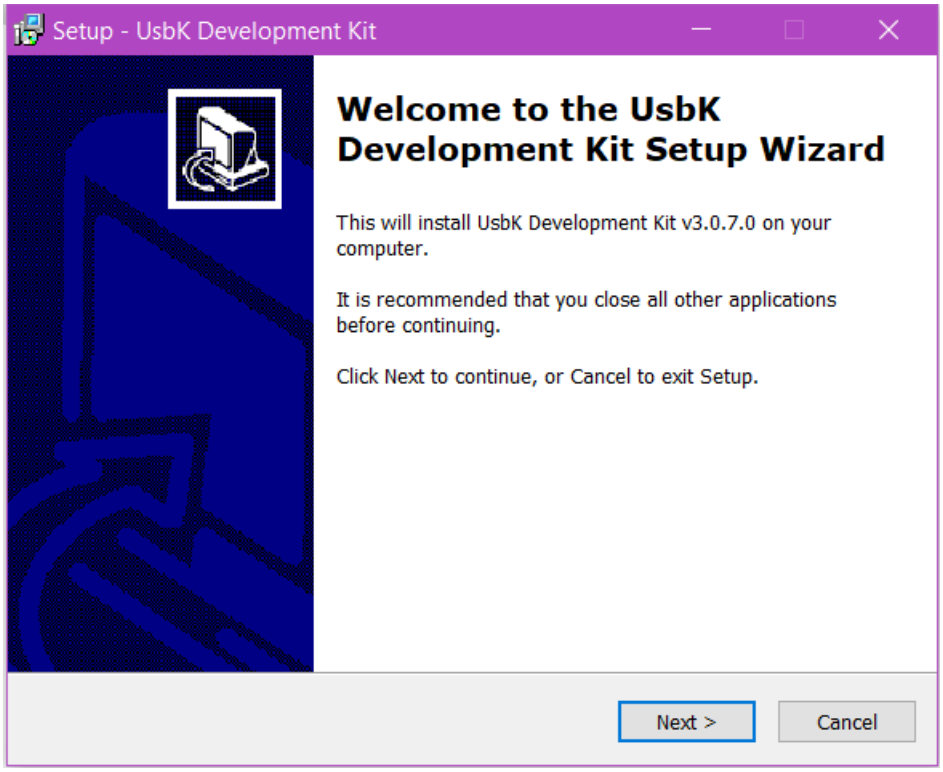
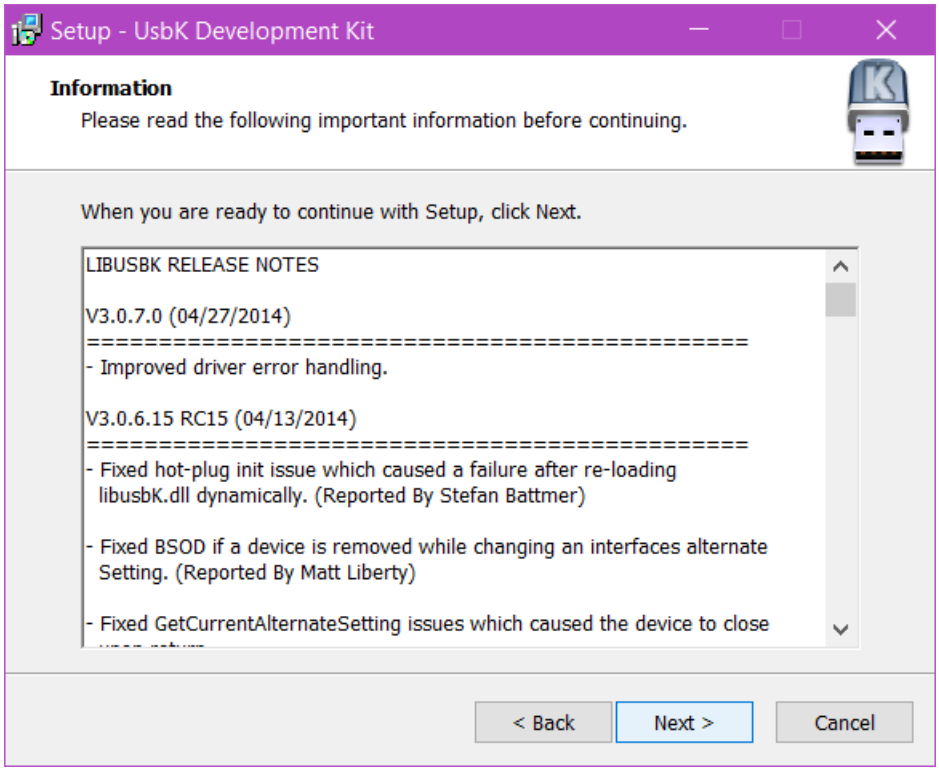
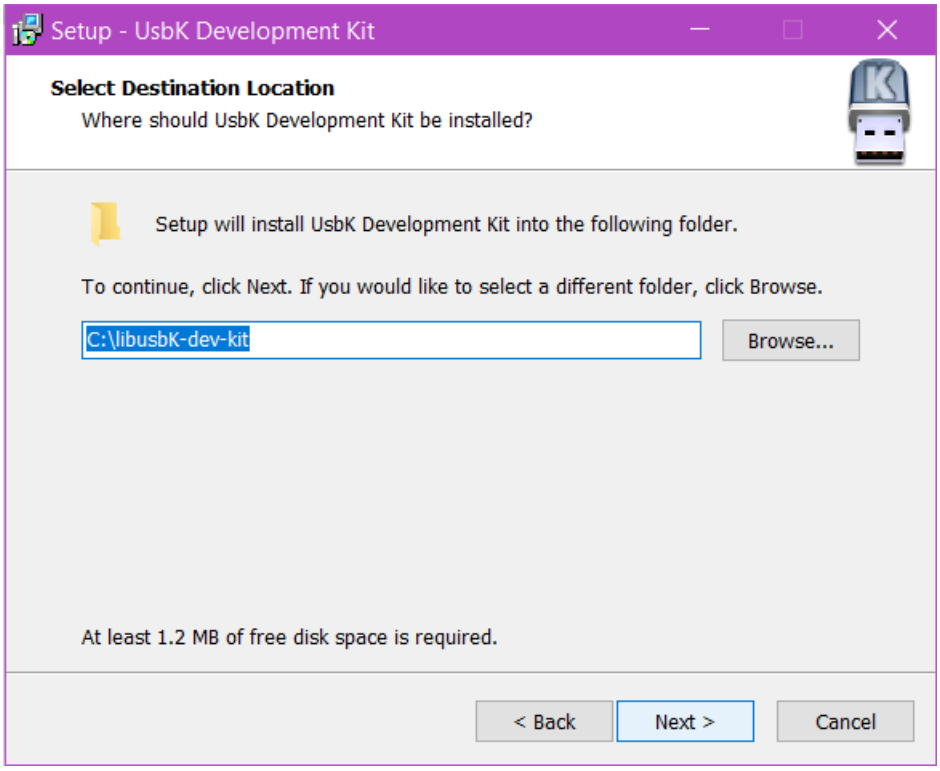
