



FEES

PROJECT

FEES MAIN BOARD PROGRAMMING

The main board has several ways to be programmed :

- Via Jtag-SWD through the St-Link programmer of any ST- Discovery Board.
- Via USB through the ST D-fuse - using the Bootloader of any STMicroelectronics produced MCU.
- Via Serial, using TX and RX pins.

In this document only the first way will be explained, since I used the second for a while and was very uneasy and required a pin connector to be inserted or removed to switch between programming or program mode. I also never tried the third and it is time expensive and also pointless to look for it since its redundancy and the handiness of the first method.

1st Method - The JTAG - SWD programming.

You have to use the ST-Link programmer built-in on every ST - Discovery Board.

The connections required to program the FEES system through the SWD are described in the following picture:

STM32 XX - Discovery Board

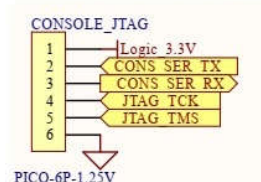
ST-Link Programmer

SWD Connector



FEES - Main Board

JTAG Connector





FEES

PROJECT

Once all the connections with the Fees System are correctly configured

You can proceed with the following procedure:

In a windows environment, open the ChibiStudio folder and start "start_gcc70.bat"

After waiting for Eclipse to start with the ChibiStudio environment select the firmware you want to upload (The Software for the TermoVoid Chamber Test is contained in the folder :

\\\\\\ GitHub\\DEIB-DAER-project\\ChibiOS\\demos\\STM32\\LASERNAV_STM32407_FEES).

The main file is main.cpp

The rest of the project is well organized, all the board configuration files are under:

\\cfg\\chconf.h || \\cfg\\halconf.h || [\\cfg\\mcuconf.h](#)

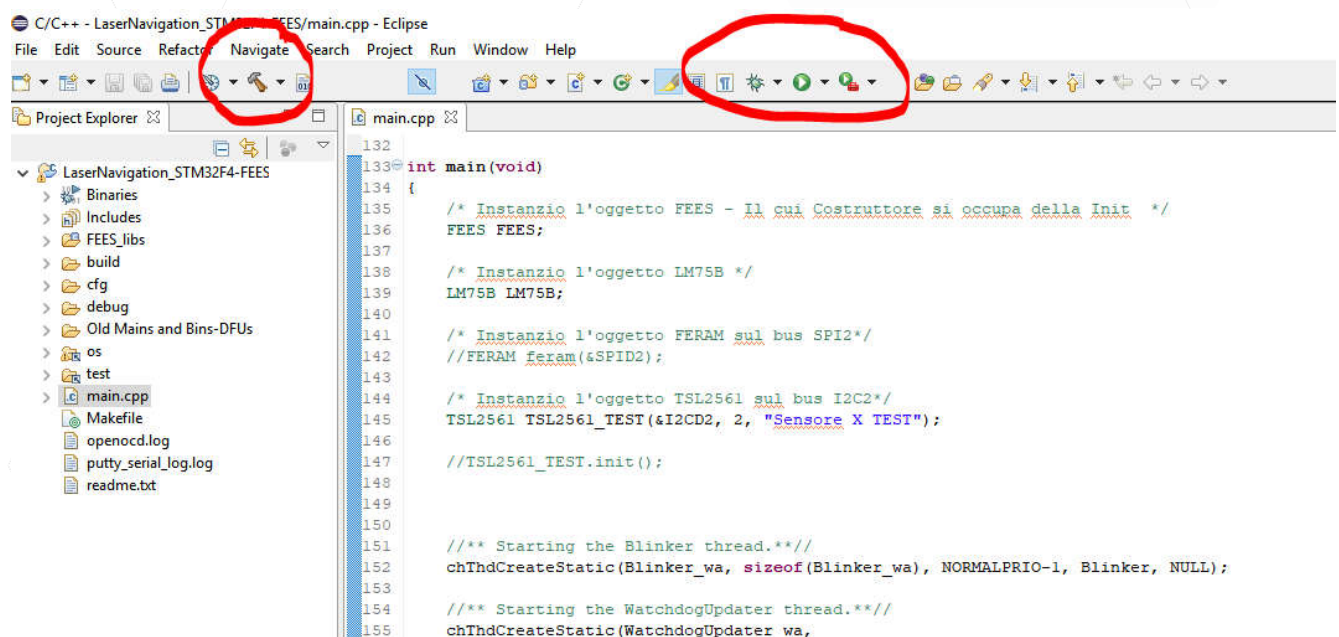
The library files for external components are under:

\\\\\\ GitHub\\DEIB-DAER-project\\ChibiOS\\os\\ex\\

where all libraries made are sorted by Device Manufacturer.

