HOT TC1797

Let’s get started now!



Programming the Internal RAM.

Exercise MEM_1

Exercise MEM_1
Program the internal RAM

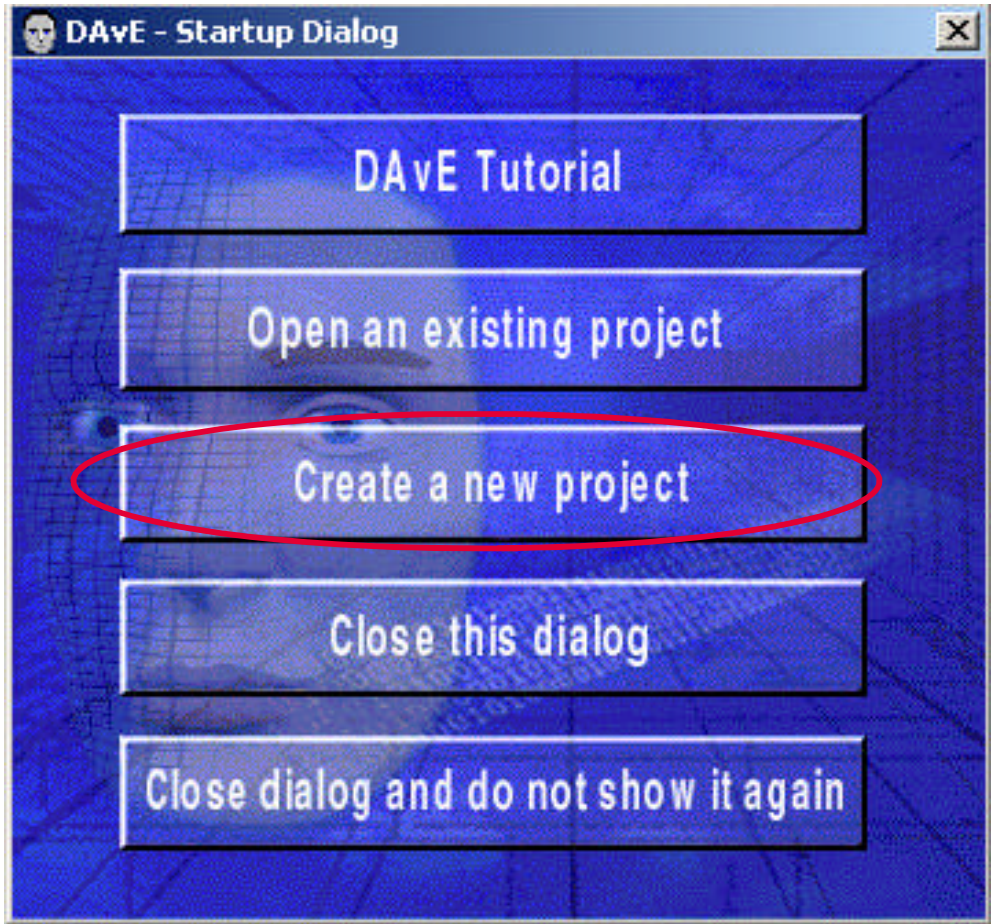
- Objective:
  - Save code in the internal SPRAM.
  - Run the code: the on board LED will blink.

Exercise MEM_1
- DAvE Configurations

- Start DAvE for TC1797.

