# ARM Compiler Toolchain voor ECSL (PRJ3Prt) en DSPESEL (DSBprt)

## for

## HAN Electrical Engineering/Embedded Systems Engineering
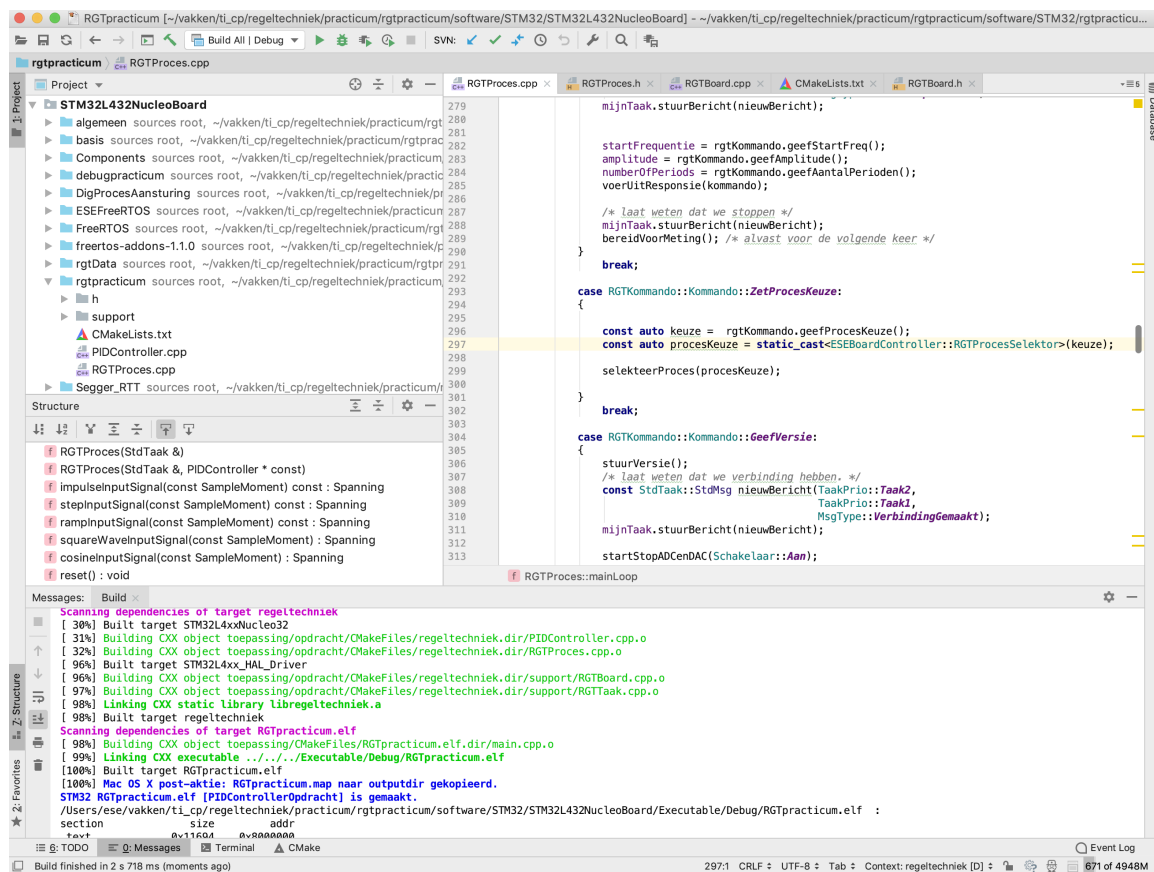
## Inhoudsopgave

# 1  Software IDE installation

There are several options for software development of embedded systems. HAN ESE offers a choice of two different development environments that are supported.

## 1.1 Jetbrains CLion + Segger J-LinkDebugger (Ozone)





*Jetbrains CLion*

An alternative to Keil is the design around the CLion and Segger Ozone packages. This setup initially requires more work on installation, but the benefits are:

- No more problems with the Keil 32 Kbyte limit.

- A superior IDE to work with modern features such as refactoring, very good navigation and full support for version management with SVN, Git and others.
- A much better support for modern C ++ variants (currently: C ++ 14).
- Flexible project design based on CMake.
- A superior and very fast debugger.

