
Pmod1_6 release v2 with MAX11300 Documentation

Release v2

Silica

Mar 16, 2017

Contents

1	New release v2: MAX11300 “Pixi” added!!!	3
1.1	Installing Codewarrior on WIN7 or WIN8	4
1.2	Firmware details	9
1.3	Firmware changes	12
1.4	More about Pmod	15
1.5	“PIXI” MAX11300	16
1.6	Quick start guide	33

Version 2.0.0 release v2

Date july 24,2014

Copyright (C)2016 by Avnet Silica company



Silica Brooklyn Board is useful system to evaluate MAXIM Pmod device and is designed for use with Freescale TWR-K70F120M tower system. This software release is working on Kinetis K70 devices but not under MQX rtos.

You can find and download TWR-K70 documentation by clicking:

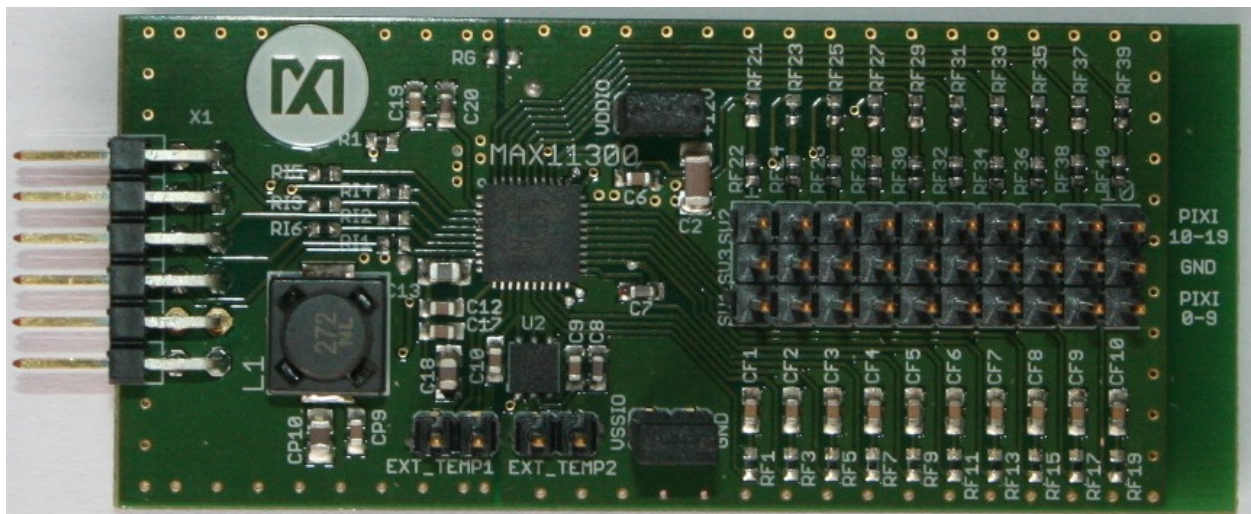
http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=TWR-K70F120M&tid=m32TWR

Developement software used is CodeWarrior MCU v10.3 Special Edition.

This application is free downloadable from Freescale site. To download it, [click here](#) and select “Special Edition: CodeWarrior for Microcontrollers 10.3 (Eclipse, Offline) (REV 10.3)” package. (note that download can take much time ...)

CHAPTER 1

New release v2: MAX11300 “Pixi” added!!!



The new release v2 is able to evaluate MAX11300 PIXI, 20-Port Programmable Mixed-Signal I/O with 12-Bit ADC, 12-Bit DAC, Analog Switches, and GPIO. Full description of Pmod Pixi's functions looking at specific chapter. For installation and configuration you can use same instructions as you can find in chapter *“Installing Codewarrior on WIN7 or WIN8”* and *“Quick Start Guide”*

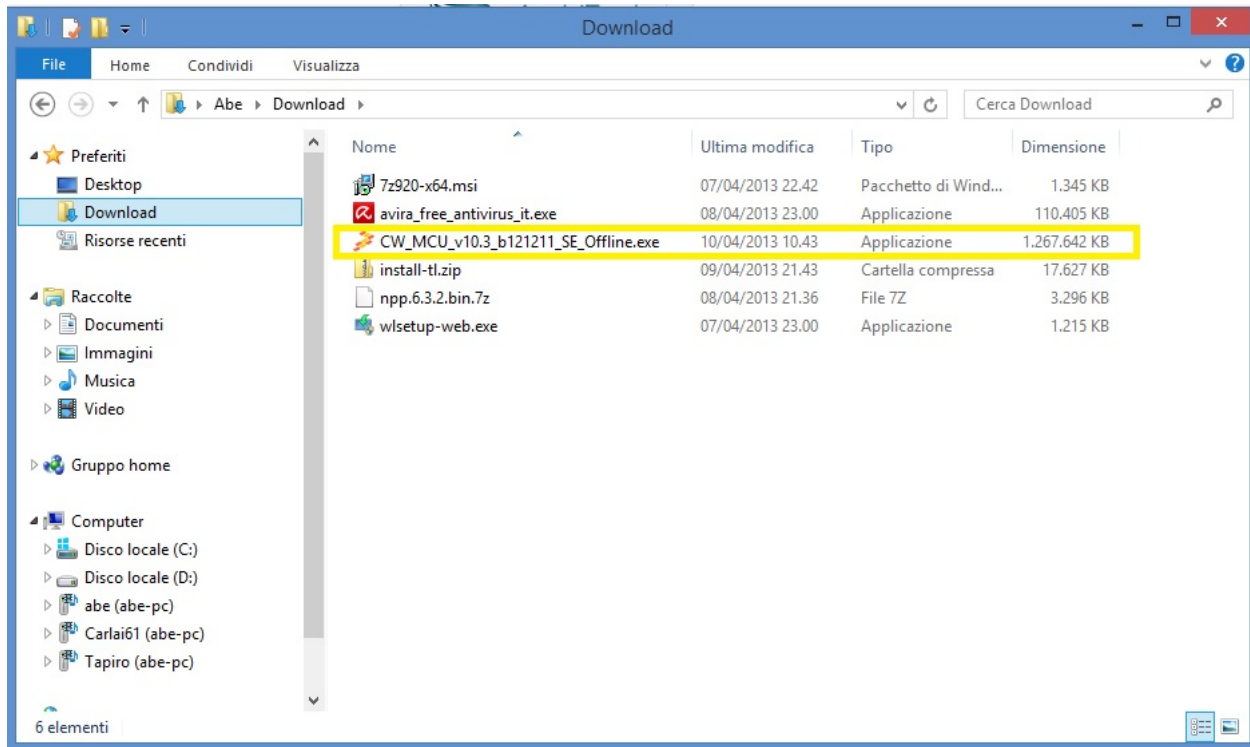
We suggest you to read the Quick Start Guide to setup your evaluation system

Quick start guide

This guide explains how to use this application and provides an overview of on the structure of the project firmware

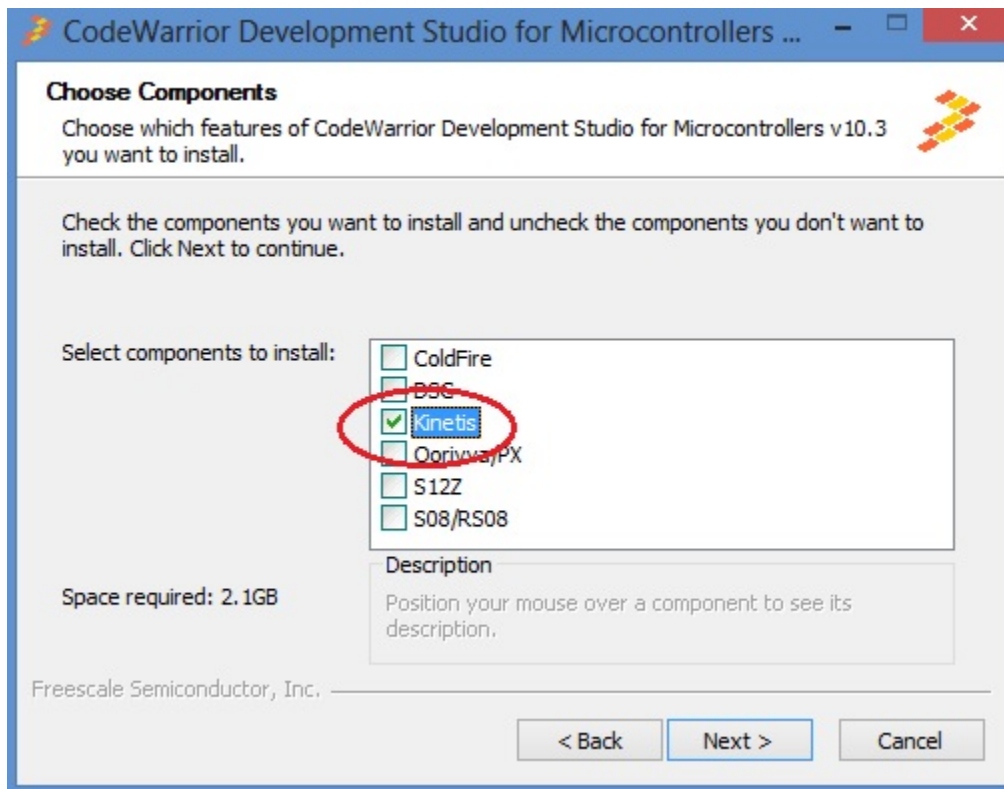
Installing Codewarrior on WIN7 or WIN8

Double click on 'CW_MCU_v10.3_b121211_SE_Offline.exe' that you have downloaded first, and follow installation instruction.

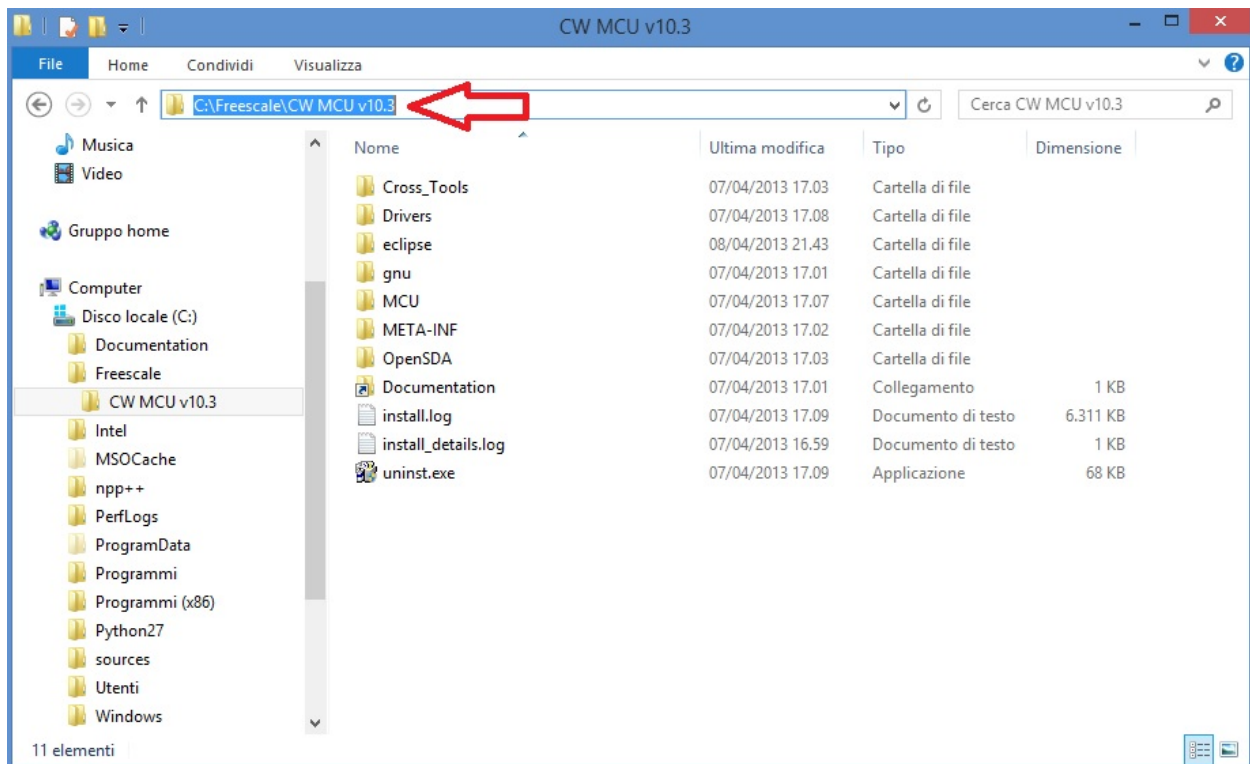


Quick start guide for Codewarrior install can be found at http://cache.freescale.com/files/soft_dev_tools/doc/quick_ref_guide/926-77846.pdf?fsrch=1&sr=3

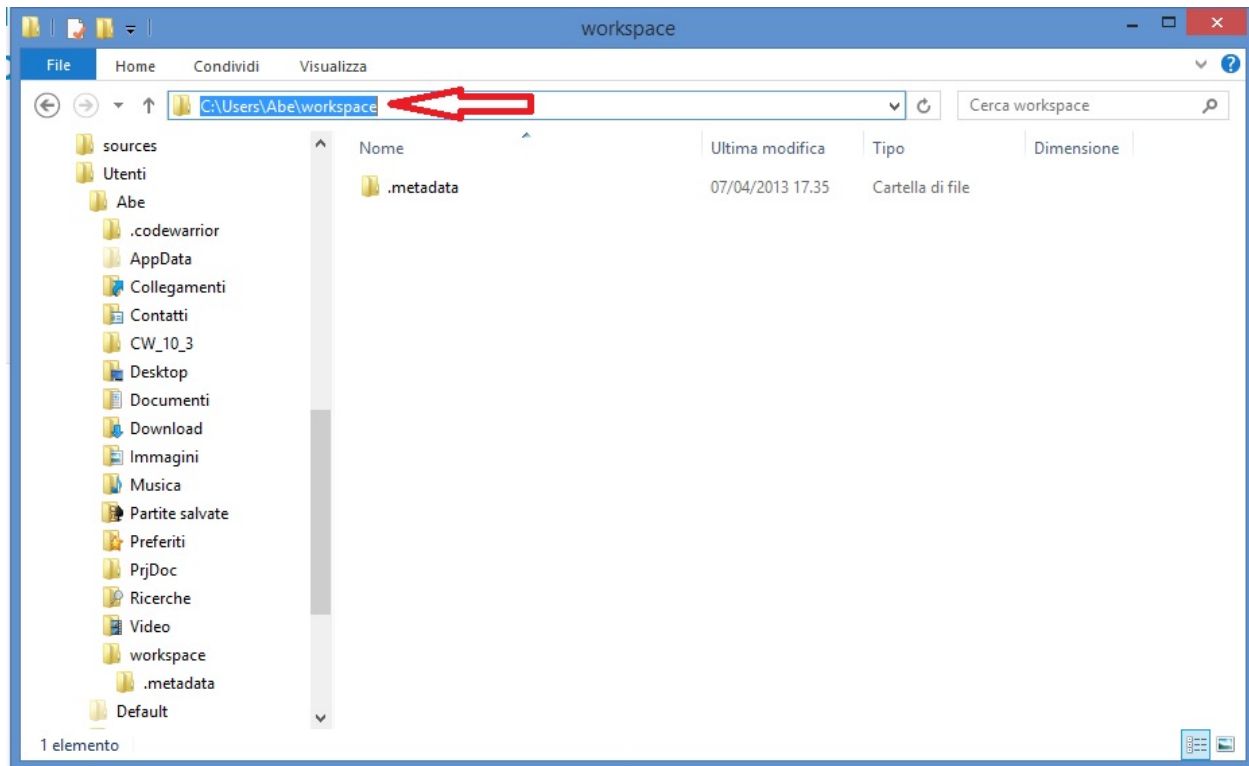
When setup ask for MCU type, select almost Kinetis as show below, then press NEXT button



If you have Windows7 or Windows8 (32 or 64 bit) Codewarrior will install into “Freescal\CW MCU v10.3” folder on the root of your system HDD.

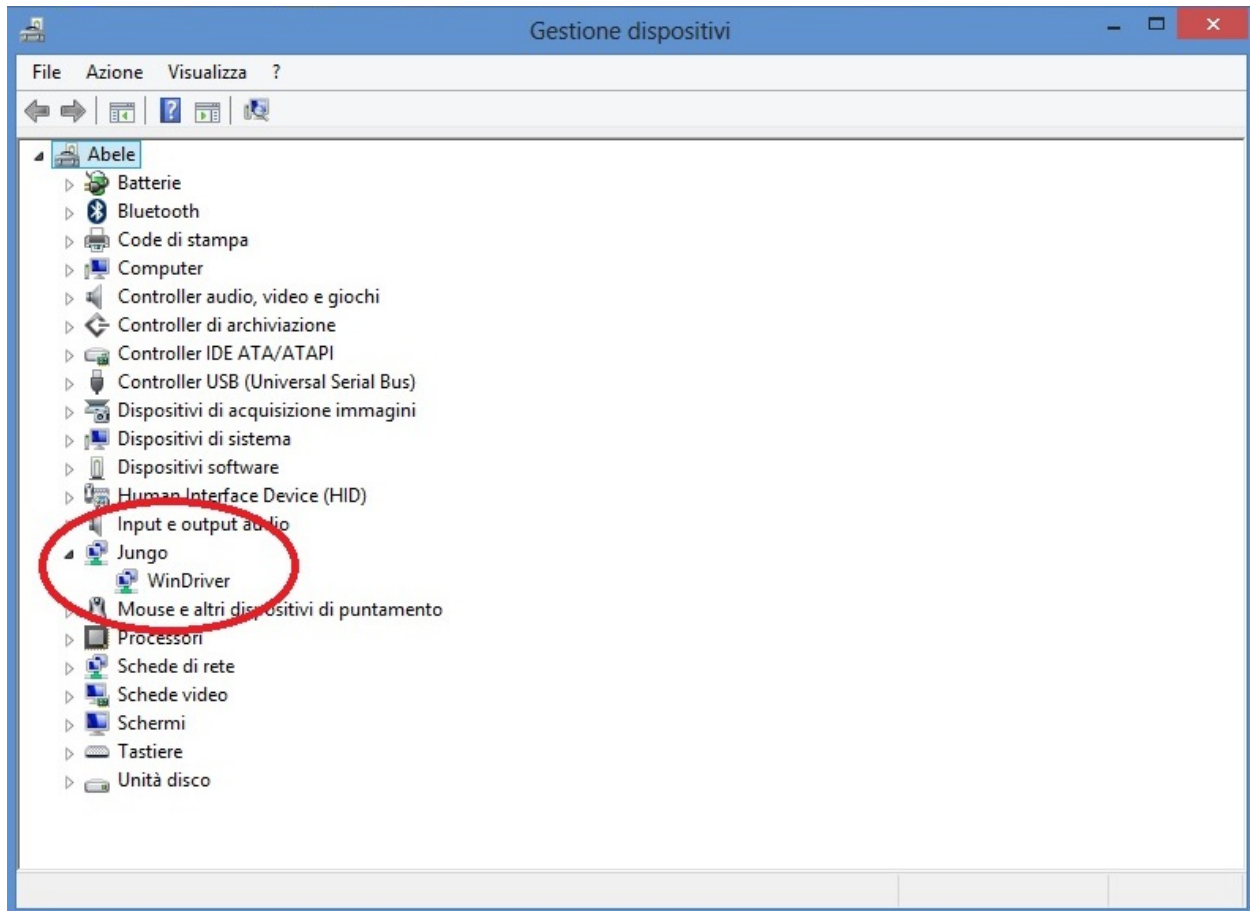


and create a default work folder named **workspace** in the path `C:\Users\your_user_name\workspace`



make shure that setup have been installed Jungo Driver. See your system configuration (righ-click on Computer -> Properties -> Device Manager)

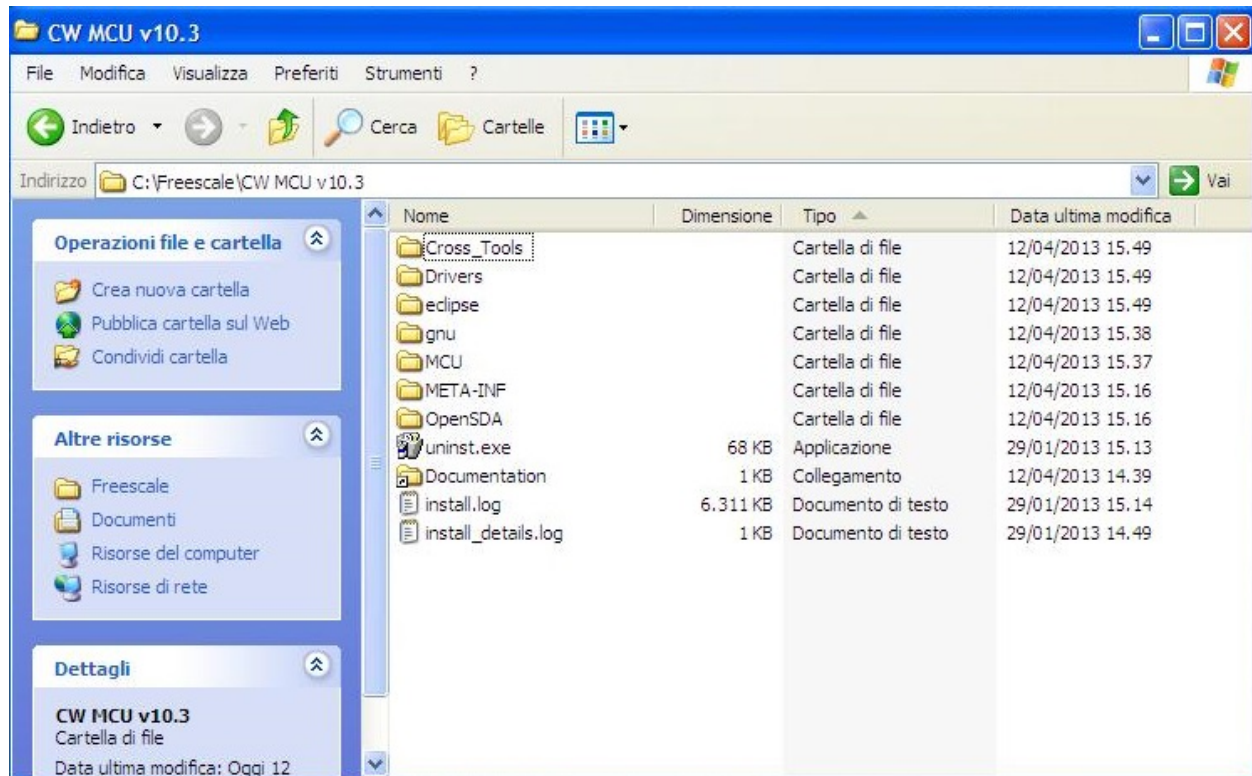
if you have any troubles about, read Codewarrior install Guide