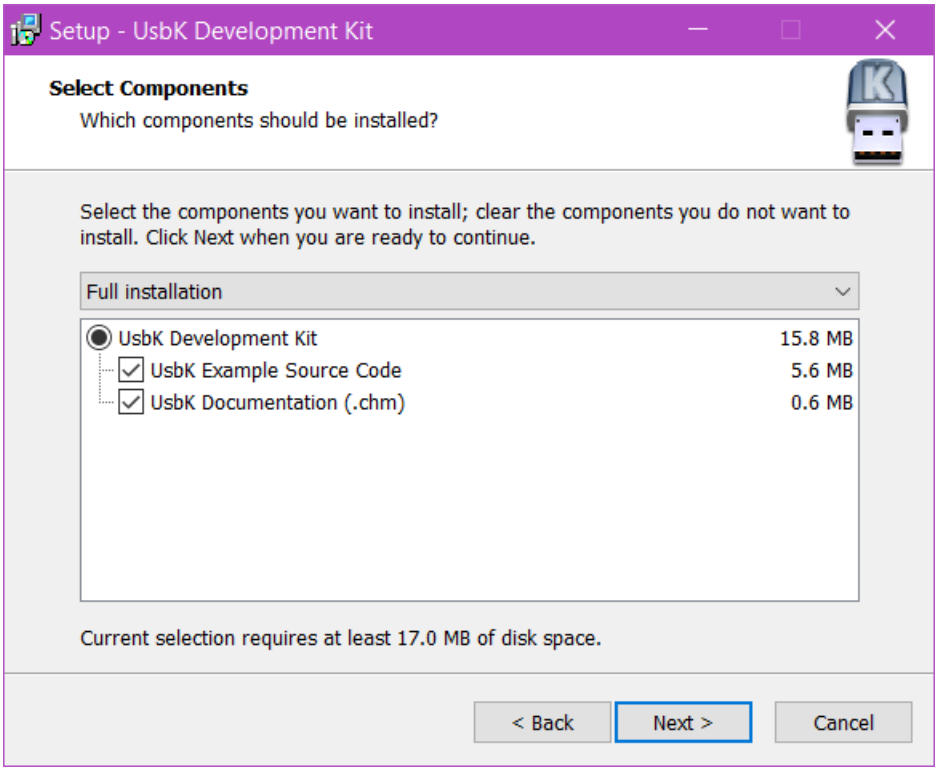
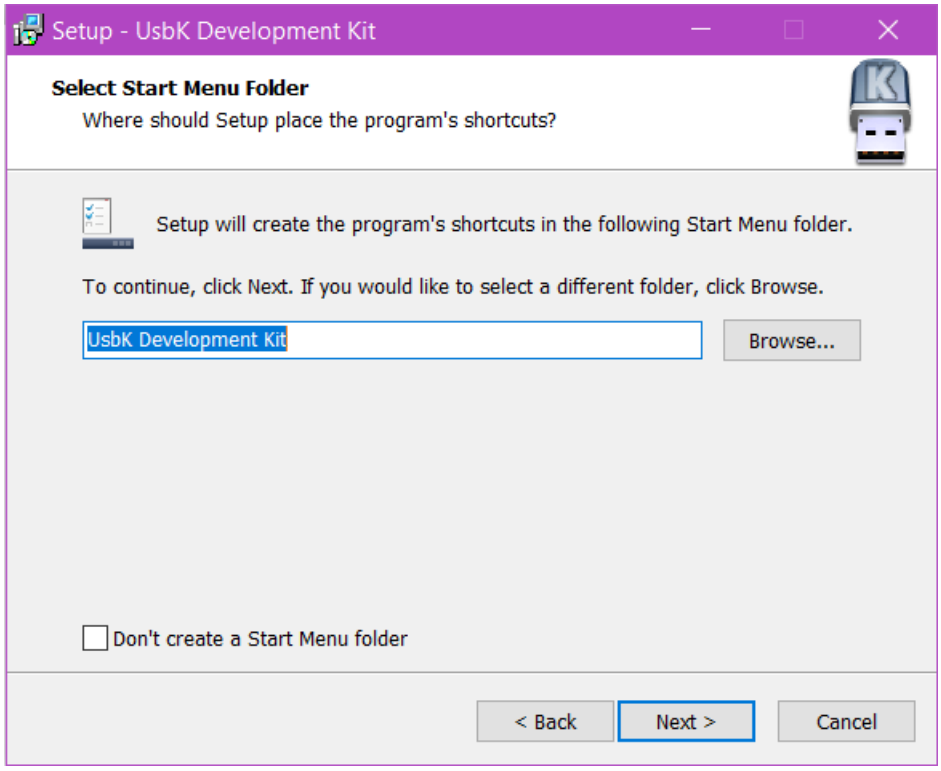
Double click on file ‘libusbK-3.0.7.0-setup.exe’ to start installation.

