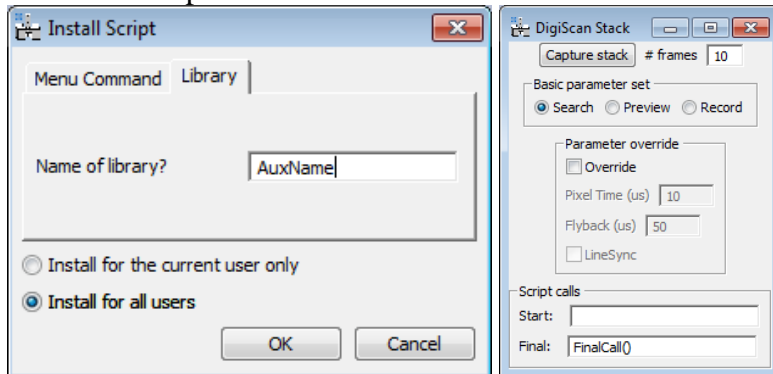
Script example 1: Close stacks (prompt to save each)

The following script added as "Final" call will automatically close all acquired stack-images after acquisition, prompting for each if the data has to be saved first:

```
void FinalCall()
{
        image img:=GetFrontImage()
        string UEID
        if ( img.IMDGetUEID(UEID) )
                EGUPerformAction(UEID,"close")
}
```

Script example 2: GroupSave and close data, then restart Search acquisition

The following script added as "Final" call will automatically save stacks in the root folder **C:\Data** with a dated subfolder. Each images is automatically numbered by 3 leading digits to avoid overwriting data. Then, the images are deleted from memory and the digiscan View acquisition is restarted.

```
void FinalCall()
{
        image img:=GetFrontImage()
        string UEID
        if ( img.IMDGetUEID(UEID) )
        {
                EGUSave(UEID,"C:\\Data\\","[D]","[###]_[N]")
                EGUPerformAction(UEID,"delete")
        }
        DSInvokeButton(1)
}
```