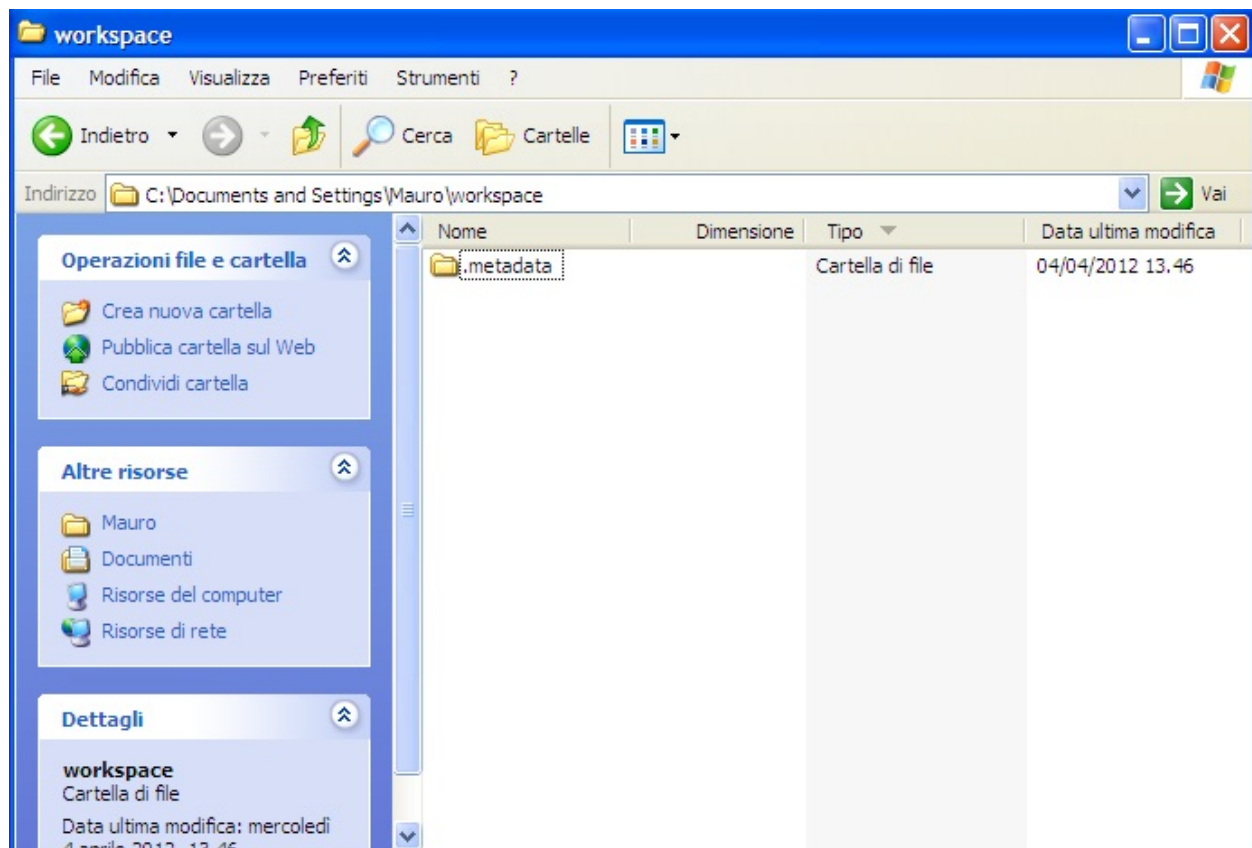
Codewarrior on WINXP

For WinXP, after installation, you have:

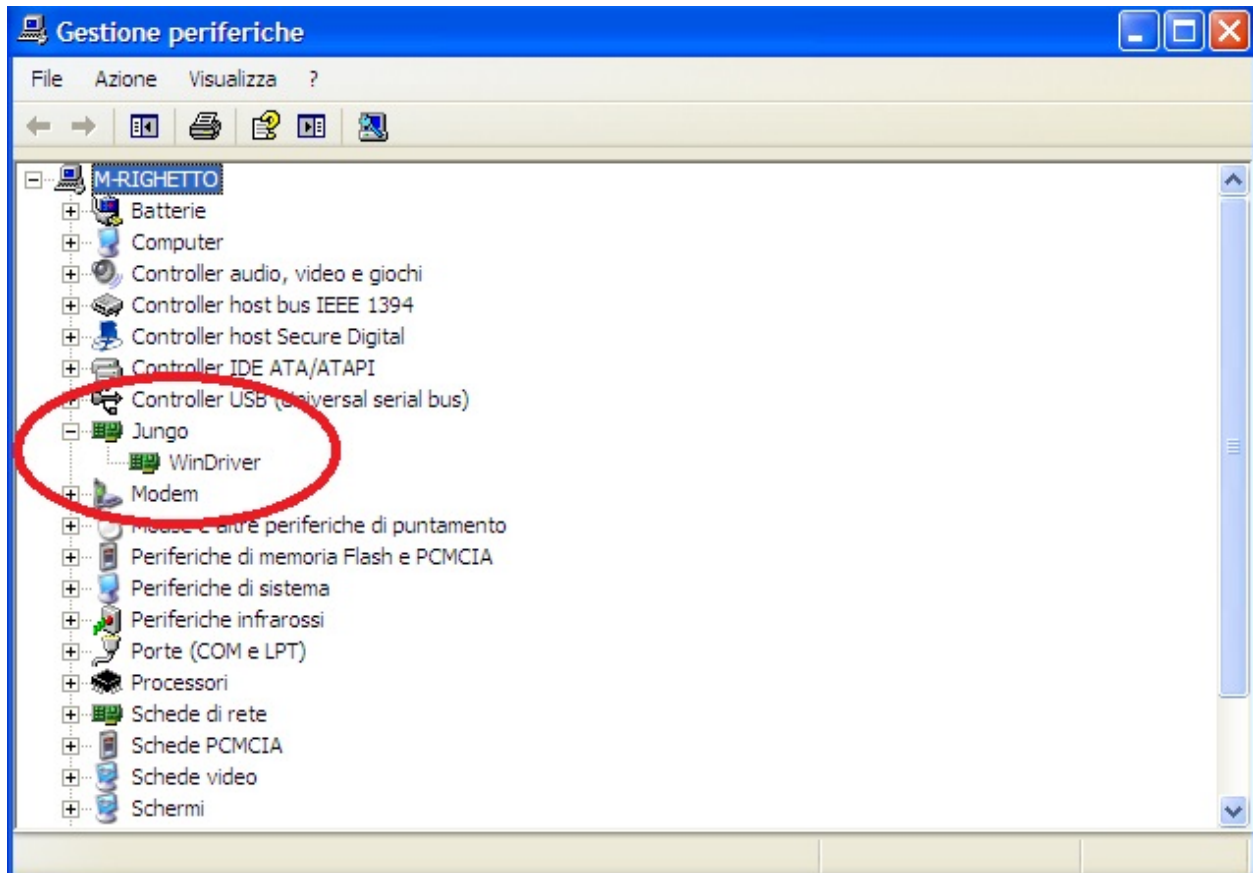
- installation folder



- default work folder



- Jungo driver:



Firmware details

Brookling Board firmware comes from original Maxim Maxim Zenboard Platform project revision 1.6, by using the file listed above.

Main project files from Maxim

- MaximPmod.c
- menu.c.
- maximDeviceSpecificUtilities.c
- platform.c
- utilities.c

and related include files

- MaximPmod.h
- menu.h
- maximDeviceSpecificUtilities.h
- platform.h

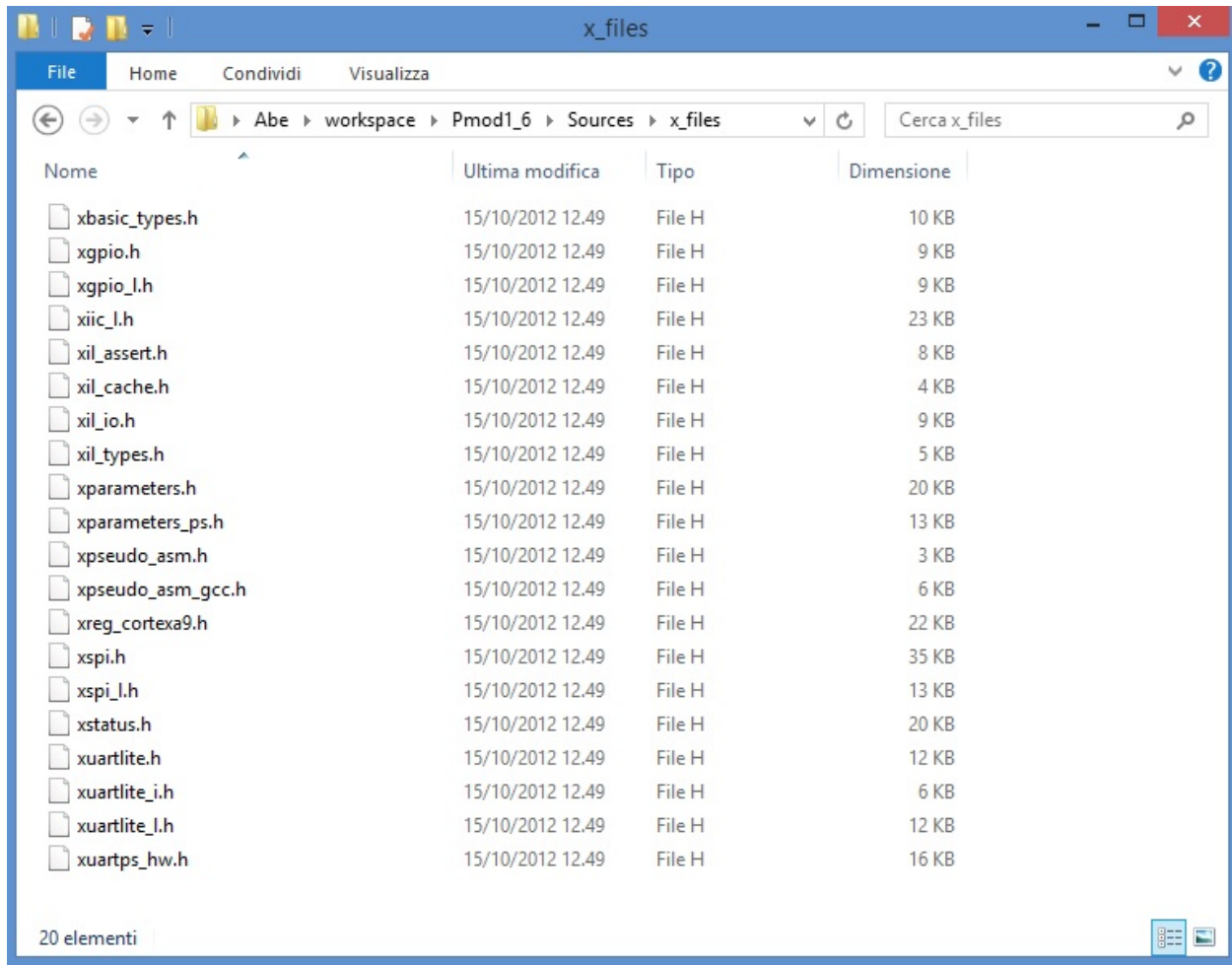
- utilities.h
- platform_config.h

You can find all this file in the “Source” folder of the project

General include files

- xbasic_types.h
- xgpio.h
- xgpio_l.h
- xiic_l.h
- xil_assert.h
- xil_cache.h
- xil_io.h
- xil_types.h
- xparameters.h
- xparameters_ps.h
- xpseudo_asm.h
- xpseudo_asm_gcc.h
- xreg_cortex9.h
- xspi.h
- xspi_i.h
- xspi_l.h
- xstatus.h
- xuartlite.h
- xuartlite_i.h
- xuartlite_l.h
- xuartps_hw.h

You can find all this file in the “Source\x_files” folder of the project



Main Project files added

In source folder you find application specific files:

- ProcessorEspert.c (containing Main())
- Events.c
- driver.c (low-level function replacement)

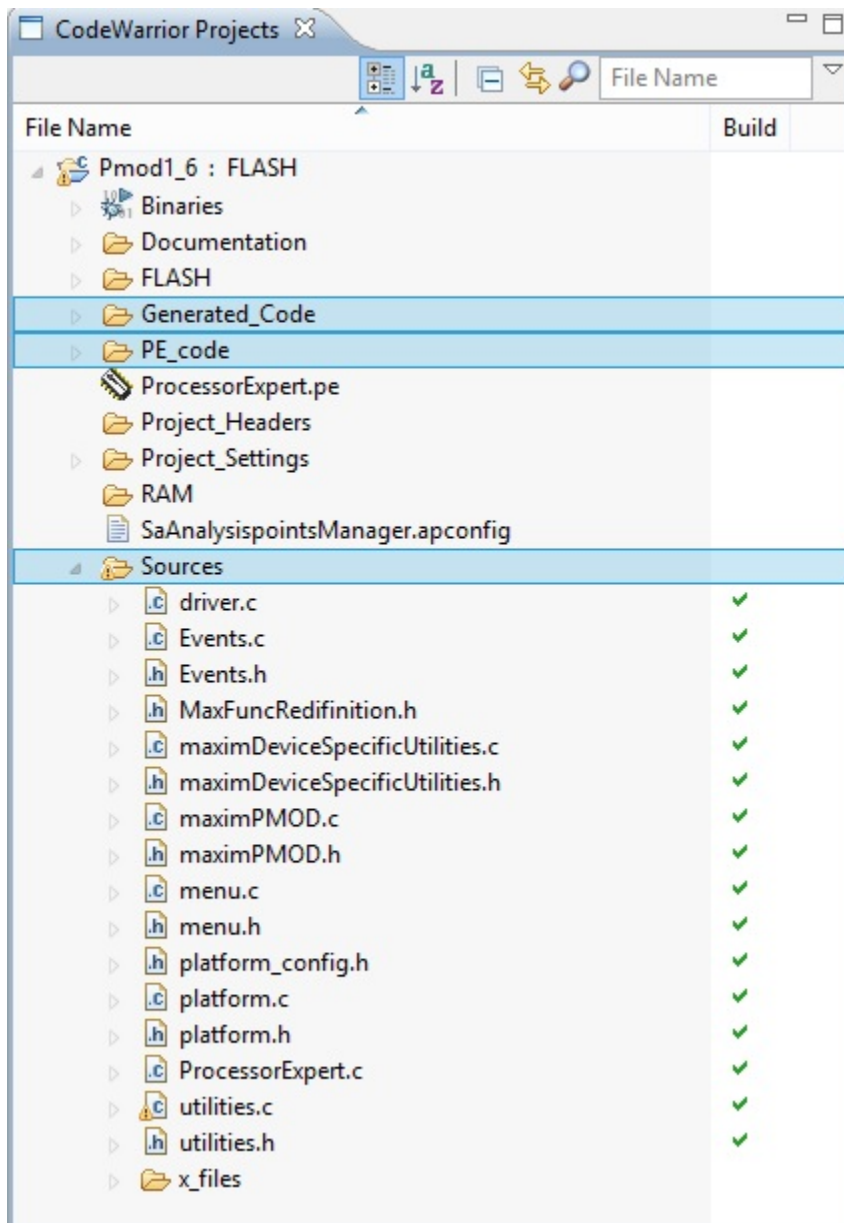
and include files

- MaxFuncRedefinition.h (start menu function redefinition)
- Events.h

This project is developed under Codewarrior 10.3 using Processor Expert tools. You can also find two folders, one named “Generated_Code” and second named “PE_code”

The first one contain auto-generated file that depend on Logical Device component inside Processor Expert Components, and may rebuild each occurrence (when a device settings modification has been made)

The second contains file one-time generated by Processor Expert and “frozen” in this folder. No modification can be invoked by Processor Expert.



Detailed documentation of firmware can be downloaded from: <http://www.architechboards.org/product/brooklyn-board>

Firmware changes

Changes in Pmod1_6 revision

1 - added include file **MaxFuncRedifinition.h** at the top of **MaximPmod.c** file. This file must be the first include in list.

```
#include "MaxFuncRedifinition.h" <----
#include <stdio.h>
```



```
#include "platform.h"
#include "menu.h"
#include "utilities.h"
#include "maximDeviceSpecificUtilities.h"
#include "maximPmod.h"

#define MAJOR_REVISION 1
#define MINOR_REVISION 6
```

2 - renamed **main()** function inside **MaximPmod.c** file with new name **main_pmod()**.

```
int main_pmod()          <----
/**
 * \brief      Main() function for Analog Essentials example program.
 * \par       Details
 *            This function sets up and initializes the FPGA and hardware, displays
↳ the root menu via
 *            Hyperterminal, then dispatches inidividual demo programs for specific
↳ module based on
 *            user's keypress selection.
 *
 * \param      None
 *
 * \retval     Always TRUE
 */
{
    // Variables for the main() function
    u8 uchInput=0;
    int nMenuState=0;
    int i=0;
    char tempString[256];
```

3 - commented function **led_knight_rider** inside **MaximPmod.c** file to obtain application fast start.

```
// Toggle the LEDs so that the user knows the board is awake
XGpio_Initialize(&g_xGpioLed, XPAR_AXI_GPIO_LED_DEVICE_ID);
XGpio_SetDataDirection(&g_xGpioLed, 1, 0x00000000); // Set the LED peripheral
↳ to outputs
// led_knight_rider(&g_xGpioLed,2);          <----
```

4 - changed costant definition **ABOUT_ONE_SECOND** inside **MaximPmod.h** file as follow:

```
#define ABOUT_ONE_SECOND 74067512/8/3    <----
// #define ABOUT_ONE_SECOND 74067512      //!< approx 1 second delay when used as
↳ argument with function delay(numberCyclesToDelay)
// Update this if uBlaze/Zynq CPU core frequency is changed, or if the external
↳ memory timing changes.
// Although emprirically tested to 1.0000003 seconds, it is not meant to be used for
↳ precise timing purposes
```

5 - changed triangle wave ramp value for MAX5216 (file menu.c, row 765 anf 769)

```
case 12:
    printf("Triangle Wave started\r\n");
    fflush(stdout);
    for(i=0; i<300; i++)
    {
        for(j=0; j<65535; j=j+23)
```

```
↪definition                                for (j=0; j<65535; j=j+230)      <-- new ramp_
                                           {
                                           max_MAX5216_set_output_voltage(g_
↪pActiveGPIOPort, j);
                                           }
//                                         for (j=65535; j>=0; j=j-23)
                                         for (j=65535; j>=0; j=j-230)      <-- new ramp_
↪definition                                {
                                           max_MAX5216_set_output_voltage(g_
↪pActiveGPIOPort, j);
                                           }
                                           }
                                           nMenuState = 0;
                                           break;
```

NOTE: All these changes are tested on revision 1.6 of Maxim project files and need to be checked on further new revisions

Changes in Pmod1_6_v2 revision

(MAX11300 added)

The v2 firmware is developed in order to add evaluation capability for MAX11300 Pixi device. To make this, starting from Pmod1_6 revision, there are some changes inside main project files. You can search a comment “*added for MAX11300*” in order to detect major changes.

Main function modified

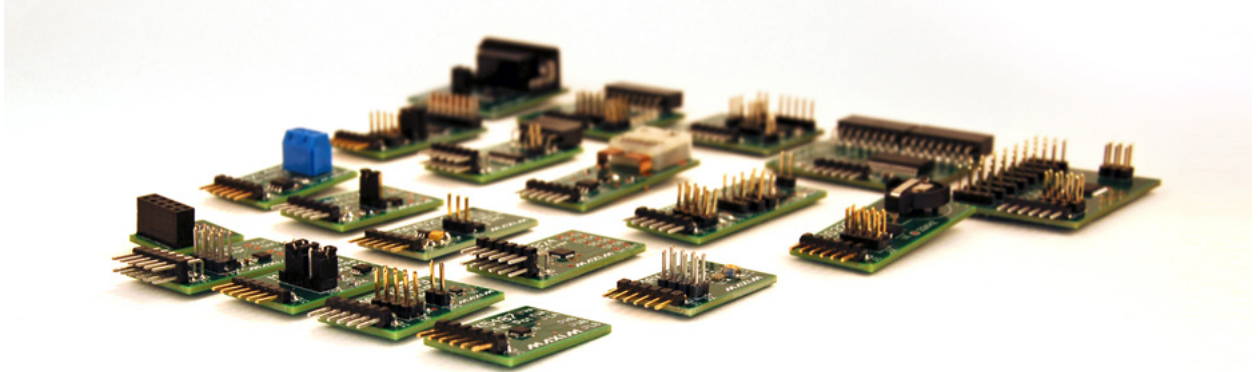
- **file driver.c**
 - xil_out32()
 - init_device()
 - __menuMAX11300() *new function*
- **file maximPMOD.c**
 - MENU_MAX11300 (added in function main_pmod())
 - changed revision message for terminal
- **file menu.c**
 - menu_get_direct_entry_func()
 - print_main_menu()
 - menu_MAX11300() *new function. Perform all evaluation capability*

bug fix

- **file maximDeviceSpecificUtilities.c**
 - added timeout during MAX11205 init function: if device is not connected, print error message and proceed.

More about Pmod

Maxim Analog Essential Collection is a collection of plug-in peripheral modules (Pmod) You can find more informations visiting [Maxim Analog Essential Collection](#) site



Important notice

At the date of issue of this review, Maxim Zenboard Platform Project files are available on version 1.6, and don't support MAX14850 Pmod module. As a result this version, that use original files from Maxim project, is not able to emulate this device.



Emulation of MAX3232

This device is a RS232 converter, and require 2 serial channel (each one connected to terminal software) for full test. The first one is used for commands and the second one must be connected, for complete testing purpose, to MAX3232 Pmod serial connector (by standard modem cable).



“PIXI” MAX11300

This guide will explain how can you work with MAX11300PMB1 Pmod Peripheral Module to evaluate the ‘Pixi’ performance.

Datasheet of MAX11300 device and MAX11300PMB1 Pmod module can be found at:

<http://www.maximintegrated.com/en/pst/run.mvp?q=max11300>

First steps with Pixi

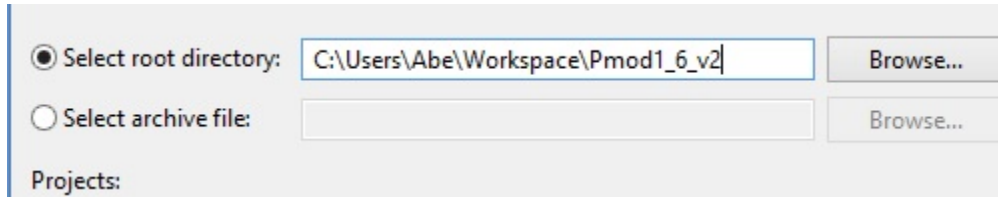
Before starting with Pixi PMB1 Pmod board, first of all you must install *CodeWarrior MCU v10.3 Special Edition*. You must have on your PC terminal application software (such as HyperTerminal) and USB to serial converter. To evaluate Pixi performance use Freescale TWR-K70F120M tower system equipped with Brooklyn Board expansion board. For correct setup of your Hardware and Firmware environment, follow instruction inside chapters:

Installing Codewarrior on WIN7 or WIN8

Quick start guide

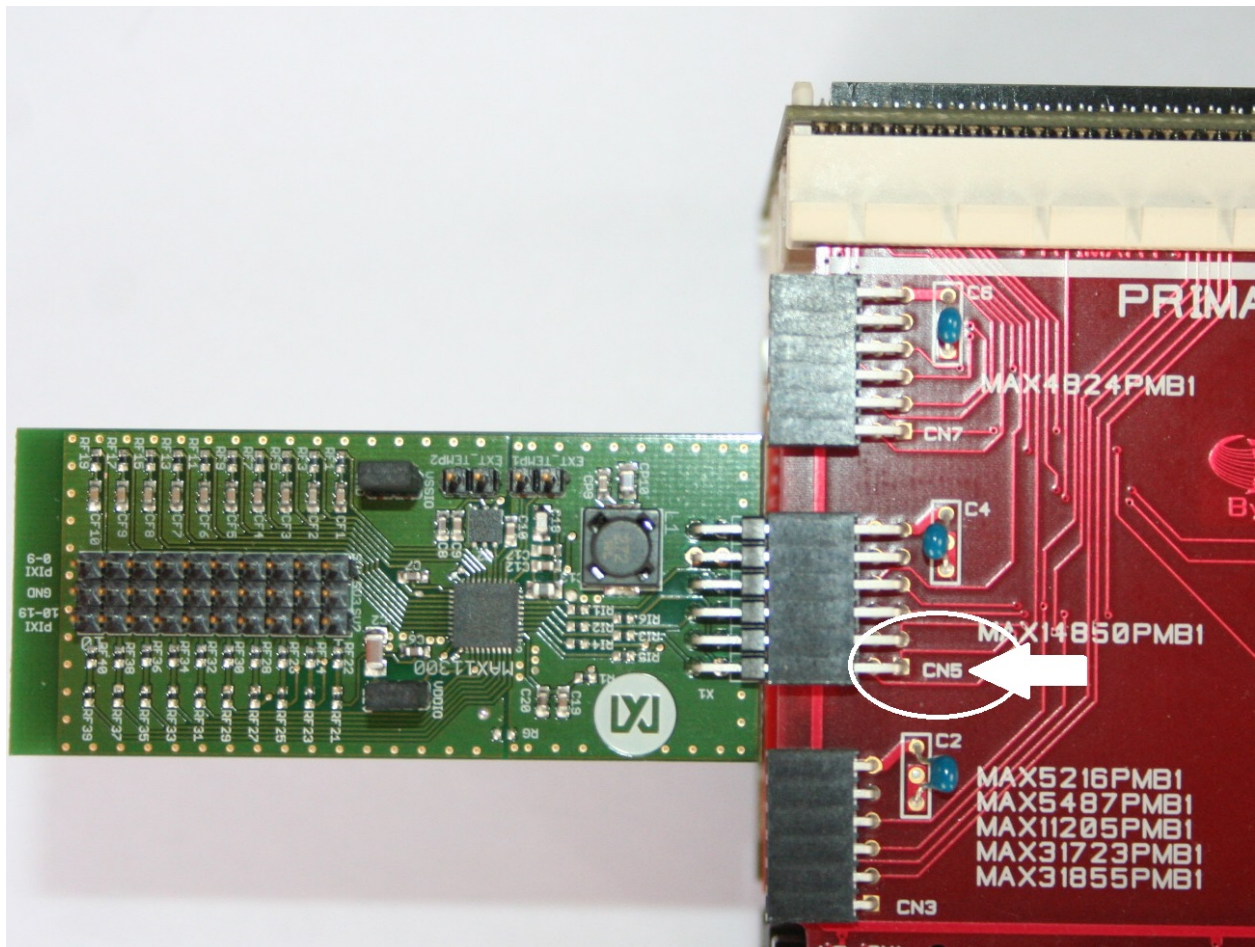
Important: Download from “Silica Architec Site” under the section “Brooklyn Board” the latest Pmod Firmware (Pmod1_6_v2.zip) and follow installation and configuration instructions specified for Pmod1_6