Well, to continue you must build your project opening the main.cpp and building from it.

To build just press the little hammer on the left of the GUI.



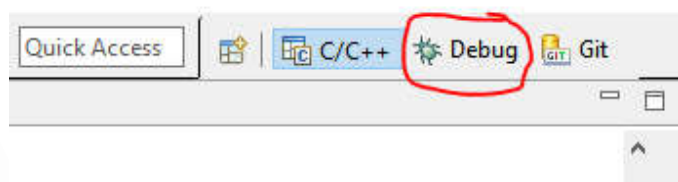
After build press the little Play Button with the red Tool case in the right, and select the Config file that you can find in \\\ChibiStudio\\tools\\openocd\\LaserNavigation_STM32F4-FEES.cfg - when prompted.

If the Upload and Debug goes well you should have a situation on the Console that looks like this :

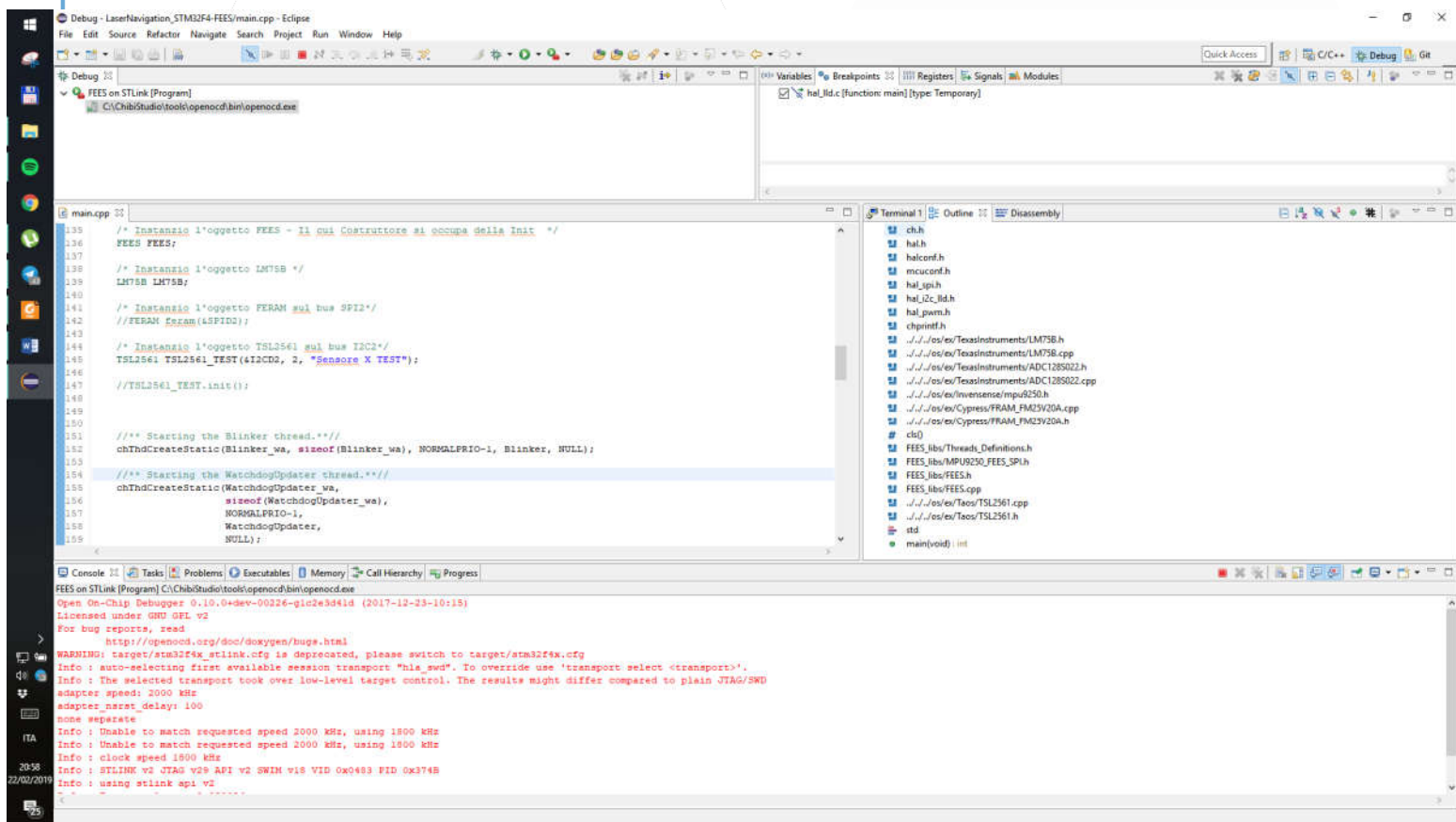


```
FEES on STLink [Program] C:\ChibiStudio\tools\openocd\bin\openocd.exe
Open On-Chip Debugger 0.10.0+dev-00226-g1c2e3d41d (2017-12-23-10:15)
Licensed under GNU GPL v2
For bug reports, read
    http://openocd.org/doc/doxygen/bugs.html
WARNING: target/stm32f4x_stlink.cfg is deprecated, please switch to target/stm32f4x.cfg
Info : auto-selecting first available session transport "hla_swd". To override use 'transport select <transport>'.
Info : The selected transport took over low-level target control. The results might differ compared to plain JTAG/SWD
adapter speed: 2000 kHz
adapter_nsrst_delay: 100
none separate
Info : Unable to match requested speed 2000 kHz, using 1800 kHz
Info : Unable to match requested speed 2000 kHz, using 1800 kHz
Info : clock speed 1800 kHz
Info : STLINK v2 JTAG v29 API v2 SWIM v18 VID 0x0483 PID 0x374B
Info : using stlink api v2
Info : Target voltage: 2.852356
Info : stm32f4x.cpu: hardware has 6 breakpoints, 4 watchpoints
```