### 1.1.1 Installation

The installation can be performed on Microsoft Windows, Apple Mac OS X or Ubuntu Linux / FreeBSD in various steps. A few steps for Windows are not necessary on the other platforms, and vice versa.
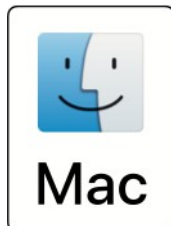
Proceed as follows:

1. [Just for Microsoft Windows]

   

   If MinGW is not present on the system, download thee [MinGW installer](). Install the following files with the installer:
   - msys base
   - mingw32-gcc-g++
   - mingw32-make
   - msys-bzip2
   - mingw32-libz

2. [Just for Apple Mac OS X]

   

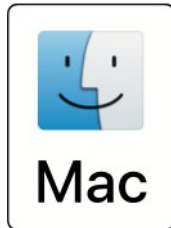   If not present, download and install Apple XCode via the App Store.

3. [Others (FreeBSD/Linux]

   

   If not present, download and install the gcc and g ++ compilers for Ubuntu Linux. FreeBSD has standard C / C ++ compilers installed.

4. [All platforms] Download and install the [GNU ARM Embedded Toolchain]().

   On Mac OS X and Linux:

Install the tarball in the / usr / local directory, otherwise the Cmake scripts will not find this gcc.

5. Download Jetbrains Clion.



Run the installer. If errors occur, then something is probably not installed correctly in the previous steps. CLion comes with a 30-day trial period, after which a license is required.

Optional: Register with Jetbrains immediately to obtain a student license. When obtaining a license, make sure you use an official student.han.nl email address (not google / hotmail, etc.) for identification.

6. [All platforms] Download and install the Segger J-Link Driver  en Segger J-LinkDebugger (Ozone) .
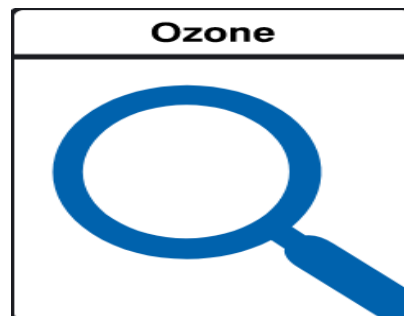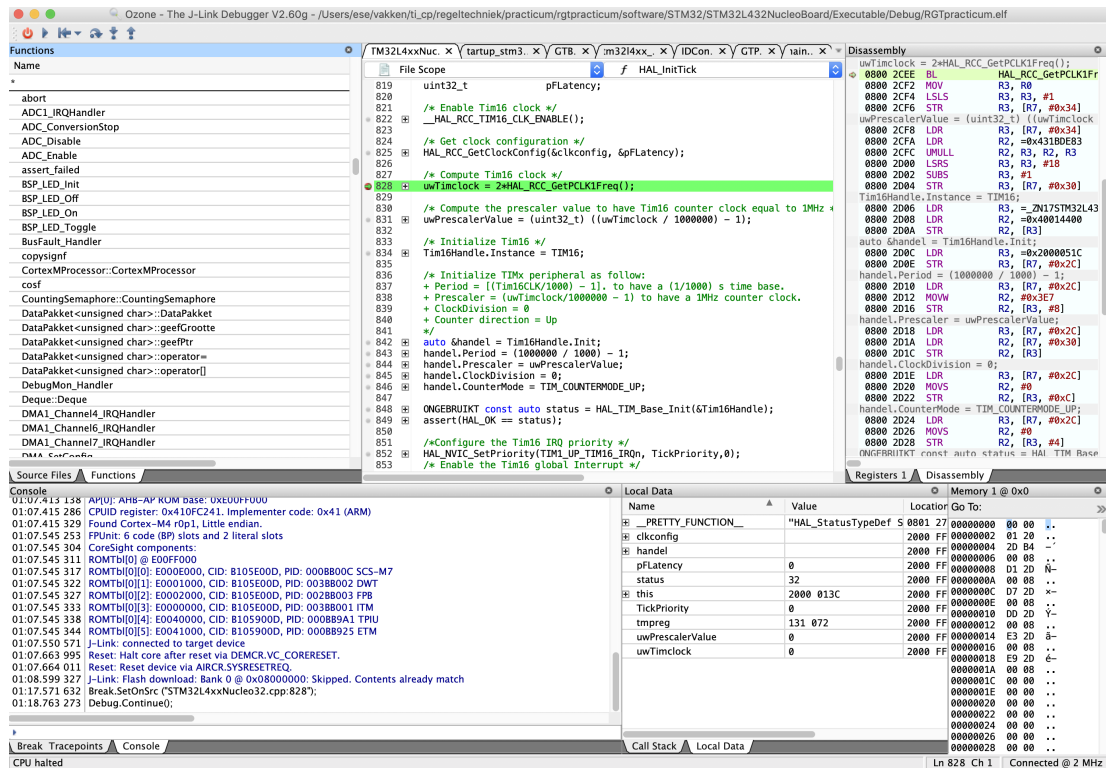
*De Segger Ozone debugger*