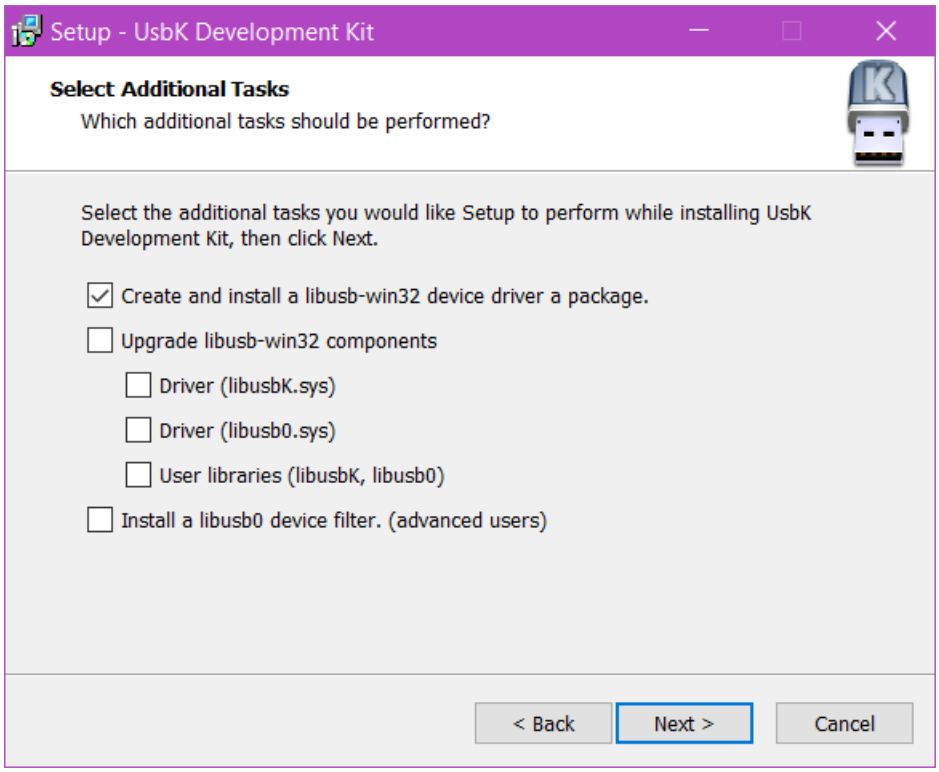


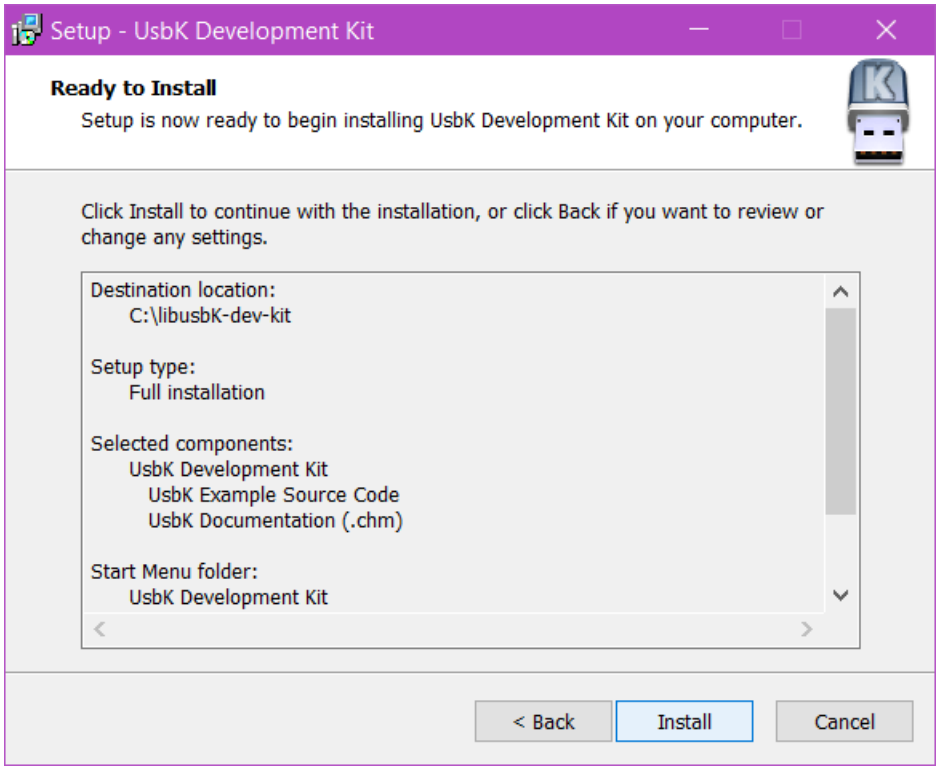


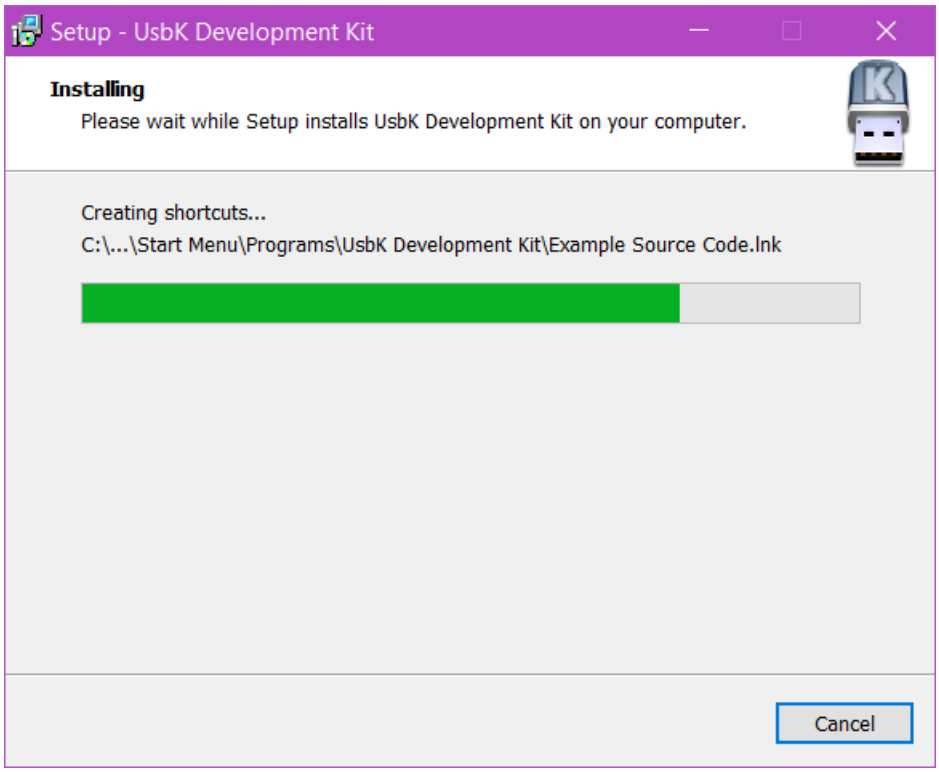

Setup - UsbK Development Kit
Select Destination Location
Where should UsbK Development Kit be installed?
Setup will install UsbK Development Kit into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\libusbK-dev-kit
Browse...
At least 1.2 MB of free disk space is required.
< Back
Next >
Cancel