The 6 breakpoints and 4 watchpoints indicate that the firmware has been uploaded and it is correctly running. You can now go on debug mode (from the top-right selector) and start debugging your firmware.

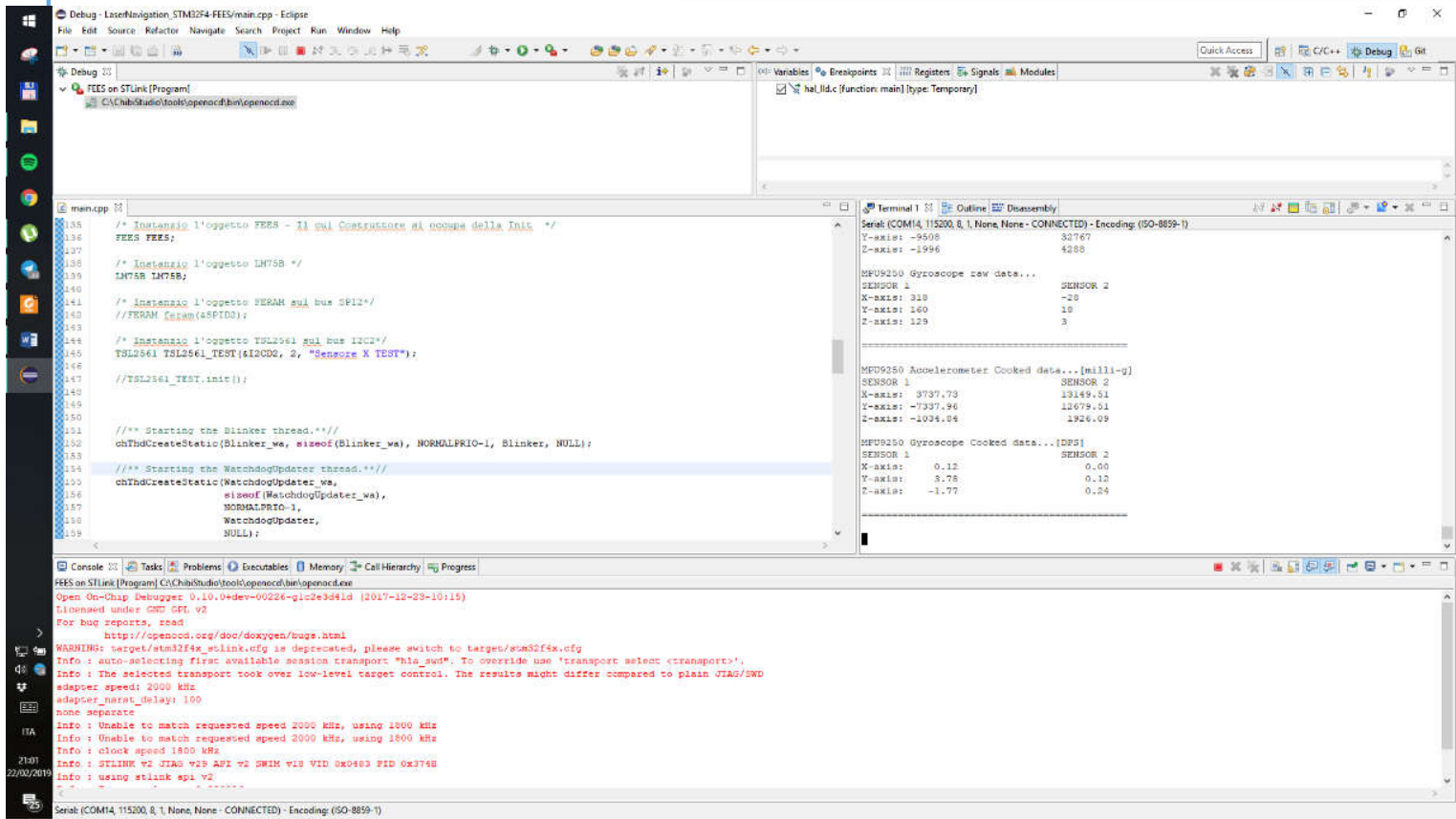


The Full view of the Debug View should be something like this:



In the Next page a view with also the Terminal from the Com Port of the Serial connection Working on the Debug.

Here it is:



```
main.cpp
135  /* Instanzia l'oggetto FEES - il cui CONSTRUCTION si occupa della Init */
136  FEES FEES;
137
138  /* Instanzia l'oggetto IM75B */
139  IM75B IM75B;
140
141  /* Instanzia l'oggetto PERAM sul bus SPI2 */
142  //PERAM peram(SPI2);
143
144  /* Instanzia l'oggetto TSL2561 sul bus I2C */
145  TSL2561 TSL2561_TEST(I2C2, 2, "Sensor X TEST");
146
147  //TSL2561_TEST.init();
148
149
150
151  /** Starting the Blinker thread.*/
152  chTndCreateStatic(Blinker_wa, sizeof(Blinker_wa), NORMALPRIO-1, Blinker, NULL);
153
154  /** Starting the WatchdogUpdater thread.*/
155  chTndCreateStatic(WatchdogUpdater_wa,
156                  sizeof(WatchdogUpdater_wa),
157                  NORMALPRIO-1,
158                  WatchdogUpdater,
159                  NULL);
```

```
Console
FEES on STLink [Program] C:\ChibiStudio\tools\openocd\bin\openocd.exe
Open On-Chip Debugger 0.10.0+dev-60226-g1c2e3d41d (2017-12-23-10:15)
Licensed under GNU GPL v2
For bug reports, see:
http://openocd.org/doc/docxygen/bugs.html
WARNING: target/stm32f4x_wolink.cfg is deprecated, please switch to target/stm32f4x.cfg