7. [All platforms]  7. Optional: download and install the JlinkProgrammer tool to be able to flash / start / stop the target independently.

## 1.1.2 Usage of CLion and Ozone

CLion can be used to build an embedded software project if a suitable CMake project file (CMakeLists.txt) is available, in which are defined:

- The compiler toolchain used.
- The target processor.
- All options for the assembler, compiler and linker in the toolchain.
- All source files that must be compiled into the binary.
- All paths for header files.
- The desired form of the binary.
- Any custom steps that are required in the project.

Various tutorials and lots of information about CMake can be found on the internet. The practicals explain how each practical project must be configured for Cmake, Clion and Ozone.

## 1.2 Keil µVision

In the school years until 2016, only the Keil µVision IDE was used.
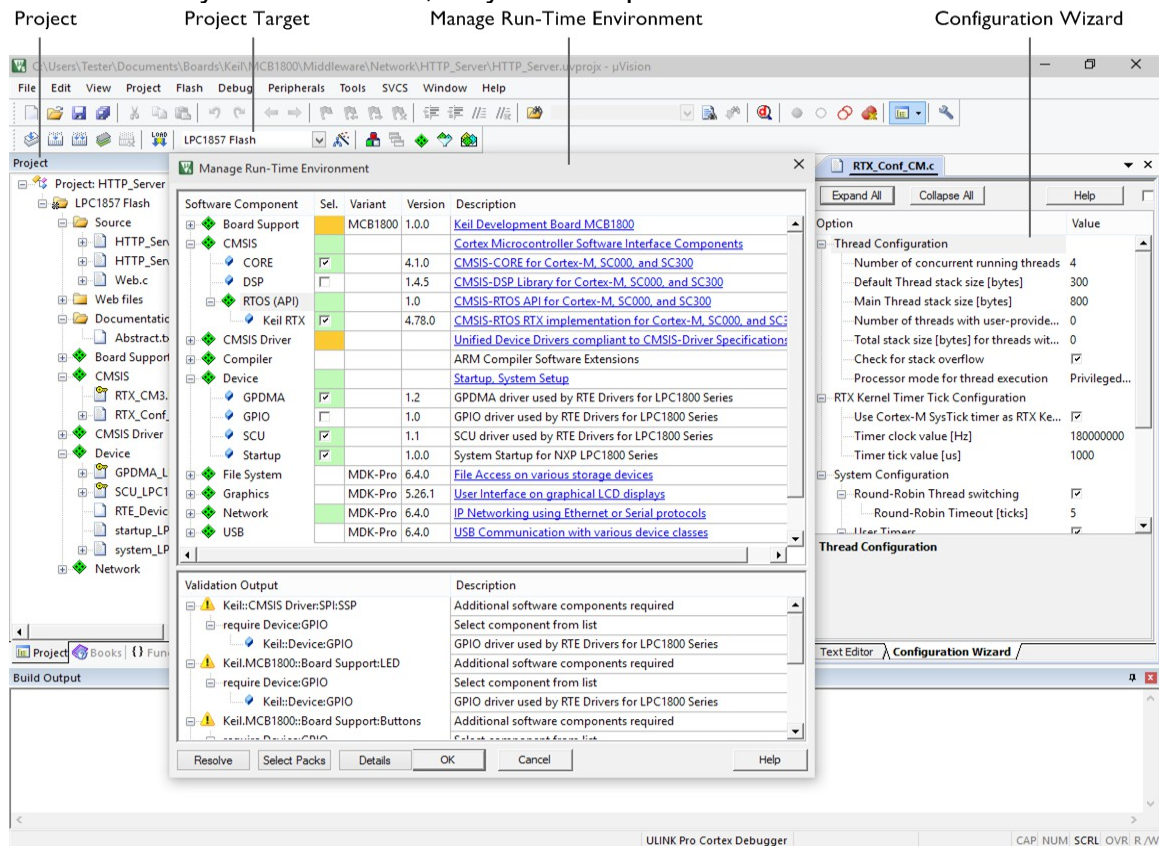


*Keil µVision*

You will find an installer for this program on the HAN ESE FTP server. You can also download the MDK from ARM (Keil) yourself, but this requires a user account.

