Building EPICS, the QE framework, and qegui on windows 10, Qt 5.9

# Environment:

      Microsoft ‘Surface’ running Windows 10 Pro

      64 bit Operating System x64-based processor

# EPICS configuration and build

      Set EPICS\_HOST\_ARCH=windows-x64

      Set EPICS\_BASE=C:\epics\base

      EPICS 3.15.5 source copied to C:\epics

      Installed strawberry perl 5.26.1 to C:\stawberry

      Installed Visual Studio 2017

      Installed mingw-w64 7.1.0 (to get make) – Note, mingw32-make.exe copied to make.exe

Additions to PATH to build EPICS

      C:\Qt\Tools\QtCreator\bin  
(jom.exe – could not be used as it does not understand ‘make’ ‘wildcard’)

      C:\Strawberry\c\bin  
(a lot of gcc stuff, but not make. Is this required?)

      C:\Strawberry\perl\bin  
(perl.exe)

      C:\Program Files (x86)\mingw-w64\i686-7.1.0-posix-dwarf-rt\_v5-rev0\mingw32\bin  
(for gcc make – last to avoid picking up other gcc stuff in preference to msvc)

Commands to build EPICS

      cd C:\epics\base

      "C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\VC\Auxiliary\Build\vcvarsall.bat" x86\_amd64

# Qt 5.9 configuration

The following was installed for building EPICSQt in QtCreator:

      Qt 5.9 development environment

      Windows SDK 10.1.0.0 (provides debugger for QtCreator)

      CMake 3.9.0 (not required?)

Qt 5.9 installed with the following kits:

      Desktop Qt 5.9.0 MSVC2017 64bit  
This kit required manual selection of:  
- Microsoft Visual C++ Compiler 15.0  (amd64) for both C and C++  
- Widows SDK 10 cdb x64 (after first installation SDK)  
- CMake (after first installing CMake)  
Still complained CMake ‘Configuration has no path to a C and C++ compiler set, even though the kit has a valid tool chain’, but worked OK.

      Desktop Qt 5.9.0 MinGW 32bit  
This kit had no errors.

      Qt 5.9.0 for UWP 32bit (MSVC 2017)  
This kit also complained of no compilers. Selected Microsoft Visual C++ Compiler 15.0  (x86) for both C and C++

      Qt 5.9.0 for UWP 64bit (MSVC 2017)  
This kit also complained of no compilers. Selected Microsoft Visual C++ Compiler 15.0  (amd64) for both C and C++

      Qt 5.9.0 for UWP armv7 (MSVC 2017)  
This kit also complained of no compilers. No arm compiler to select.

Qt 5.9 further configuration:

Build and Run – Debuggers – Add (following install of Windows SDK):

      Name: Widows SDK 10 cdb x64

      Path: C:\Program Files (x86)\Windows Kits\10\Debuggers\x64\cdb.exe

      Type: CDB

      ABIs: x86-windows-msvc2017-pe-64bit

      Version: 10.0.15063.400

Version control – Git (following installation of Git Desktop app):

      Prepend to PATH: C:\ProgramData\user\GitHubDesktop\app-0.5.9\resources\app\git\mingw64\bin

# EPICSQt

Prerequisites for epicsqt:

      QWT 6.1.3  
Source copied to C:\qwt-6.1.3  
Project built by QtCreator

      FFMPEG  
3.3.1 ‘dev’ and ‘shared’ 64 bit binary downloaded and both copied to C:\ffmpeg  
(Note there are some common files to each)

      C:\epicsqtTarget directory created

Additions to PATH to either use Qt Creator or run qegui:

      C:\Program Files\CMake\bin\  
CMake

      C:\Program Files (x86)\Windows Kits\10\Windows Performance Toolkit\  
Microsoft debugger

      C:\epicsqtTarget\bin\windows-x64  
qegui.exe

      C:\epicsqtTarget\lib\windows-x64  
qeplugin.dll

      C:\ffmpeg\bin  
ffmpeg

      C:\epics\base\bin\windows-x64  
ca.dll, com.dll

      C:\qwt-6.1.3\lib  
qwt libraries

      C:\Qt\5.9\msvc2017\_64\bin

      C:\Qt\Tools\QtCreator\bin