Note: Take care at project importer window on Codewarrior: you must import project named Pmod1_6_v2



Setting your hardware

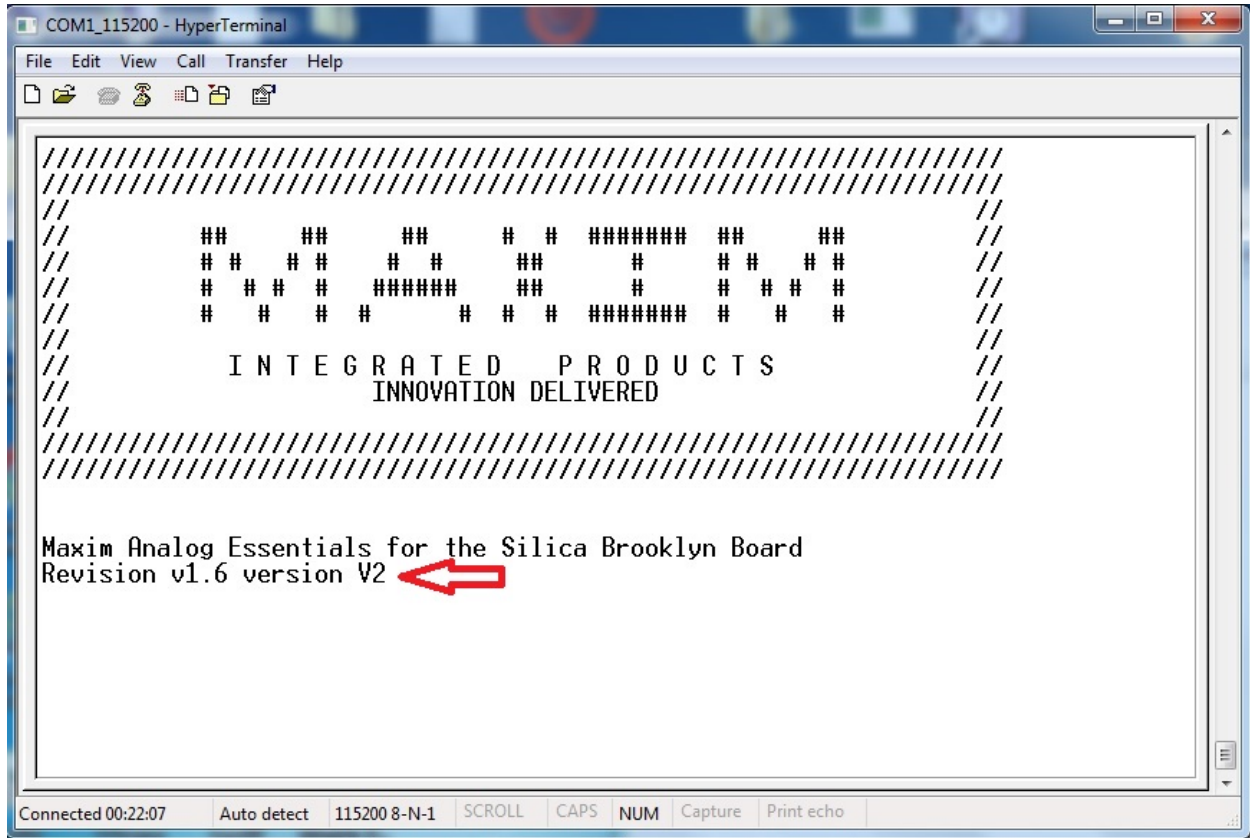
The MAX11300PMB1 Pmod board must be inserted on CN5 connector of Brooklyn Expansion Board:



Now you are ready to evaluate Pixi! Enjoy!

Starting Pmod1_6_v2 firmware

When program starts, you can see the two following screens on your terminal:



```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help
[Icons]

////////////////////////////////////
//
//      ##      ##      ##      # #      #####      ##      ##
//      # #      # #      # #      ##      #      # #      # #
//      # # #      #####      ##      #      # # #      #
//      # #      # #      # #      #####      # #      #
//
//      I N T E G R A T E D      P R O D U C T S
//      I N N O V A T I O N      D E L I V E R E D
//
////////////////////////////////////

Maxim Analog Essentials for the Silica Brooklyn Board
Revision v1.6 version V2

Connected 00:22:07  Auto detect  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo
```

and then, after about 3 seconds:

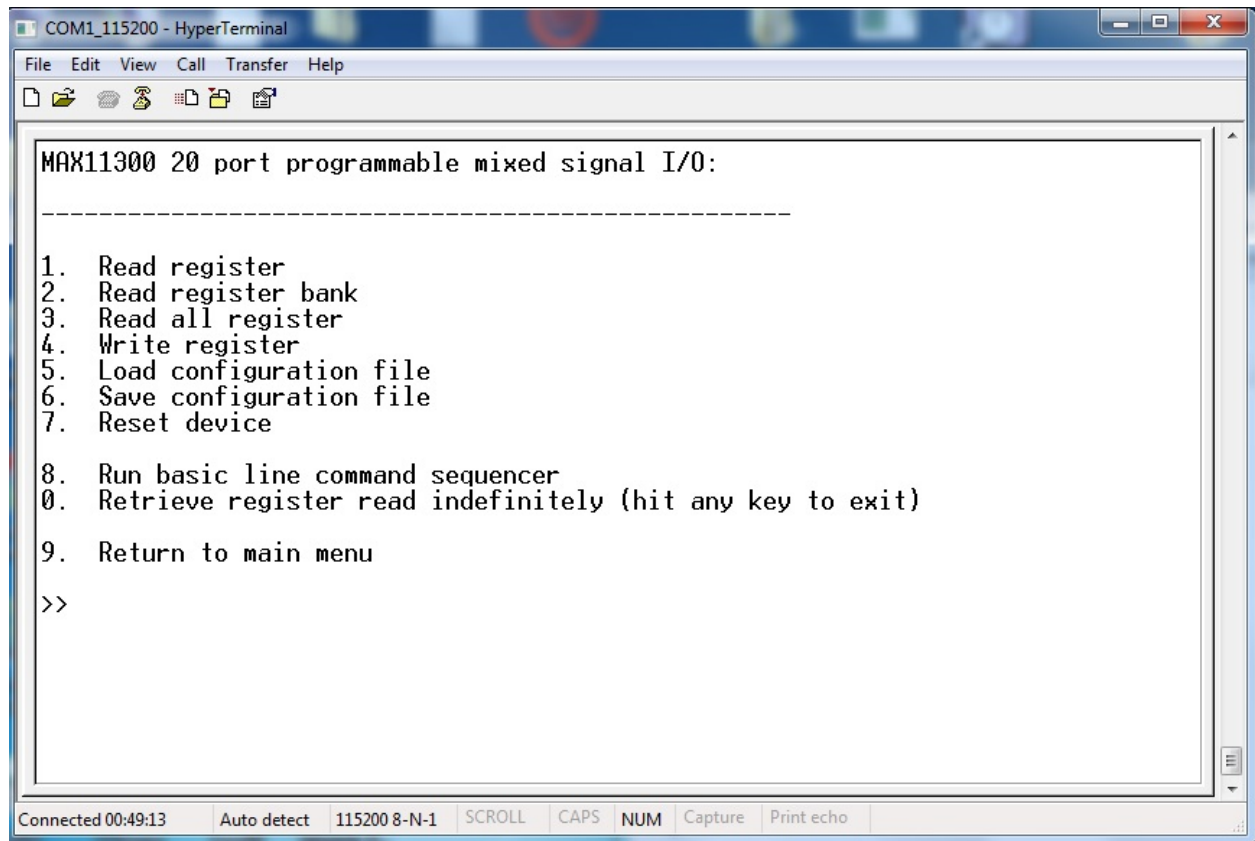
```

//      ##      ##      ##      # # ##### ##      ##      //
//      # #      # #      ##      # #      # #      //
//      # # #      #####      ##      # #      # #      //
//      # #      # #      # # # ##### # #      # #      //
#####
MAXIM PMOD1_6 release v2 - Press a key/number to test a Peripheral Module:
{X} MAX11300  PIXI programmable I/O with DAC, ADC, Analog switch and GPIO
{0} DS1086L   Spread Spectrum Econ Oscillator
{1} DS3231M   I2C Real-Time Clock
{2} MAX3232   RS-232 Transceiver
{3} MAX4824   Octal Relay Driver
{4} MAX5216   SPI-Compatible, High performance 16 bit DAC
{5} MAX5487   Dual, 256-tap SPI Digital Potentiometer
{6} MAX5825   I2C Octal DAC
{7} MAX7304   I2C-Interfaced 16-Port, GPIO and LED Driver
{8} MAX9611   Current sense amplifier with OpAmp and ADC
{9} MAX11205  16 bit Delta-Sigma ADC with 2-Wire interface
{A} MAX14840  RS-485 Transceiver
{B} MAX31723  MAX31723 Digital Thermometer
{C} MAX31855  Thermocouple to Digital Converter
{D} MAX44000  I2C-Interfaced Ambient Light and Proximity Sensor
>>_

```

Connected 00:45:35 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo

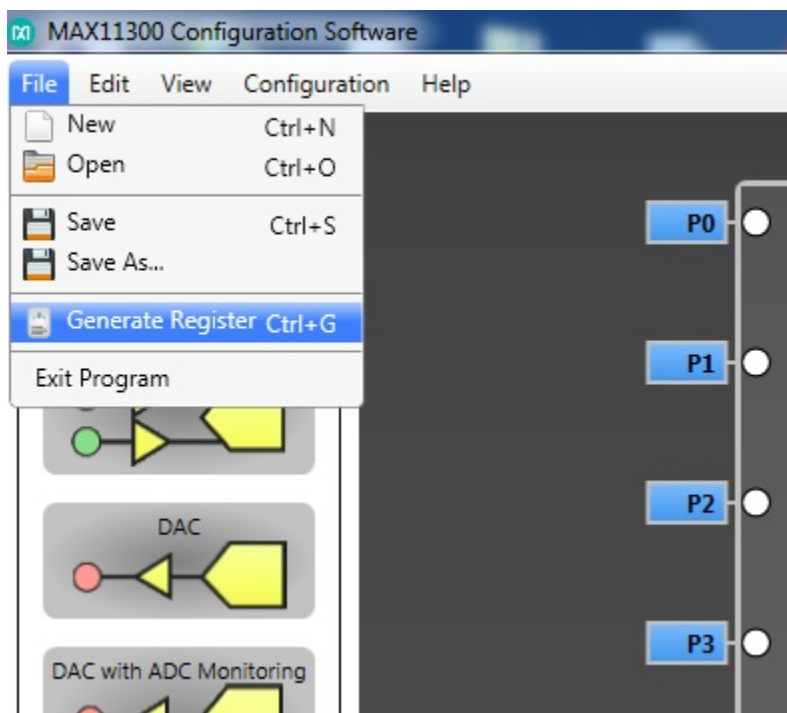
By typing “X” at terminal prompt, you can see **MAX11300 main menu** (figure below)



Follow instruction on screen to perform basic read/write functions or reset device to power-up default

Load and save configuration

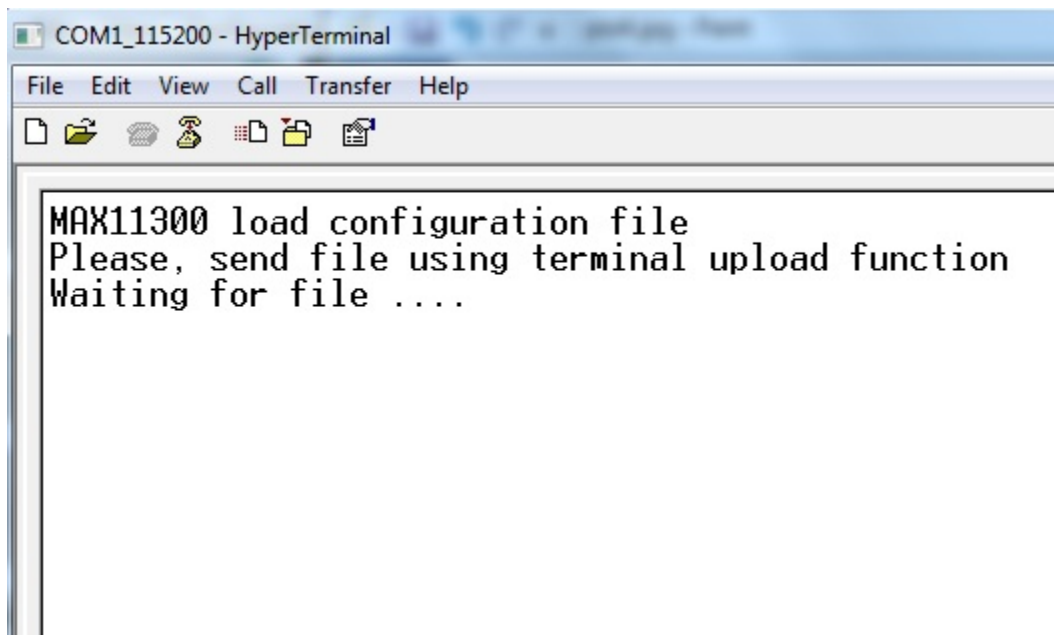
Using this functions you can load a MAX11300 configuration file generated by the MAXIM PC application “MAX11300 Configuration Software”. You can use “Generate register” function inside “File” menu from the Maxim PC application (*figure below*)



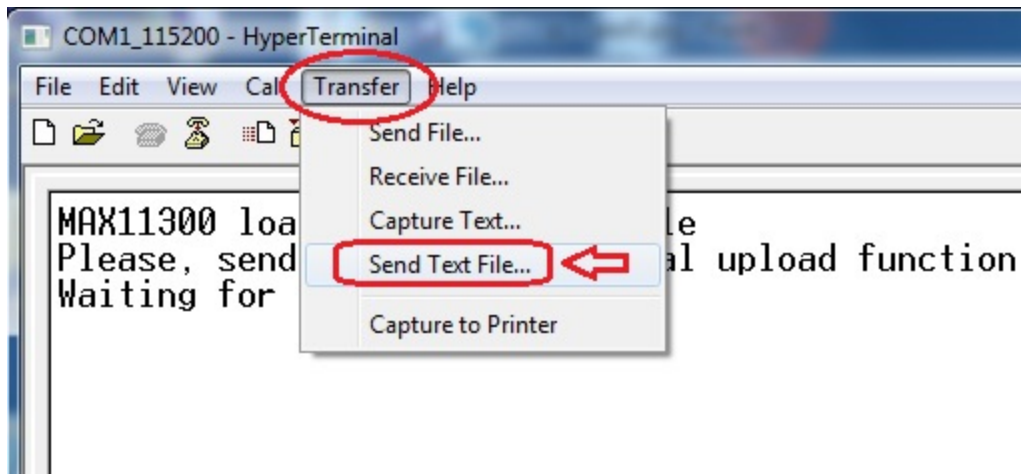
The MAXIM application will store a ".csv" file format, that can be read by Pmod1_6_v2 and stored inside the MAX113000PMB1 evaluation board.

Load Configuration instructions

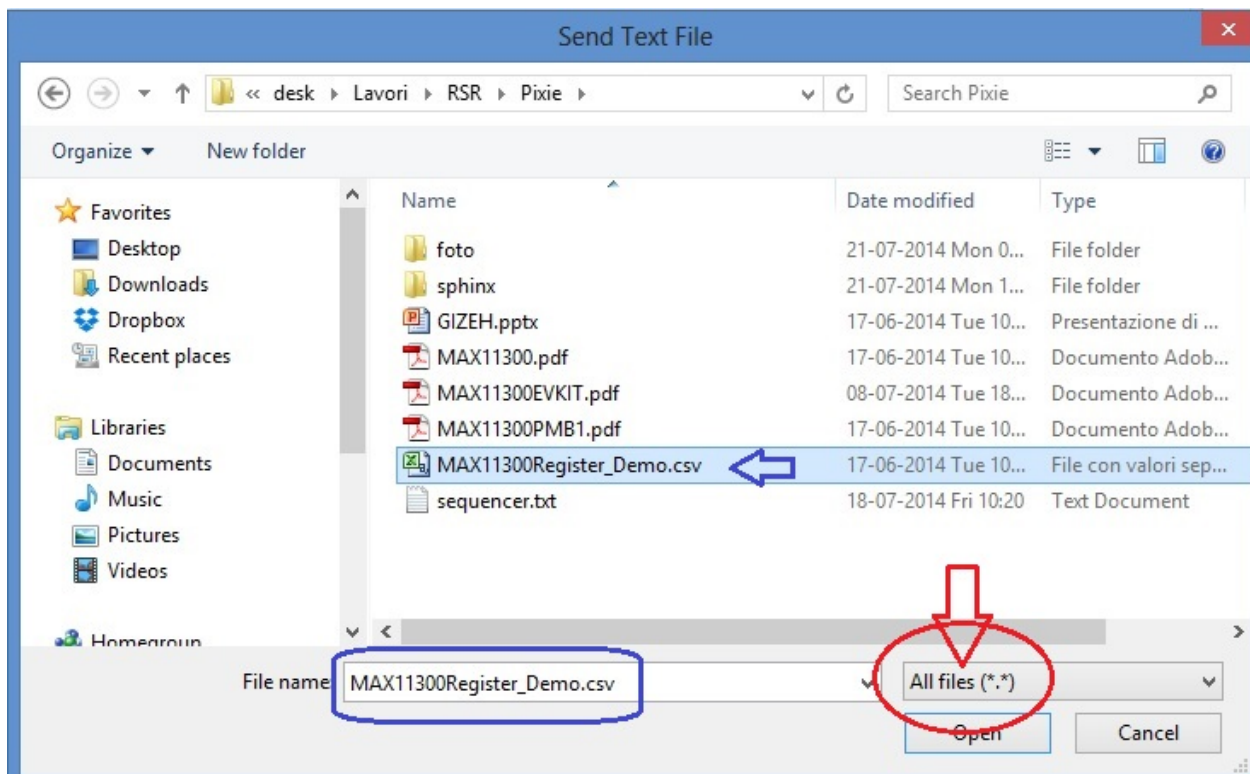
At prompt of MAX11300 main menu, type "5"



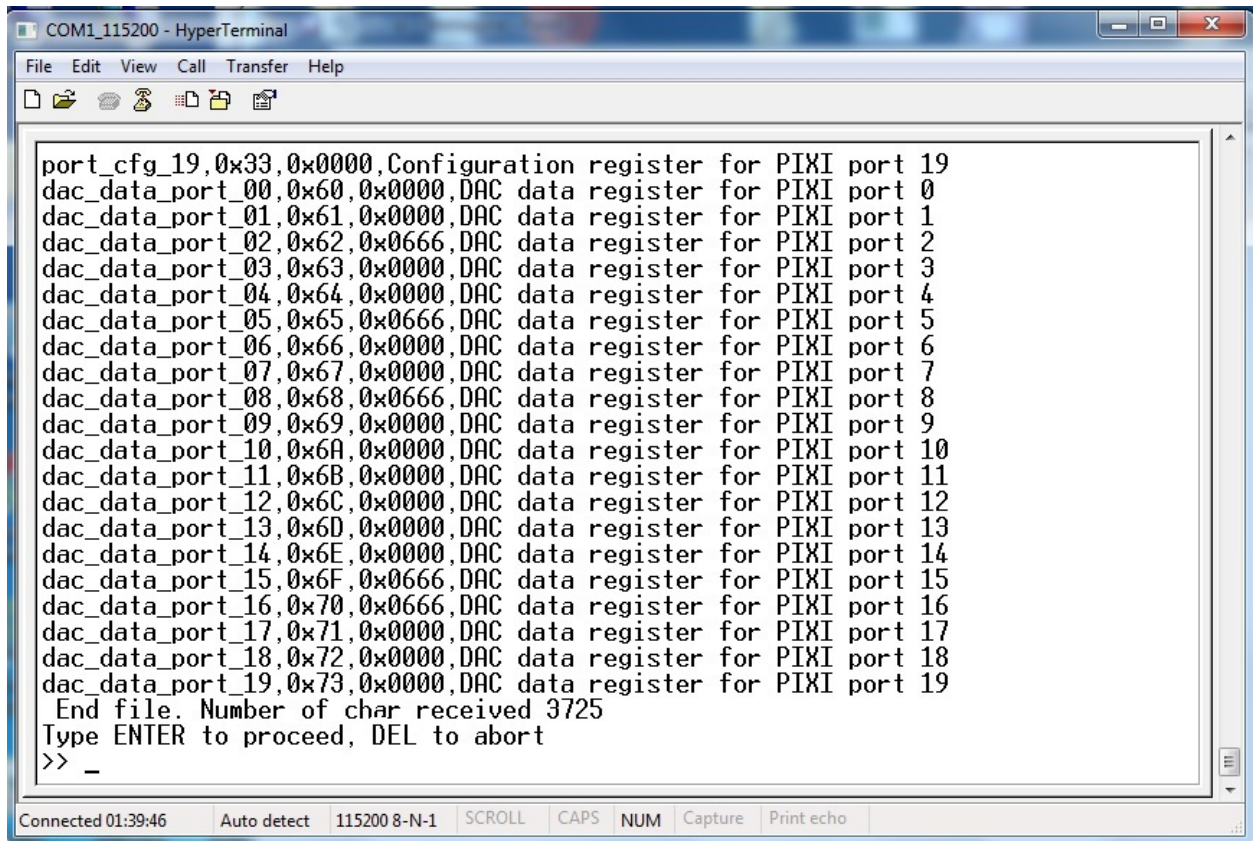
Now you can see on your terminal the message in figure below.



click on “**Transfer**” menu, select “**Send Text File**” and click over

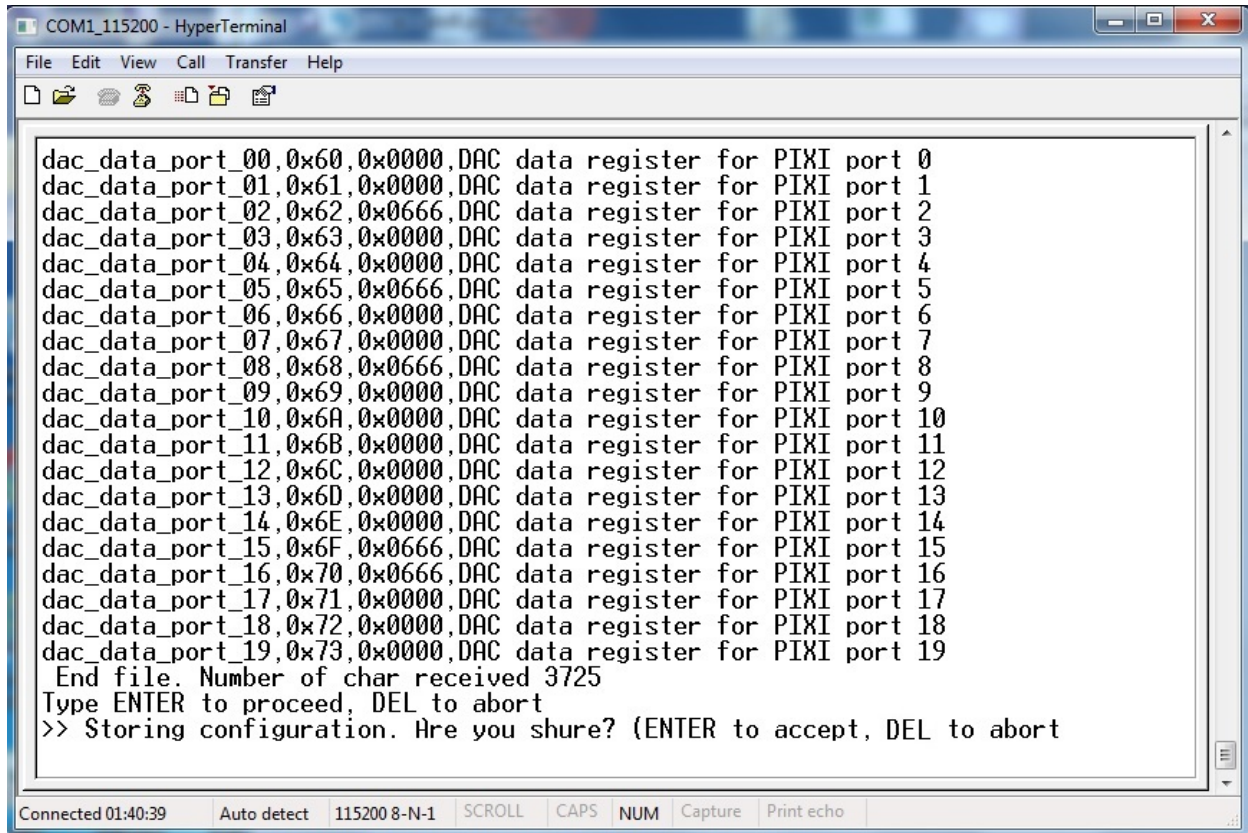


navigate to your folder containing the MAX11300 register configuration Files (an example file can be found in the Codewarrior project folder under documentation\examples), select “**All Files**” in the pulldown menu (*red-circled*) and then select file (as example blue-signed). Click “Open” and file will be transferred.



