TASKING C166/ST10 Toolset v8.0r1 OCDS Debugging with CrossView Pro

TASKING™
Embedded software development from Altium™

This Start-Up Guide describes all necessary steps for successful OCDS debugging with the C166/ST10 toolset v8.0r1. It applies to the Infineon XC16x boards, but part of the information can also be used for configuring your own hardware design. The hints and instructions in this guide are based on the Eight Queens example (queens), included in the product.
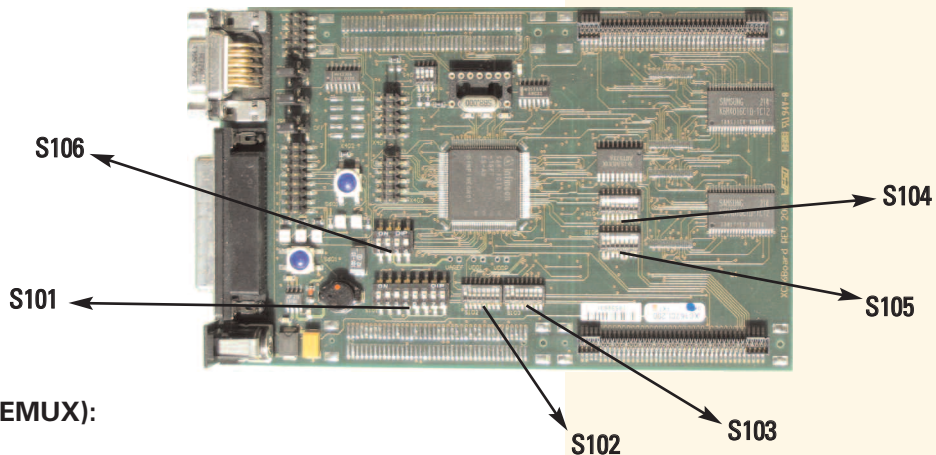
EXAMPLES RUNNING FROM EXTERNAL RAM

The following steps should be taken to run the Queens (queenss) example from external RAM. Settings are given for both a multiplexed and a non multiplexed bus.

DIP-Switch Settings

The correct board switch settings are:

For Multiplexed Bus (MUX):

Switch	1	2	3	4	5	6	7	8
S101	Off	Off	Off	Off	Off	On	On	Off
S102	Off	Off	On	Off	Off	Off	Off	Off
S103	Off	Off	Off	**Off**	Off	Off	Off	Off
S104	On	On	Off	Off	Off	Off	Off	Off
S105	Off	Off	On	On	On	On	On	On
S106	Off	Off	Off	**Off**				

✓ **VERSIONS**

This Start-Up Guide is based on the following versions:
- *TASKING C166/ST10 toolset, v8.0r1*
- *Infineon XC161CJ Evaluation Board, rev 200*
- *Infineon XC164CS Evaluation Board, rev 202B*
- *Infineon XC167CI Evaluation Board, rev 200*

✓ **NOTE**

A Start-Up Guide for v7.5r6 of the TASKING toolset is available in the C166 support section of the TASKING website.

For Non Multiplexed Bus (DEMUX):

Switch	1	2	3	4	5	6	7	8
S101	Off	**On**	**On**	Off	Off	**Off**	**Off**	Off
S102	Off	Off	On	Off	Off	Off	Off	Off
S103	Off	**On**	Off	**Off**	Off	Off	Off	Off
S104	On	On	Off	Off	Off	Off	Off	Off
S105	Off	Off	On	On	On	On	On	On
S106	Off	Off	Off	On				

Note: the **bold blue** switches mark deviations from the default settings as given in the Infineon XC16x Board Manual (v1.2, Oct. 2002).

Startup Register Settings

The correct system register settings for the Bus Configuration, External Bus Controller and address selection should be set as follows:

Register	Multiplexed Bus	Non Multiplexed Bus
SYSCON0	0x0000	0x0000
SYSCON1	0x0000	0x0000
VECSEG	0x0000	0x0000
CPUCON1	0x0007	0x0007
CPUCON2	0x8FBB	0x8FBB
EBCMOD0	0x4022	0x6022
FCONCS0	0x0031	0x0021
TCONCS0	0x7A68	0x6240

These registers are automatically set when the execution environment is selected as described in the next paragraph.

