**Сборка библиотеки OpenCV с помощью CMAKE и Microsoft Visual Studio 2019**

1. Скачать и установить cmake: https://cmake.org/download/
2. Cкачать и установить Visual Studio Community 2019 (на момент написания документа доступной для скачивания версией без авторизации была Visual Studio Community 2022. Последний cmake ее поддерживает, поэтому изменения в инструкции затрагивают только выбор компилятора)
3. Скачать исходники OpenCV: https://opencv.org/releases/
4. Открыть cmake. Внешний вид окна показан на рис. 1

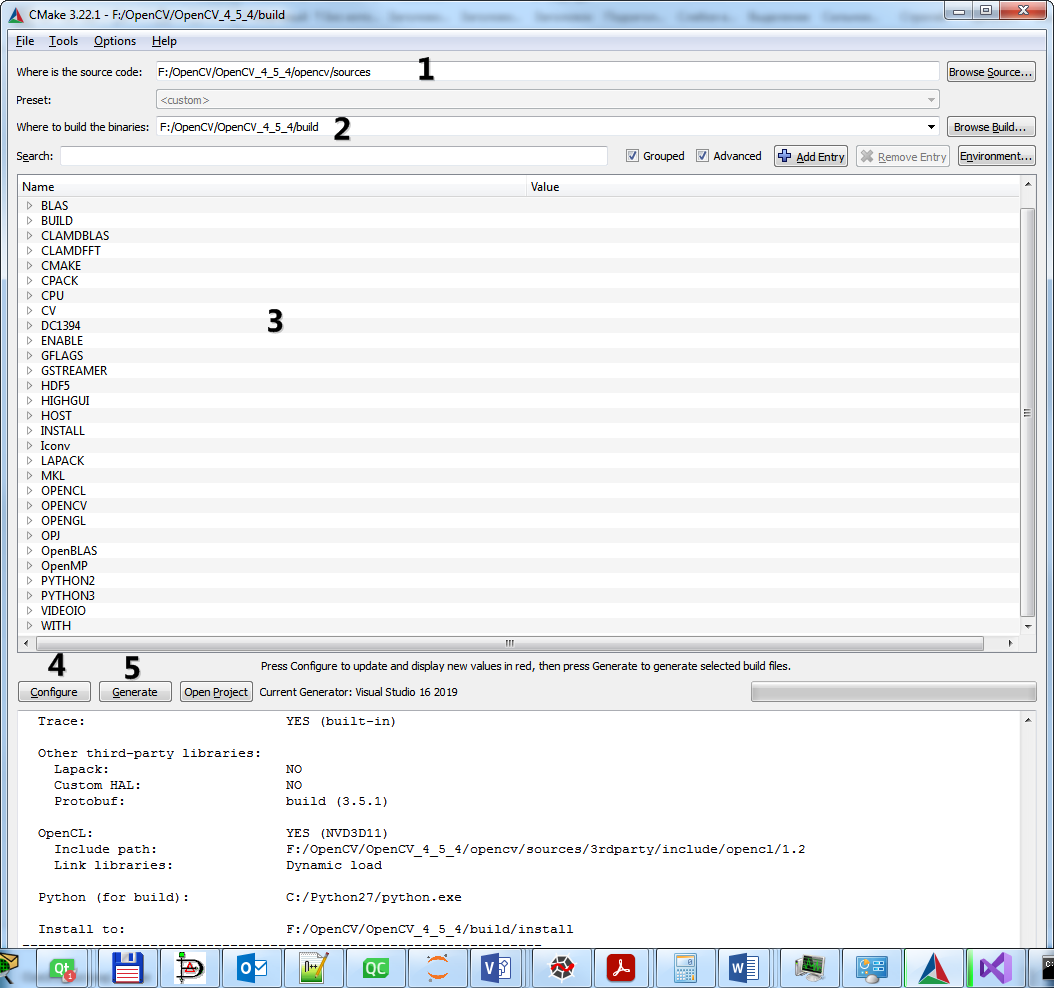


Рис. 1. Главное окно CMAKE

На рис. 1 цифрами обозначены:

1 – директория с исходными кодами для сборки

2 – директория, в которую производится сборка

3 – область, в которой будут отображены конфигурируемые параметры. Изначально пустая. Появляется после успешной процедуры конфигурации.