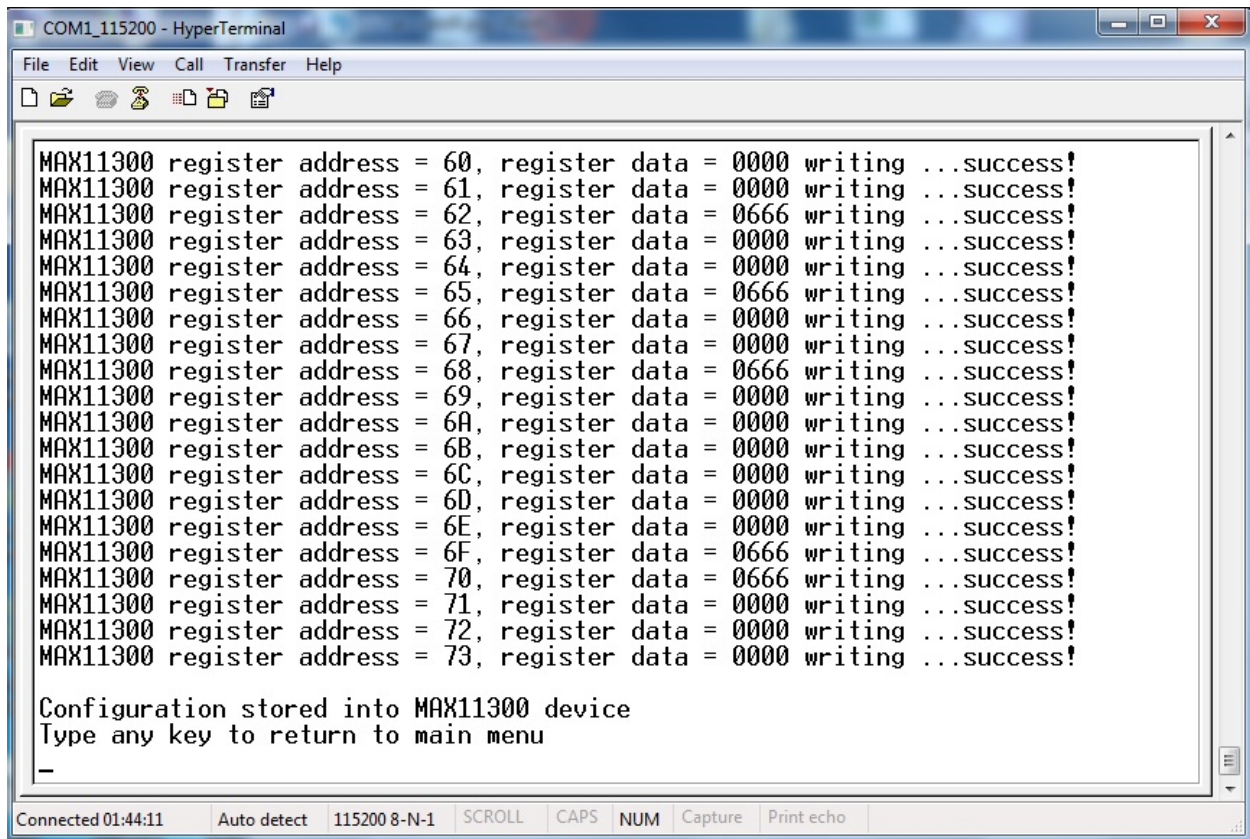
```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help
port_cfg_19,0x33,0x0000,Configuration register for PIXI port 19
dac_data_port_00,0x60,0x0000,DAC data register for PIXI port 0
dac_data_port_01,0x61,0x0000,DAC data register for PIXI port 1
dac_data_port_02,0x62,0x0666,DAC data register for PIXI port 2
dac_data_port_03,0x63,0x0000,DAC data register for PIXI port 3
dac_data_port_04,0x64,0x0000,DAC data register for PIXI port 4
dac_data_port_05,0x65,0x0666,DAC data register for PIXI port 5
dac_data_port_06,0x66,0x0000,DAC data register for PIXI port 6
dac_data_port_07,0x67,0x0000,DAC data register for PIXI port 7
dac_data_port_08,0x68,0x0666,DAC data register for PIXI port 8
dac_data_port_09,0x69,0x0000,DAC data register for PIXI port 9
dac_data_port_10,0x6A,0x0000,DAC data register for PIXI port 10
dac_data_port_11,0x6B,0x0000,DAC data register for PIXI port 11
dac_data_port_12,0x6C,0x0000,DAC data register for PIXI port 12
dac_data_port_13,0x6D,0x0000,DAC data register for PIXI port 13
dac_data_port_14,0x6E,0x0000,DAC data register for PIXI port 14
dac_data_port_15,0x6F,0x0666,DAC data register for PIXI port 15
dac_data_port_16,0x70,0x0666,DAC data register for PIXI port 16
dac_data_port_17,0x71,0x0000,DAC data register for PIXI port 17
dac_data_port_18,0x72,0x0000,DAC data register for PIXI port 18
dac_data_port_19,0x73,0x0000,DAC data register for PIXI port 19
End file. Number of char received 3725
Type ENTER to proceed, DEL to abort
>> _
Connected 01:39:46 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```

When transfer ends, press “**ENTER**” to start programming device, “**DEL**” to abort



```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help
dac_data_port_00,0x60,0x0000,DAC data register for PIXI port 0
dac_data_port_01,0x61,0x0000,DAC data register for PIXI port 1
dac_data_port_02,0x62,0x0666,DAC data register for PIXI port 2
dac_data_port_03,0x63,0x0000,DAC data register for PIXI port 3
dac_data_port_04,0x64,0x0000,DAC data register for PIXI port 4
dac_data_port_05,0x65,0x0666,DAC data register for PIXI port 5
dac_data_port_06,0x66,0x0000,DAC data register for PIXI port 6
dac_data_port_07,0x67,0x0000,DAC data register for PIXI port 7
dac_data_port_08,0x68,0x0666,DAC data register for PIXI port 8
dac_data_port_09,0x69,0x0000,DAC data register for PIXI port 9
dac_data_port_10,0x6A,0x0000,DAC data register for PIXI port 10
dac_data_port_11,0x6B,0x0000,DAC data register for PIXI port 11
dac_data_port_12,0x6C,0x0000,DAC data register for PIXI port 12
dac_data_port_13,0x6D,0x0000,DAC data register for PIXI port 13
dac_data_port_14,0x6E,0x0000,DAC data register for PIXI port 14
dac_data_port_15,0x6F,0x0666,DAC data register for PIXI port 15
dac_data_port_16,0x70,0x0666,DAC data register for PIXI port 16
dac_data_port_17,0x71,0x0000,DAC data register for PIXI port 17
dac_data_port_18,0x72,0x0000,DAC data register for PIXI port 18
dac_data_port_19,0x73,0x0000,DAC data register for PIXI port 19
End file. Number of char received 3725
Type ENTER to proceed, DEL to abort
>> Storing configuration. Are you shure? (ENTER to accept, DEL to abort)
Connected 01:40:39 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```

confirmation screen: hit “ENTER” to continue, “DEL” to abort



```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help
MAX11300 register address = 60, register data = 0000 writing ...success!
MAX11300 register address = 61, register data = 0000 writing ...success!
MAX11300 register address = 62, register data = 0666 writing ...success!
MAX11300 register address = 63, register data = 0000 writing ...success!
MAX11300 register address = 64, register data = 0000 writing ...success!
MAX11300 register address = 65, register data = 0666 writing ...success!
MAX11300 register address = 66, register data = 0000 writing ...success!
MAX11300 register address = 67, register data = 0000 writing ...success!
MAX11300 register address = 68, register data = 0666 writing ...success!
MAX11300 register address = 69, register data = 0000 writing ...success!
MAX11300 register address = 6A, register data = 0000 writing ...success!
MAX11300 register address = 6B, register data = 0000 writing ...success!
MAX11300 register address = 6C, register data = 0000 writing ...success!
MAX11300 register address = 6D, register data = 0000 writing ...success!
MAX11300 register address = 6E, register data = 0000 writing ...success!
MAX11300 register address = 6F, register data = 0666 writing ...success!
MAX11300 register address = 70, register data = 0666 writing ...success!
MAX11300 register address = 71, register data = 0000 writing ...success!
MAX11300 register address = 72, register data = 0000 writing ...success!
MAX11300 register address = 73, register data = 0000 writing ...success!

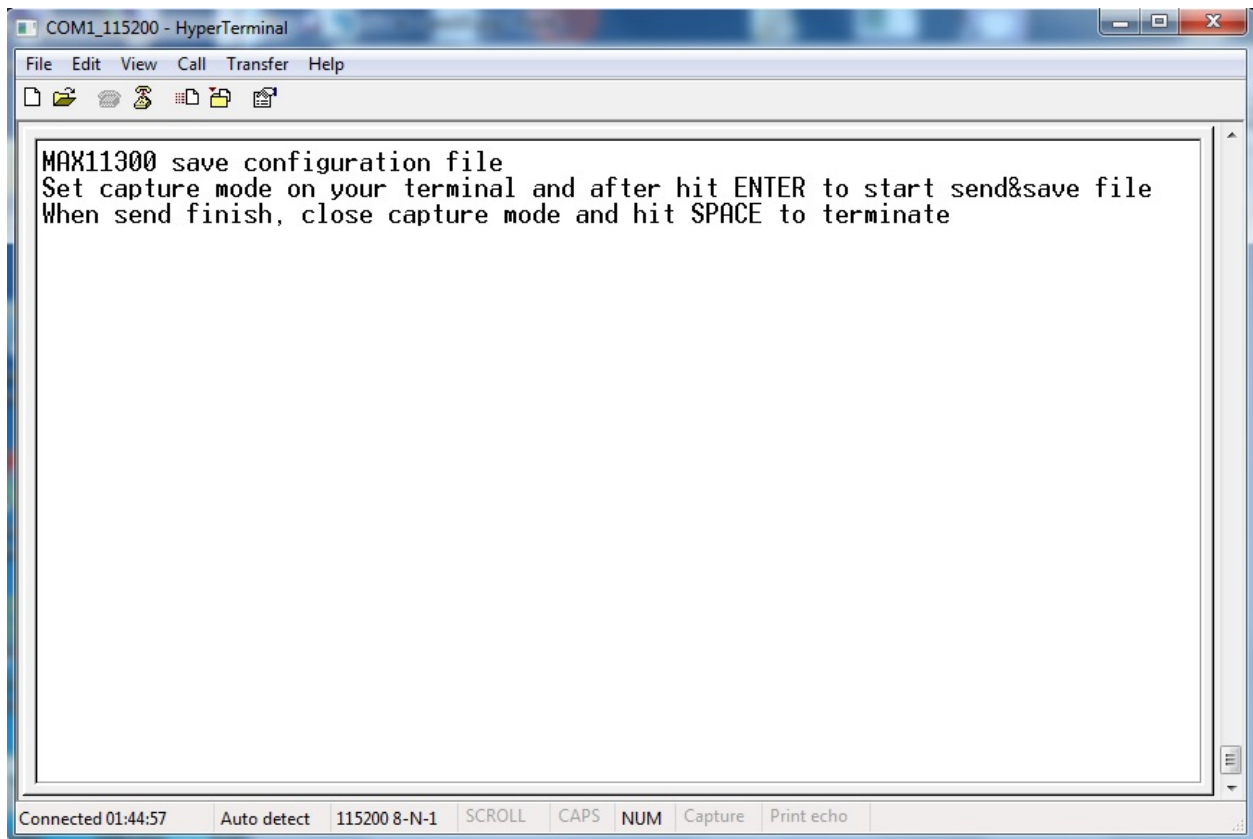
Configuration stored into MAX11300 device
Type any key to return to main menu
_

Connected 01:44:11 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```

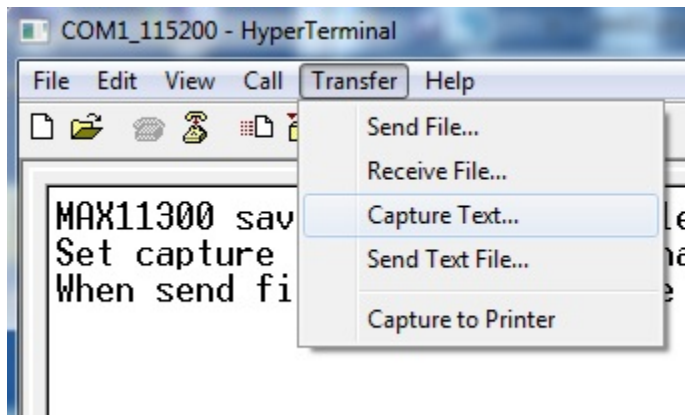
MAX11300 succesfull programmed!!

Save Configuration instructions

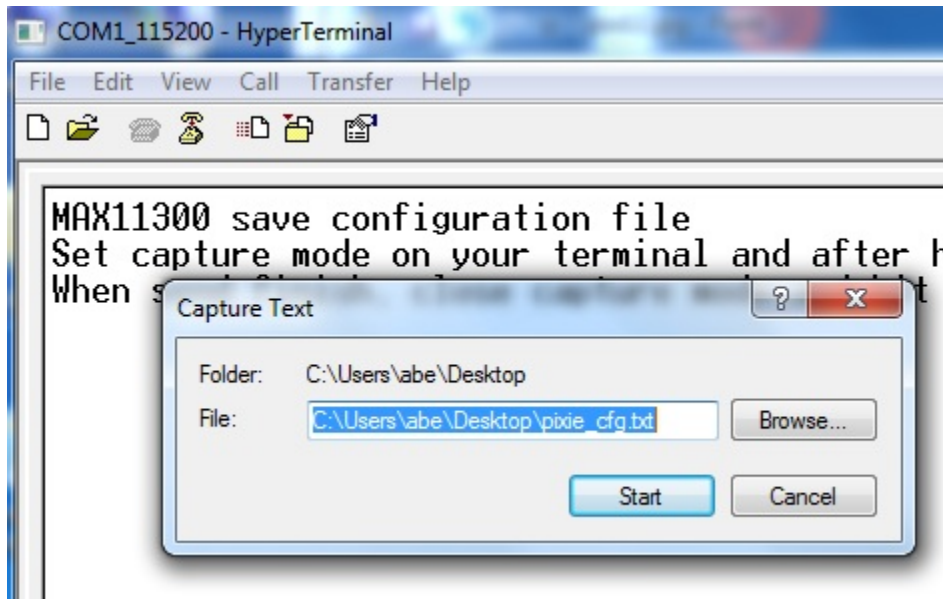
At prompt of MAX11300 main menu, type “6”



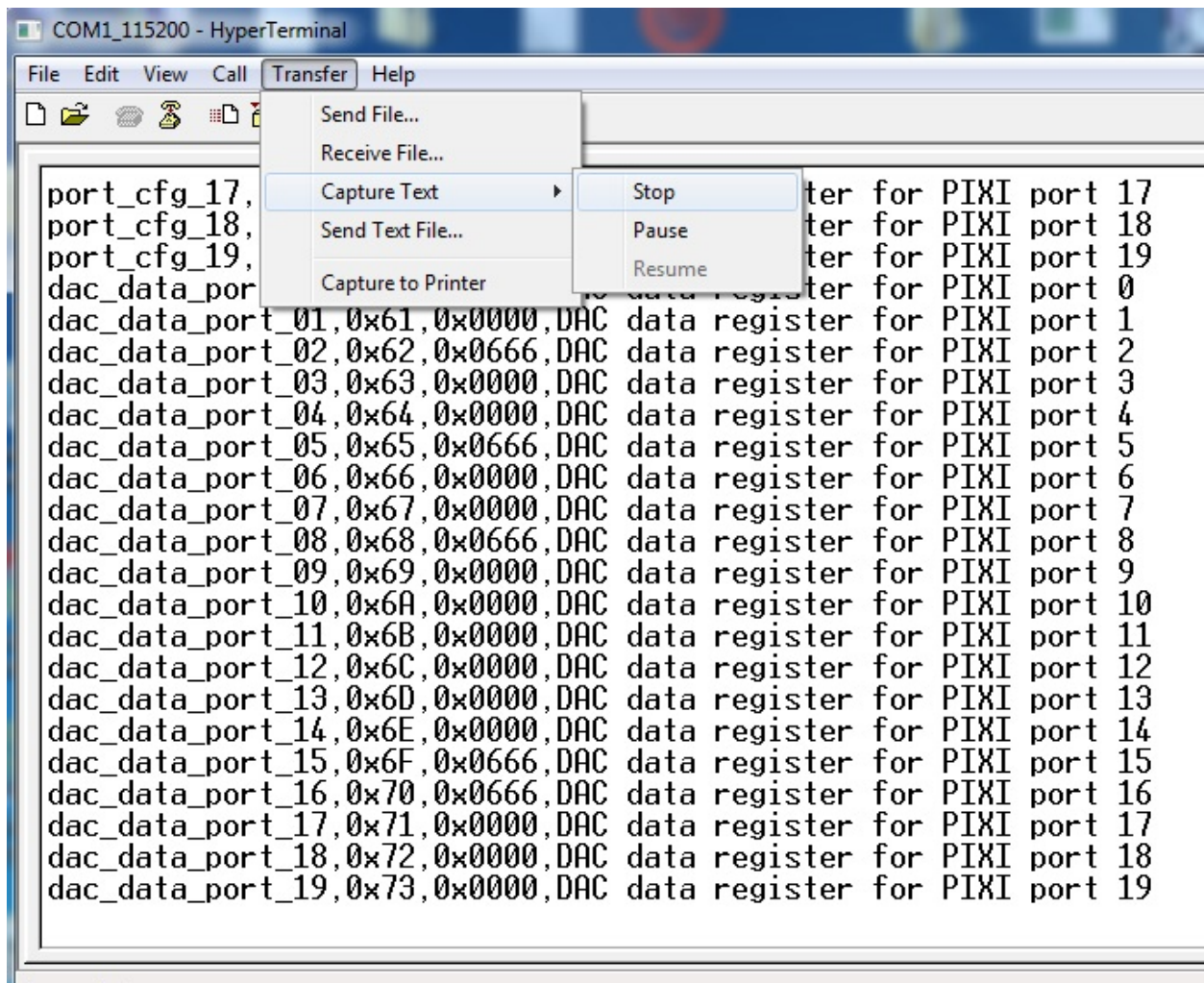
Take care at simple instruction on your terminal window (figure above) and remember it



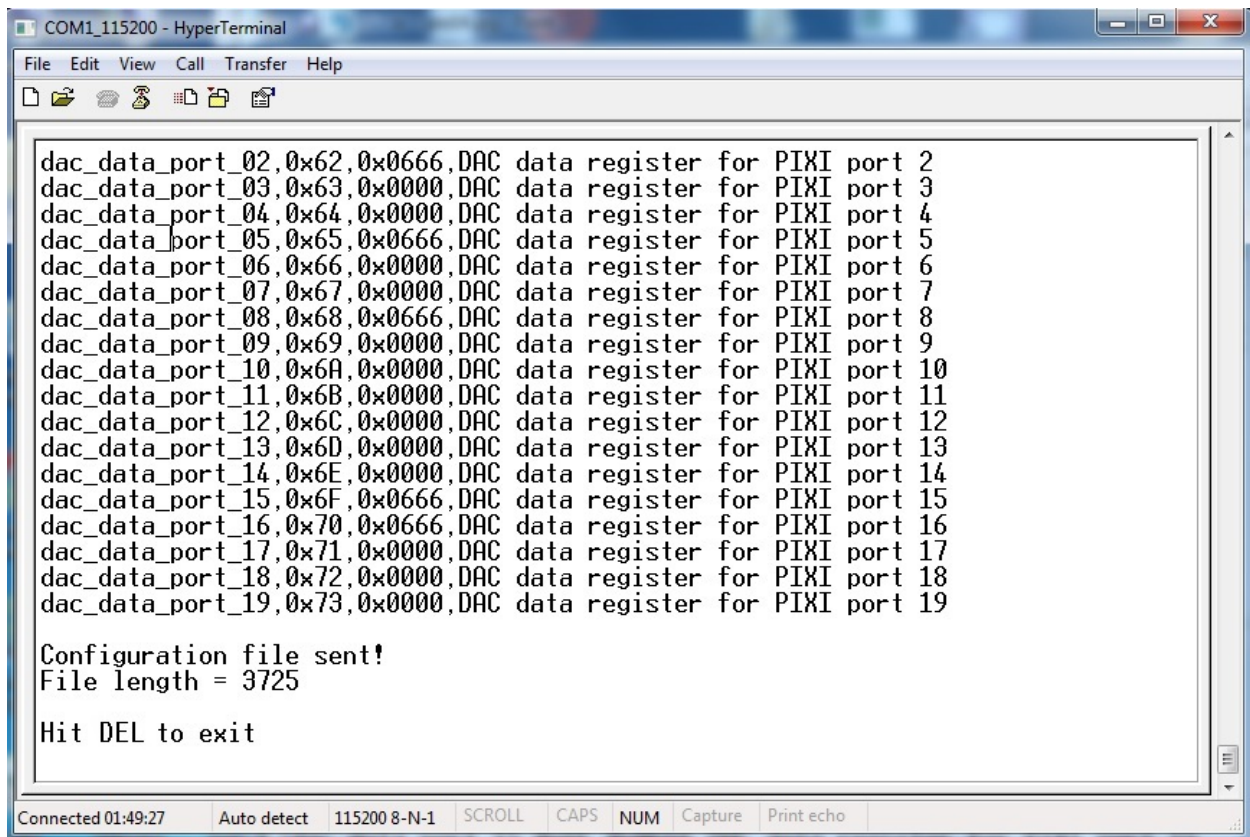
click on “**Transfer**” menu, select “**Capture Text**” and click over



Type path and name of file that you will store (also using Browse button), then click “Start” and after hit ENTER



When transfer ends, click on “Transfer” menu, select “Capture Text” → “Stop” and click over