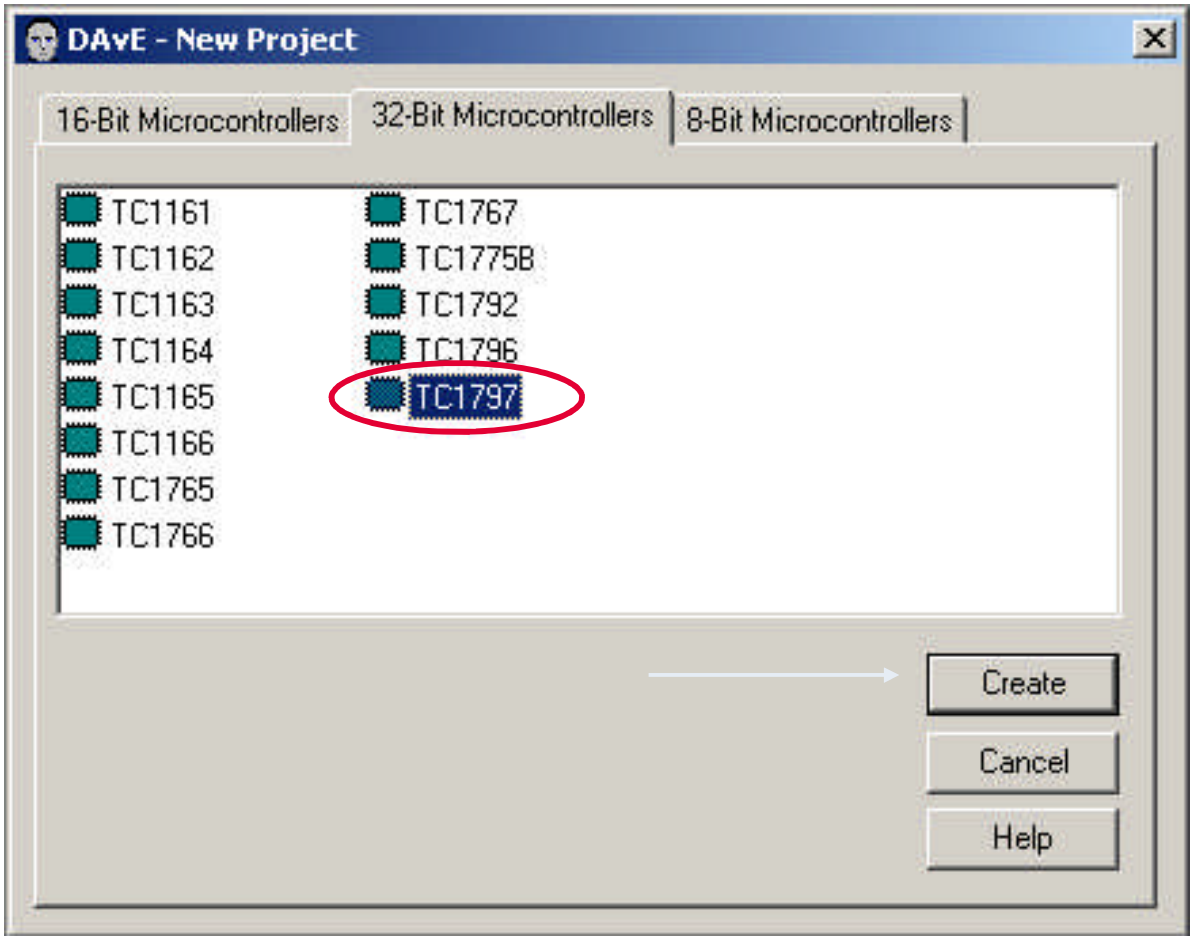


- Create a new project:
  - From DAvE menu bar, Select **File → New**
  - Or click 'Create a new project'



