

Keil unter Wine ausführen mit externem Flash-Tool und externem Debugger:

1. Installation von Keil (wine)

Einfach installieren unter wine oder PlayonLinux

2. Installation von st-link (Linux)

-> sudo apt-get install st-link

3. gdbgui installieren (Linux)

-> Siehe <https://www.gdbgui.com>

3. Datei uvision-flash.sh (z.B. in ~/bin) erstellen (Adresse evtl. anpassen) und ausführbar machen:

```
--snip-->
#bin/bash
# uvision-flash for Keil uVision
# parameters:
# $1: Filename without path and without extension (Keil: @L)
# $2: Adress (STM32F429: 0x08000000)
echo converting $1.axf to $1.bin and $1.elf
wine /data/PlayOnLinux/wineprefix/Keil/drive_c/Keil_v5/ARM/ARMCC/bin/fromelf.exe --bin --output $1.bin
$1.axf
echo flashing $1.bin to adress $2
#flash CT-Board (STM32F429)
st-flash write $1.bin $2
read -p "Press enter to continue" k
<--snap--
```

4. Datei run.gdb im jedem Projektverzeichnis erstellen mit Inhalt (Pfade müssen ggf. angepasst werden):

```
--snip-->
target remote localhost:4242
set substitute-path app\\ app/
set substitute-path RTE\\Device\\CT_Board_HS14_M0\\ RTE/Device/CT_Board_HS14_M0/
<--snap--
```

5. Datei uvision-gdb.sh (z.B. in ~/bin) erstellen und ausführbar machen:

```
--snip-->
#bin/bash
# uvision-gdb for Keil uVision
# parameters:
# $1: Filename of linker outputfile without path and without extension (Keil: @L)
# $2: relative path to linker outputfile (e.g. "build")
st-util&
gdbgui_0.13.2.1 -g gdb-multiarch --gdb-args="--command=run.gdb" $2/$1&
sleep 1
firefox localhost:5000&
read -p "Press enter to close" k
<--snap--
```

6. Tools-Menüeinträge in Keil erstellen:

Flashen:

Command: Z:\usr\bin\xfce4-terminal

Initial Folder: \$L

Arguments: -x uvision-flash.sh @L 0x08000000

Debuggen:
Command: Z:\usr\bin\xfce4-terminal
Initial Folder: \$L
Arguments: -x uvision-gdb.sh %L build

Weiterführende Infos:

- Keil key sequences:
http://www.keil.com/support/man/docs/uv4/uv4_ut_keysequence.htm
- Keil bietet eine kostenfreie Lizenz für einige STM32-Controller auf der HP
- Debugger nemiver kann leider keine gdb-Befehle ausführen und daher keine Pfade ersetzen
https://www.jann.cc/2012/04/14/using_nemiver_for_remote_debugging_on_arm_microcontrollers.html
- Debugger ddd verliert regelmässig die Verbindung zum gdb-Server