EDE Quick Start

In the EDE the processor and target board must be selected. The application must be configured so that the register values are set to match the board settings and memory must be configured. The following steps must be taken to achieve these settings and to get a successful debugger connection to the target board.

1. Open the Queens project
   a. Select the **Project** menu
   b. Select **Set current** and set queens.pjt as active project or right-click on the queens project in the Project bar at the left
   c. Select **Set as Current Project**

2. From the **Project** menu, select **Project Options...**

2.1 a. In the left pane expand the **Application** entry and select **Processor**
   b. Set **Processor** to the XC16x processor that you use
   c. When the popup window shows up Click **Yes** to set the registers to default values

2.2 a. In the left pane select **Memory Model**
   b. Set **Memory Model** to: small
   c. Set **Small model memory layout** to: Paged
   d. Fill in the page numbers:
      DPP0: 0
      DPP1: 1
      DPP2: 4
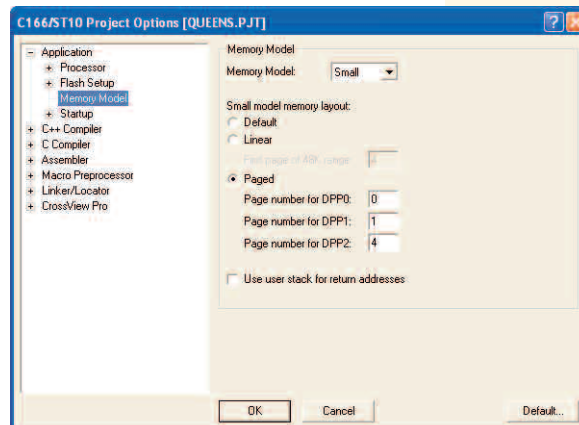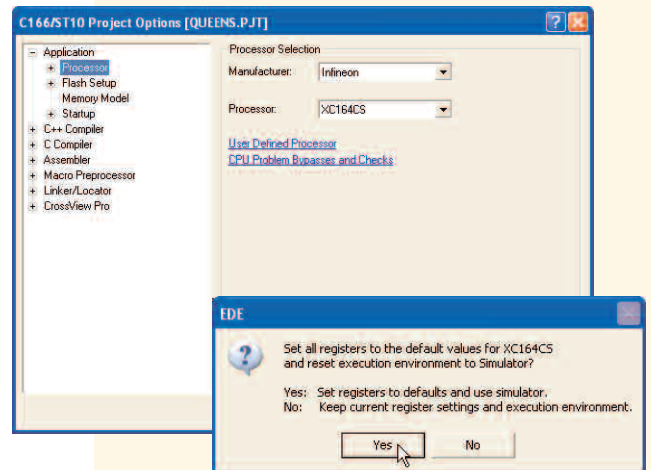
2.3 a. In the left pane expand the **Linker/Locator** entry and select **Memory**

b. Set **Type**=ROM, **Start**=0x0, **End**=0x069FF
Set **Type**=RAM, **Start**=0x06A00, **End**=0x07FFF
Set **Type**=RAM, **Start**=0x0C000, **End**=0x0CFFF
Set **Type**=RAM, **Start**=0x10000, **End**=0x1FFFF
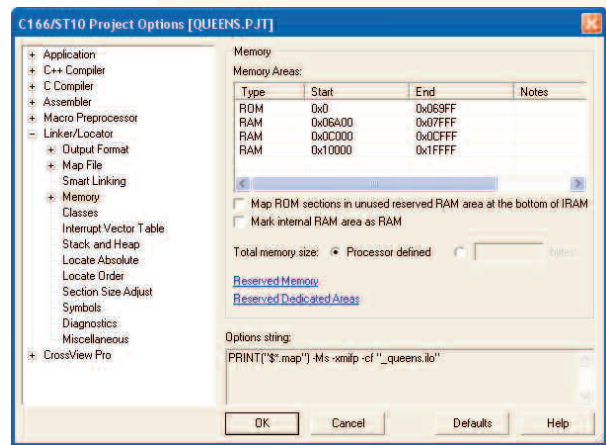
2.4 a. In the left pane expand the **CrossView Pro** entry and select **Execution Environment**

b. Set the **Manufacturer** to: Infineon

c. Set the **Execution environment** to the evaluation board and configuration you use