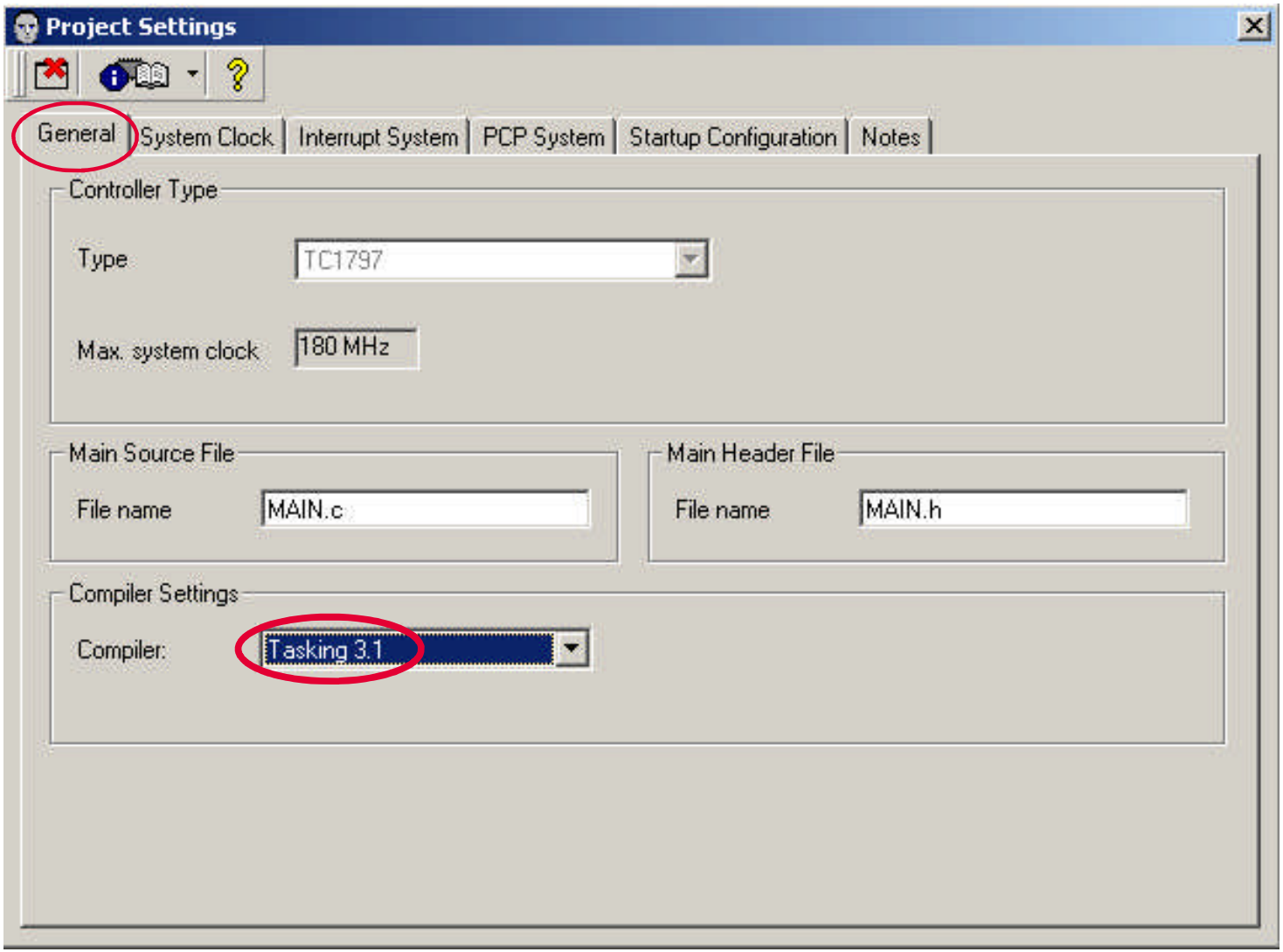
Exercise MEM_1
- DAvE Configurations

- Create a new project:
  - Select 32-Bit microcontroller: ‘TC1797’.



