english
review

Note: Attention: Work in progress - page under construction

UHSDR development - Build using eclipse

Preconditions, Assumptions, Scope

- UHSDR toolchain installed, see here

Pull UHSDR source from Github, import into eclipse

- Goto https://github.com/df8oe/UHSDR and click button „clone or download“
- Unpack the downloaded .zip file on your hard disk. Do not change the name, do not change the internal zip structure
- In Eclipse, choose „File→Import“.
- .. then choose „General→Existing Projects into Workspace“.
- Then choose the folder „mchf-eclipse in the path where you just downloaded the zip file as „root-directory“
- The project is now recognized and can be imported

Build UHSDR using eclipse

- Project → Build Configurations → Set Active → DebugLibOVI40 (OVI49 DSP Library (STM32F7))
- Click on the „hammer“ in the tool bar
- Watch Console window in Eclipse - if all ok console output ends with “ … Build finished …„

- Project → Build Configurations → Set Active → DebugOVI40 (OVI40 Firmware (STM32F7) …)
- Click on the „hammer“ in the tool bar
- Watch Console window in Eclipse - if all ok console output ends with “ … Build finished …„

Newly built binary fw-ovi40.bin can be found in the eclipse project explorer (left sidebar): mchf-eclipse → DebugOVI40 → fw-ovi40.bin

Download new firmware .bin to OVI40 UI

- Please refer to https://www.amateurfunk-sulingen.de/wiki/doku.php?id=en:uhsdr:uhsdrfw#st-link_v2_firmware_upgrade
- UHSDR FW build should run.
- Hit switch „Menu“, then select „System Info“ and „Build“. Check the build date and time corresponds to your compile time in eclipse approximatelly

Generate .dfu file from .bin file

With .bin file the FW can be installed via USB stick of ST-Link/V2 adapter. You may want to create a dfu file from the bin file so that the DfuSE FW install method can be used.

Section 4.2.1 of STM manual UM0412 describes step-by-step the generation of a .dfu file.

The ST-Micro UM0412 manual on DfuSe can be found

here

. Original source is this.

Eclipse user guide

- User guide can be found here.
- Profiling is described here

This covers basic usage as well as advanced tools such as profiling and others.

Debug, tracing, real-time diagnostics

- nice summary of Cortex M tools here
- OpenOCD & Eclipse debug step-by-step instructions
- AN4989 Application note STM32 microcontroller debug toolbox
- Concurrent use of ST-Link debugger and SWO Viewer not possible, see here - but this link states Eclipse support in debugger for SWO?
- J-Link Debug Probe here

Further reading

- https://www.amateurfunk-sulingen.de/forum/index.php?board=15;action=display;threadid=313;start=0
- https://www.amateurfunk-sulingen.de/forum/index.php?board=15;action=display;threadid=263
- SWV and SWO and Eclipse explaind here