Compiling

- **Install the following packages before starting the compilation:**
  - CodeBlocks IDE (http://www.codeblocks.org/).
  - CMake (https://cmake.org/)

- **If you are building on Windows, install the following packages before starting the compilation**
  - TDM GCC compiler (http://tdm-gcc.tdragon.net/). Choose the 32bit or 64bit compiler.
  - MSYS (http://www.mingw.org/wiki/msys).
  - Run MSYS and type: **`gcc -v`**. Check COLLECT_GCC path. This should be something like **`c:\TDM-GCC-64\bin\gcc.exe`**
  - Setup CodeBlocks IDE to use the previously installed TDM-32 bit or TDM-64 bit compiler. Open CodeBlocks and go to [settings]->[compiler]->[toolchain executables]. Change [compiler installation directory] path to TDM GCC compiler installation folder like C:\TDM-GCC-64 for 64 bit build and choose auto detect.
  - Install NSIS (https://nsis.sourceforge.io/Download). This is for creating installation package.

- **If you are building on MACOSX install Xcode and Xcode command line tools**

- **On Linux (Ubuntu) install:**
  - Synaptic Package Manager.
  - Install the following packages using synaptic package manager:
    - Gnu c++ compiler: build-essential
    - Gtk+ development libraries: libgtk-3-dev
    - Libudev: libudev-dev
    - OpenGL development libraries: libgl1-mesa-dev

- **Extract sources to the directory of your choosing**
  - `git clone https://gitlab.com/scopefun/scopefun-software.git`

- **Create a build directory in the 'scopefun-software' directory and enter the directory**
  - `mkdir scopefun-software/build && cd scopefun-software/build`

- **First step (Windows): Configure and generate 'MinGW makefiles' using Cmake.**
  - Make sure you enter the correct build type for CMAKE_BUILD_TYPE variable (this can be **Debug** or **Release**), after performing 'Configure' step.
  - Select 'Configure' step two times.
  - Check that all SCOPEFUN variables are properly set after 'Configure' step. There should be no red colored configuration variables or errors in cmake prior to generating choosen build system scripts.
  - Choose 'Generate'.
  - Open MSYS (Windows), go to your build folder and type: **mingw32-make package.** This command will build libraries in the ScopeFun source folder.

- **First step (Linux and Mac): Configure and generate 'Unix makefiles' using Cmake.**
  - **Linux:** run terminal, go to your build folder and type:
    - `chmod +x ../lib/wxWidgets-3.0.4/src/stc/gen_iface.py`
    - `cmake -G "Unix Makefiles" -D CMAKE_BUILD_TYPE="Release" -D CPACK_BINARY_DEB="true" -D CPACK_BINARY_TZ="false" -D CPACK_BINARY_TGZ="false" -D CPACK_BINARY_STGZ="false" ..`
    - `make`
  - **Mac:** run terminal, go to your build folder and type:
    - `chmod +x ../lib/wxWidgets-3.0.4/src/stc/gen_iface.py`
    - `chmod +x ../lib/libusb-1.0.22/install-sh`
    - `PATH="/Applications/Cmake.app/Contents/bin":"$PATH"`
    - `cmake -G "Unix Makefiles" -D CMAKE_BUILD_TYPE="Release" -D CMAKE_VERBOSE_MAKEFILE="true" -D CPACK_BINARY_DRAGNDROP="true" -S "$CI_BUILDS_DIR.." -B.`
    - `make`

- **Second step (Windows): Configure and generate ’CodeBlocks-MinGW Makefiles’ using Cmake.**
  - Change ’Where to build the binaries’ folder in Cmake (this must be different folder as in first step).
  - Make sure you enter the correct build type for CMAKE_BUILD_TYPE variable (this can be **Debug** or **Release**), after performing ’Configure’ step.
  - Select ’Configure’ step two times.
  - Check that all SCOPEFUN variables are properly set after ’Configure’ step. There should be no red colored configuration variables or errors in cmake prior to generating choosen build system scripts.
  - Choose ’Generate’.
  - Open generated project with Codeblocks and select target ’sfScope’
  - Go to Project options -> Build targets tab: change ’Execution working dir’ to project source folder
  - You can now build the project

- **Second step (Linux or Mac): Configure and generate ’CodeBlocks-Unix Makefiles’ using Cmake.**
  - Change ’Where to build the binaries’ folder (this must be different folder as in first step).
  - **Linux:** run terminal, go to your build folder and type:
    - **`cmake`** `-G "CodeBlocks - Unix Makefiles" -D CMAKE_BUILD_TYPE="Release" -D CPACK_BINARY_DEB="true" -D CPACK_BINARY_TZ="false" -D CPACK_BINARY_TGZ="false" -D CPACK_BINARY_STGZ="false" ..`
  - **Mac:** run terminal, go to your build folder and type:
    - **`cmake`** `-G "CodeBlocks - Unix Makefiles" -D CMAKE_BUILD_TYPE="Release" -D CMAKE_VERBOSE_MAKEFILE="true" -D CPACK_BINARY_DRAGNDROP="true" -S "$CI_BUILDS_DIR.." -B.`
  - Open generated project with Codeblocks and select target ’sfScope’
  - Go to Project options -> Build targets tab: change ’Execution working dir’ to project source folder
  - You can now build the project

- **Open CodeBlocks IDE and run compile (or optionally run below commands in command shell)**
  - Go to build directory and type:
    - **Windows (MSYS)**
      - Build executable files: `mingw32-make -f makefile`
      - Build installer package: `mingw32-make package`
      - Build source files package: `mingw32-make package_source`
    - **Linux and Mac**
      - Build executable files: `make`
      - Build installer package: `make package`
      - Build source files package: `make package_source`

- **Questions ?** Go to www.scopefun.com where you can join a forum or contact us by e-mail.