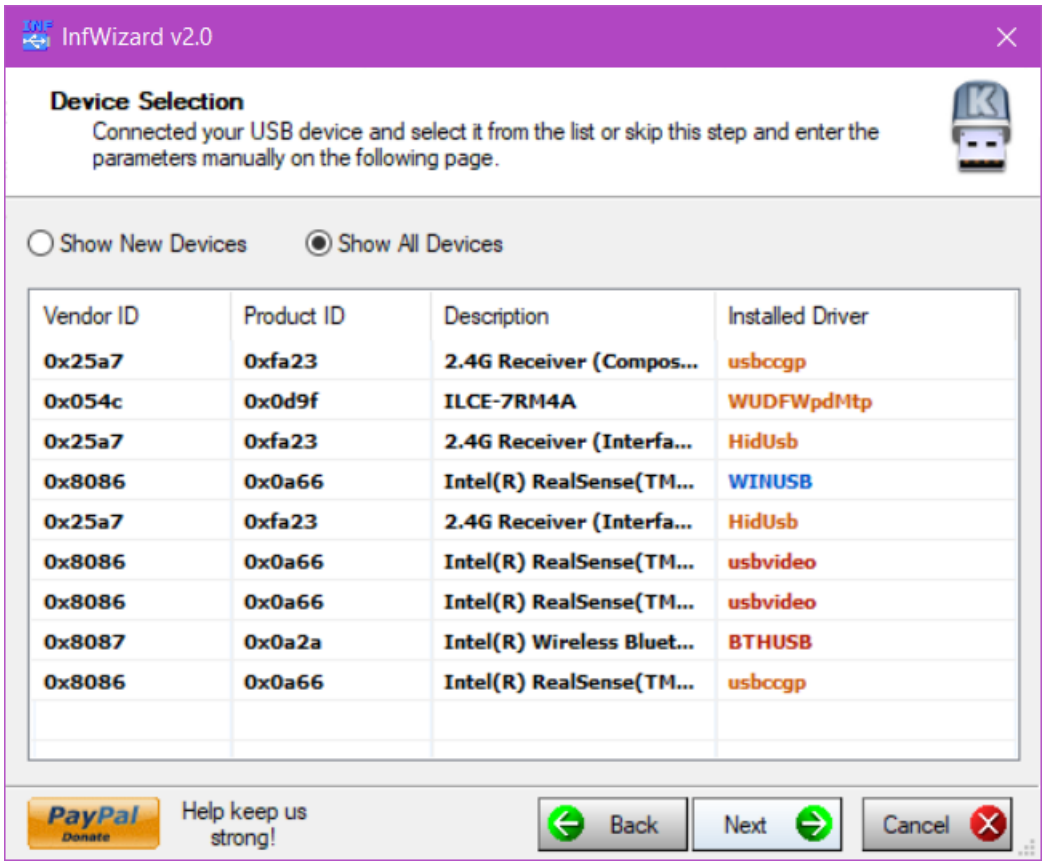

Setup - UsbK Development Kit
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Full installation
UsbK Development Kit 15.8 MB
UsbK Example Source Code 5.6 MB
UsbK Documentation (.chm) 0.6 MB
Current selection requires at least 17.0 MB of disk space.
< Back
Next >
Cancel


Setup - UsbK Development Kit
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
UsbK Development Kit
Browse...
Don't create a Start Menu folder
< Back
Next >
Cancel

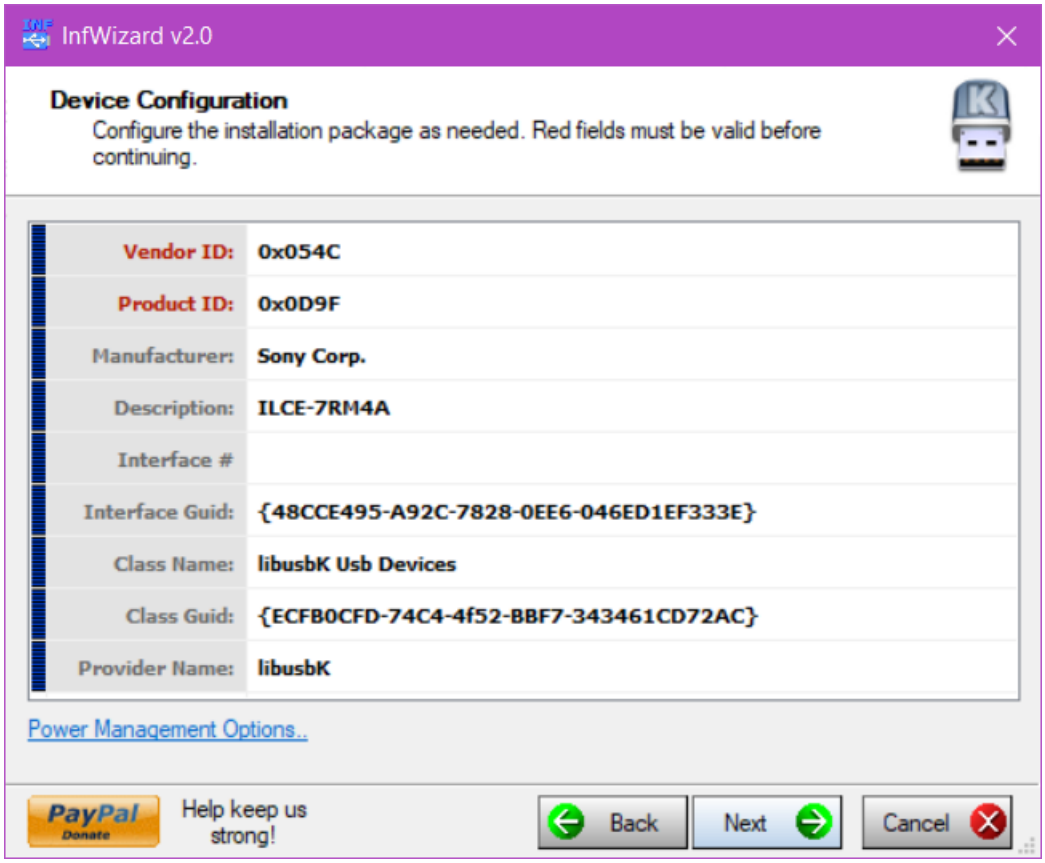

Setup - UsbK Development Kit
Select Additional Tasks
Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing UsbK Development Kit, then click Next.
Create and install a libusb-win32 device driver a package.
Upgrade libusb-win32 components
Driver (libusbK.sys)
Driver (libusb0.sys)
User libraries (libusbK, libusb0)
Install a libusb0 device filter. (advanced users)
< Back
Next >
Cancel

