**Notes on bringing up a completely new PC as an acousto-optic scanning device**

**Microsoft Windows 10 PC Setup**

BIOS Settings:

* Main Tab
  + BIOS Version -> 1.1.9
* Advanced Tab
  + BIOS Auto-Recovery -> [Disabled]
* Advanced Tab -> Power Options
  + AC Recovery -> [Power On] (so computer reboots after a power loss, **most important setting!**)
  + Deep Sleep Control -> [Enabled in S4 and S5]
  + Auto Power On -> [Enabled]
* Boot Tab
  + Secure Boot -> [Disable]

**3rd Party Software Setup**

This section details how to set up the PC for scanning.

Install [Anaconda](https://www.anaconda.com/distribution/) ( <https://www.anaconda.com/distribution/> )

* Install Anaconda version 2020.02 64-bit ( [Anaconda3-2020.02-Windows-x86\_64.exe](https://repo.anaconda.com/archive/) )
  + If Anaconda is already present on the computer, update to the 2020.02 version
    - If possible, fully [uninstall all python and anaconda versions](https://docs.anaconda.com/anaconda/install/uninstall/)
    - Confirm Anaconda version with: conda list anaconda$
      * Should be version 2020.02 and build py37\_0
    - Update version with: conda update anaconda=2020.02
    - If that doesn’t work try: conda install anaconda=2020.02
    - Confirm conda version with: conda info
      * Should be version 4.8.2 and build 3.18.11
* Click installer (in browser download bar)
  + Select installation type: all users
  + Don’t check any advanced options
  + Install
* Any computer that’s using Thorlabs drivers will need the pyvisa package installed
  + In Anaconda Powershell run: pip install pyvisa
  + Check version (should be at least 1.11.3) with: pip list

Install [Microsoft Visual Studio Community 2019](https://visualstudio.microsoft.com/downloads/)

* Community version (free)
  + Currently Microsoft doesn’t seem to support downloading older specific versions of the Community version of Visual Studio (options for Enterprise and Professional), unfortunately we’ll be resigned to just having different versions on each computer for the meantime
* Workloads
  + Python
  + Desktop development with C++
* After installation, run Visual Studio 2019 (click: Start after installation)
  + [Don’t sign in, click “Not now, maybe later.”]
  + <Get started>
  + Choose Dark color theme
  + <Start Visual Studio>
  + <Continue without code -> 2
  + Tools -> Options -> Text Editor -> C/C++ -> Tabs -> Tab size: 2, Indent size: 2, Insert Spaces
  + Tools -> Options -> Text Editor -> C/C++ -> Code Style -> Formatting -> General -> Default formatting style -> Google
  + Tools -> Options -> Text Editor -> C/C++ -> Formatting -> Indentation -> Position of preprocessor directives -> Leave indented
  + Tools -> Options -> Text Editor -> C/C++ -> Formatting -> New Lines ->
    - Position of open braces for namespaces: check “Keep on the same line …”
    - Same for types, functions, control blocks, lambdas
    - Position of Keywords: uncheck "place 'else' on a new line"
  + OK
  + At top of UI, change dropdown from Debug to Release
  + Quit Visual Studio

Install [7-zip](https://www.7-zip.org/) ( <https://www.7-zip.org/> )

* Download: 64-bit X64
* Take the latest non-alpha build

Install [CONEX-CC\_Installer\_Win10](https://www.newport.com/f/linear-actuators-with-conex-controller) [Structural Only]

* ( <https://www.newport.com/f/linear-actuators-with-conex-controller>
* click “RESOURCES” at top right
* Click CONEX-CC\_Installer\_Win10 ZIP to download
* Right-click to extract
* Click CONEX-CC Utility Installer Win64.exe (note: only see CONEX-PP Utility Installer Win32.exe)
* <Next><Next>etc.<Finish>

Rigol: Need [NI-VISA drivers](http://www.ni.com/en-us/support/downloads/drivers/download/packaged.ni-visa.305862.html) to communicate with Rigols

* Go to [v19.0](http://www.ni.com/en-us/support/downloads/drivers/download/packaged.ni-visa.305862.html)
* Note: this driver will disable Windows Fast Startup
* Do not include any “Additional items you may wish to install” <deselect/uncheck all>
* <next><accept><next><etc.>
* No, do not allow NI to check for updates automatically.
* reboot/restart
* Log back in as Lab

NiDAQ: USB-6002 (used to capture photodiode data and moglabs PID) [Structural Only]

