Getting Started With PICam

Sabbir Liakat

Applications Specialist

Outline

What is PICam?

Hardware requirements

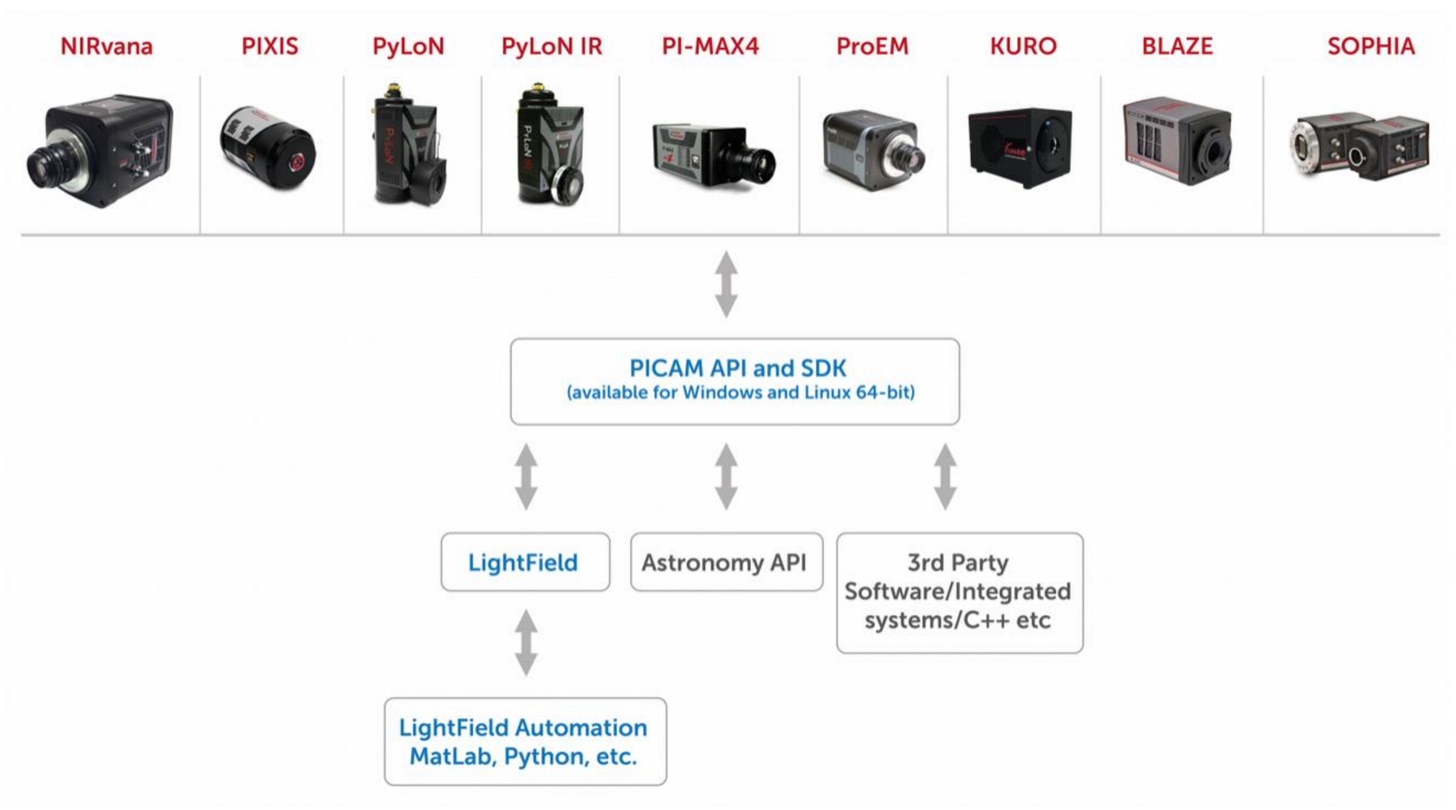
Installation Information

Included reference material: Sample projects and code

What is PICam?

PICam is an SDK that allows one to control Teledyne Princeton Instruments cameras. PICam is the engine that powers our LightField application.

PICAM Integration



NIRvana

PIXIS

PyLoN

PyLoN IR

PI-MAX4

ProEM

KURO

BLAZE

SOPHIA

PICAM API and SDK
(available for Windows and Linux 64-bit)

LightField

Astronomy API

3rd Party Software/Integrated systems/C++ etc

LightField Automation MatLab, Python, etc.