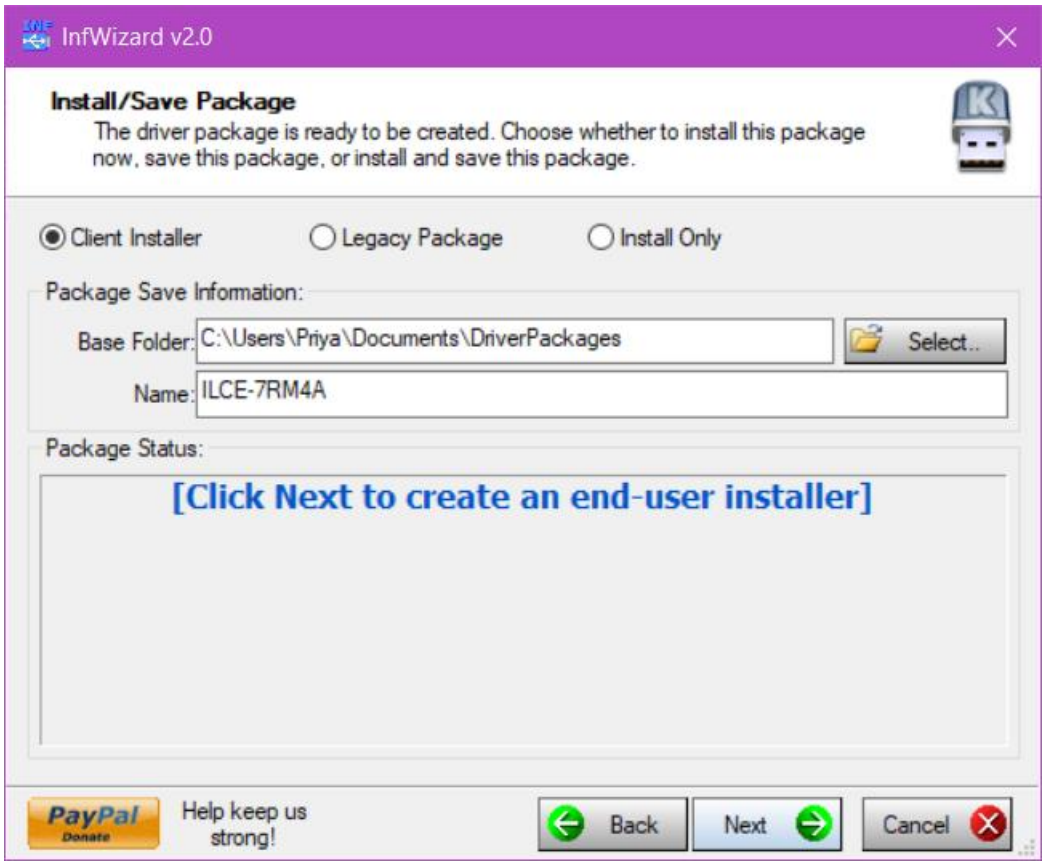

Setup - UsbK Development Kit
Ready to Install
Setup is now ready to begin installing UsbK Development Kit on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
C:\libusbK-dev-kit
Setup type:
Full installation
Selected components:
UsbK Development Kit
UsbK Example Source Code
UsbK Documentation (.chm)
Start Menu folder:
UsbK Development Kit
< Back
Install
Cancel

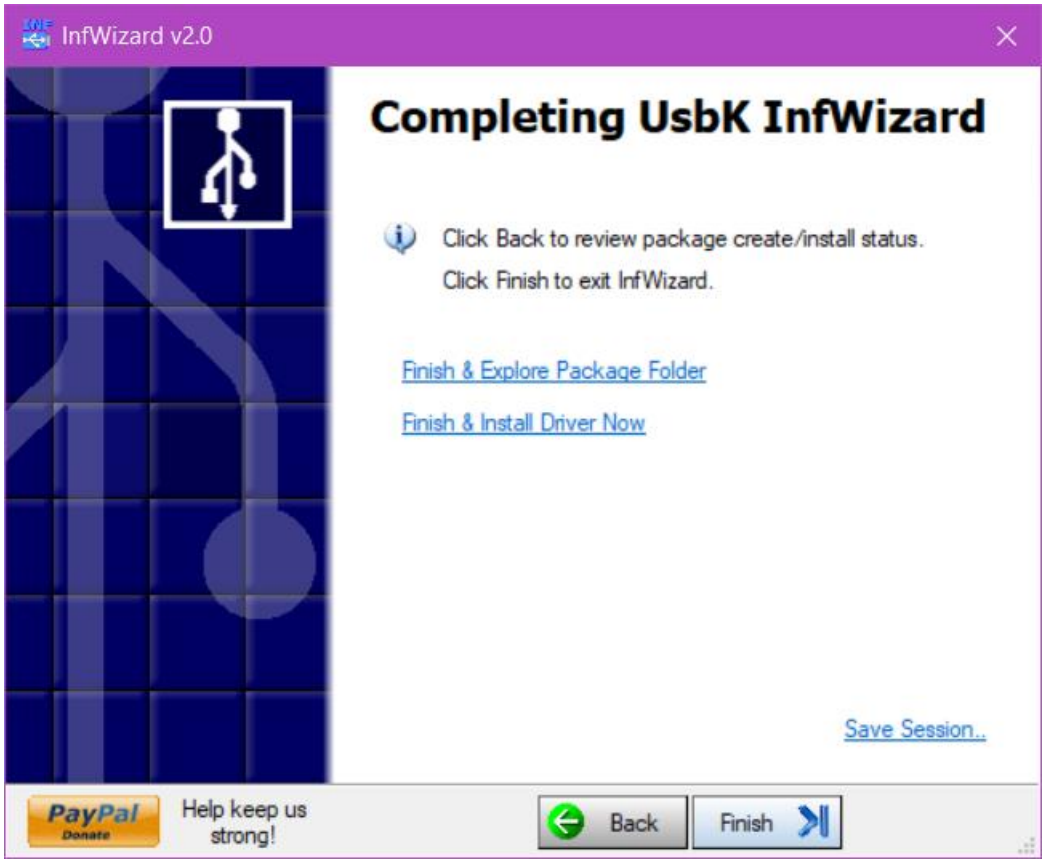

Setup - UsbK Development Kit
Installing
Please wait while Setup installs UsbK Development Kit on your computer.
Creating shortcuts...
C:\...\Start Menu\Programs\UsbK Development Kit\Example Source Code.lnk
Cancel



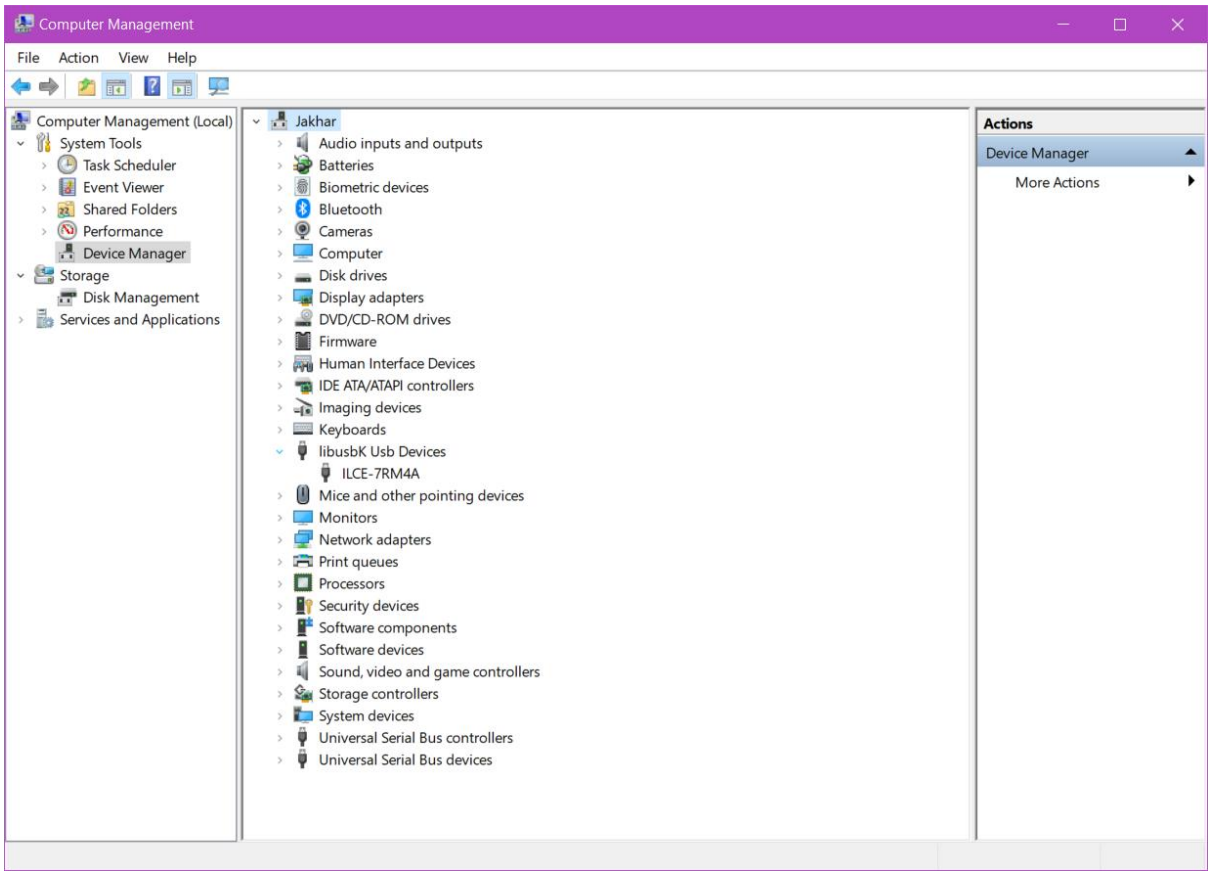
Choose the camera (in my case it’s ILCE-7RM4A)