4 – кнопка запуска процедуры конфигурации

5 – кнопка запуска генерации проекта для MS Visual Studio по заданной конфигурации.

1. Задать пути к исходным кодам и к директории сборки и нажать Configure
2. В отобразившемся окне выбрать генератор и указать 4 пункт настроек (рис. 2)

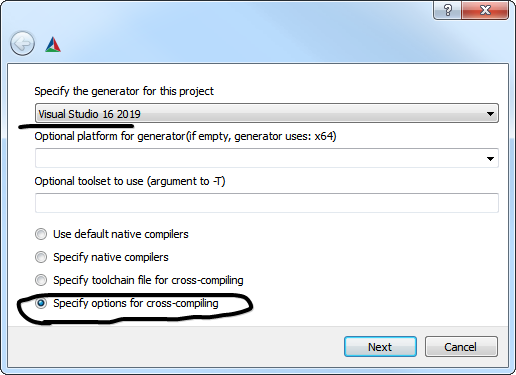


Рис. 2. Окно конфигурации

1. Далее следует указать путь к компилятору, а также тип ОС (рис. 3).

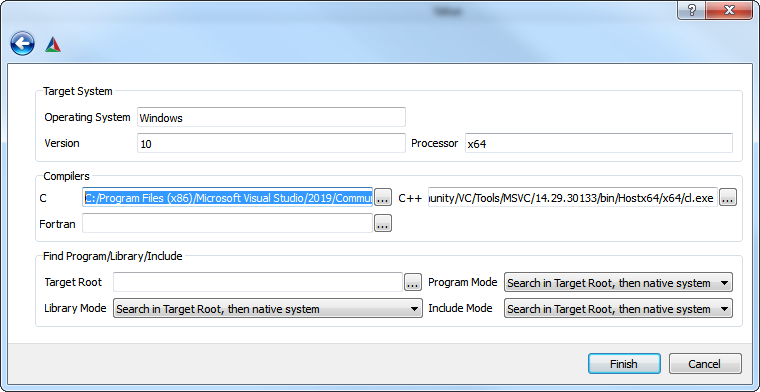


Рис. 3. Окно настройки конфигуратора

В моем случае была Window7, но я указал 10, т.к. в процессе конфигурации столкнулся с ошибкой линкера (link fatal error lnk1104 cannot open file 'ucrt.lib'). Дело было в отсутствии нужной библиотеки в SDK. Пришлось устанавливать Windows SDK (<https://developer.microsoft.com/en-us/windows/downloads/windows-sdk/>), которая имеет 10 в номере.

1. В области 3 (см. рис. 1) следует выбрать компоненты, по которым будет произведена генерация. В частности, были выбраны WITH\_QT, WITH\_OPENGL. Также в качестве дополнительных модулей могут быть поставлены opencv\_contrib (исходники скачиваются отдельно из <https://github.com/opencv/opencv_contrib>, далее прописывается путь к модулям (Path\_to\_opencv\_contrib/modules) в OPENCV\_EXTRA\_MODULES\_PATH и устанавливается отметка в OPENCV\_FORCE\_3RDPARTY\_BUILD (рис. 4)). Если требуется сменить параметры конфигурации, то следует удалить кеш (File->Delete Cache)
2. Далее нажать на кнопку генерации. При отсутствии ошибок будет выдано сообщение “Generating done”

