Build Instructions for the PNT Integrity Toolkit

This repository contains the top level CMake project for buidling the IS4S PNT Integrity Library, Toolkit and their dependencies.

System Requirements

The PNT Integrity Toolkit has been tested with on the following operating systems:

```
* Ubuntu Linux 18.04 & 20.04
* MacOS 10.15
```

The following additional tools are needed to build the library:

```
* CMake 3.5 or greater
* C++14 compliant compiler (e.g. Clang 3.3+, GCC 4.7+, MSVC 2015+)
```

Dependencies

The PNT Integrity Library is designed to require as few third party dependencies as possible to support building on a wide variety of platforms. Two dependencies are required in addition to the libraries provided in the package. The Eigen (https://eigen.tuxfamily.org) C++ template library for linear algebra is required by the base PNT Integrity library. The FFTW (http://fftw.org) package is optionally required and is needed to use the acquisition check. QT (https://www.qt.io) is required to build and run the user interface

Both packages can be installed following instructions on their respective websites. Eigen is a header-only package and can be installed by downloading a release from the project web site and extracting to a local folder. FFTW binaries are avaiable for a range of platforms from the project web site.

Alternatively, a package manager can be used to install the dependencies. For MacOS the Homebrew (https://brew.sh) package manager is recommended. The Chocolatey (https://chocolatey.org) package manager is recommended for Windows. Instructions on installing the required and optional dependencies using package managers on the supported operating systems are provided in the following sections.

Ubuntu / Debian

Install Boost by running:

```
sudo apt-get install libboost-all-dev
```

Install Eigen by running:

```
sudo apt install libeigen3-dev
```

Optionally install FFTW by running:

```
sudo apt install libfftw3-dev
```

Install Qt5 on Ubuntu by running:

```
sudo apt install qtdeclarative5-dev qtwebengine5-dev libqt5charts5-dev
```

Install YAML-CPP by running:

```
sudo apt install libyaml-cpp-dev
```

Install UHD by running:

```
sudo apt-get install libuhd-dev libuhd003 uhd-host
```

MacOS (for PNT Integrity Library only)

Install boost by running:

```
brew install boost
```

Install Eigen by running:

```
brew install eigen
```

Optionally install FFTW by running:

```
brew install fftw
```

Install Qt5 on MacOS by running:

```
brew install qt
```

If you encounter CMake build errors when finding QT this may help:

```
export CMAKE_PREFIX_PATH=/usr/local/Cellar/qt/[version]/
```

Install YAML-CPP by running:

```
brew install yaml-cpp
```

Follow the Ettus instructions (https://files.ettus.com/manual/page_install.html) to install UHD.

Unpackaging

Skip this step if cloning directly from GitHub / Gitlab

Extract the release archive:

```
unzip release.zip
cd release
```

Building

Building can be performed ether with a script or manually with commands.

Script

Execute the provided build script to build

```
./build.sh
```

Build Commands

Generate build files using cmake

```
mkdir build
```

To build only the PNT Integrity library

```
cmake ../ -DCMAKE_BUILD_TYPE=Release -DBUILD_KIT=FALSE
```

To build the full toolkit with sample application and hardware drivers

```
cmake ../ -DCMAKE_BUILD_TYPE=Release -DBUILD_KIT=TRUE
```

By default, this will generate Unix Makefiles for the package. Project files can be generated for other build systems or IDEs by selecting an alternative CMake generator (https://cmake.org/cmake/help/v3.15/manual/cmake-generators.7.html).

Build the libraries by running:

```
make
```

Installing

The libraries can be optionally installed to the user's system by running:

```
make install
```