InfWizard v2.0
Device Configuration
Configure the installation package as needed. Red fields must be valid before continuing.
Vendor ID: 0x054C
Product ID: 0x0D9F
Manufacturer: Sony Corp.
Description: ILCE-7RM4A
Interface #
Interface Guid: {48CCE495-A92C-7828-0EE6-046ED1EF333E}
Class Name: libusbK Usb Devices
Class Guid: {ECFB0CFD-74C4-4f52-BBF7-343461CD72AC}
Provider Name: libusbK
Power Management Options..
PayPal Donate
Help keep us strong!
Back
Next
Cancel


InfWizard v2.0
Install/Save Package
The driver package is ready to be created. Choose whether to install this package now, save this package, or install and save this package.
Client Installer
Legacy Package
Install Only
Package Save Information:
Base Folder: C:\Users\Priya\Documents\DriverPackages
Select..
Name: ILCE-7RM4A
Package Status:
[Click Next to create an end-user installer]
PayPal Donate
Help keep us strong!
Back
Next
Cancel



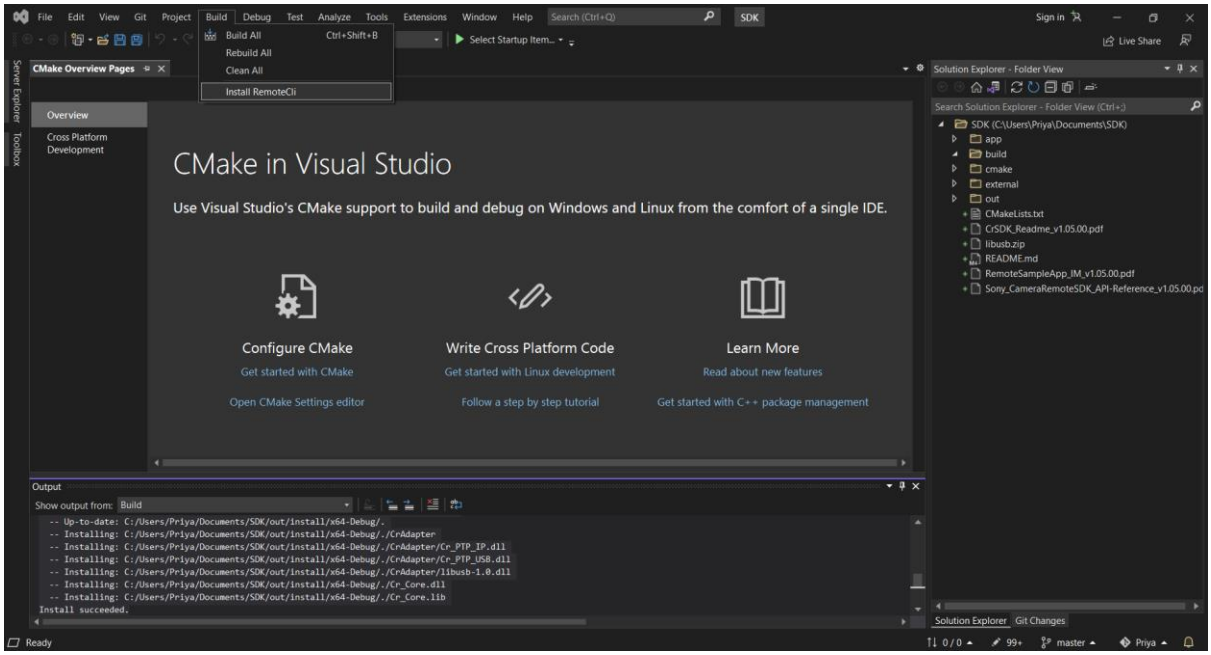
Check the device is now listed under ‘libusbK Usb Devices’ in Device Manager



Use VisualStudioSetup.exe to install Visual Studio if already not present. Download and extract file ‘CrSDK_v1.05.00_20211207a_Win64.zip’ in a folder called ‘SDK’. Or download one from