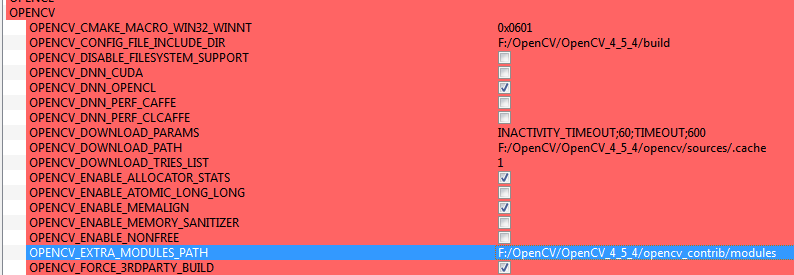


Рис. 4. Выбор опций в конфигурации

1. Далее следует собрать полученное решение в Visual Studio. Для этого открываем решение OpenCV.sln (или нажимаем на кнопку OpenProject в CMake). После чего нужно изменить сборку с Debug на Release, а затем собрать проект ALL\_BUILD. Сборка занимает определенное количество времени. Если ошибок не было, то следует собрать проект INSTALL (рис. 5)

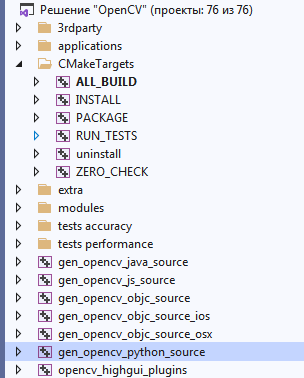


Рис. 5. Выбор проектов для сборки

Стоит отметить, что для сборки opencv под python следует сначала установить numpy, а затем вручную настроить пути к numpy в соответствующих полях конфигуратора (рис. 6)



Рис. 6. Пример задания путей к пакетам Numpy

Также следует убедиться в том, что установлена отметка напротив BUILD\_opencv\_python\_2/BUILD\_opencv\_python\_3 (рис. 7)

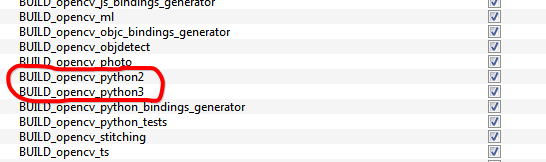


Рис. 7. Проверка наличия сборки OpenCV для python

К сожалению, на Windows7 OpenCV не была корректно поставлена для python 3.8 (рис. 8), но установилась успешно для Python 2.7 (рис. 9)

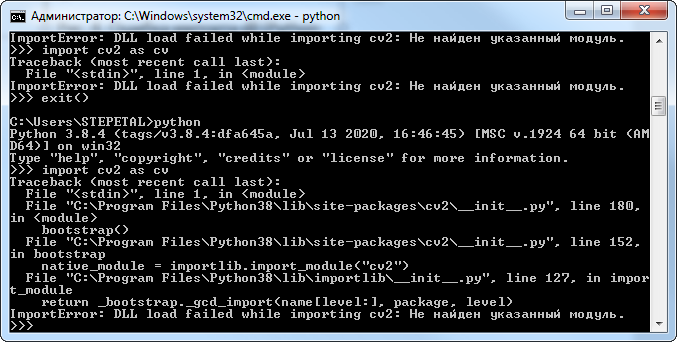


Рис. 8. Ошибка импорта dll (Python 3.8)

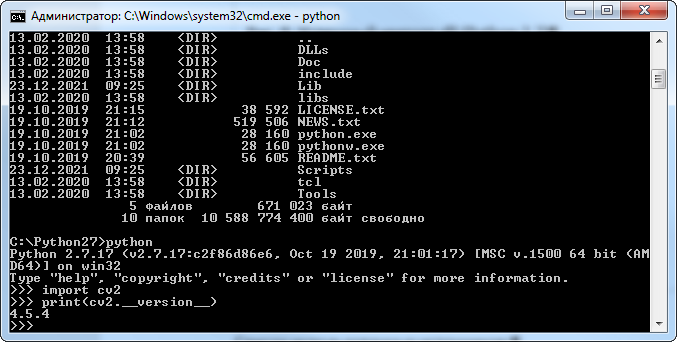


Рис. 9. Успешный импорт dll (Python 2.7)

Список использованных источников:

1. <https://docs.opencv.org/4.x/d5/de5/tutorial_py_setup_in_windows.html>
2. https://cv-tricks.com/how-to/installation-of-opencv-4-1-0-in-windows-10-from-source/