FLIMage Hardware Setting

# Init files

Documents/FLIMage/Init\_Files/FLIM\_deviceFile\_VX.txt (X = 1 for version 1.X, X = 2 for version 2.X)

Documents/FLIMage/Init\_Files/Default.txt

# Manual

Documents/FLIMage/Manual/

# Python Script examples

Documents/FLIMage/Python\_Scripts

# Default FLIM setting

## Operation system:

Windows 10

## NIDAQ:

### Driver:

NIDAQmx 18.6. If you want to use different NIDAQ, please install NIDAQ with DotNet4.5 support and replace NIDAQ DLL files in FLIMage application folder. FLIMage was tested only on NIDAQmx 18.6

### Board:

* PCIe-6232 / PCIe-6231
* PCI-6731 / PCI-6733 (Can be used for EOM. Not recommended, but works)

## Port setting for scan mirrors:

* Dev4/AO0 : X mirror
* Dev4/AO1 : Y mirror
* Dev4/port/line0 : trigger output
* Dev4/port/line1 : main shutter
* Dev4/port/line2 : line clock signal
* Dev4/PFI6 : trigger input
* Dev4/PFI4 : clock signal output
* Dev4/ctr0: line clock signal output (often mapped to Dev4/PFI12)

## Port setting for EOM (You can use the same board with scan mirror):

* Dev2/AI0 : Photodiode input for EOM 1 (for calibration etc)
* Dev2/AI1 : Photodiode input for EOM 2 (for calibration etc)
* Dev2/AO0 : EOM 1 control (In some version default may be Dev4. Please check device file)
* Dev2/AO1 : EOM 2 control
* Dev2/PFI4: clock signal input

## Uncaging shutter:

* Dev4/port/line3 : uncaging shutter (if State.Init.DO\_uncagingShutter = true)
* Dev2/AO3 : uncaging shutter (if State.Init.AO\_uncagingShutter = true)

## FLIM card:

Please set State.Init.FLIM\_mode in the device file as follows:

* State.Init.FLIM\_mode = “PQ” (for PicoQuant TimeHarp 260)
* State.Init.FLIM\_mode = “MH” (for PicoQuant MultiHarp)
* State.Init.FLIM\_mode = “BH” (for Becker and Hickl SPC series)

## Set number of FLIM card:

Currently software support 2 channels only – does not work with 1 channel or more than 2 channels.

### Pico Quant:

State.Spc.spcData.n\_devicesPQ = 1

State.Spc.spcData.channelPerDevicePQ = 2

### Becker and Hickl:

Default is:

State.Spc.spcData.n\_devicesBH = 2

State.Spc.spcData.channelPerDeviceBH = 1

If you have only 1 Becker and Hickl card, please set to:

State.Spc.spcData.n\_devicesBH = 1

State.Spc.spcData.channelPerDeviceBH = 2 (Should work with router device)

## Digital signal connections:

* Board trigger: Dev4/port/line0 – Dev4/PFI6 – Dev2/PFI6
* Clock sync: Dev4/PFI4 – Dev2/PFI4
* Line clock sync: Dev4/ctr0 – M3 of PicoQuant, line clock of Becker and Hickl

Set State.Spc.spcData.lineID\_PQ or State.Spc.spcData.lineID\_BH if you want use different lines.

# Electrophysiology setting

## Default port setting

* Dev3/AO0 : Patchclamp output port 1
* Dev3/AO1 : Patchclamp output port 2
* Dev3/AI0 : Patchclamp input port 1
* Dev3/AI1 : Patchclamp input port 2
* Dev3/AO2 : stimulation output port 1
* Dev3/AO3 : stimulation output port 2
* Dev3/PFI2 : Trigger input (connect with Dev4/port/line0 to receive trigger from FLIMage)
* Dev3/PFI6 : Self-trigger input (connect with Dev3/port/line0)
* Dev3/port0/line0: Trigger output

## Digital signal connections:

* Board self-trigger: Dev3/port/line0 – Dev3/PFI6
* Trigger from FLIMage: Dev4/port/line0 (FLIMage trigger) – Dev3/PFI2

# Motor setting

## Sutter

Default motor is MP-285. Set as follows. Default COM port is COM1

State.Init.MotorHWName = “MP-285A”

State.Init.MotorComPort = “COM1”

State.Init.MotorConversionFactor = [-0.04, -0.04, 0.005] (This depends on MP setting)

## Thorlabs

You can use Thorlab motors. It will use COM port written in XML file provided by Thorlabs. You don’t need to set State.Init.MotorComPort.

State.Init.MotorHWName = “ThorBScope”

State.Init.MotorHWName = “ThorMCM3000”

# Microscope control

Only Thorlabs scope is supported for now.

## Switch light path for Galvo, Camera, and Resonant Galvo.

It can be ThorBCM or ThorBScope.

State.Init.MicroscopeFlipper = "ThorBCM"

or

State.Init.MicroscopeFlipper = "ThorBScope"

## PMT gain control

PMT control can be ThorECU. This requires COM port setting.

State.Init.PMTModule = "ThorECU"

State.Init.PMTModule\_COMPort = “COM29”

Or

State.Init.PMTModule = " ThorBScope "

# Resonant Scanner (Experimental)

You can connect NiDAQ board with ThorECU.

You need to turn on resonant scanner (default is off), and set resonant scanner frequency

State.Init.enableResonantScanner = True

State.Init.resonantFreq = 8000

## Default port setting

* Dev5 is resonant scanner control board
* Dev5/AO0 : X-valvo signal.
* Dev5/AO1 : Y-galvo signal.
* Dev5/PFI5 : Line clock input. (Perhaps not necessary)

## Digital signal connections:

* Line clock signal: Dev5/PFI5 – ECM Line clock in – ECM Line clock out – TCSPC card line clock (M3 for PicoQuant, D1 for Becker and Hickl)

# Tag lens (Experimental)

### COM port setting:

Set COM port as following example:

State.Init.TagLensPort = “COM6”

## Digital signal connections:

* One of RGB output - TCSPC card marker (M2 for PicoQuant, D2 for Becker and Hicl)