https://support.d-imaging.sony.co.jp/app/sdk/en/index.html

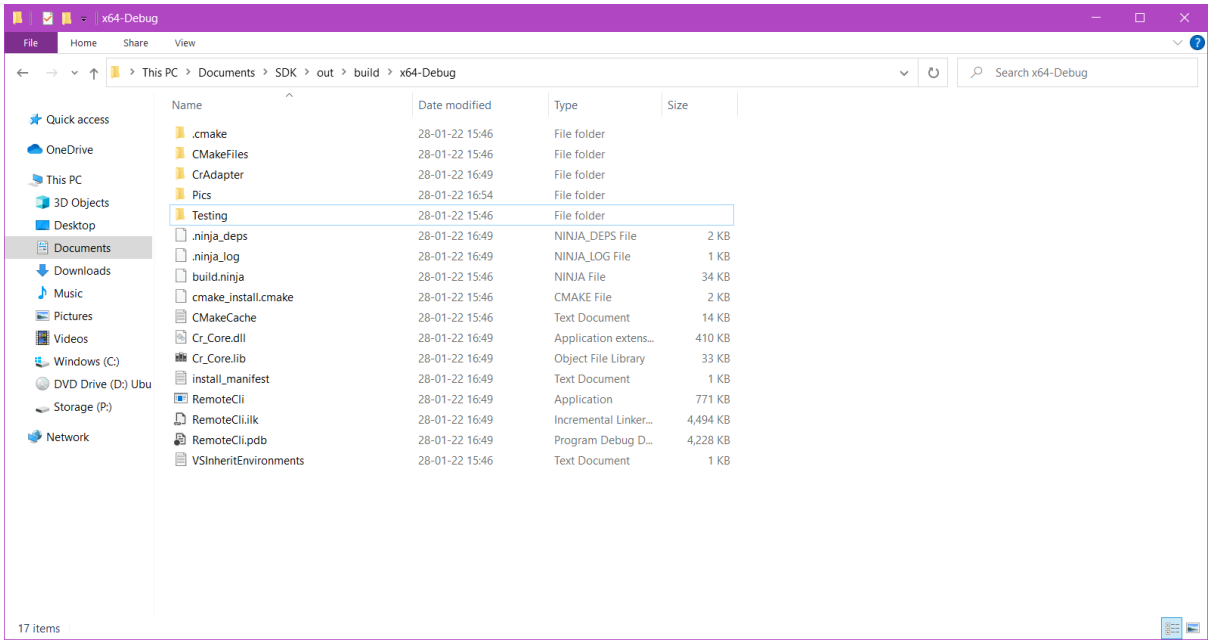
Create an empty folder called ‘build’ inside the SDK folder. Open the camera SDK directory in Visual Studio, and in the menu under ‘Build’ option, click on ‘Install RemoteCli’. You will get a success message on installation.




```
>------ Build started: Project: CMakeLists, Configuration:  ------
  [1/8] Building CXX object CMakeFiles\RemoteCli.dir\app\ConnectionInfo.cpp.obj
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
C:\Users\Priya\Documents\SDK\app\ConnectionInfo.cpp(50): warning C4996:
'sprintf': This function or variable may be unsafe. Consider using sprintf_s
instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for
details.
C:\Users\Priya\Documents\SDK\app\ConnectionInfo.cpp(57): warning C4996:
'mbstowcs': This function or variable may be unsafe. Consider using mbstowcs_s
instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for
details.
C:\Users\Priya\Documents\SDK\app\ConnectionInfo.cpp(65): warning C4996:
'sprintf': This function or variable may be unsafe. Consider using sprintf_s
instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for
details.
C:\Users\Priya\Documents\SDK\app\ConnectionInfo.cpp(71): warning C4996:
'mbstowcs': This function or variable may be unsafe. Consider using mbstowcs_s
instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for
details.
  [2/8] Building CXX object CMakeFiles\RemoteCli.dir\app\Text.cpp.obj
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
  [3/8] Building CXX object
CMakeFiles\RemoteCli.dir\app\PropertyValueTable.cpp.obj
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
  [4/8] Building CXX object CMakeFiles\RemoteCli.dir\app\MessageDefine.cpp.obj
```

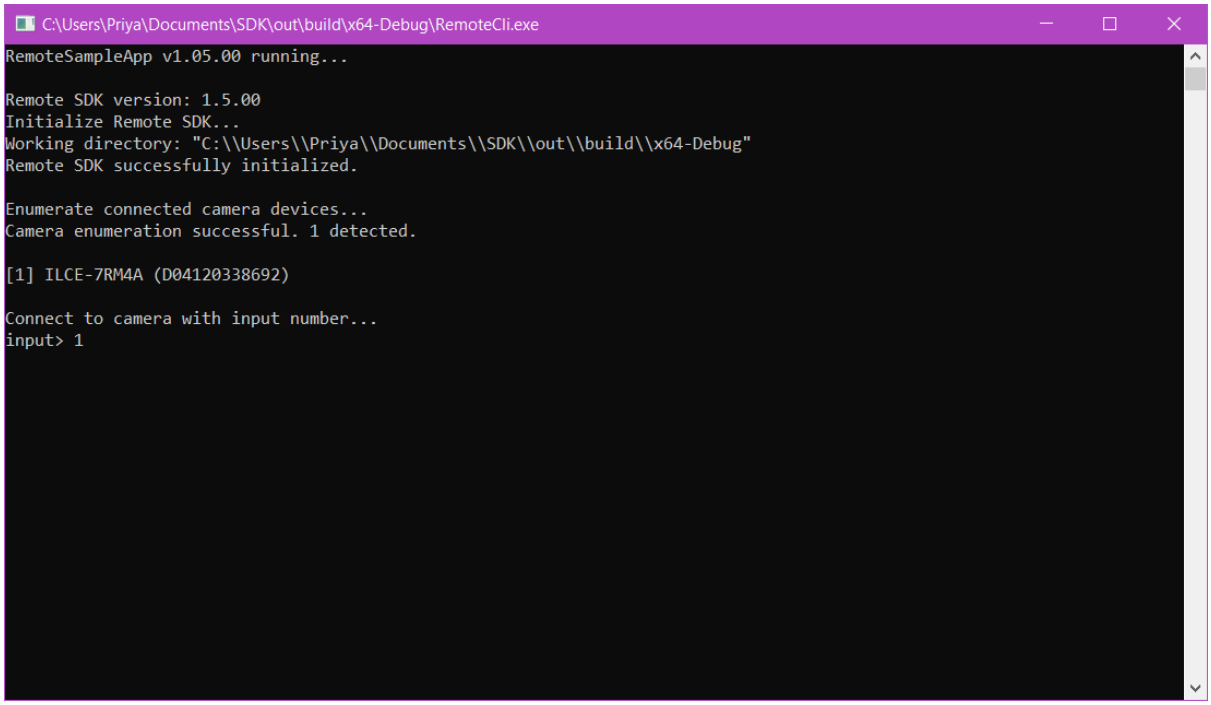
```
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
  [5/8] Building CXX object CMakeFiles\RemoteCli.dir\app\CameraDevice.cpp.obj
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
C:\Users\Priya\Documents\SDK\app\CameraDevice.cpp(1087): warning C4996:
'sprintf': This function or variable may be unsafe. Consider using sprintf_s
instead. To disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for
details.
C:\Users\Priya\Documents\SDK\app\CameraDevice.cpp(2272): warning C4018: '<':
signed/unsigned mismatch
C:\Users\Priya\Documents\SDK\app\CameraDevice.cpp(2302): warning C4018: '<':
signed/unsigned mismatch
  [6/8] Building CXX object CMakeFiles\RemoteCli.dir\app\RemoteCli.cpp.obj
C:\Users\Priya\Documents\SDK\cl : Command line warning D9002: ignoring unknown
option '-fsigned-char'
C:\Users\Priya\Documents\SDK\app\RemoteCli.cpp(291): warning C4267:
'initializing': conversion from 'size_t' to 'int32_t', possible loss of data
  [7/8] Linking CXX executable RemoteCli.exe
  [7/8] Install the project...
  -- Install configuration: "Debug"
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-
Debug/./RemoteCli.exe
  -- Up-to-date: C:/Users/Priya/Documents/SDK/out/install/x64-Debug/.
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-Debug/./CrAdapter
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-
Debug/./CrAdapter/Cr_PTP_IP.dll
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-
Debug/./CrAdapter/Cr_PTP_USB.dll
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-
Debug/./CrAdapter/libusb-1.0.dll
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-Debug/./Cr_Core.dll
  -- Installing: C:/Users/Priya/Documents/SDK/out/install/x64-Debug/./Cr_Core.lib
Install succeeded.
```