```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help

dac_data_port_02,0x62,0x0666,DAC data register for PIXI port 2
dac_data_port_03,0x63,0x0000,DAC data register for PIXI port 3
dac_data_port_04,0x64,0x0000,DAC data register for PIXI port 4
dac_data_port_05,0x65,0x0666,DAC data register for PIXI port 5
dac_data_port_06,0x66,0x0000,DAC data register for PIXI port 6
dac_data_port_07,0x67,0x0000,DAC data register for PIXI port 7
dac_data_port_08,0x68,0x0666,DAC data register for PIXI port 8
dac_data_port_09,0x69,0x0000,DAC data register for PIXI port 9
dac_data_port_10,0x6A,0x0000,DAC data register for PIXI port 10
dac_data_port_11,0x6B,0x0000,DAC data register for PIXI port 11
dac_data_port_12,0x6C,0x0000,DAC data register for PIXI port 12
dac_data_port_13,0x6D,0x0000,DAC data register for PIXI port 13
dac_data_port_14,0x6E,0x0000,DAC data register for PIXI port 14
dac_data_port_15,0x6F,0x0666,DAC data register for PIXI port 15
dac_data_port_16,0x70,0x0666,DAC data register for PIXI port 16
dac_data_port_17,0x71,0x0000,DAC data register for PIXI port 17
dac_data_port_18,0x72,0x0000,DAC data register for PIXI port 18
dac_data_port_19,0x73,0x0000,DAC data register for PIXI port 19

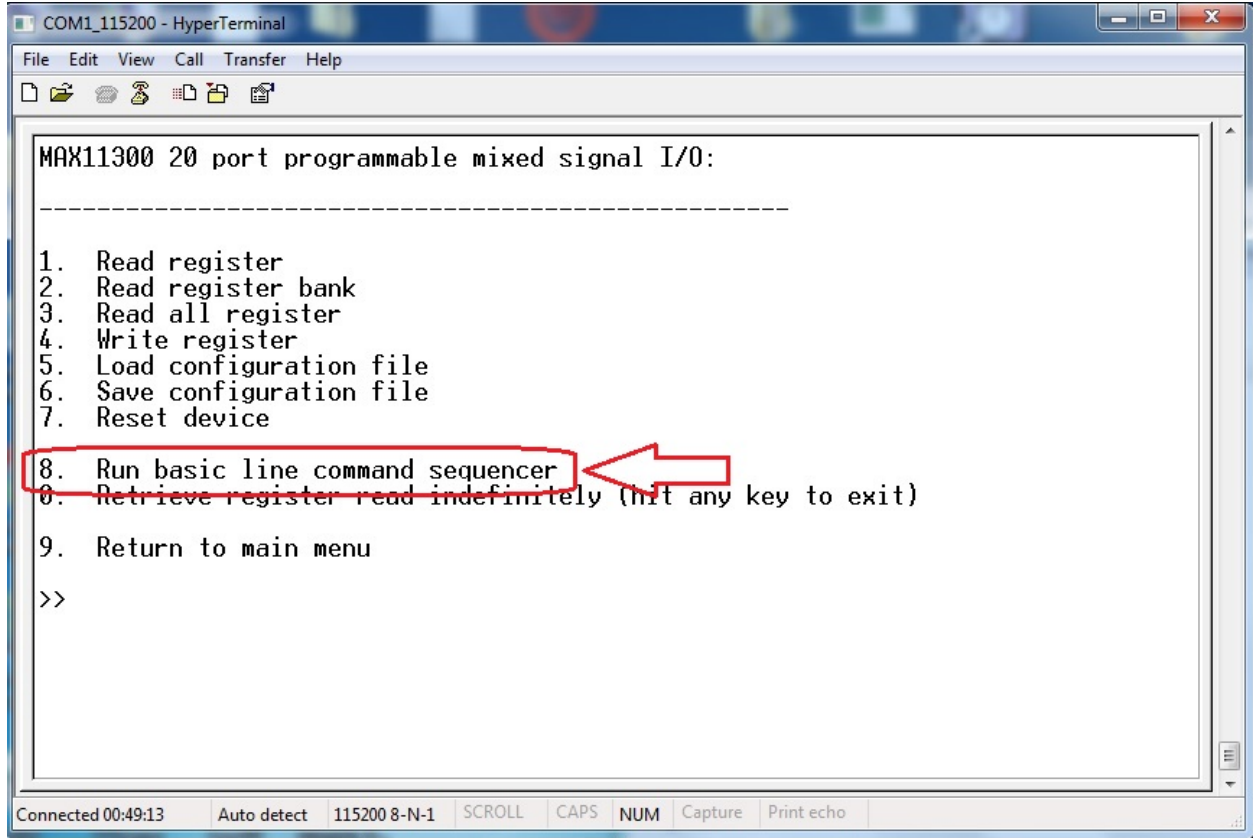
Configuration file sent!
File length = 3725

Hit DEL to exit

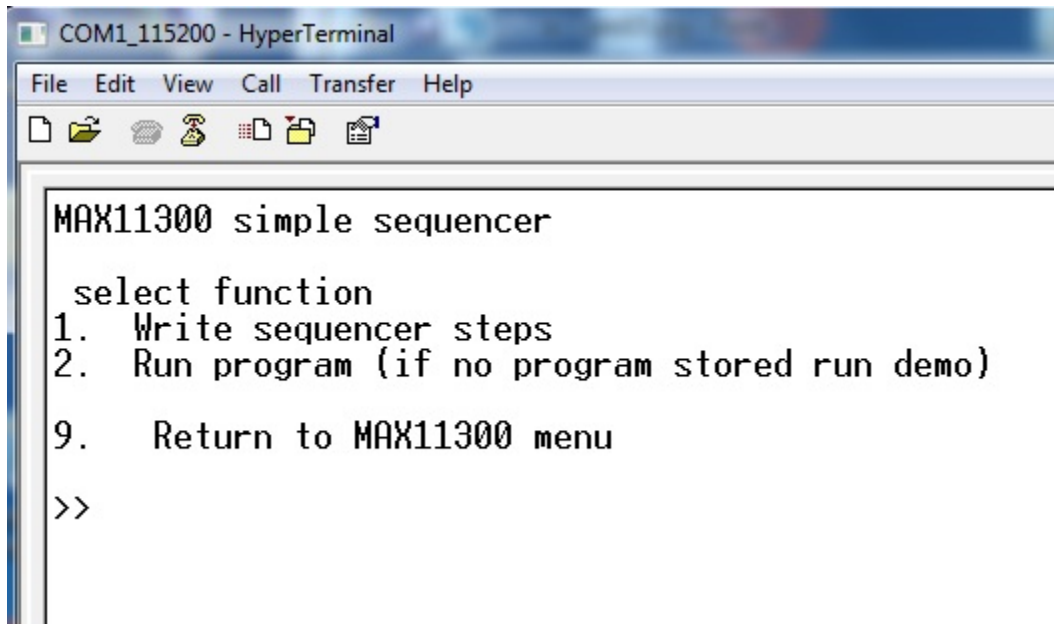
Connected 01:49:27 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```

Configuration file successfully saved. You can use this file with “Load Configuration” function to backup/restore your custom configuration.

Simple sequencer



In MAX11300 main menu you can find the function “**Run basic line command sequencer**”. This is a simple *step by step* sequencer that performs READ and WRITE of Pixi register. It can be used to dynamically check basic input/output sequences and see on terminal the result of each step. It is possible to activate one or more *loop* to retry up to 65535 times any step group defined during line command insertion. By typing “8” in main menu screen, you go to sequencer sub-menu.

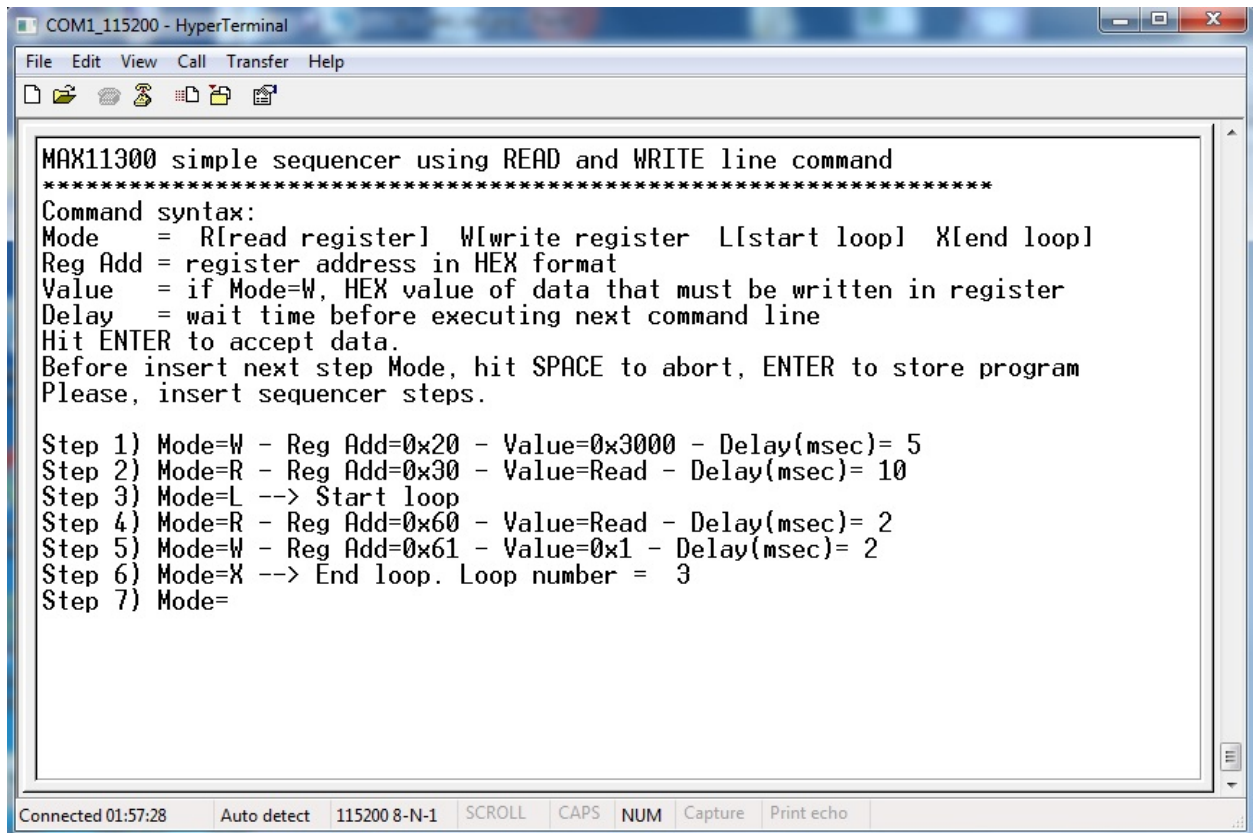


In the Sequencer sub-menu there are two functions: | **Write sequencer steps** (*enter by typing 1*) | **Run program** (*enter by typing 2*)

The first one is a wizard for entering sequencer steps. The second is the “launch” function that runs the program. If no program is written by user (using Write sequencer steps) the firmware will run the demo program. This demo program will perform a basic setting of port0 and generate port0 output as in figure below

Writing sequences

When you enter in **Write sequencer steps** sub-menu, you can see the screen below:



```

COM1_115200 - HyperTerminal
File Edit View Call Transfer Help

MAX11300 simple sequencer using READ and WRITE line command
*****
Command syntax:
Mode    = R[read register] W[write register] L[start loop] X[end loop]
Reg Add = register address in HEX format
Value   = if Mode=W, HEX value of data that must be written in register
Delay   = wait time before executing next command line
Hit ENTER to accept data.
Before insert next step Mode, hit SPACE to abort, ENTER to store program
Please, insert sequencer steps.

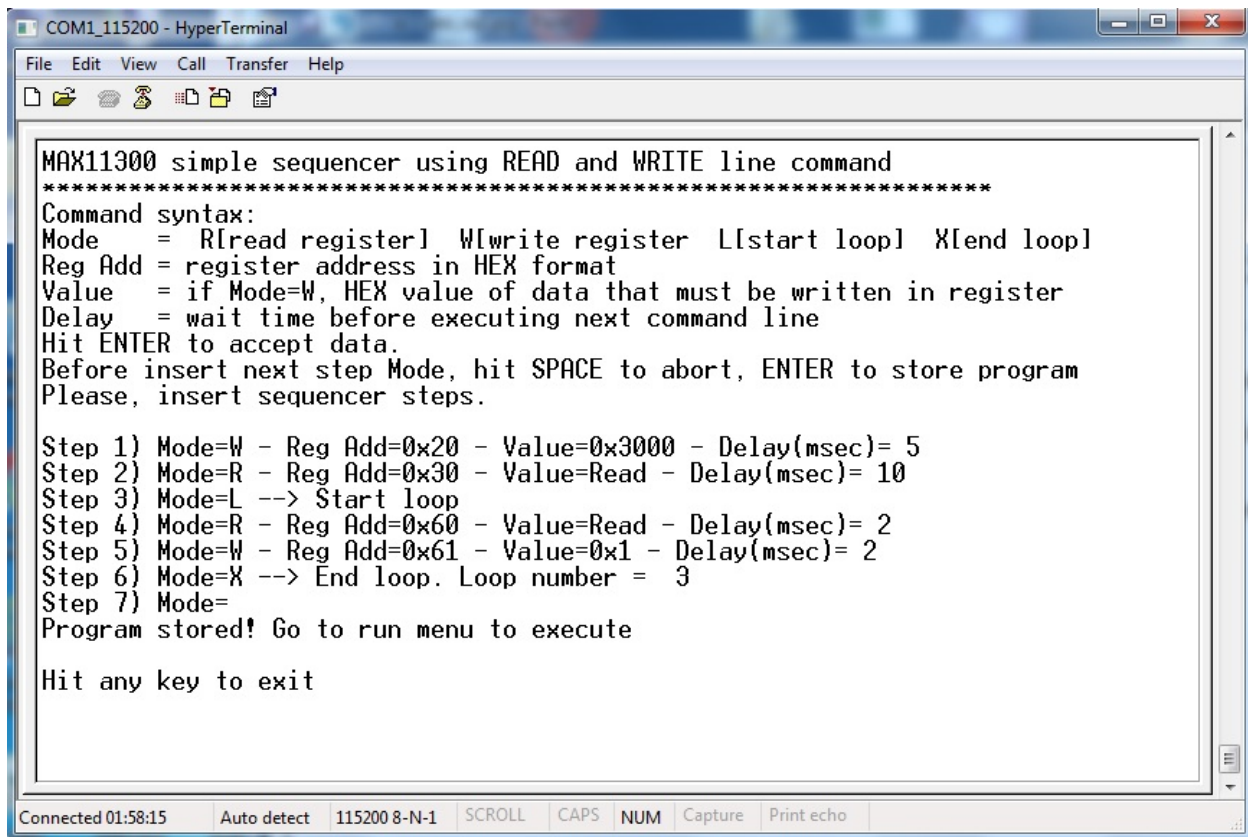
Step 1) Mode=W - Reg Add=0x20 - Value=0x3000 - Delay(msec)= 5
Step 2) Mode=R - Reg Add=0x30 - Value=Read - Delay(msec)= 10
Step 3) Mode=L --> Start loop
Step 4) Mode=R - Reg Add=0x60 - Value=Read - Delay(msec)= 2
Step 5) Mode=W - Reg Add=0x61 - Value=0x1 - Delay(msec)= 2
Step 6) Mode=X --> End loop. Loop number = 3
Step 7) Mode=

Connected 01:57:28  Auto detect  115200 8-N-1  SCROLL  CAPS  NUM  Capture  Print echo

```

Read simple help lines to understand how to write command line. In the figure above you can see one program example. Data will be accepted when you hit ENTER. By typing ENTER when the sequencer editor will wait for MODE, the program will be ended and stored inside program memory.

Note: maximum sequencer steps is 200. If you reached this number, the editor will automatically ends and stores the program.



```

COM1_115200 - HyperTerminal
File Edit View Call Transfer Help

MAX11300 simple sequencer using READ and WRITE line command
*****
Command syntax:
Mode    = R[read register] W[write register] L[start loop] X[end loop]
Reg Add = register address in HEX format
Value   = if Mode=W, HEX value of data that must be written in register
Delay   = wait time before executing next command line
Hit ENTER to accept data.
Before insert next step Mode, hit SPACE to abort, ENTER to store program
Please, insert sequencer steps.

Step 1) Mode=W - Reg Add=0x20 - Value=0x3000 - Delay(msec)= 5
Step 2) Mode=R - Reg Add=0x30 - Value=Read - Delay(msec)= 10
Step 3) Mode=L --> Start loop
Step 4) Mode=R - Reg Add=0x60 - Value=Read - Delay(msec)= 2
Step 5) Mode=W - Reg Add=0x61 - Value=0x1 - Delay(msec)= 2
Step 6) Mode=X --> End loop. Loop number = 3
Step 7) Mode=
Program stored! Go to run menu to execute

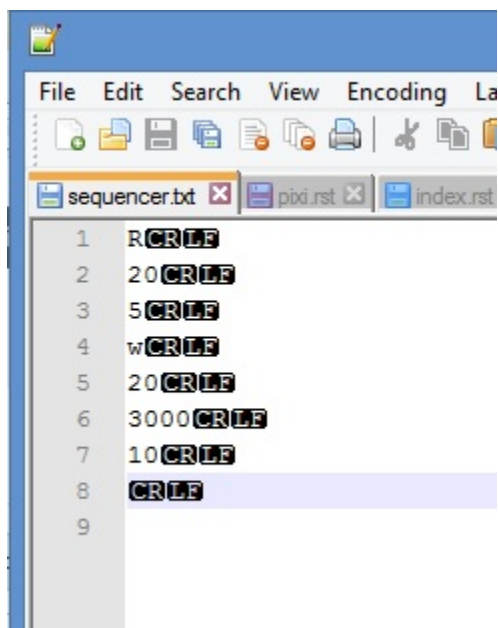
Hit any key to exit

Connected 01:58:15 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo

```

Hit ENTER when editor will wait for MODE. The program ends and will be stored.

Note: It is possible to create a file containing all entries for sequencer steps and send it through “terminal send text file”. Here below a file example. This file is stored in the Codewarrior project folder documentation\examples



```

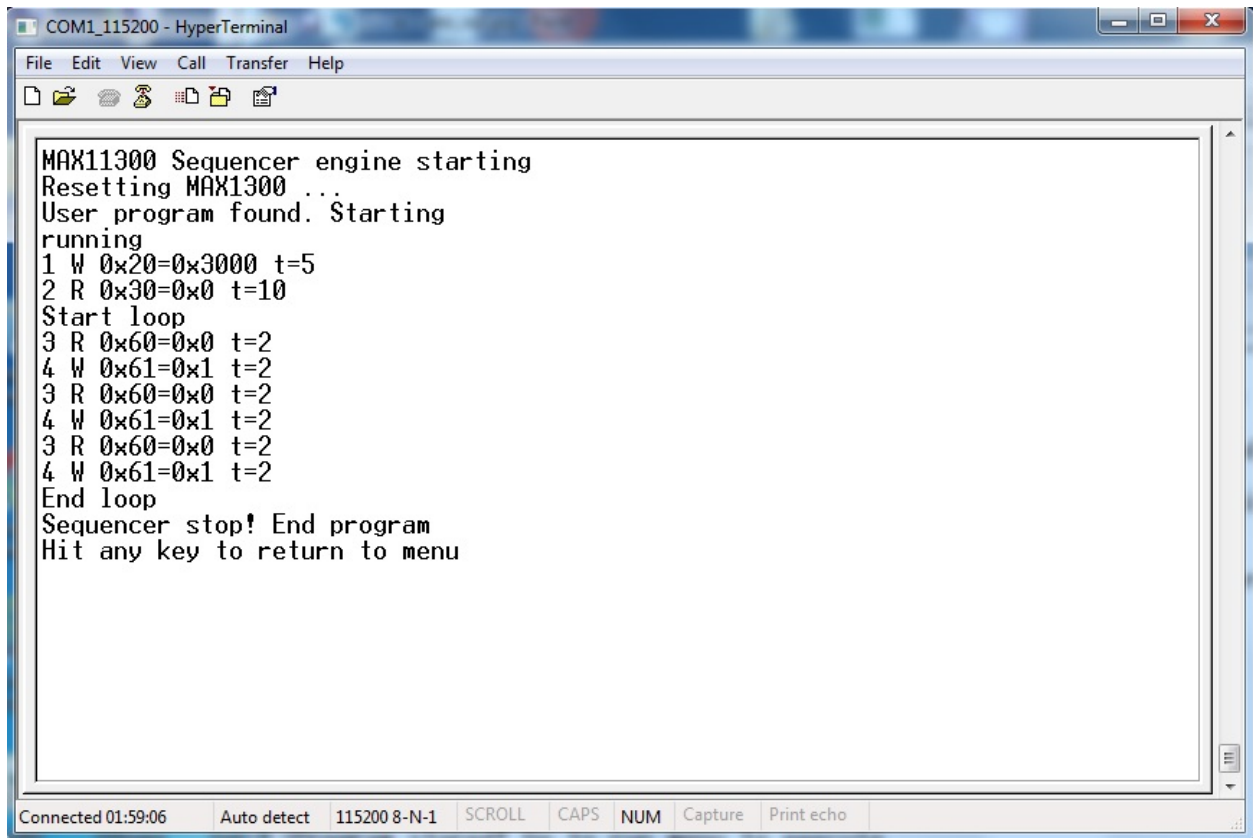
File Edit Search View Encoding La
sequencer.txt pxi.rst index.rst

1 RCR LF
2 20CR LF
3 5CR LF
4 wCR LF
5 20CR LF
6 3000CR LF
7 10CR LF
8 CR LF
9

```

Executing sequences

Typing **2** from sequencer sub-menu, the system will perform a MAX11300 software reset and then starts to execute steps. Program will end automatically when all steps has been executed.