Exercise MEM_1
- DAvE Configurations

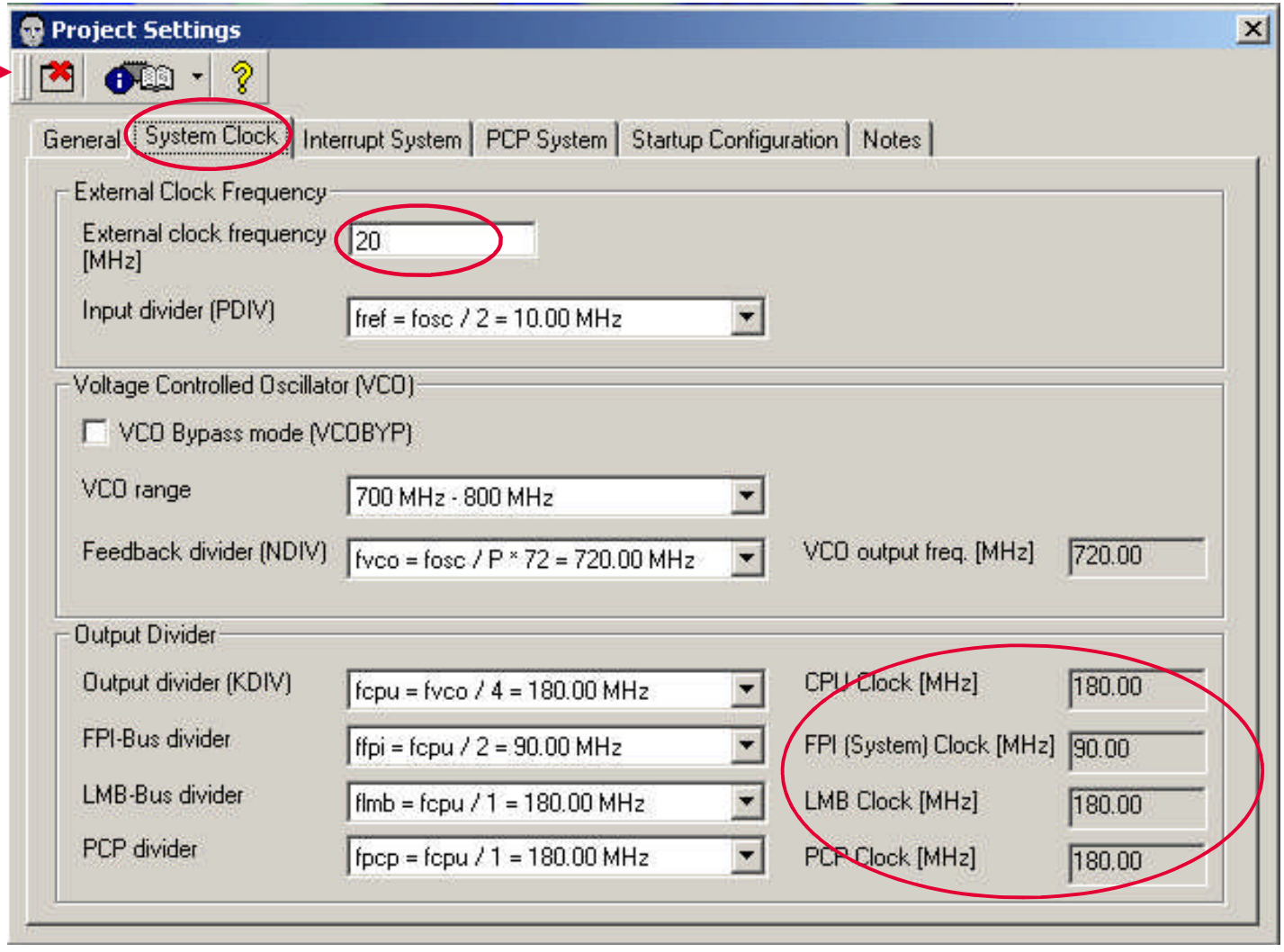
- **Project Settings** - opens automatically
  - **General**:
    - Compiler Settings: Tasking 3.1



