



# TIVALAB

TIVA EK-TM4CI23GXL MICROCONTROLLER PROJECT

Embedded Systems Project

Agnese Salutari, [agnese.salutari@student.univaq.it](mailto:agnese.salutari@student.univaq.it)

# TIVALAB PROJECT OVERVIEW

In TivaLab Project I performed some TM4C123G LaunchPad Workshop labs (proposed in TM4C123G\_LaunchPad\_Workshop\_Workbook.pdf), writing C code and running it on the Tiva board.

Actually that Workshop has been tested on Windows XP, 7 and 8. I've tested it on Windows 10, so in this short report I'm mostly going to explain the problems I found and their solutions. The code files are rich of comments by the way, and I performed homeworks too (they are all available on Github).

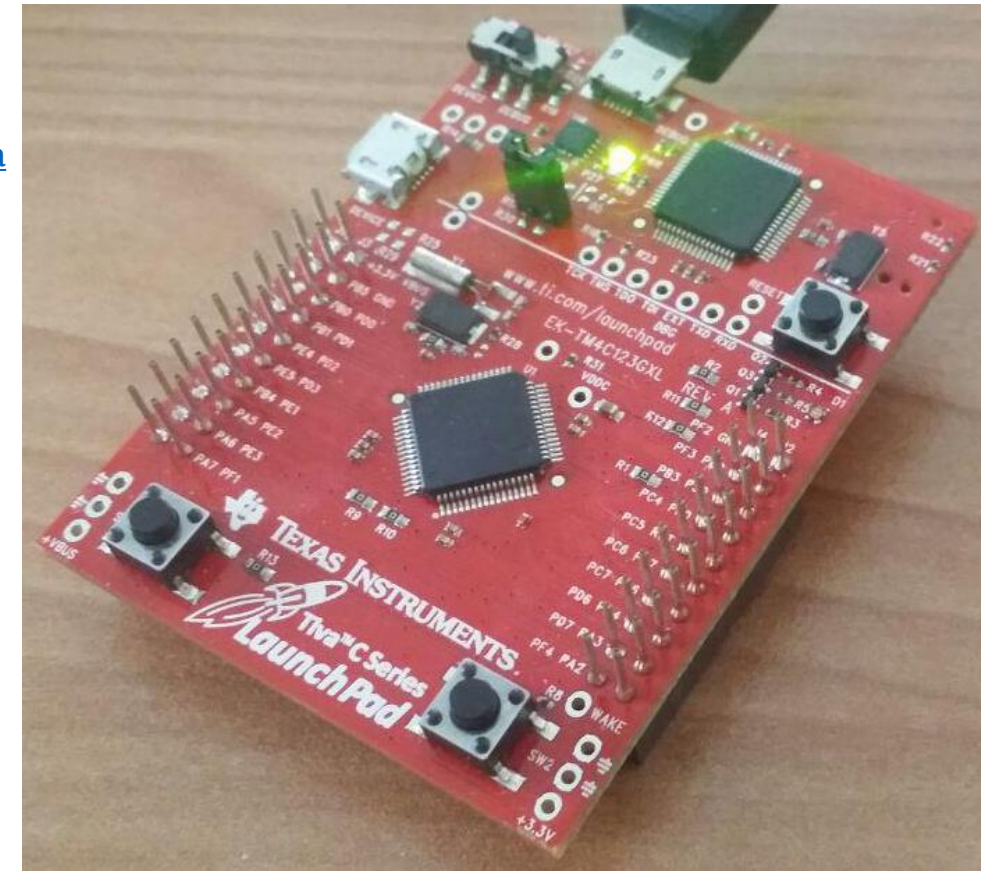
Each lab correspond to a different folder (1, 2, 6, and 7 are for setting up and testing the environment, so they don't have a folder):

- 3: GPIO
- 4: Interrupts and timers
- 5: ADC
- 6: Hibernation Module
- 8: Memory
- 9: Floating-Point Unit
- 12: UART
- 13: Micro DMA

# ENVIRONMENT

In addition to the Tiva board, I used the following Software tools:

- IDE:
  - Code Composer Studio: [http://processors.wiki.ti.com/index.php/Download\\_CCS](http://processors.wiki.ti.com/index.php/Download_CCS)
- Drivers:
  - Stellaris Drivers: [http://www.ti.com/tool/STELLARIS\\_ICDI\\_DRIVERS#descriptionArea](http://www.ti.com/tool/STELLARIS_ICDI_DRIVERS#descriptionArea)
- C libraries for Tiva board:
  - TivaWare for C series: <http://www.ti.com/tool/sw-tm4c>
- (Optional) Flash Programmer:
  - LM Flash Programmer: <http://www.ti.com/tool/lmflashprogrammer>
- Labs material:
  - Workshop files: <http://www.ti.com/TM4C123G-Launchpad-Workshop>
  - Workshop Workbook: <http://www.ti.com/TM4C123G-Launchpad-Workshop>
- Terminal:
  - Putty: <https://www.putty.org/>



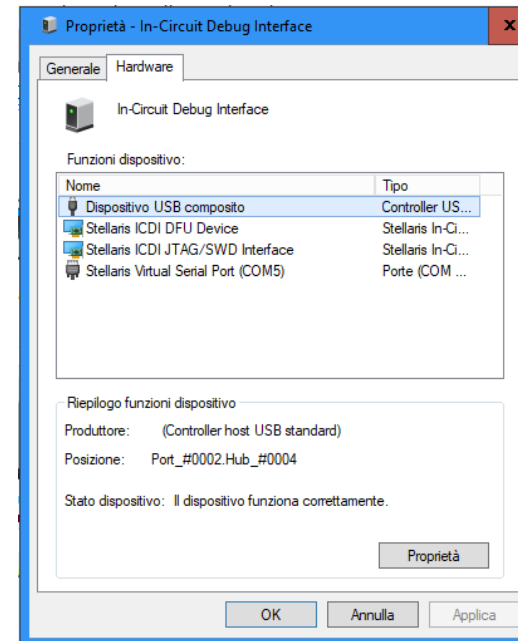
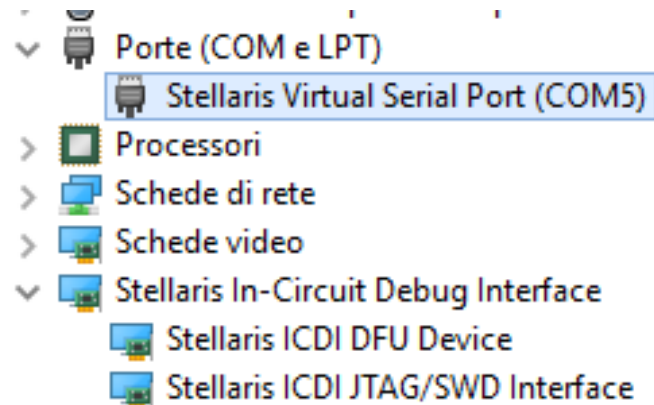
# I - STELLARIS DRIVERS

The first thing I had to pay attention to was Stellaris Drivers installation on Windows 10.

Solution:

- Open Computer > Devices Management (the board as to be plugged in, obviously)
- The board is detected as 3 different components, one under COM Ports and the other two under Unknown Devices: find them all and install the downloaded drivers.

Finally, I had the following configuration:





### 3 - VARS.INI

The next problem I found was because the configuration file of the Workshop workspace, vars.ini, was made for an older version of TivaWare, so I changed it and I added it to the variable paths like this:

The screenshot shows the Code Composer Studio workspace with the following components:

- vars.ini - Blocco note:** A text editor window showing the configuration line: `TIVAWARE_INSTALL = c:\TI\TivaWare_C_Series-2.1.4.178`.
- File Explorer:** A window showing the directory structure of the workspace, with the path `C:\ti` selected. The contents of the directory are listed as follows:

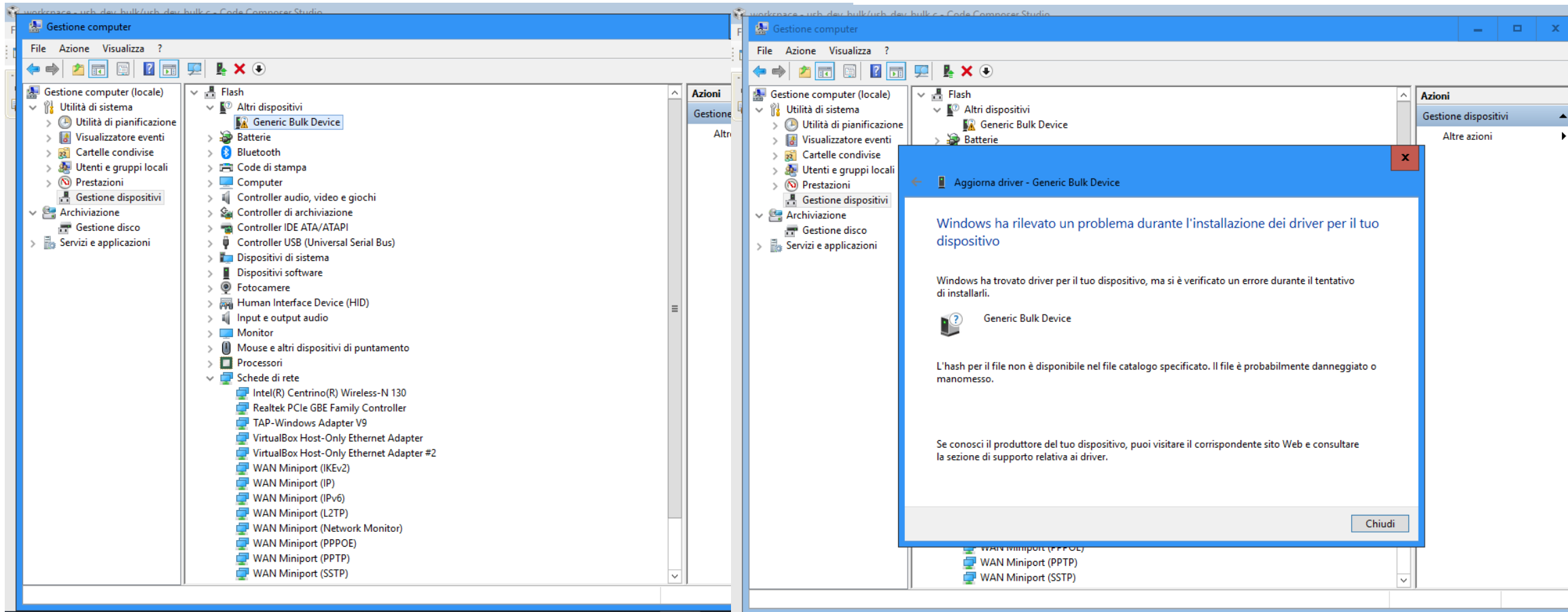
| Nome                        |
|-----------------------------|
| ccs910                      |
| tirex-product-tree          |
| tirtos_tivac_2_16_00_08     |
| TivaWare_C_Series-2.1.4.178 |
| xdctools_3_32_00_06_core    |
| zips                        |

- Select Variable:** A dialog box titled "Select Variable" showing the "Defined variables for build-configuration 'Debug'". The variables are listed in a table:

| Name                         | Type        | Value   |
|------------------------------|-------------|---|
| system_property              | String      | <ECLIPSE DYNAMIC VARIABLE>  |
| SystemDrive                  | String      | C:  |
| SystemRoot                   | String      | C:\WINDOWS  |
| target_config_active         | String      | <ECLIPSE DYNAMIC VARIABLE>  |
| target_config_active_default | String      | <ECLIPSE DYNAMIC VARIABLE>  |
| target_config_default        | String      | <ECLIPSE DYNAMIC VARIABLE>  |
| TARGET_CONTENT_BASE          | Directory   | C:/ti/ccs910/ccs/ccs_base   |
| TEMP                         | String      | C:\Users\Agnese\AppData\Local\Temp  |
| TI_PRODUCTS_DIR              | Path        | C:/ti   |
| TI_PRODUCTS_DIR_TIREX        | Path        | C:/ti   |
| TI_TRACE_DATA_DIR            | String      | C:\Users\Agnese\TI-trace\data   |
| TI_TRACE_DEBUGCLIENT_PID     | String      | 17524   |
| TI_TRACE_LOGGING_DIR         | String      | C:\Users\Agnese\TI-trace  |
| TI_TRACE_SETTINGS_DIR        | String      | C:\Users\Agnese\TI-trace\settings   |
| <b>TIVAWARE_INSTALL</b>      | <b>Path</b> | <b>c:/TI/TivaWare_C_Series-2.1.4.178</b>                                  |
| TMP                          | String      | C:\Users\Agnese\AppData\Local\Temp  |
| TMS470_CG_ROOT               | Directory   | C:/ti/ccs910/ccs/tools/compiler/ti-cgt-arm_18.12.2.LTS                    |
| ToolChainVersion             | String      | 0.0.4   |
| ToolVersion                  | String      | 0.0.4   |
| TPI                          | String      | C:\ti\ccs910\ccs\ccs_base\emulation\analysis\bin\...\tpi                  |
| TRACE_XMLDB_DEVICES          | String      | C:\ti\ccs910\ccs\ccs_base\emulation\analysis\bin\...\xmlldb\trace_conf... |