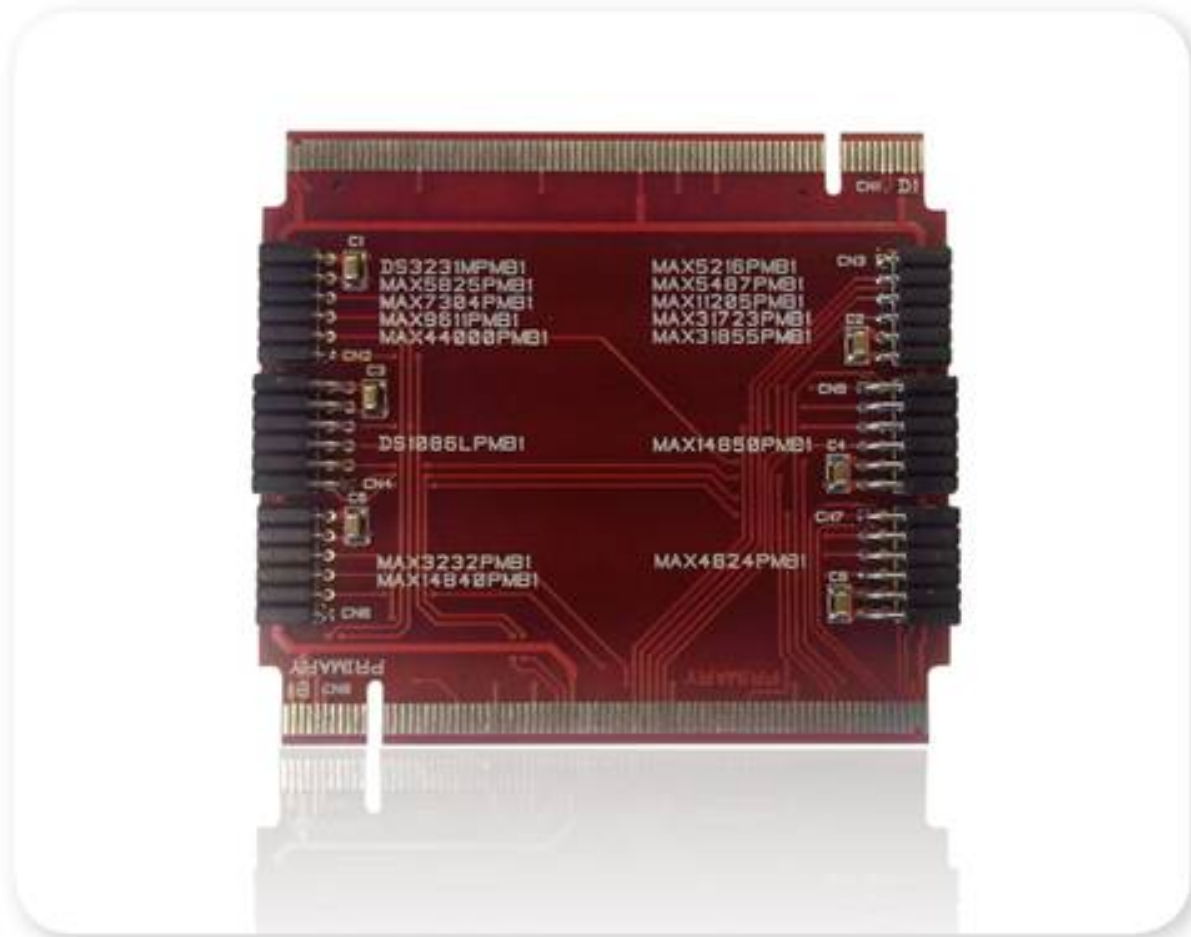


```
COM1_115200 - HyperTerminal
File Edit View Call Transfer Help
MAX11300 Sequencer engine starting
Resetting MAX1300 ...
User program found. Starting
running
1 W 0x20=0x3000 t=5
2 R 0x30=0x0 t=10
Start loop
3 R 0x60=0x0 t=2
4 W 0x61=0x1 t=2
3 R 0x60=0x0 t=2
4 W 0x61=0x1 t=2
3 R 0x60=0x0 t=2
4 W 0x61=0x1 t=2
End loop
Sequencer stop! End program
Hit any key to return to menu
Connected 01:59:06 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo
```

Quick start guide

Hardware requirements

- Tower system for Kinetis K70F120M (with TWR-SER expansion)
- Mini USB type-B cable
- Silica BrooklynBoard
- PC with at least one RS232 serial port and terminal software (two serial port for MAX3232 emulation)
- RS232 DB9 serial cable (modem type)
- Maxim Analog Essential Collection

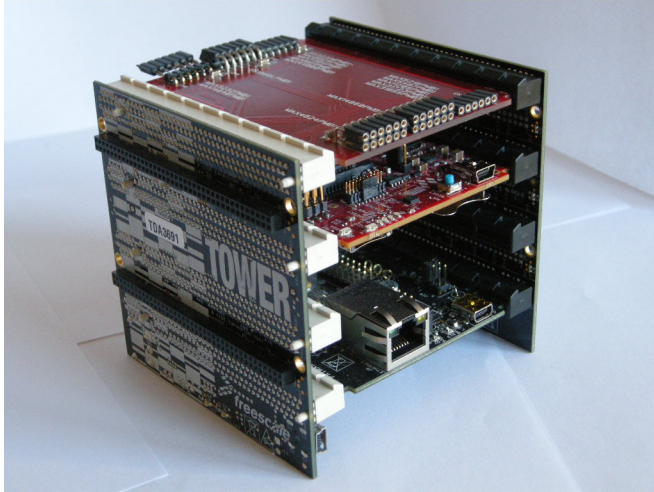


Software requirements

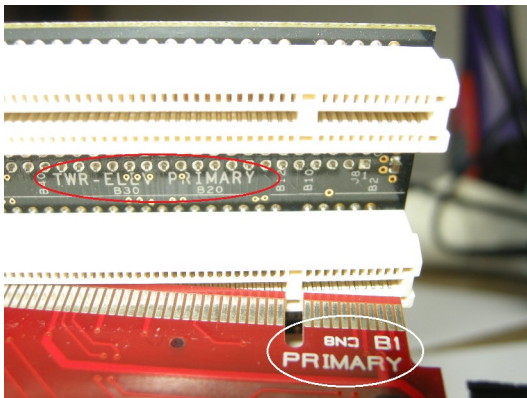
- CodeWarrior MCU v10.3 Special Edition ([download here](#)).
- Brooklyn Board application firmware for TWR-K70F120M system ([download here](#))

Hardware setup

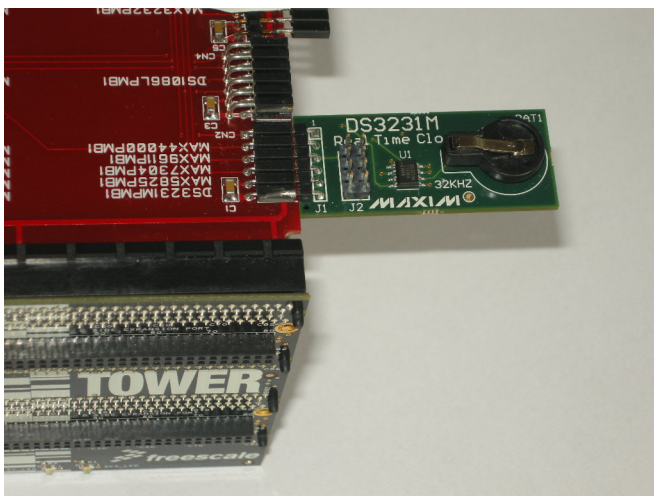
- Assemble tower system TWR-K70F120M and Brooklyn Board as in figure below.



Don't care slot position, but be careful to connect Primary and Secondary connector properly. Take care at reference signed near PCI board connectors.



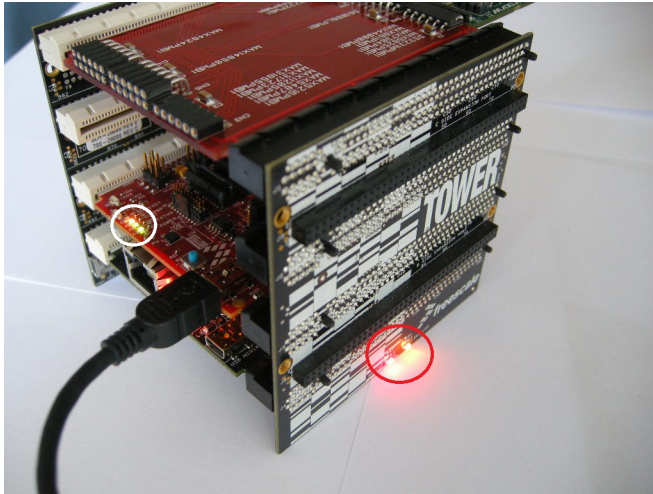
- Plug a Pmod Device (i.e. DS3231M Real Time Clock) inside properly connector.



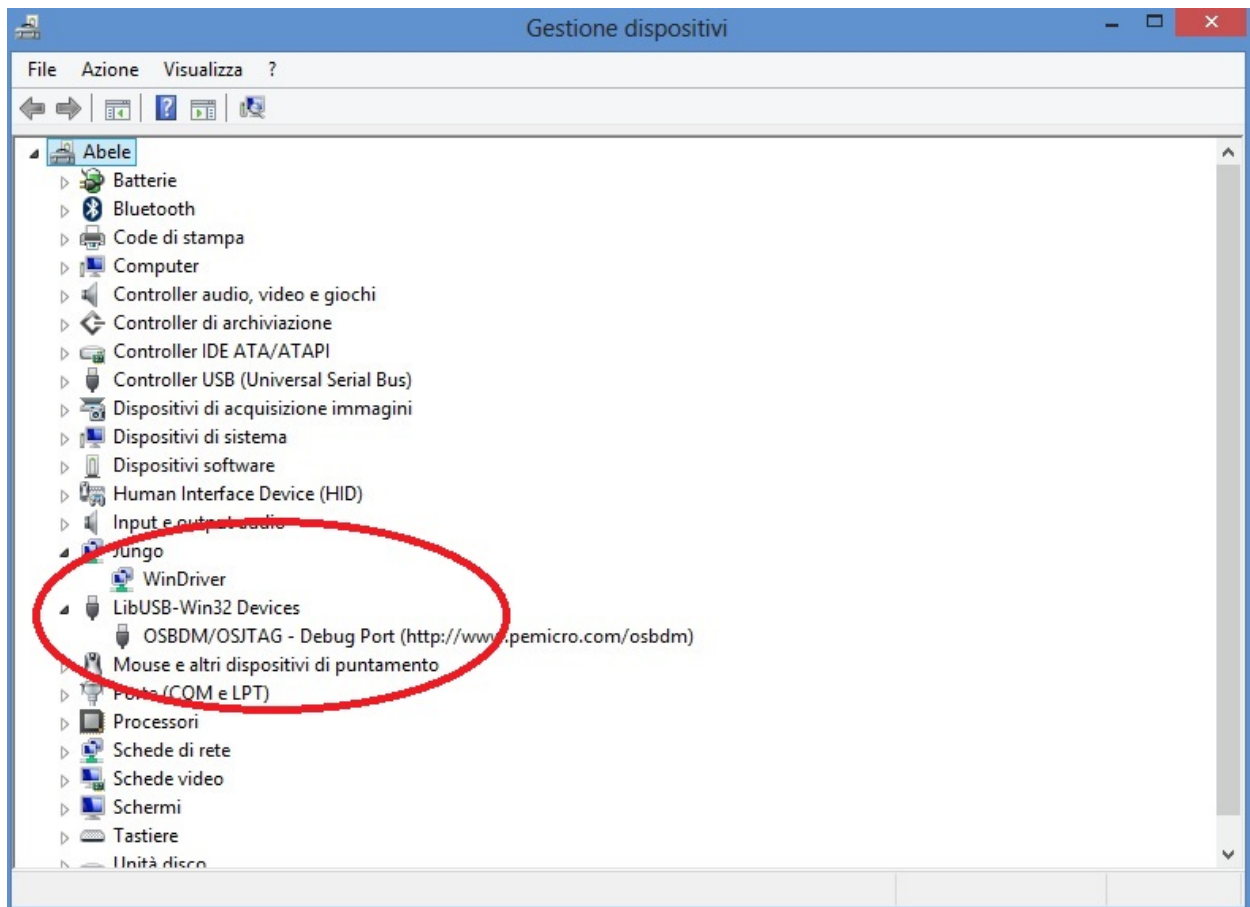
→ Be careful to see device reference next to connector. Each connector is designed for one or more devices and

will only accept dedicated modules.

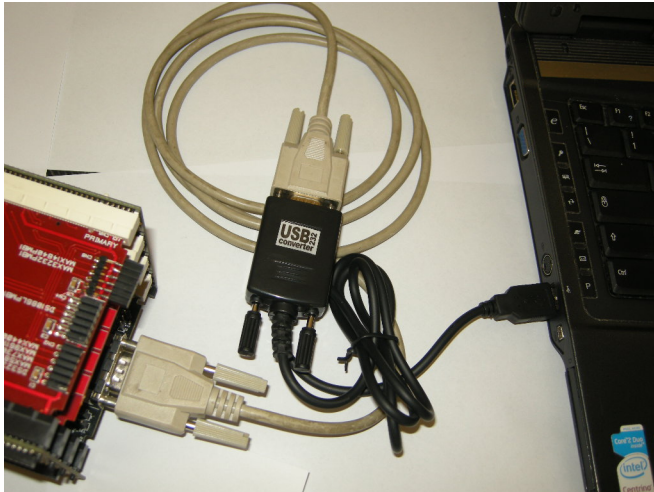
- Plug Mini USB type-B cable into Cpu Board plug and connect to PC with Codewarrior. TWR power led will on



- If you see device tab, you will find OSBDM/OSJTAG debug port



- plug the standard serial DB9 cable into serial connector on Tower System
- connect serial cable to terminal PC (equipped with terminal SW)



- On your terminal PC setup COMx parameter:

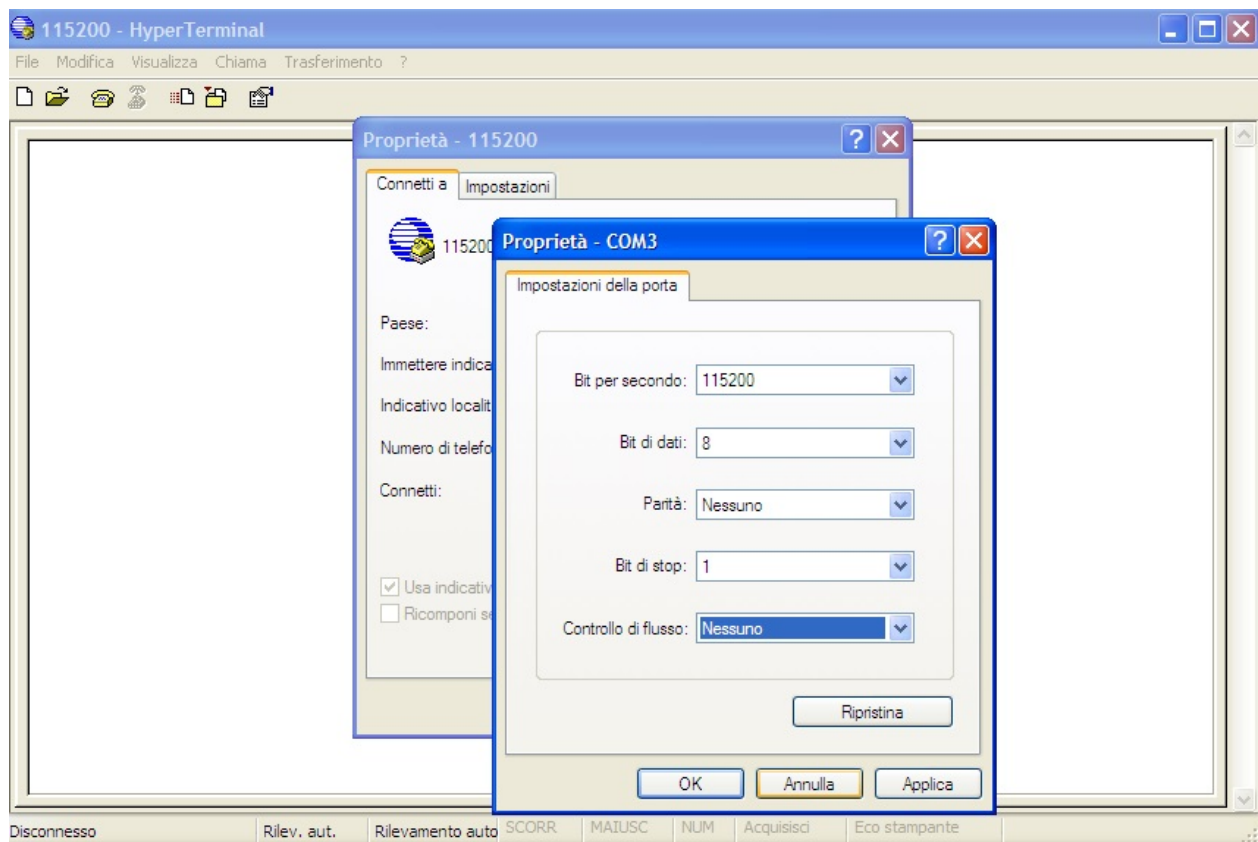
speed = 115200 baud

data with = 8

parity = none

stop bit = 1

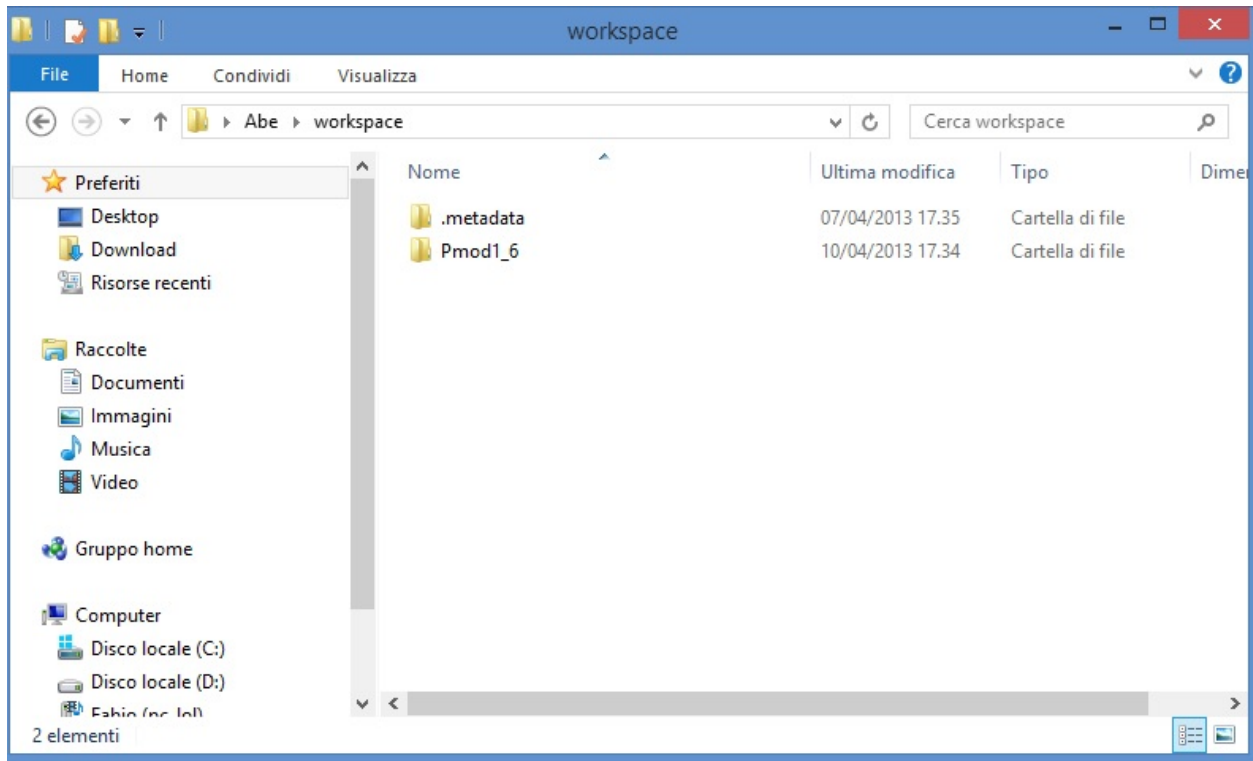
flow control = none



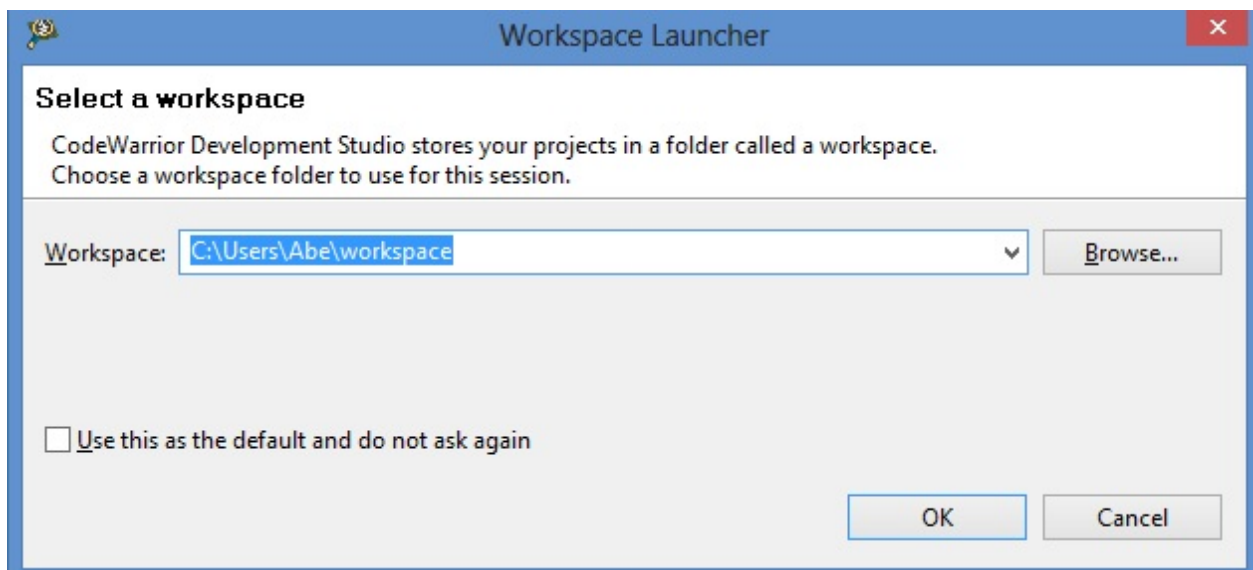
Now you are ready for install FW project.

Brooklyn Board FW setup

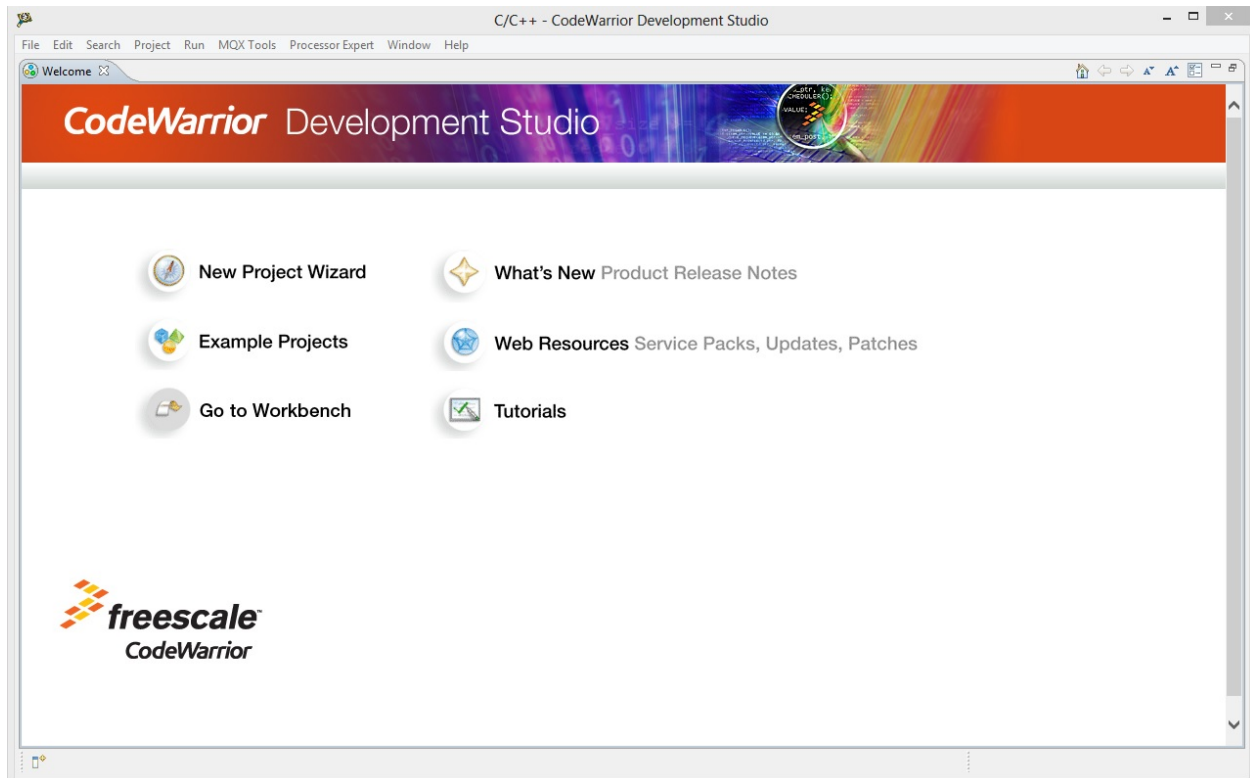
- Extract from Pmod1_6.zip the folder **Pmod1_6** and place it into Codewarrior default workspace



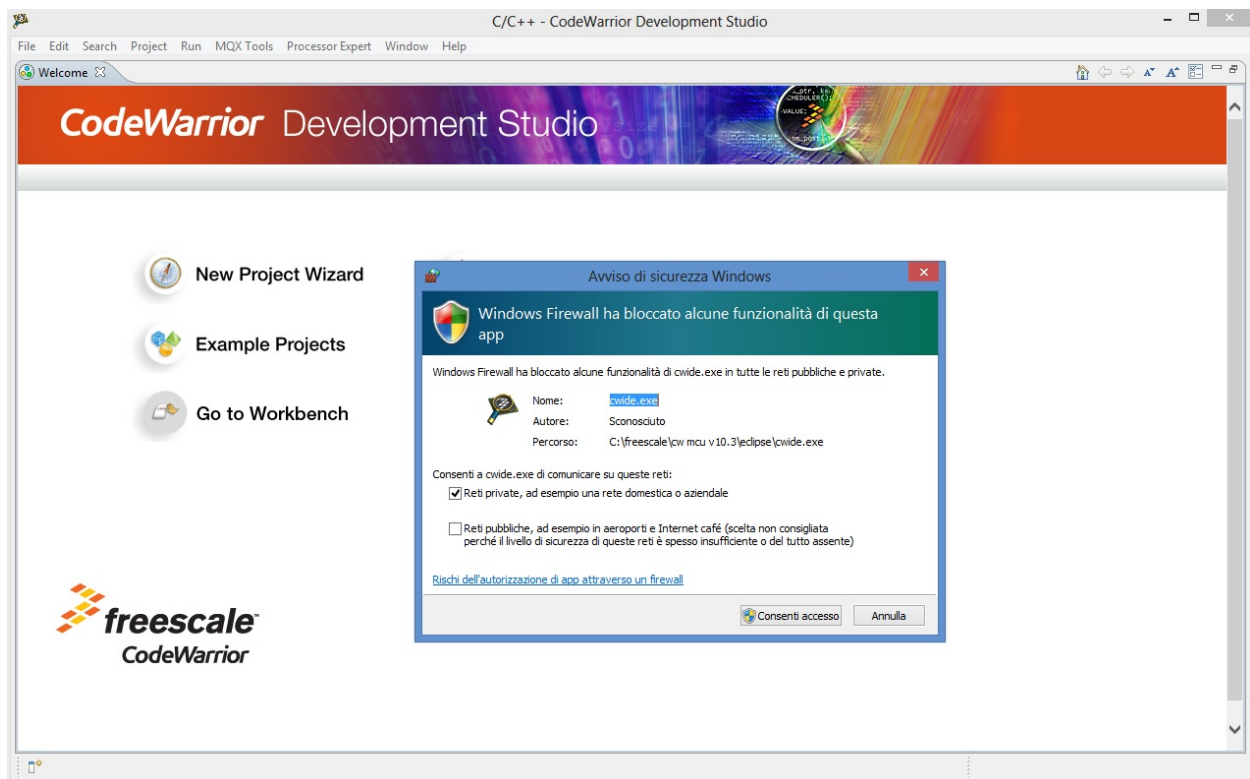
- start Codewarrior. Select “workspace” as in default window and click OK.