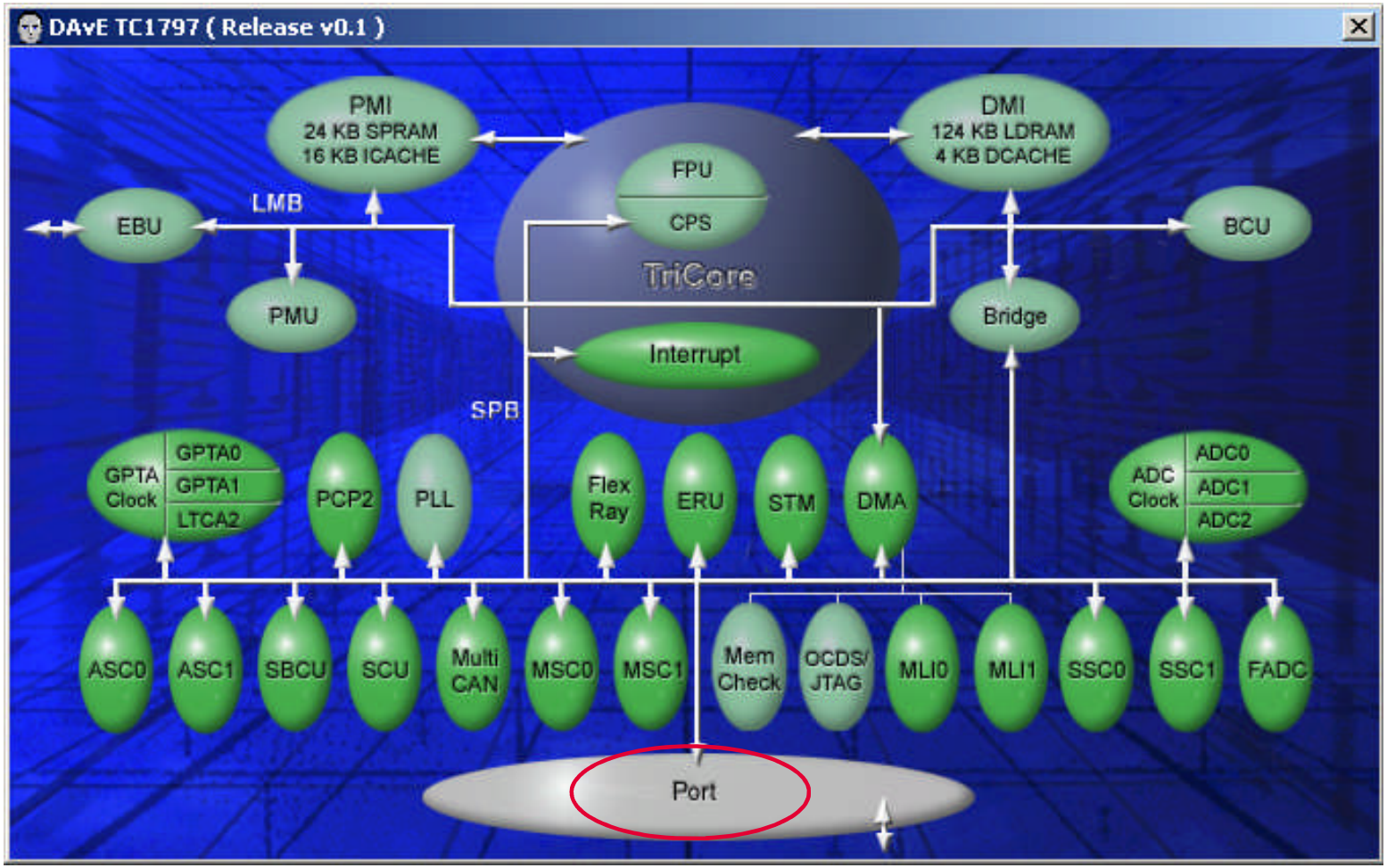
Exercise MEM_1
- DAvE Configurations

- **Project Settings** - opens automatically
  - **General**:
    - System Clock

Click to close



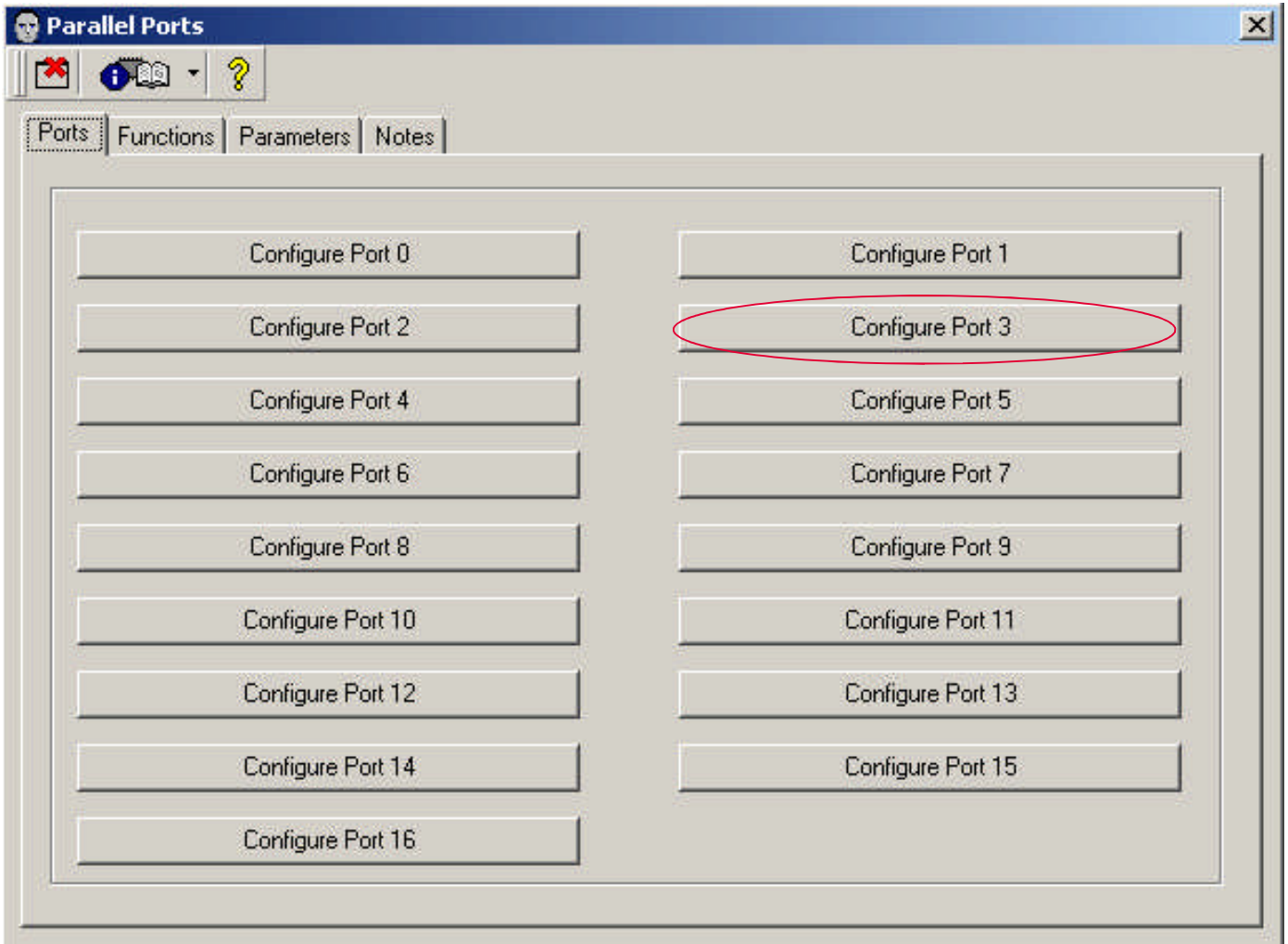
- **Project Settings (cont.)**
  - **Port:**



Project Settings (cont.)