Hardware Requirements

You will not need a supercomputer (though more memory will not hurt!)

Minimum system requirements:

- Camera: A Teledyne Princeton Instruments camera with internal controller and the appropriate interface card installed in the host computer:
  - USB 3.0
  - USB 2.0
  - GigE
- 64-bit operating system
  - Windows 7 / 8 / 8.1 / 10
  - Linux (we officially support the CentOS 7 distribution)
- Computer Specs (Minimum):
  - 2GHz dual core processor
  - 4GB RAM
  - 1GB storage

We recommend operating PICam on Windows 10, with as much RAM and storage as possible.

Installation Information

Windows Applications

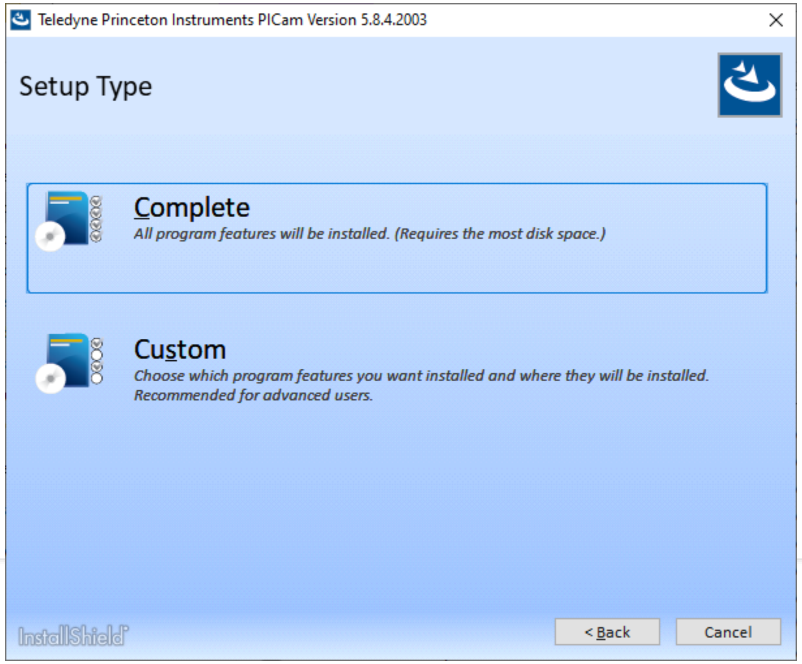
The PICam installer can be found here:

- ftp://ftp.princetoninstruments.com/Public/Software/Official/PICam/PICam%20Install.exe
- This link will always contain the latest release of PICam.

After downloading the installer, just run and follow the prompts. You will need Administrator privileges to properly install the software.

We recommend performing a Complete install of PICam.

Reboot PC after the install finishes.



Teledyne Princeton Instruments PICam Version 5.8.4.2003

Setup Type

Complete

All program features will be installed. (Requires the most disk space.)

Custom

Choose which program features you want installed and where they will be installed. Recommended for advanced users.

Installation Information

Linux Applications

The Linux PICam installer can be found here:

- ftp://ftp.princetoninstruments.com/Public/Software/Official/PICam/Picam_SDK.run
- This link will always contain the latest release of PICam.

After downloading the installer, convert to an executable with the chmod +x command.

Run the installer with the sudo command, or as a root user.

Reboot PC after the install finishes.

Included Reference Material

Programmer’s Manual and Sample Code

Teledyne Princeton Instruments supports PICam with a library of reference material, which includes:

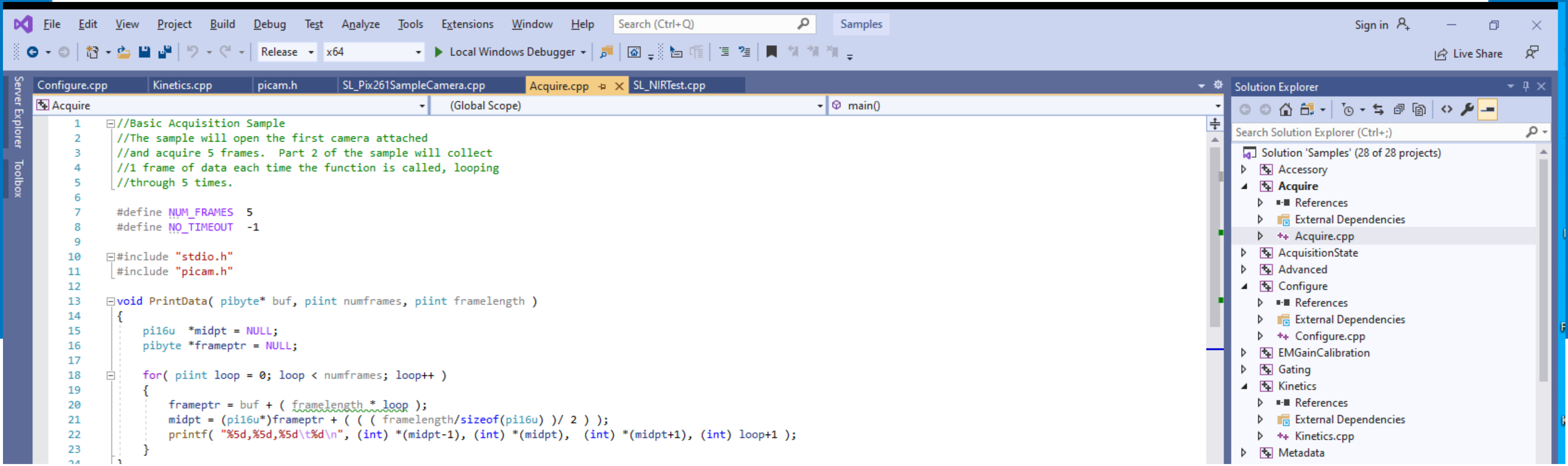
- A comprehensive programmer’s manual which includes a full description of the SDK structure, as well as details about all functions, parameters, and types used in the SDK
  - Link to Programmer’s Manual: https://www.princetoninstruments.com/products/software-family/pi-cam
  - Scroll near the bottom of the page; you will find the link to the latest manual under the ‘Resources’ → ‘Manuals’ section.
- A library of sample code and ready-to-compile projects.
  - C++ source code that covers the vast majority of applications with the SDK, including: acquisition (asynchronous and synchronous), regions of interest, camera configuration, event call-backs, and more!
  - Sample code and projects will automatically install to the following locations:
    - Windows: C:\Users\Public\Documents\Princeton Instruments\Picam\Samples
    - Linux: \opt\Princeton Instruments\Picam\samples\

Using the Sample Projects

Visual Studio

In the example below, I load the ‘Samples.sln’ solution from “C:\Users\Public\Documents\Princeton Instruments\Picam\Samples\projects\vs2010” into Visual Studio.

A great way to learn PICam is to walk through the sample source code and corroborate the function / parameter/ type with the details in the Programmer’s Manual.



```
//Basic Acquisition Sample
//The sample will open the first camera attached
//and acquire 5 frames.  Part 2 of the sample will collect
//1 frame of data each time the function is called, looping
//through 5 times.

#define NUM_FRAMES  5
#define NO_TIMEOUT  -1

#include "stdio.h"
#include "picam.h"

void PrintData( pibyte* buf, piint numframes, piint framelength )
{
    pi16u  *midpt = NULL;
    pibyte *frameptr = NULL;

    for( piint loop = 0; loop < numframes; loop++ )
    {
        frameptr = buf + ( framelength * loop );
        midpt = (pi16u*)frameptr + ( ( ( framelength/sizeof(pi16u) )/ 2 ) );
        printf( "%5d,%5d,%5d\t%d\n", (int) *(midpt-1), (int) *(midpt),  (int) *(midpt+1), (int) loop+1 );
    }
}
```

TELEDYNE PRINCETON INSTRUMENTS
Everywhereyoulook™

Extended Applications

You can use PICam to take your experiment to new heights!

You are not limited to just C / C++; after becoming familiar with the API, you can consider projects such as:

- Building GUIs for camera control and image analysis
- Porting to Python for integration with image processing and machine learning libraries
  - Translating the c++ source to Python with the ctypes library is fairly easy and opens up a wealth of applications possibilities.

PICam also allows for the use of software emulated “demo” cameras. Use of such emulated models is covered in several examples, including the ‘Acquire’ project.

- Using the demo cameras, one can build an application without needing a live camera.
- This is excellent for productive remote development!

If You Need Additional Help…

Cross-referencing the sample code and details the Programmer's Manual should allow you to get going running the camera in no time!

If you have additional issues, we can help!

Just ping us at pi.techsupport@Teledyne.com, and we'll take care of your issues and answer your questions.

Our support team includes PhD level scientists who are as enthusiastic about cutting-edge scientific research as you are!