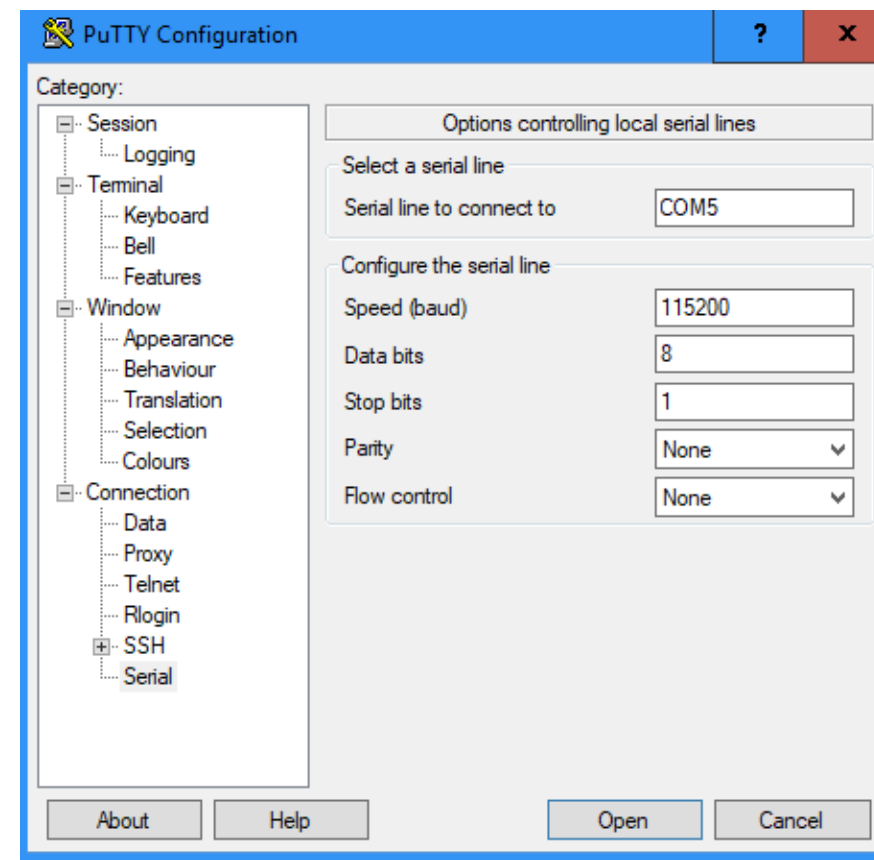
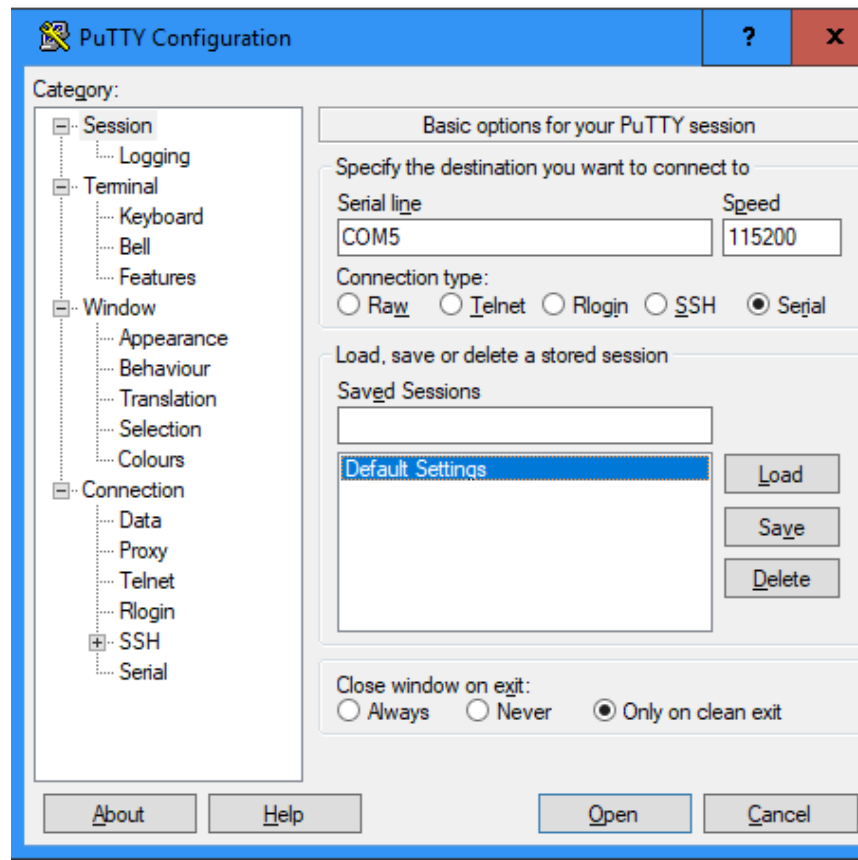
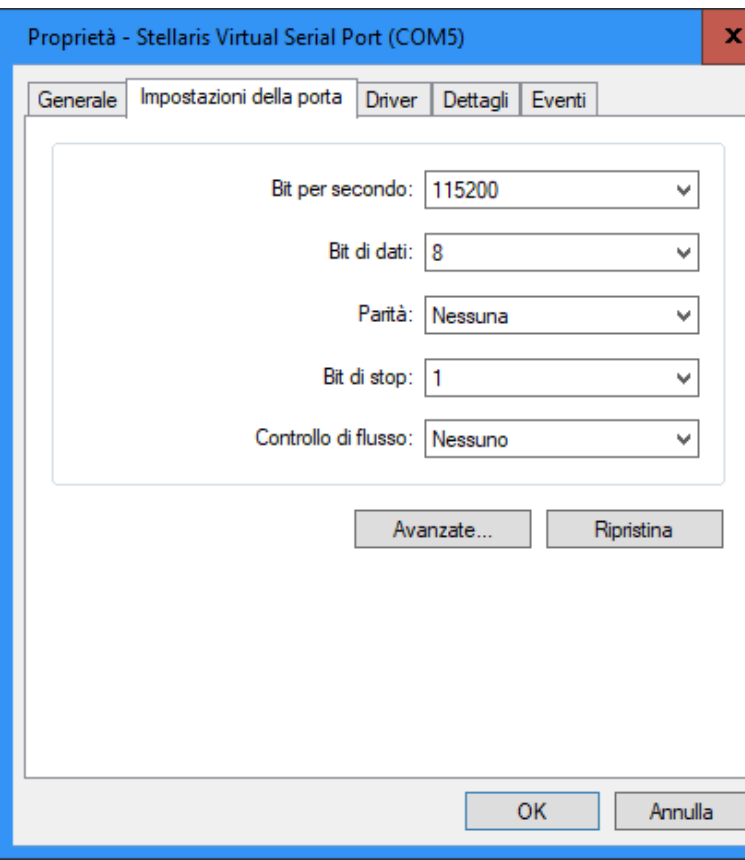
## 7 - USB BULK