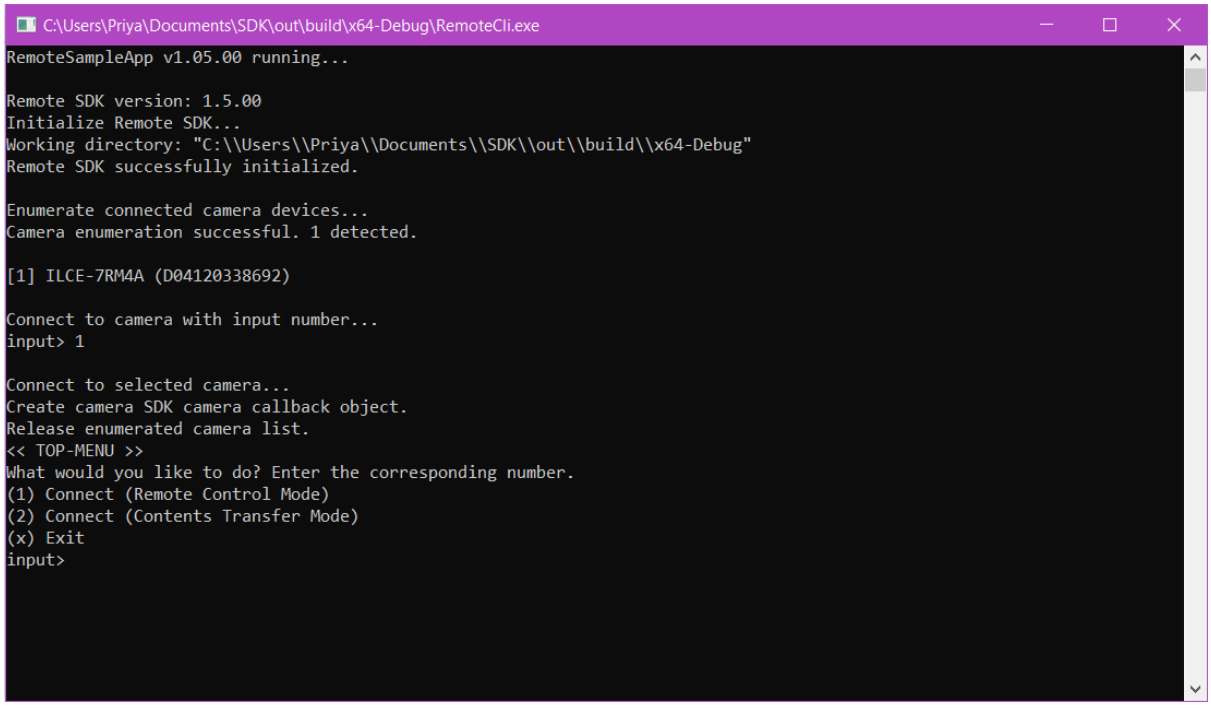
Go to SDK->out->build->x64-Debug directory

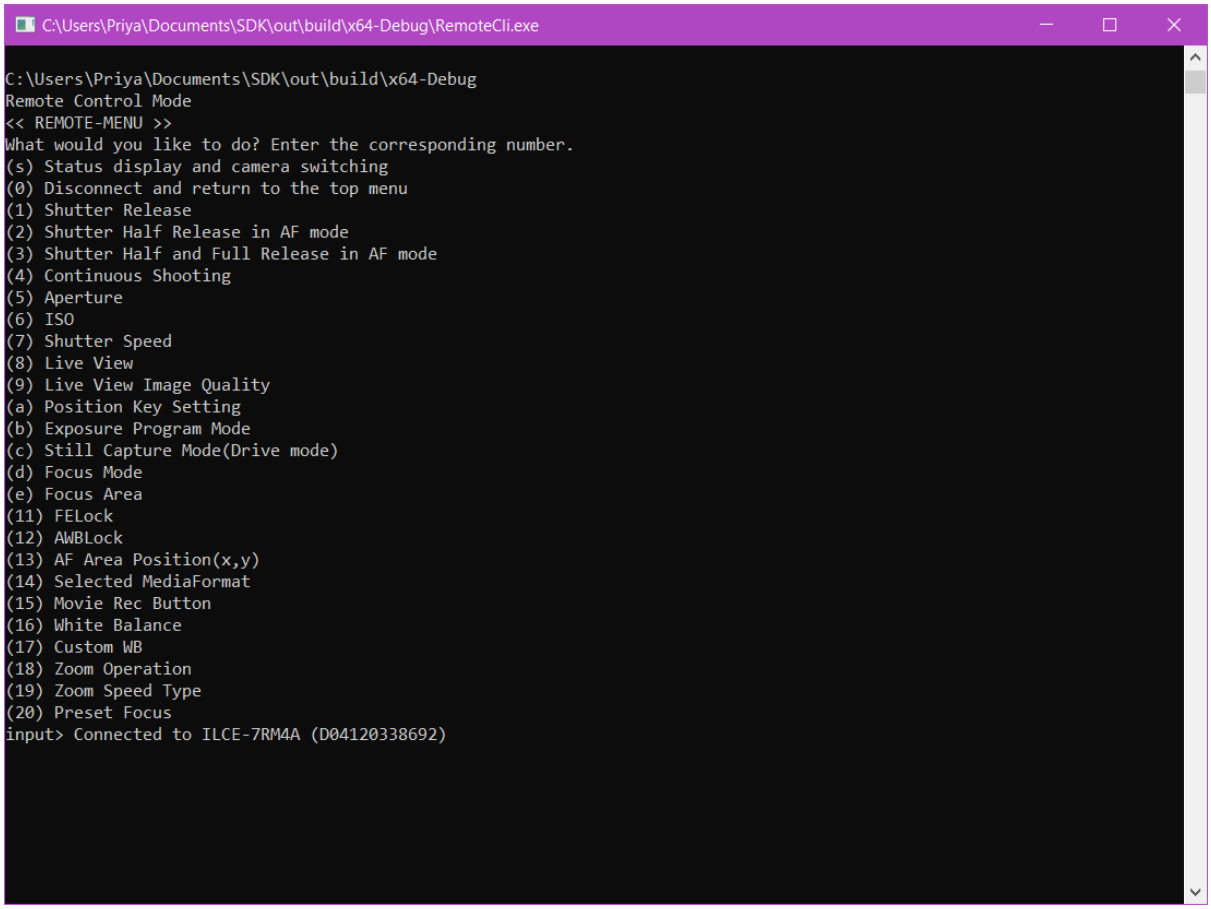


Double click on RemoteCli.exe (the one with type ‘Application’ here) to launch the demo script.

Select options from menus to explore different functions possible through this sample application.







Inside SDK/app/ folder, replace files RemoteCli.cpp and CameraDevice.cpp with the ones provided in the repository Setup Files directory. Then do the 'Install RemoteCli' step again in Visual Studio. This time when you run the RemoteCli.exe, it will capture and save image from the camera in the current directory without any user input.

Setup for GUI:

Install python
Install pip:

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python get-pip.py
```

Create and activate a python virtual environment:

```
python3 -m venv myenv
myenv\Scripts\activate.bat
pip install -U socketIO-client
pip install pathlib2
pip install -U paho-mqtt
```

Copy files 'GUI.py' and 'MachineMotion.py' from Setup Files directory into SDK. Run the GUI script:

```
python3 GUI.py
```