Info : auto-selecting first available session transport 'this_swf'. To override use 'transport select <transport>'.
Info : The selected transport took over low-level target control. The results might differ compared to plain JTAG/SWD
adapter speed: 2000 kHz
adapter_nsrst_delay: 100
some separator
Info : Unable to match requested speed 2000 kHz, using 1800 kHz
Info : Unable to match requested speed 2000 kHz, using 1800 kHz
Info : clock speed 1800 kHz
Info : STLINK v2 JTAG v29 API v2 SWIM v18 VID 0x0483 PID 0x374B
Info : using stlink api v2
Info : *****
Serial (COM14, 115200, 8, 1, None, None - CONNECTED) - Encoding: (ISO-8859-1)
```

```
Terminal 1
Serial (COM14, 115200, 8, 1, None, None - CONNECTED) - Encoding: (ISO-8859-1)
T-axi: -5508      32167
Z-axi: -1996      4288
MF09250 Gyroscope raw data...
SENSOR 1          SENSOR 2
X-axi: 318        ~26
Y-axi: 160        18
Z-axi: 129        3
=====
MF09250 Accelerometer Cooked data...[milli-g]
SENSOR 1          SENSOR 2
X-axi: 3737.73    13149.51
Y-axi: -7337.96   12679.51
Z-axi: -1036.04   1926.09
=====
MF09250 Gyroscope Cooked data...[DPS]
SENSOR 1          SENSOR 2
X-axi: 0.12       0.00
Y-axi: 5.76       0.12
Z-axi: -1.77      0.24
=====
```

This file is part of the FEES project documentation,
This particular one is intended for instruction purpose only, in the eventuality of
the TVTC test and following firmware updating/debugging by the
GPAdvancedProject company and/or Associates.

Author: Ing. Stefano Ampolo
Reproduction is Reserved. (2019)