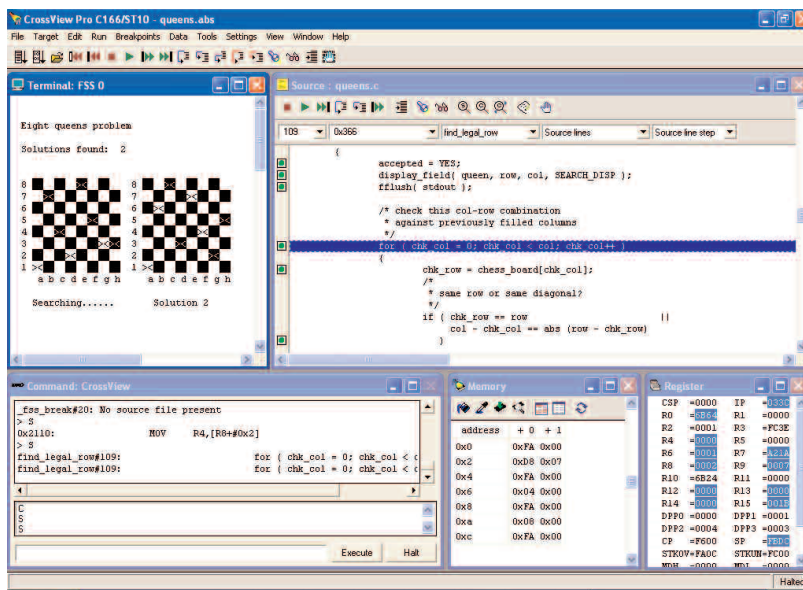d. When the popup window shows up click **Yes** to set the startup registers to new values
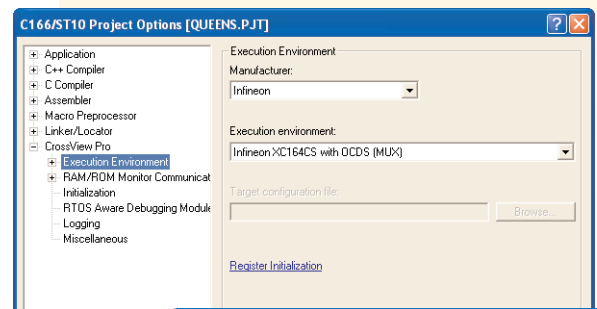
2.5 Click on **OK**

3. Rebuild the application
a. Select the **Build** menu
b. Select **Rebuild**

4. You can now start debugging with CrossView Pro
a. Select the **Build** menu
b. Select **Debug**
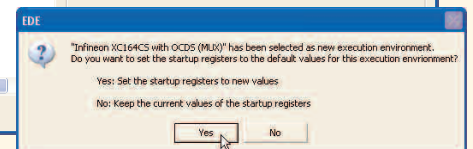
Output of the queens example, shown by CrossView Pro Debugger

HARDWARE AND SOFTWARE BREAKPOINTS

The OCDS only has four hardware breakpoints available, which is often too limited for debugging an application. With software breakpoints this limitation does not exist. The usage of OCDS software breakpoints is limited to 64k breakpoints. A software breakpoint is set by injecting DEBUG instructions into the memory, so this is only possible within RAM. For debugging code in EEPROM or FLASH memory only hardware breakpoints can be used.

From CrossView Pro you can configure the use of hardware and software breakpoints:
a. Select **Breakpoints** menu
b. Select **Breakpoints…**
c. Click on **Preferences…**

d. You will now see the dialog from which you can configure the use of breakpoints.
e. If your application is allocated in FLASH memory, select **Hardware breakpoints only**
If your application is allocated in RAM, select **First hardware breakpoints then software**
f. Click **OK**
g. Click **OK**

TROUBLE SHOOTING

1. CrossView Pro issues the following error:
GdiInitIO(): OCDS module can't be enabled (connect cable or correct jumper setting)

Solution: Set JP501 off -> on

2. CrossView Pro issues the following error:
GdiInitIO(): No valid JTAG cable found (target could be in Bootstrap mode)

Solution: Set S103.4 on -> off

3. Check the TASKING support web: www.tasking.com/support/C166-ST10

NOTE

The OCDS has a limited number of hardware breakpoints:
■ 4 code or data write breakpoints, of which one can be used for a code range or data range read or write breakpoint
■ or 1 value write breakpoint

NOTE

By default the peripherals will not be stopped upon a breakpoint.

NOTE

For data breakpoints, CrossView Pro will always use hardware breakpoints.

JP501

S103.4