Compiling software from source code

Requirements

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

- **If you are building on Windows, download and install required packages:**
  - Download and install **MSYS2** from https://www.msys2.org/
    Select default installation folder (C:\msys64)
    Before finishing installation, tick "Run MSYS2 64 bit now". This will launch the "MSYS2 **MSYS**" terminal.
    Udate packages using the following command in the MSYS terminal:
    ```
    pacman -Syu
    ```
    MSYS will close after this upgrade.
    Add "C:\msys64\mingw64\bin" folder to Windows PATH variable!
    Run "MSYS2 **MinGW 64-bit**" application and proceed to install required packages using the following commands:
    ```
    pacman -Su
    pacman -S mingw-w64-x86_64-gcc
    pacman -S git
    pacman -S make
    pacman -S mingw-w64-x86_64-make
    pacman -S msys/automake-wrapper
    pacman -S msys/autoconf
    pacman -S mingw-w64-x86_64-nsis
    ```

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

- **If you are building on Linux (Ubuntu) install the required packages:**
  ```
  sudo apt-get install cmake protobuf-compiler
  sudo apt-get install build-essential
  sudo apt-get install libgtk-3-dev
  sudo apt-get install libwxgtk3.0-gtk3-dev
  sudo apt-get install libudev-dev
  sudo apt-get install libgl1-mesa-dev
  sudo apt-get install autoconf
  sudo apt-get install automake
  sudo apt-get install libxxf86vm-dev
  ```

- **Extract software sources from GitLab**
  - **On Windows** run "MSYS2 **MinGW 64-bit**", on **Linux/Mac** open "Terminal":
    ```
    git clone --branch develop 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
  ```

1st step: compiling the libraries

- **Windows: Configure and generate ’MinGW makefiles’ using CMake**
  - Set the source code path ... C:/msys64/home/<*your_username*>/scopefun-software
  - Set build the binaries path ... C:/msys64/home/<*your_username*>/scopefun-software/build
  - ... where <*your_username*> is your Windows username
  - Click Configure and specify **MinGW makefiles** as the generator (Use default native compillers)
  - Enter the correct build type for SCOPEFUN_BUILD_TYPE variable (this can be **Debug** or **Release**), after performing ’Configure’ step.
  - Select ’Configure’ step two times.
  - Check that there is no red colored variables in CMake prior to generating choosen build system scripts.
  - Select ’Generate’.
  - Run **MSYS2 MinGW 64-bit**, and type

```
cd  ~/scopefun-software/build
cd ../lib/libusb-1.0.22 && aclocal && automake && cd .. && cd ../build
mingw32-make package
```

- **Linux and Mac: Configure and generate ’Unix makefiles’ using CMake**

  Linux: run Terminal, go to your "build" directory and type:

```
cd ../lib/libusb-1.0.22 && aclocal && automake && cd .. && cd ../build
chmod +x ../lib/wxWidgets-3.0.5/src/stc/gen_iface.py
chmod +x ../lib/libusb-1.0.22/install-sh
cmake -G "Unix Makefiles" -D SCOPEFUN_VERSION_MAJOR="2" -D SCOPEFUN_VERSION_MINOR="1" \
  -D SCOPEFUN_VERSION_MICRO="0" -D SCOPEFUN_BUILD_TYPE="Release" \
  -D CPACK_BINARY_DEB="true" -D CPACK_BINARY_TZ="false" \
  -D CPACK_BINARY_TGZ="false"-D CPACK_BINARY_STGZ="false" ..
make package
```

  - **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":"$PATH"
cmake -G "Unix Makefiles" -D SCOPEFUN_BUILD_TYPE="Release" \
  -D CMAKE_VERBOSE_MAKEFILE="true" -D CPACK_BINARY_DRAGNDROP="true" \
  -S "$CI_BUILDS_DIR.." -B.
make
```

2nd step: compiling source code in Codeblocks

- **Make sure you have successfully finished the 1st step: compiling the libraries**
- **Windows: Configure and generate 'CodeBlocks-MinGW Makefiles' using CMake**
  - Set the source code path ... C:/msys64/home/<*your_username*>/scopefun-software
  - Set build the binaries path ... C:/msys64/home/<*your_username*>/scopefun-software/codeblocks-build
  - ... where <*your_username*> is your Windows username
  - Note: 'Where to build the binaries' folder **is different folder as in first step**.
  - Click Configure and specify **CodeBlocks-MinGW Makefiles** as the generator
  - Enter the correct build type for SCOPEFUN_BUILD_TYPE variable (this can be **Debug** or **Release**), after performing 'Configure' step.
  - Select 'Configure' step two times.
  - Check that there is no red colored variables in CMake prior to generating choosen build system scripts.
  - Select 'Generate'.
  - You can now build the project
  - Open the generated project with Codeblocks and build project. Then select target 'ScopeFun' and Run.

- **Linux or Mac: Configure and generate 'CodeBlocks-Unix Makefiles' using Cmake.**
  - Go to "scopefun-software" folder and **create a new folder "codeblocks-build"**
    `mkdir codeblocks-build && cd codeblocks-build`
  - **Linux:** run Terminal, go to your "codeblocks-build" folder and type:
    ```
    cmake -G "CodeBlocks - Unix Makefiles" -D SCOPEFUN_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 the generated project with Codeblocks and build project. Then select target 'ScopeFun' and Run.

- **Open CodeBlocks IDE and run Build (or optionally run below commands in command shell)**
  - Go to build directory and type:
    - **Windows (MSYS2 MinGW 64-bit)**
      - Build executable files: `mingw32-make -f makefile`
      - Build installer package: `mingw32-make package`
      - Build source files package: `mingw32-make package_source`
    - **Linux/Mac using Terminal**
      - Build executable files: `make`
      - Build installer package: `make package`
      - Build source files package: `make package_source`

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