Keil pros and cons:

- **+** active support by ARM itself. Integrated debugger and many example projects can be found at ARM licensees such as NXP and ST.

- **-** Very expensive for student use (€ 5000 per license), without license limited to 32KByte flash size.

- **-** IDE and Editor from 1995 ("Windows 95 look") with few auxiliary functions for fast and efficient code writing.

- **-** Poor support for modern C ++ variants.

- **-** Difficult project management - paths in a µVision project are often absolute. If a path in the project is no longer correct, then it is endless work to put this path right again.

- **-** Limited to Microsoft Windows PCs.

Because of the disadvantages mentioned here, the use of Keil is increasingly being phased out in favor of the method described above.

# 2 Boards

This chapter describes the boards that are used in education at E / ESE.

## 2.1 The STM32F412 RGT+DSB Board
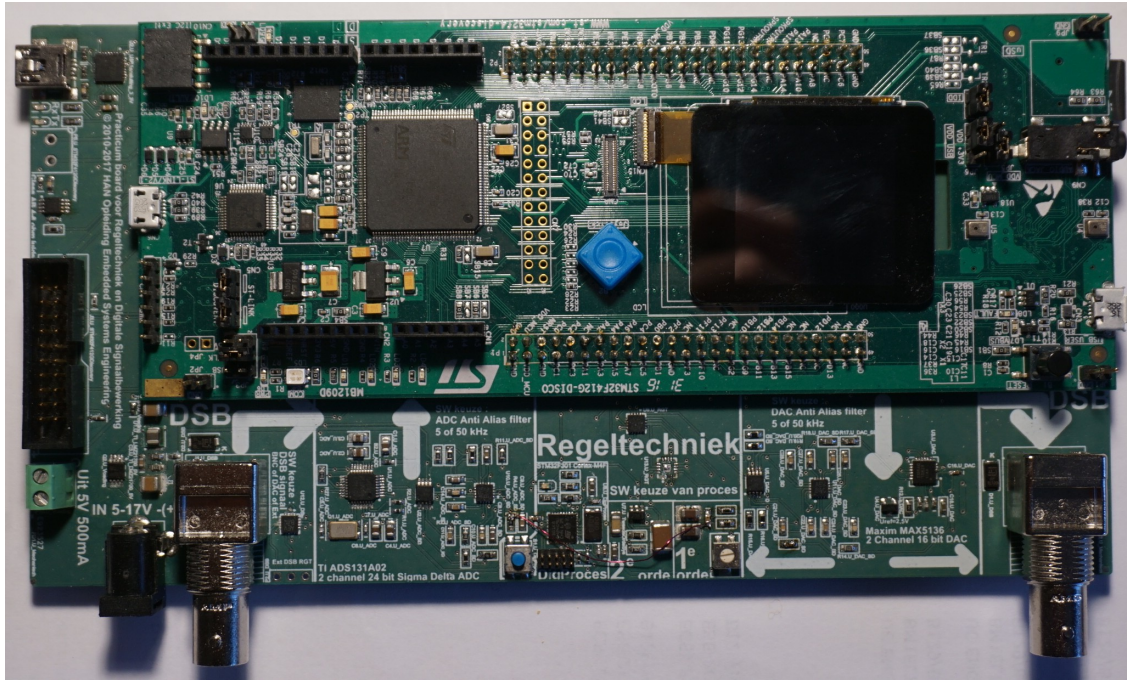


*HAN ESE RGT+DSB board with the STM32F412ZG microcontroller.*

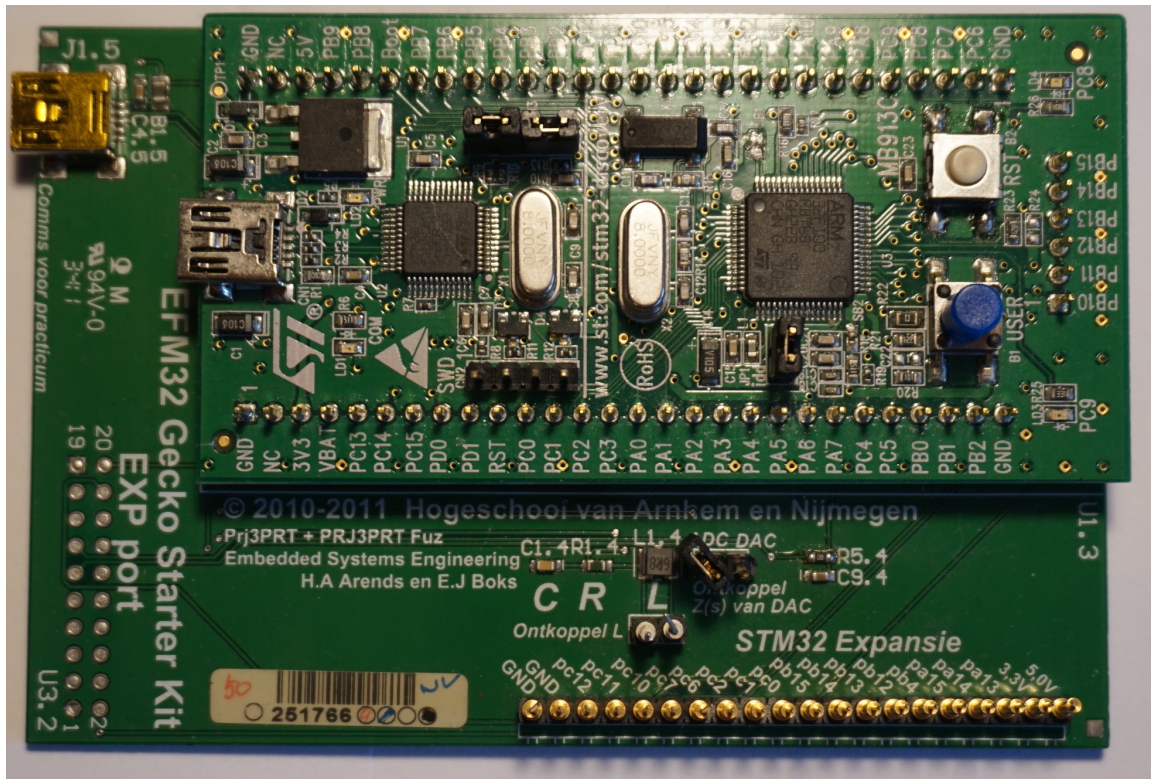The board has two debug / programming connections:

1. The standard 20-pin JTAG / SWD connector (large black connector on the left side of the board). This can be connected to any external debug probe that supports JTAG or SWD. The preference for educational usages goes to the Segger JLink-EDU programmer:



*Segger JLink-EDU*

This debug probe is very inexpensive to purchase, very fast and very

reliable.

2. 2. The built-in SWD link (micro USB-B connector on the left side of the board). This connector is set either as the STLink v2, or the Segger JLink variant of the STLink v2. In terms of speed, it is less capable than the loose debug probe, but otherwise it meets all requirements.

The training is preferred for using the Segger JLink variant, because it is faster and more versatile than the STLink v2 version. JLink is required for using Segger software such as Ozone and RTT. Every STLink v2 version can be converted to the Segger variant with the help of this software program. A restore to STLink v2 can also be done with this small program.

## 2.2 The STM2L432Nucleo board



*HAN ESE STM32L432Nucleo board with the STM32L432KC microcontroller*

The board has a debug connection, namely the micro USB-B connection on the right side of the board. This connector offers the STLink v2 or the Segger JLink variant, as described above for the RGT + DSB board.

## 2.3 The STM32VLDISCOVERY board



*HAN ESE PRJ3Prt+PRJ3Prt Fuz STM32VLDiscovery board with the STM32F100RB microcontroller*

This board is **DEPRECATED**, and is only supported for education that still uses this board.

Arnhem, 29.11.2019

ir drs E.J Boks (ewout.boks@han.nl)