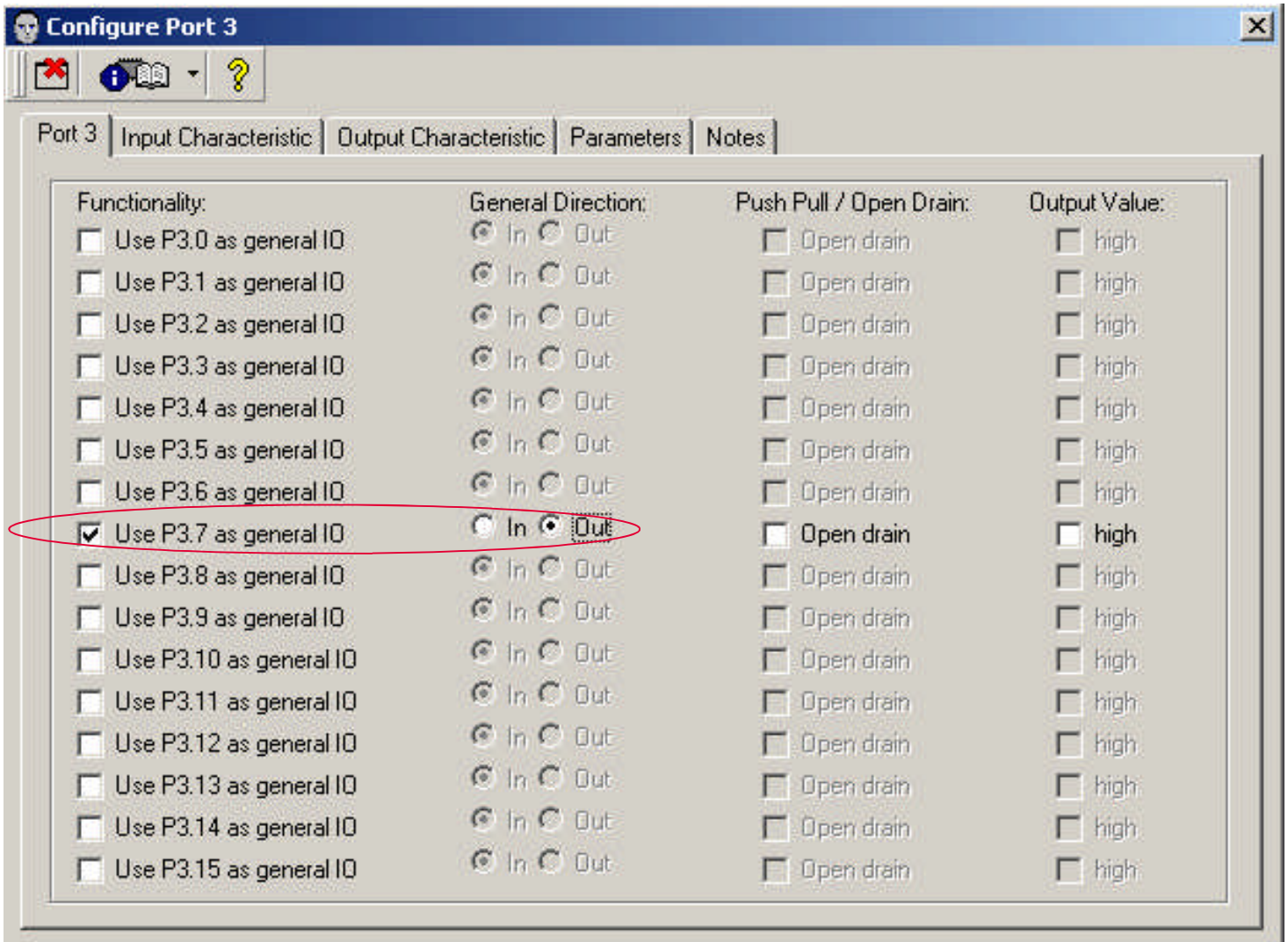
Port

- Configure Port 3

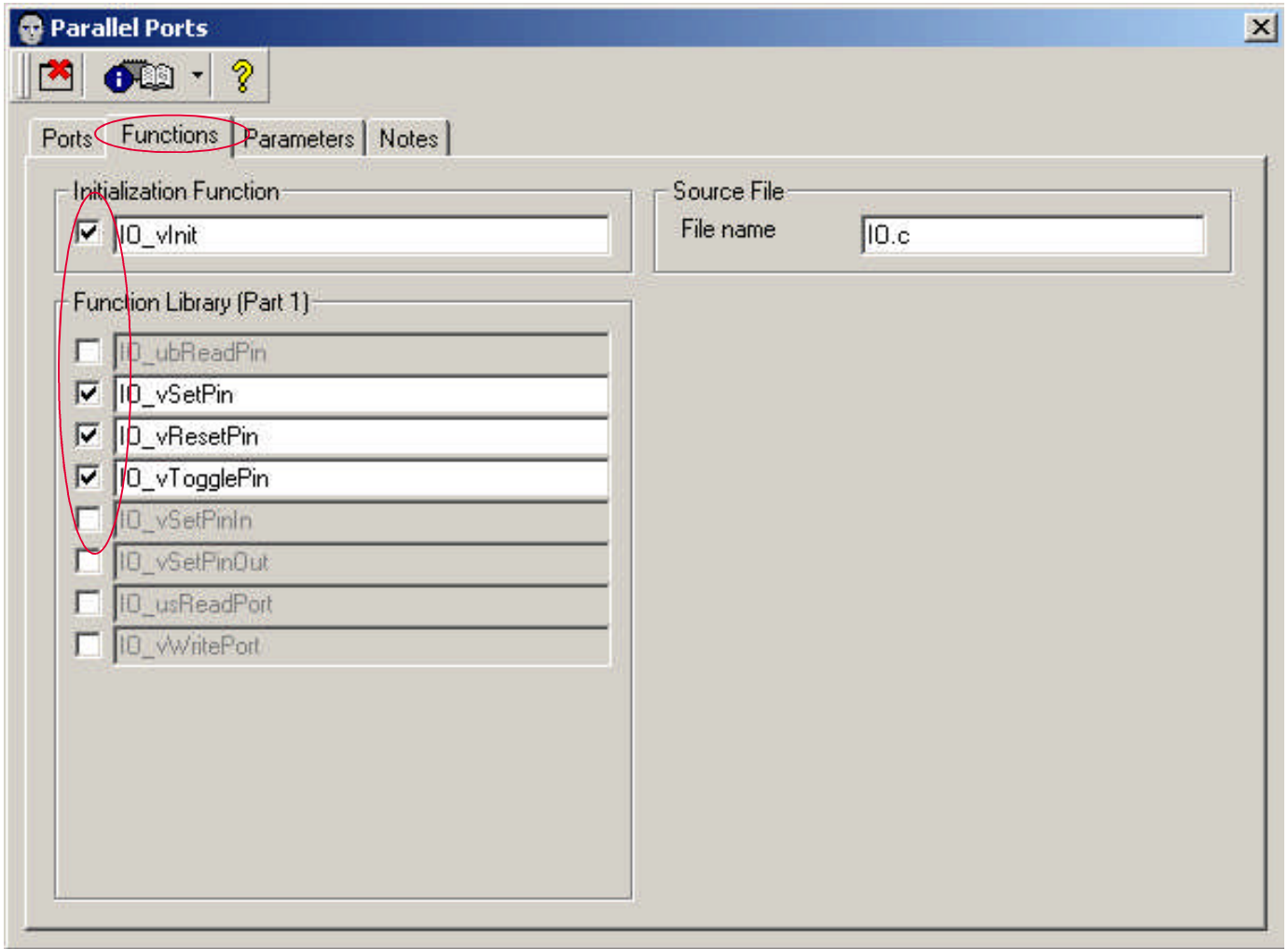


Exercise MEM_1 - DAvE Configurations (cont.)

- **Project Settings (cont.)**
  - **Port**
    - Configure Port 3.7 as output



- **Project Settings (cont.)**
  - **Port**
    - Parallel ports -> Functions