Now we could see the welcome window of Codewarrior Development Suite

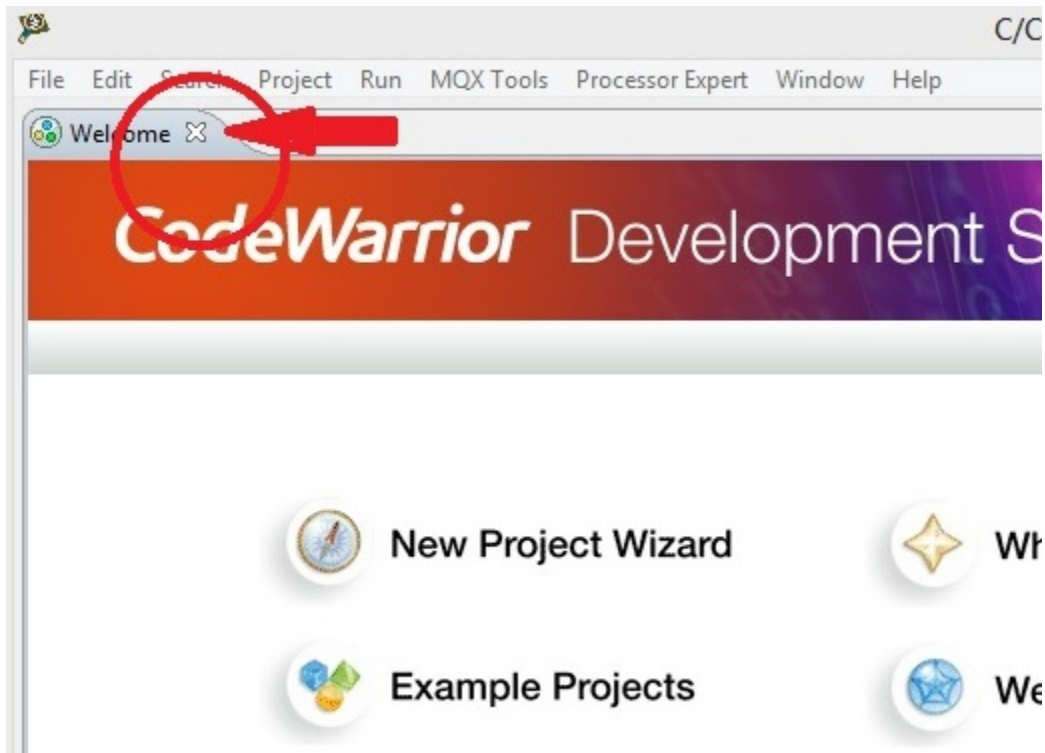


maybe will open firewall popup as below

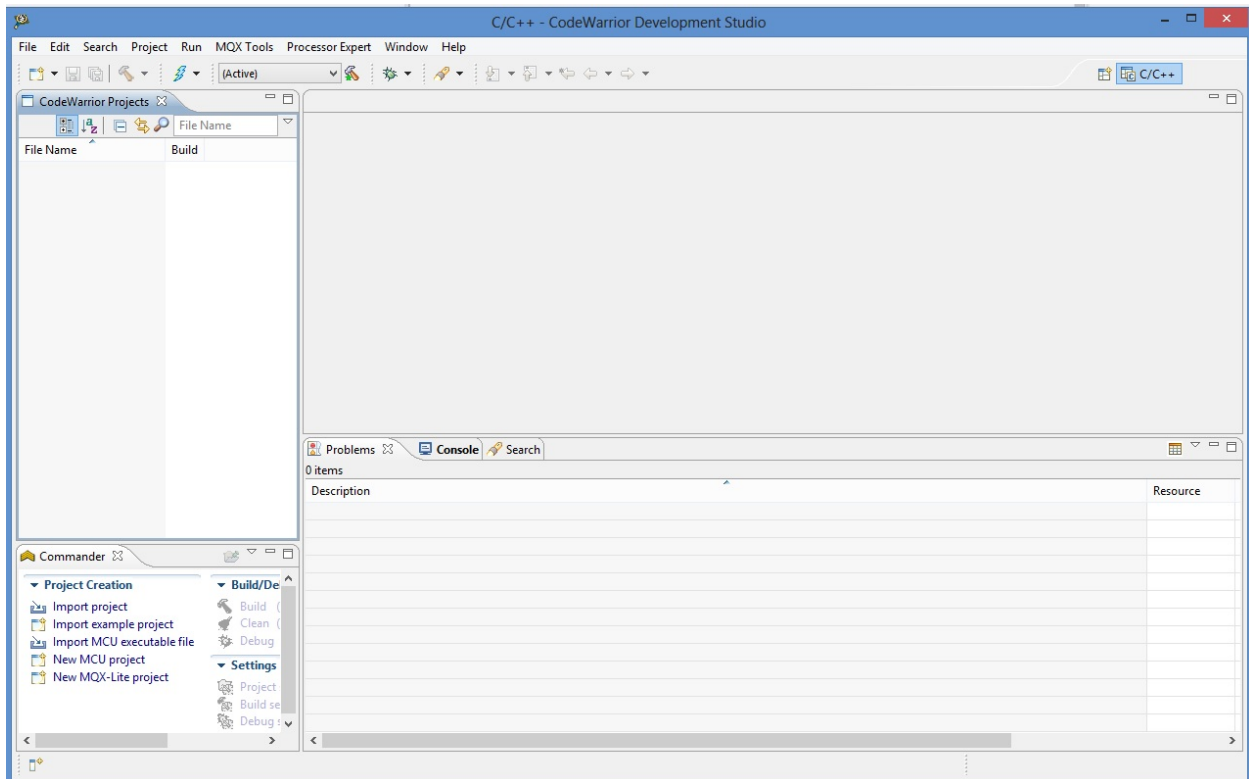


if yes, left-click on **enable access** and proceed

- close the welcome window by clicking 'X' in the Welcome tab



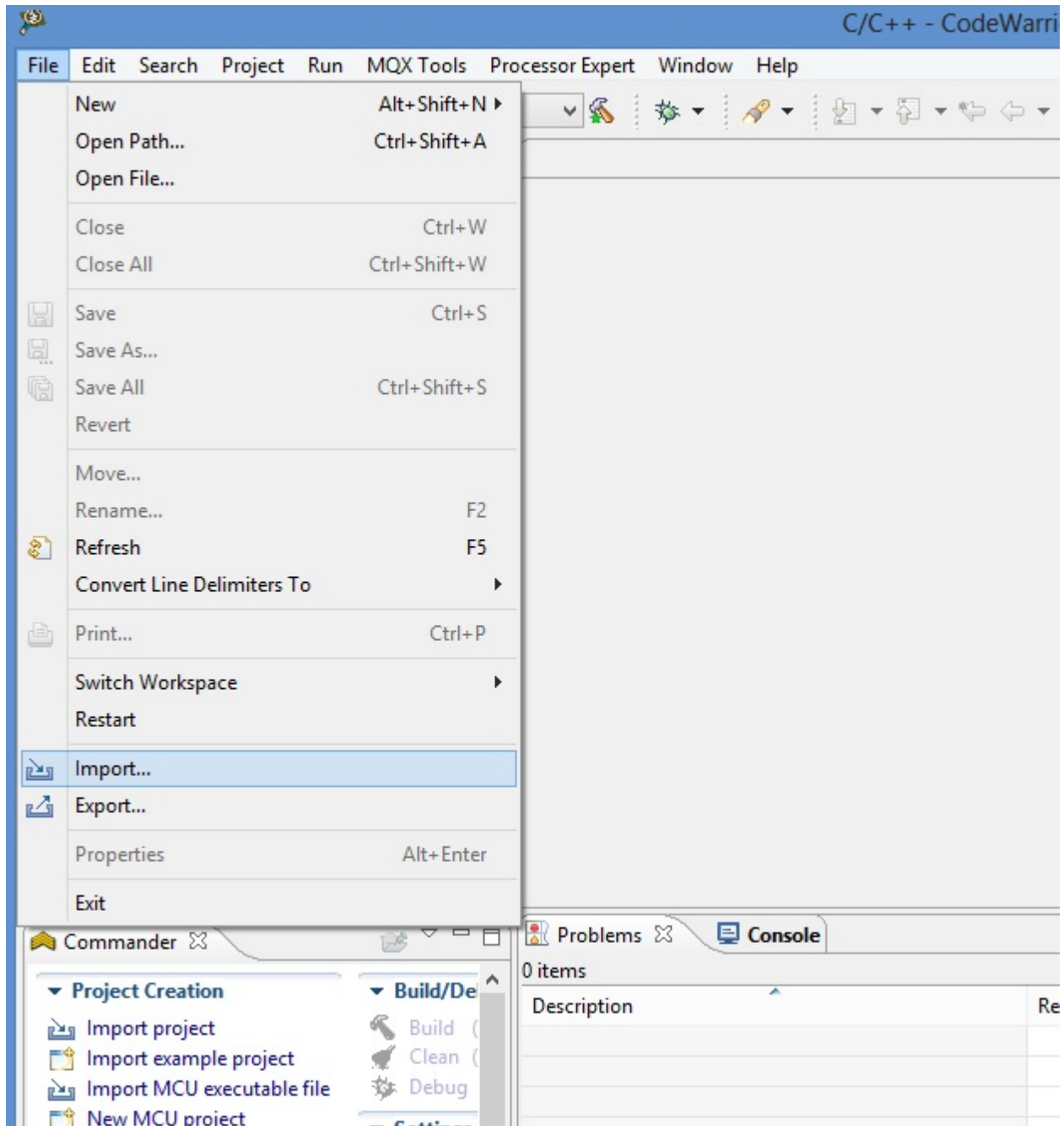
Now we can see the Codewarrior main window



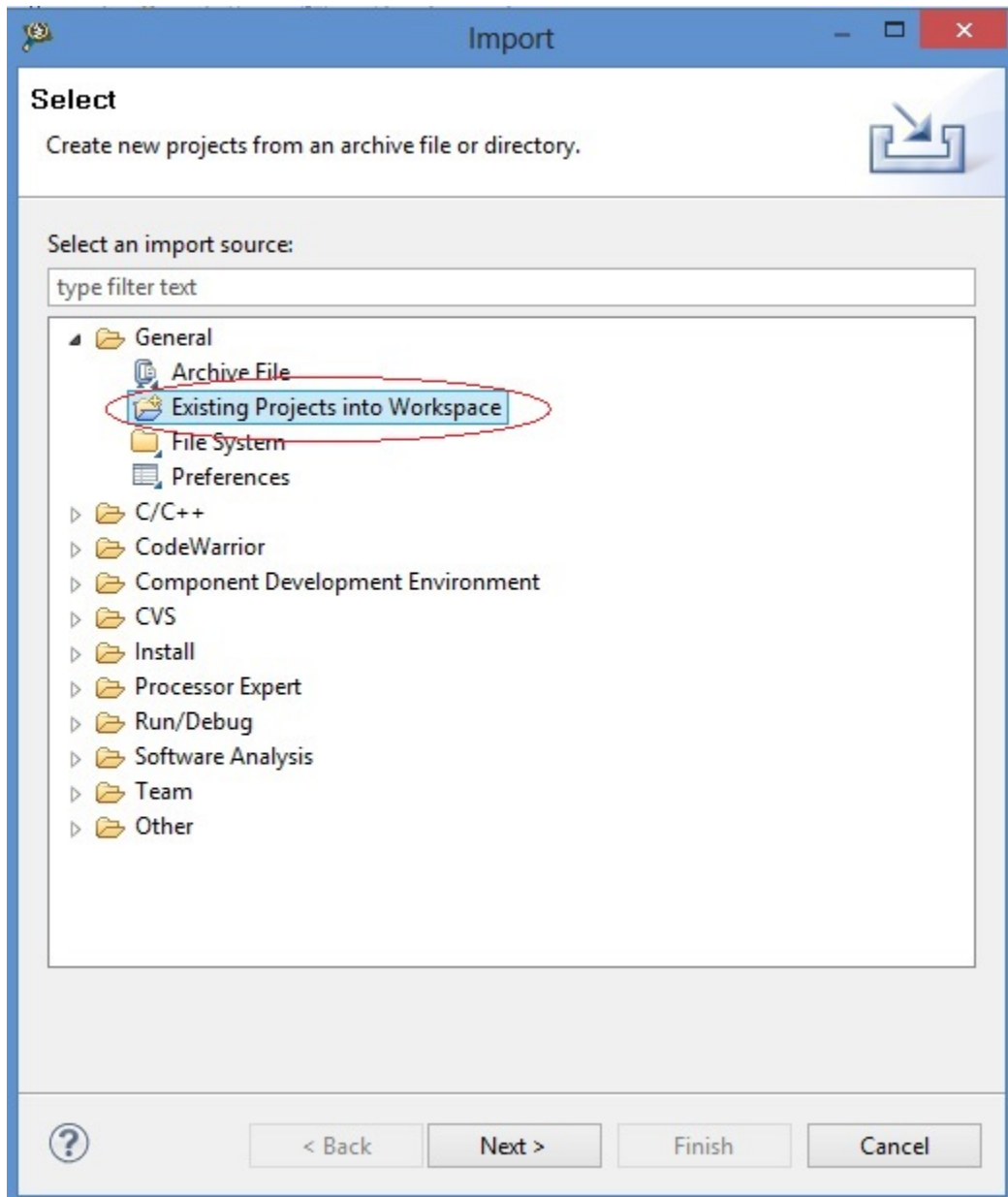
Codewarrior is ready to import the project

Pmod FIRMWARE IMPORT

- Select File → Import and click

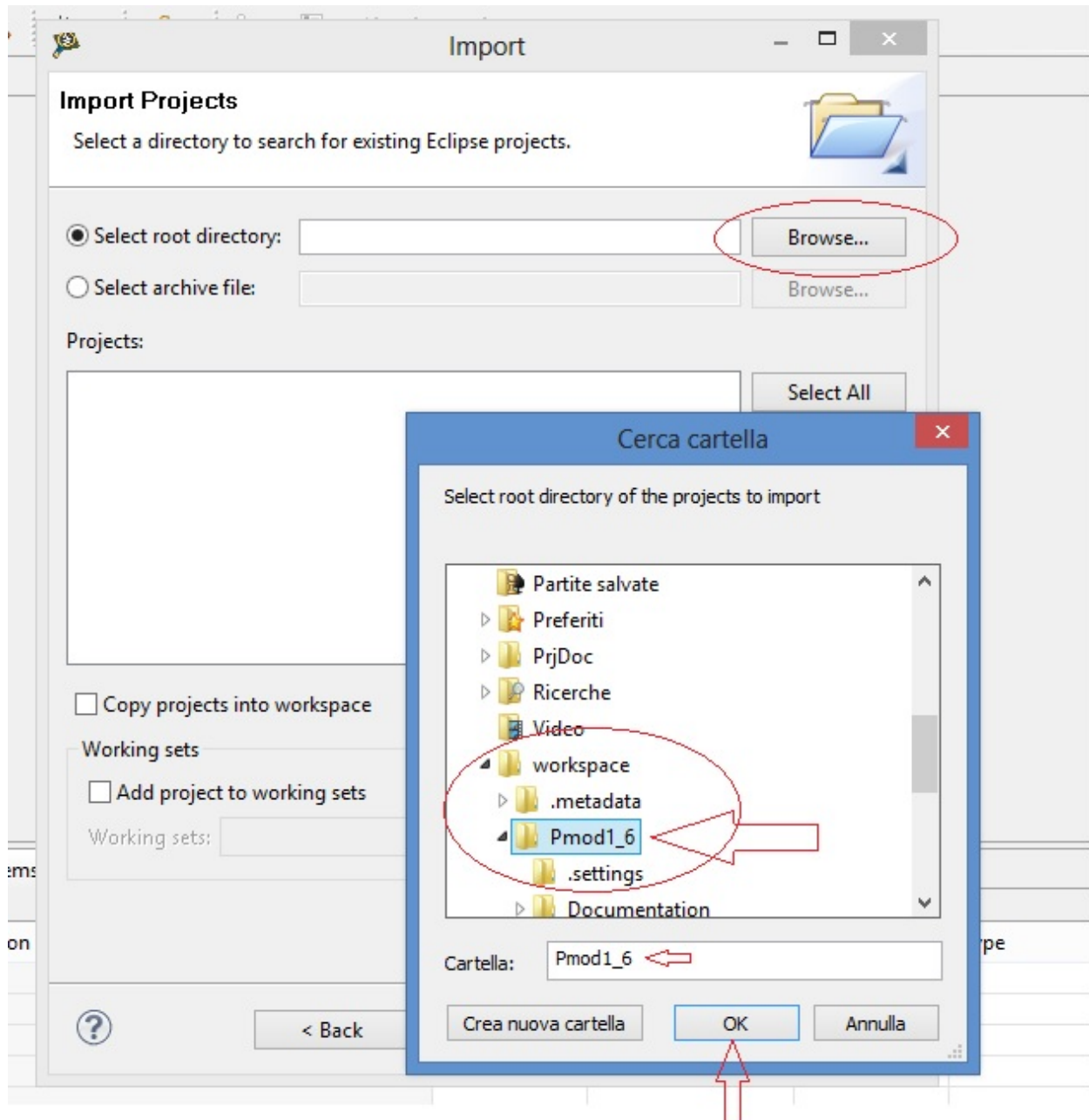


- in the next tab select “Existing Project into Workspace” and click “NEXT”

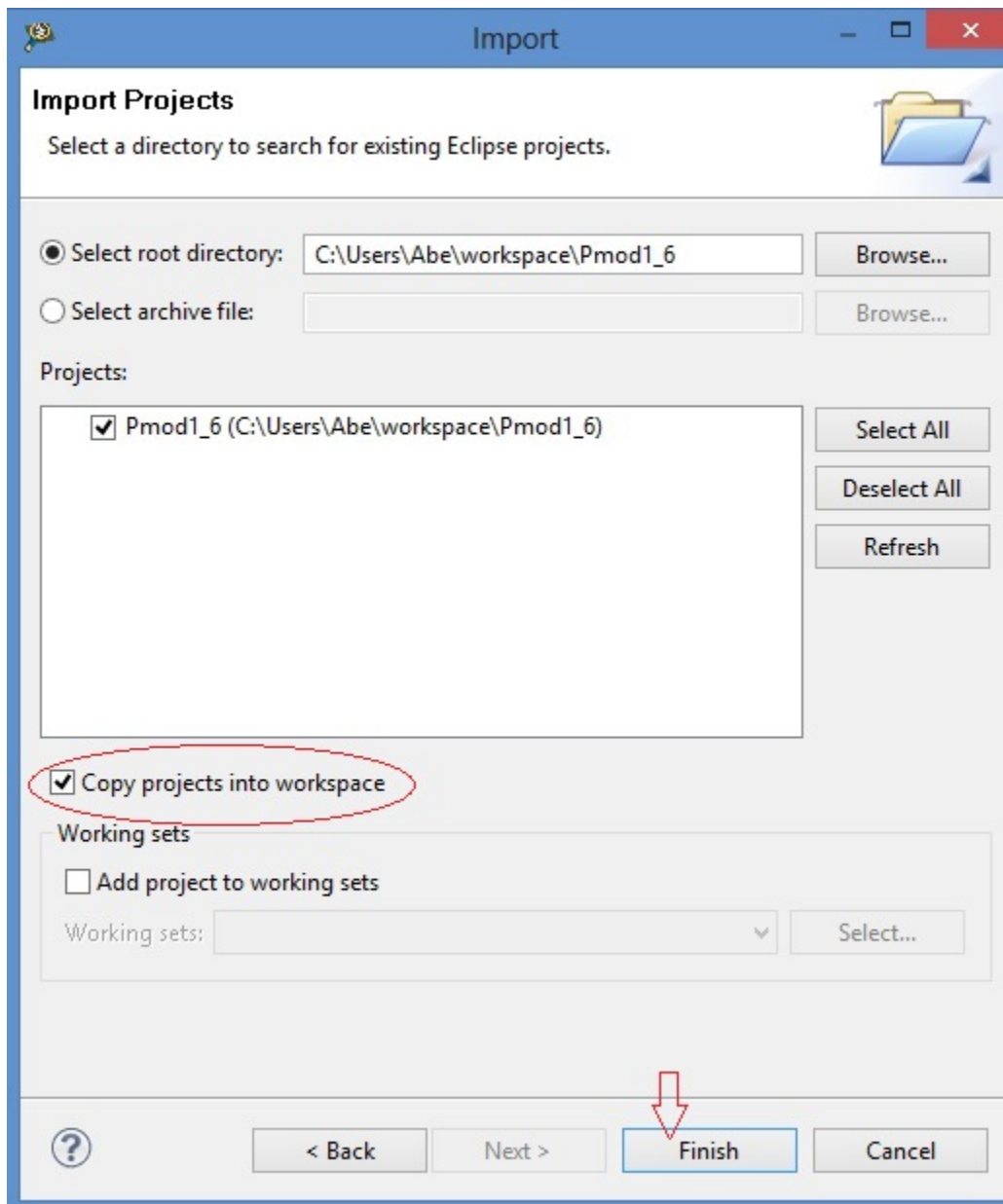


- in the next window make the following step

- 1 - click on **Browse** button.
- 2 - select folder "Pmod1_6" as below.
- 3 - click on OK button.