The TivaWare drivers, necessary to run USB Bulk example, are not compatible with Windows 10:



## 8 - PUTTY CONFIGURATION

I had to configure the board COM Port and Putty like this:



## 9 – BUILD ERROR

In lab 9, I had to make some changing in build settings in order to run Workshop code. The error I faced with and the solution are the following:

The screenshot displays an IDE interface with several panels. On the left, the 'Console' panel shows the output of a build process, including a table of undefined symbols and their first referenced files, followed by error messages and a 'Build Finished' status.

| undefined symbol          | first referenced in file |
|---------------------------|--------------------------|
| ROM_FPUEnable             | ./main.obj               |
| ROM_FPULazyStackingEnable | ./main.obj               |
| ROM_SysCtlClockSet        | ./main.obj               |

error #10234-D: unresolved symbols remain  
error #10010: errors encountered during linking; "lab9.out" not built

>> Compilation failure  
makefile:135: recipe for target 'lab9.out' failed  
gmake: \*\*\* [lab9.out] Error 1  
gmake: Target 'all' not remade because of errors.

\*\*\*\* Build Finished \*\*\*\*

The 'Problems' panel in the center shows a list of 7 errors and 5 warnings. The 'Properties for lab9' window on the right is open, showing the 'Predefined Symbols' section. The 'Configuration' is set to 'Debug [Active]'. The 'Pre-define NAME (--define, -D)' section lists symbols like 'PART\_TM4C123GH6PM', 'ccs="ccs"', and 'TARGET\_IS\_TM4C123\_RB1'. The 'ARM Compiler' section is expanded, showing various options like 'Processor Options', 'Optimization', 'Include Options', 'MISRA-C:2004', and 'Advanced Options'.





THANKS FOR  
YOUR  
ATTENTION