* Go to [NI-DAQmx Download](https://www.ni.com/en-us/support/downloads/drivers/download.ni-daqmx.html#325032) and click Download.
  + If for some reason, the downloaded exe doesn’t work, instead of clicking download, find the Note in the bottom right that says “If you need to download individual versions or patches… click on: [Individual Offline Installers](https://www.ni.com/en-us/support/downloads/drivers/download.ni-daqmx.html#show-offline-installers)
  + Download 19.5.0 NI-DAQmx 19.5 - this will save an .iso image. Double-click that to mount it, then click “autorun.exe”.
* Deselect all “Additional items you may wish to install:” (button, bottom left).
* Select only “NI-DAQmx Support for C”.
* <Next><Accept><Accept><Next>etc.
* Always trust content from NI.com, Install.
* No I do not want to participate.
* <Reboot>

**Openwater’s Software Setup**

Log in as lab

In C:/ make “Openwater” dir

In C:/Openwater/ make ”scan” dir

In C: make data\_scans dir

Open a shell window: Start menu -> Git -> Git Bash, then enter these commands:

$ cd /c/Openwater/scan

$ git clone https://github.com/OpenwaterInternet/opw\_acousto-optic\_sw.git

**OW Camera Setup**

To allow installation of unsigned drivers (needed for camera):

* Windows start menu >> change advanced startup options >> Advanced Startup [click restart now] >> Troubleshoot >> Advanced Options >> Startup Settings
* When computer reboots, click F7 (or whatever option allows unsigned driver)
* \*\*\* Note: if the computer screen stays black following the reboot, try switching to the other hdmi port, or use a display port cable \*\*\*
* In File Explorer, open C:/Openwater/scan/opw\_acousto-optic\_sw/system/component/driver/
  + <right click> on ow-fx3-driver.zip, 7-Zip -> Extract to “ow-fx3-driver”
  + Get a Gumstick attached.
  + Plug the Gumstick into a USB 3 port. Windows10 will probably say something like “We’re setting up device Westbridge.”
* Search box: type “Device Manager” (Control panel) and launch it
* Other Devices -> WestBridge, right-click Update driver -> Browse my computer for driver software: C:/Openwater/scan/opw\_acousto-optic\_sw/system/component/driver/ow-fx3-driver
* Click Include subfolders
* <Next>
* <Close>

Camera USB Cable Requirements

* Must be USB-A to USB-C cables meant for both data and charging
* Model that we have found to consistently work:
  + [Digi-Key part number: USBC22KM302M0USBAMBKR from GlobTek Inc.](https://www.digikey.com/short/w3wq2v11) (preferred)
  + [Amazon Basics Double Braided Nylon USB Type-C to Type-A 3.1 Gen 1 Cable, 6 feet](https://www.amazon.com/dp/B07D7TXTPP) (alternate)

# Gumstick Cameras

## **1. How do I program a new serial number onto a new (out of the box) gumstick (anything that uses an FX3 communication scheme)?**

Amended by EM (01/21/2021)

* you need to have the cypress fx3 dev kit from here: <https://www.cypress.com/documentation/software-and-drivers/ez-usb-fx3-software-development-kit>
* Make sure computer has a clone of the Openwater opw\_acousto-optic\_sw:

https://github.com/OpenwaterInternet/opw\_acousto-optic\_sw

In anaconda3 prompt:

cd C:/Openwater/scan/opw\_acousto-optic\_sw/system/util

python makeserial.py <device> <serial number>

*In this case <device> should either be ‘gumstick’ and serial number should be the next in the sequence of (ie: typically first digit refers to version number) numbers in the equipment log. Please remember to add your new device to the equipment log.*

**To unprogram serial number from fx3 device**

Cypress -> Control Center

Expand “Openwater Gumstick” in menu on left hand side [note: must show up as openwater gumstick (programmed), if not, run fftutil or scanner to program it]

Expand “Config 1”

Click control endpoint

Go to data transfer tab

Text to send:

Data to send:

Bytes to transfer: 0

Direction: out

Request type: Vendor

Target: Device

Request Code: 0xE2

Value: 0x0

Index: 0x0

Click transfer data

Give it a minute (no indicator)

**Build the scan executable(s)**

In the still-open bash shell:

$ git clone https://github.com/OpenwaterInternet/openwater-imaging.git

$ cd opw\_acousto-optic\_sw

$ git pull

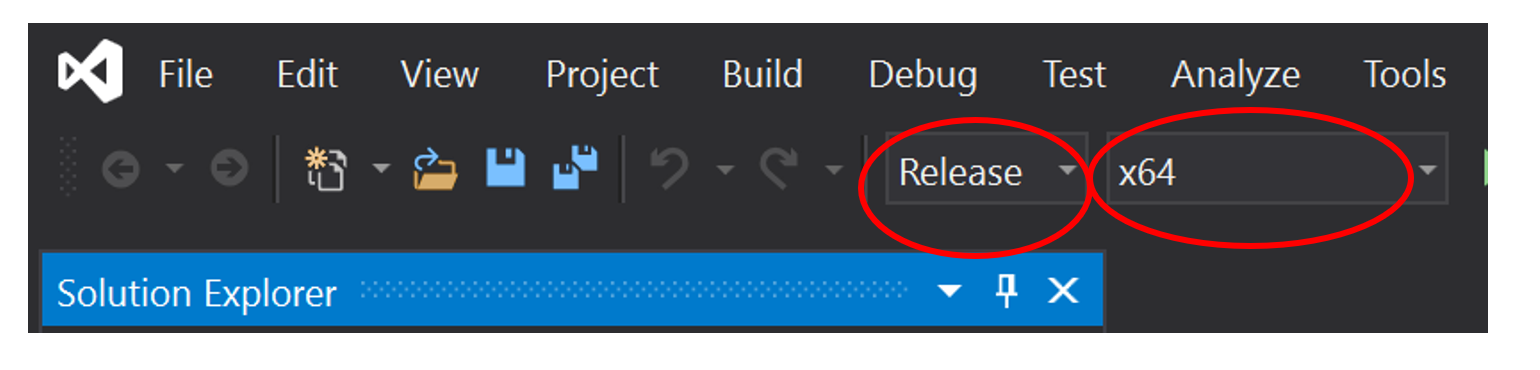
Edit system env variables: type “env” in search box

(It pulls up Control Panel : System Properties, click “Environment Variables” button at bottom)

Under system variables, click “Path”, click Edit button, then click New and type in C:/Openwater/scan/opw\_acousto-optic\_sw/system/third\_party/dll <OK>

<OK>

Start menu -> Visual Studio 2019

* <Don’t log in to microsoft>
* Dark mode
* Continue without code (at bottom)
* File->Open -> Project/Solution, select C:/Openwater/scan/opw\_acousto-optic\_sw/system/scanner/OpenwaterScaningSystem\_Pulsed (Note this has both pulsed and CW, we should probably clean up/rename...>)
* Change the build configuration: ‘Release’ and ‘x64’
* Build->Project Only->Build Only OpenwaterScanningSystem\_Pulsed
* Build->Project Only->Build Only fftutilExplora

Check that the Environment variable just set above (adding PATH) includes C:/Openwater/scan/opw\_acousto-optic\_sw/system/third\_party/dll

Control Panel: System -> Advanced system settings -> Environment Variables

(hopefully it should be there)

Start->Anaconda3->Anaconda Powershell Prompt

(base) > cd C:/Openwater/scan/opw\_acousto-optic\_sw/system

(base) > conda config --add channels conda-forge

(base) > conda create -n owi --file requirements\_scanner.txt

(base) > conda activate owi

(owi) > pip install -e . <Note: yes, include the “.” as it means this local dir>

(owi) > conda install python==3.7.3

(owi) > conda install pywin32 <although looks like this might have already been installed, so nothing is done>

(owi) > conda update numpy <already done as well>

(owi) > conda update scikit-image <already done as well>

(owi) > conda list <Note: use this command to see package versions, if you are curious>

(owi) > exit

Ok, Let’s test that things are working…

To do this, plug in a camera (e.g. gumstick camera as was used prior/above)

Fire up Anaconda Power Shell:

(base) > conda activate owi

(owi) > cd C:/Openwater/scan/opw\_acousto-optic\_sw/system/app/scanUI

(owi) > jupyter notebook

<brings up chrome browser (should be set as default browser, if not, make it so)>

Click “scanUI.ipynb”

Note: In # Configuration, check that “aligner = ‘fftutil.exe’” if you are using a CW system, master defaults to ‘fftutilExplora.exe’

Note: In # Configuration, also check that “SCAN\_DATA = ‘C:/data\_scans/’ (vs. D:) if you have put the ‘data\_scans’ dir there

Kernel -> Restart & Run All

<wait a bit>

In Align tab at bottom of page, click “Align” grey box to run fftutil.

Note: if this is your first time, the fftutil might take a long time to run as it is testing code to determine how to optimize the ftt computation. (Be patient.)

If everything is working, your camera should be responding to light and dark.

Quit/Exit.