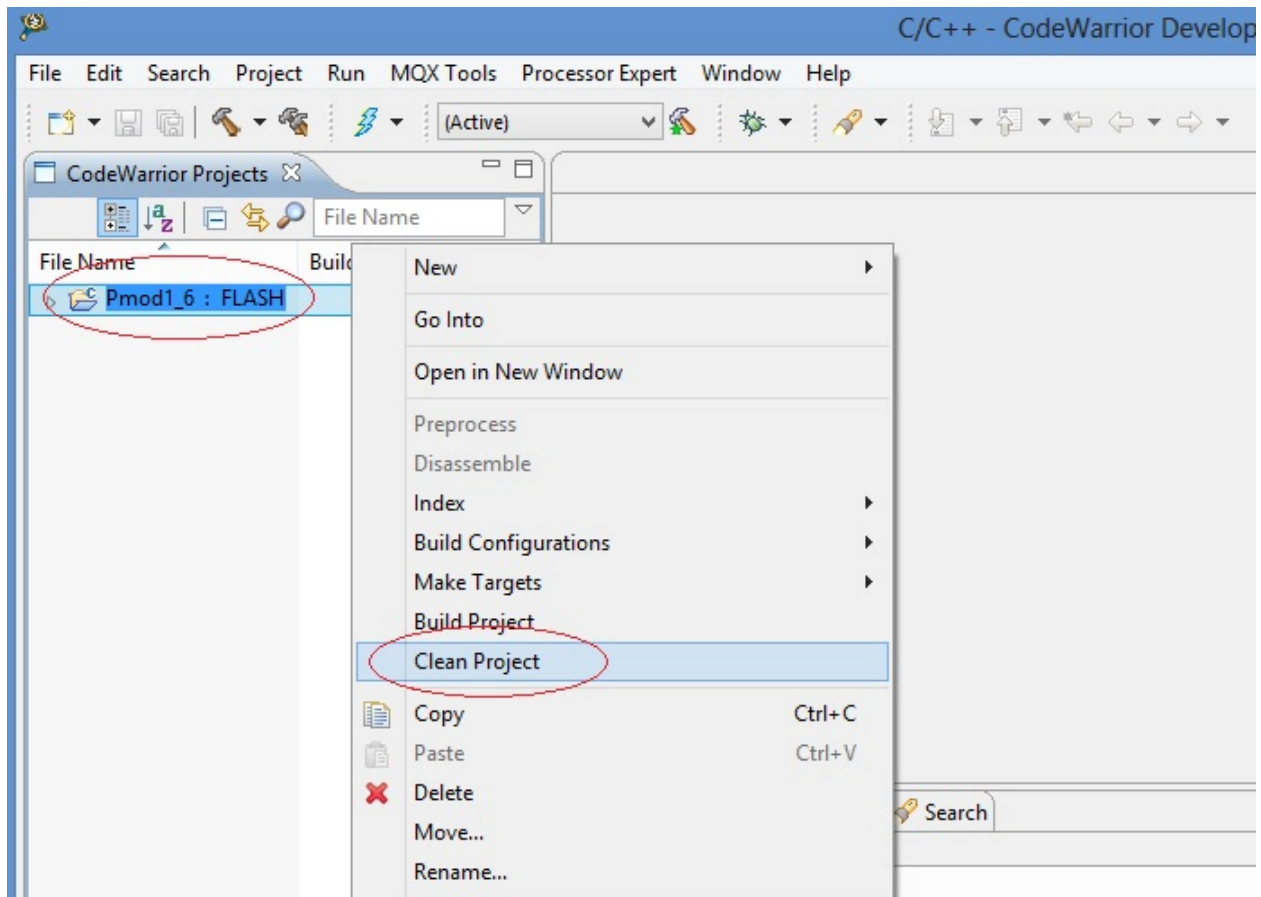


- select checkbox “Pmod1_6(C:\User\ *my_foder_name* \workspace\Pmod1_6) and “Copy projects into workspace”
- click “Finish”

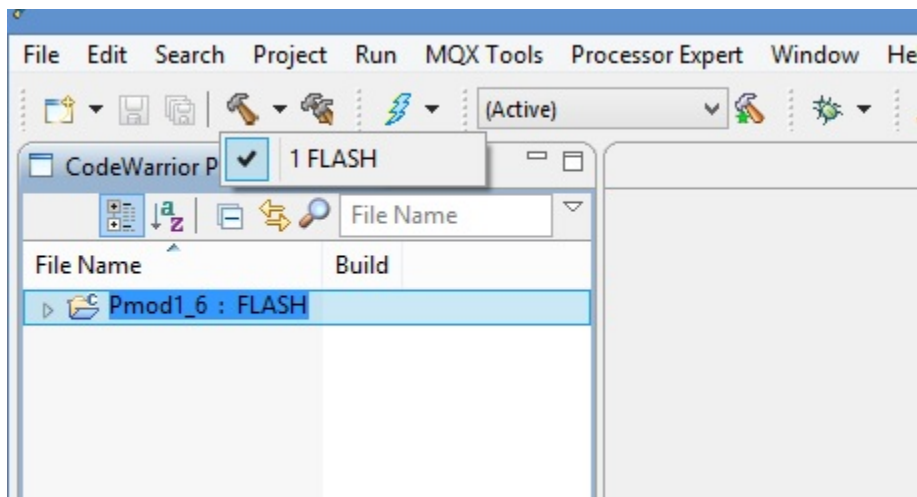


How to build Brooklyn Board FW

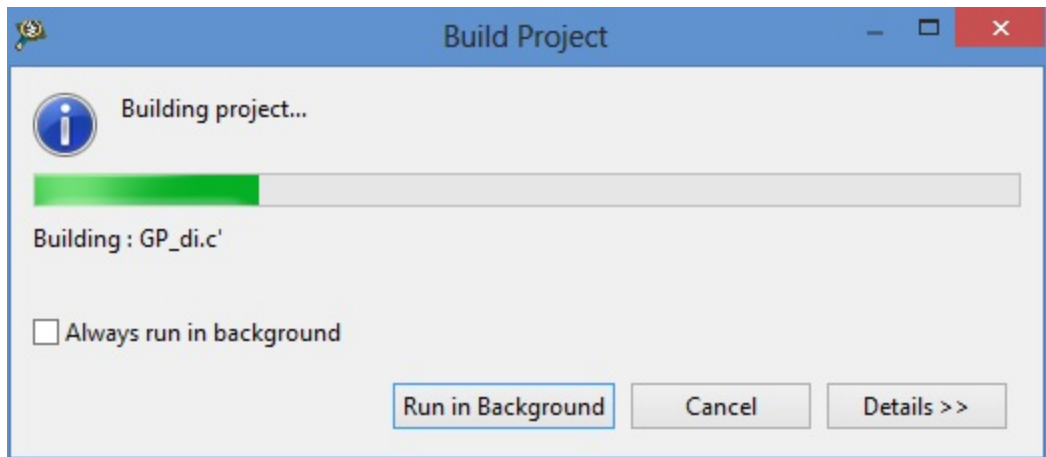
- see Codewarrior Project tab and select the project “Pmod1_6”, right-click over, select “Clean Project” and click



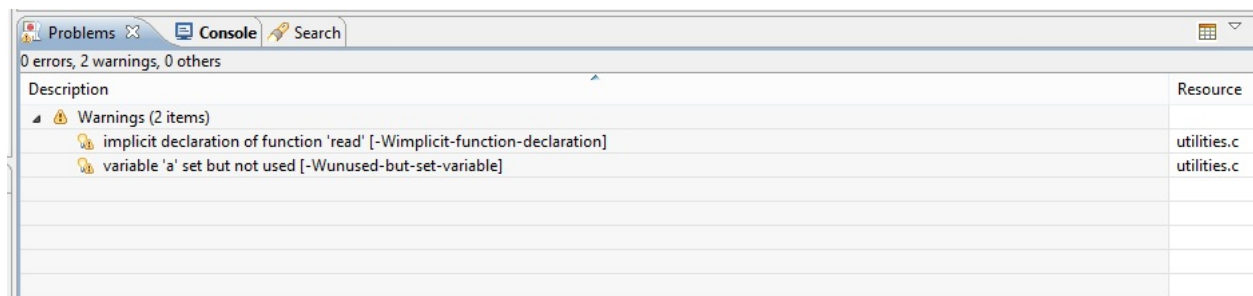
when process finish, see at hammer icon and click right arrow: in the tab you can see “1 FLASH” checked



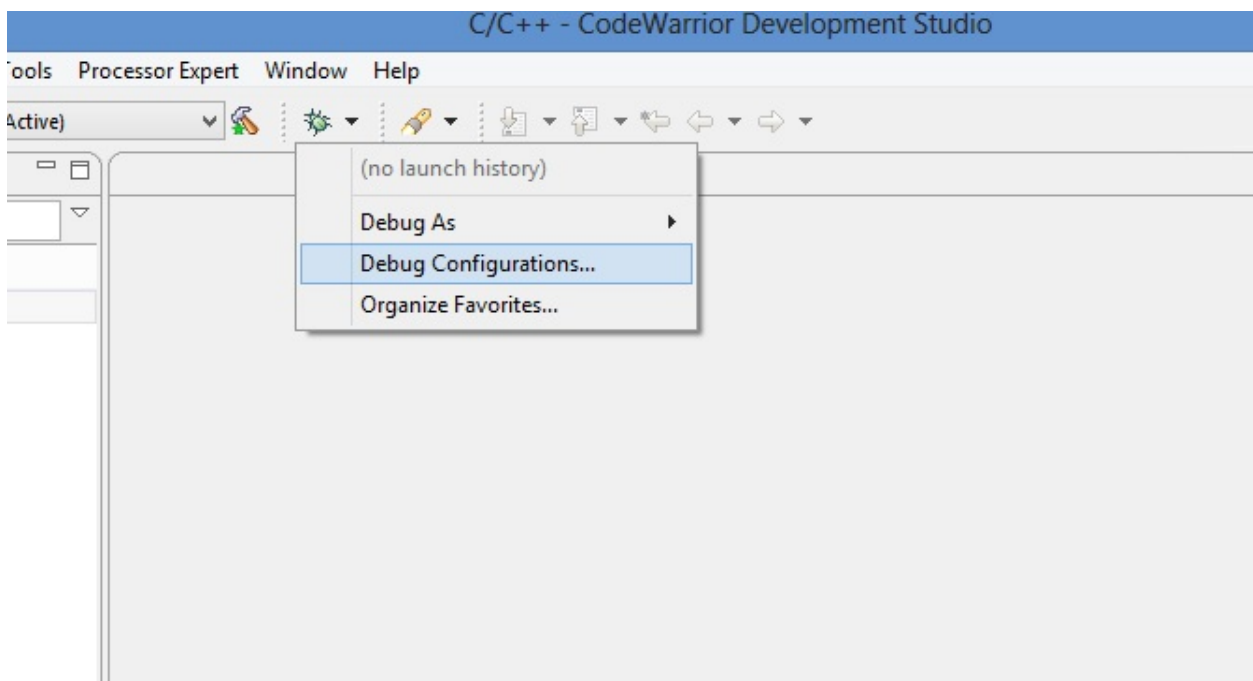
now click single-hammer icon to build entire project



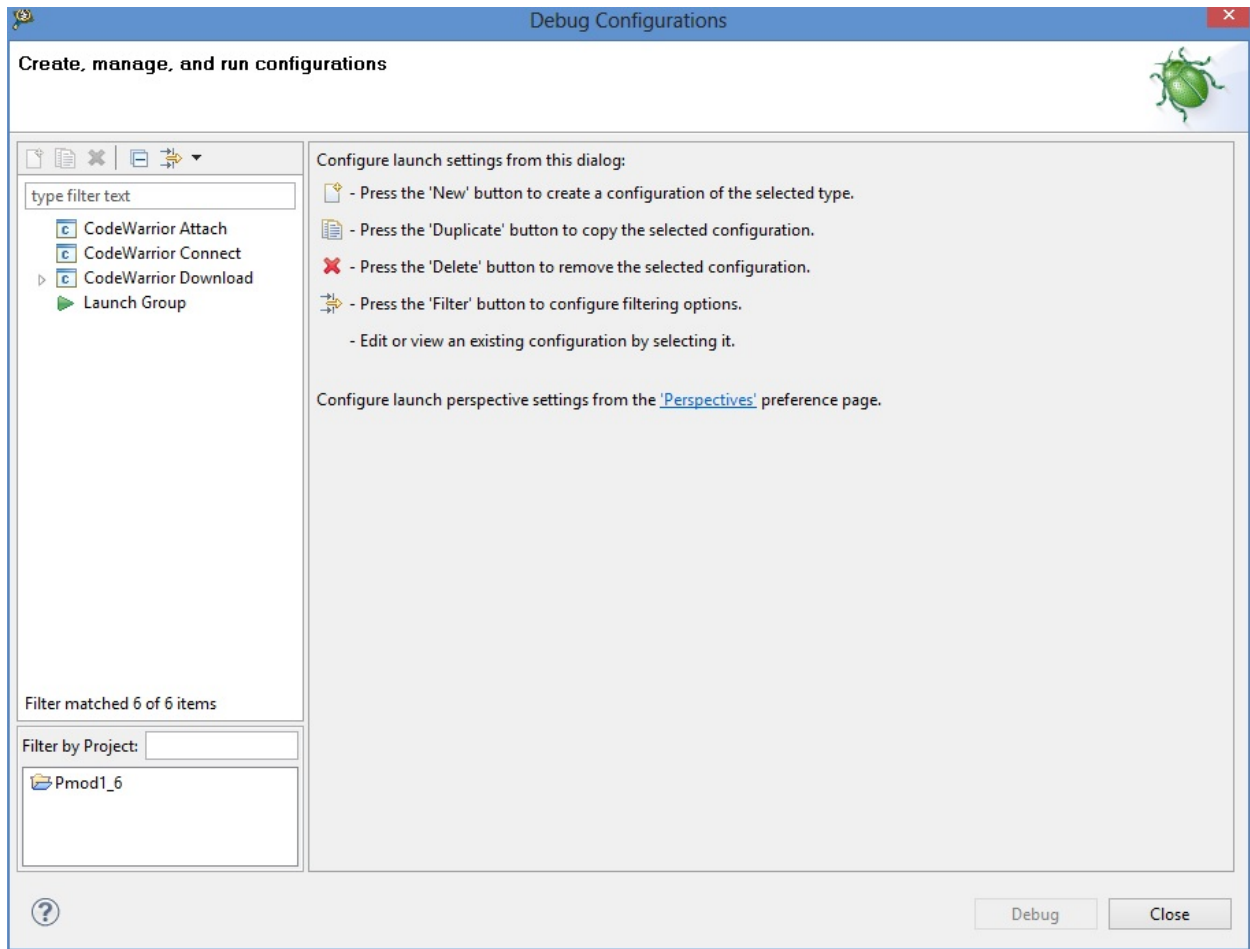
See the “problems” tab. There are 2 warnings derived from original project. They have no functionality effects.



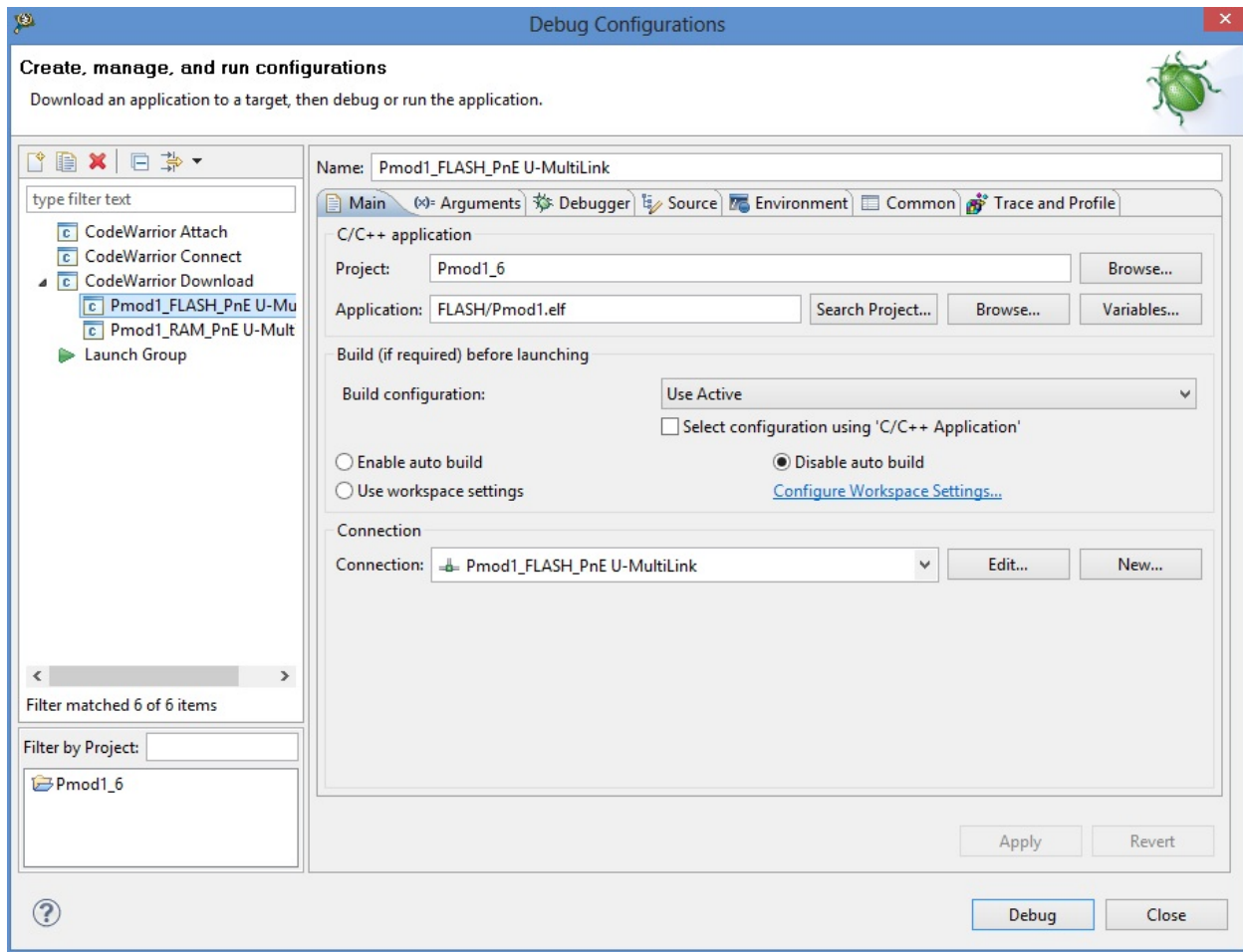
see the bug icon and click right arrow, select “Debug configurations” and click



Debug Configurations tab will open



expand Codewarrior Download, select “Pmod1_FLASH_PnE U-Multilink” and click. Now we see



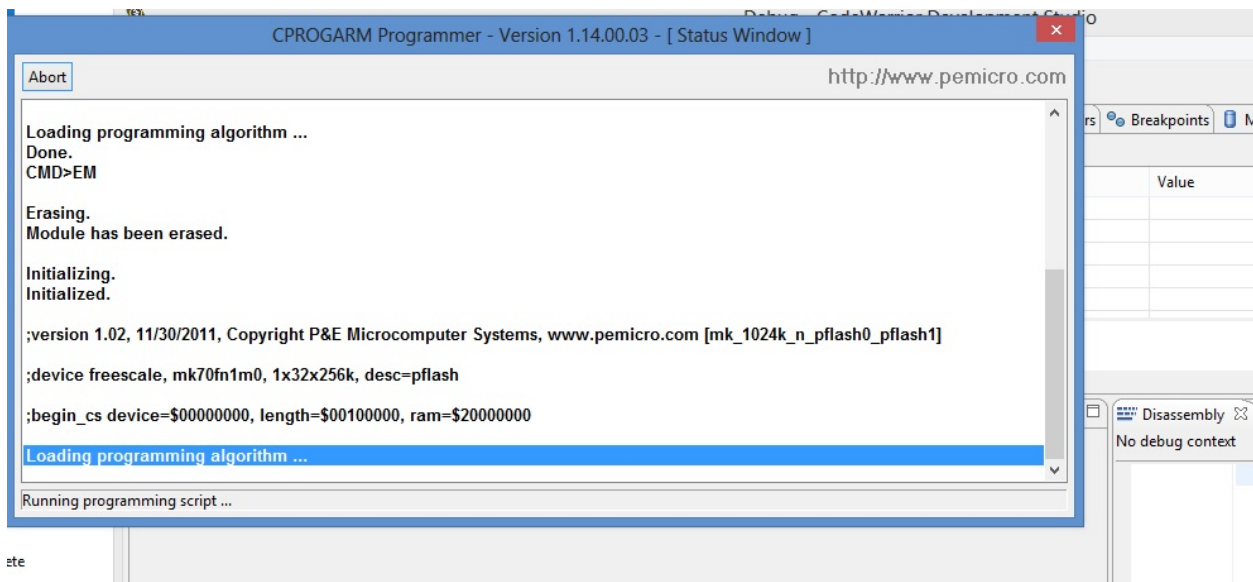
click on “Debug” button and wait

maybe will open firewall popup as below

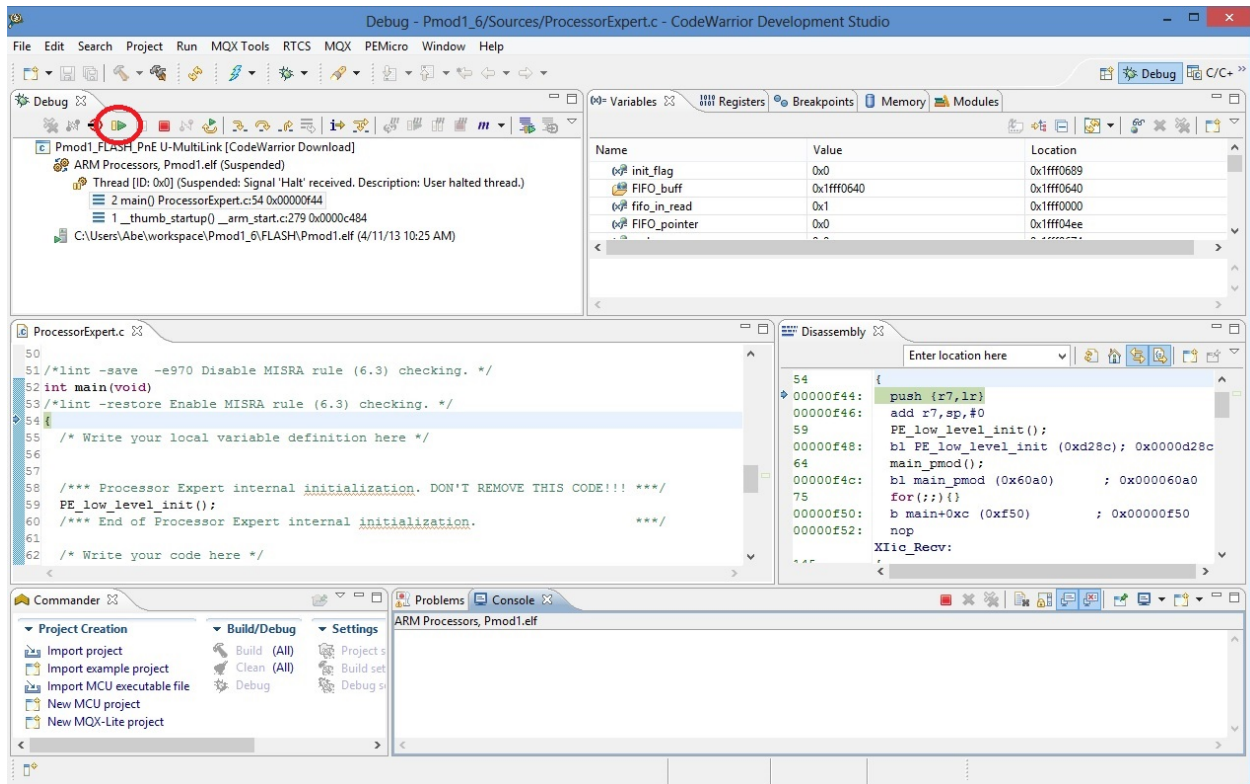


if yes, left-click on **enable access** and proceed

during firmware download this tab will open



and when download finish you see the main debug windows of Codewarrior



to start program you can press “F8” or click on Icon red-circled in image above

NOTE: for full Codewarrior functionality please refer to Freescale Official Guide

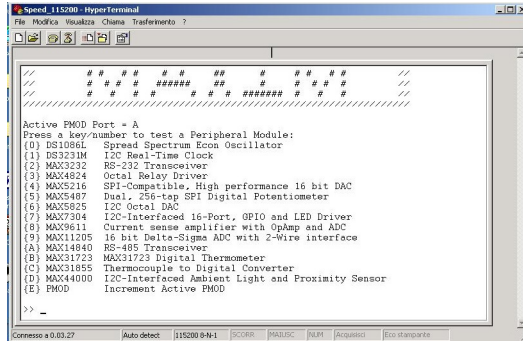
[download here Codewarrior Guide](#)

Running Brooklyn Board FW

When you start program, in terminal window you can see for few seconds this screen



and after you can see the main menu



Now select device menu (typing selection key in the terminal window) and follow menu option to test device.

It' strongly recomended to change or insert Pmod Modules when Tower System is off (without power).

Then, turn off the power by disconnecting the Mini USB B-type cable, remove device (if present) and insert new module in properly connector.

Turn on the power by plug the Mini USB B-type cable. The program will restart. Follow same steps used before to test new device

We also suggest you to see documentation [Maxim Pmod-Compatible Plug-In Peripheral Modules](#) for any specific further detail.

- search

C

CwInst, [3](#)

M

MaxFiles, [9](#)

P

Pixi, [16](#)