Exercise MEM_1 - DAvE Configurations Generate Code

- Save (**File → Save)** this DAvE file in :

  c:\hot\tc1797\mem_1\mem_1.dav

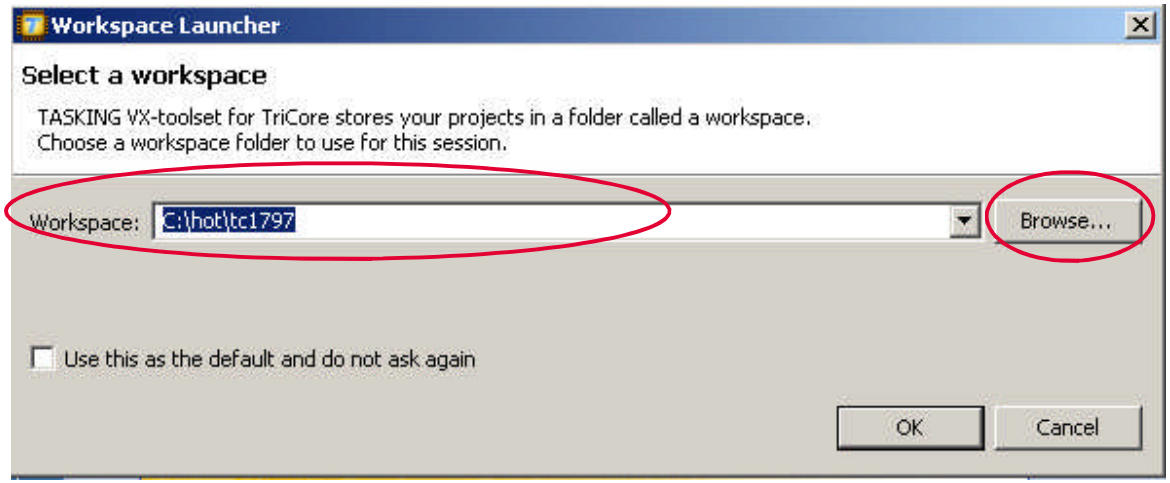
- Let DAvE now generate the code for you
  - Go to **File → Generate Code** or click: 
  - DAvE opens a new window with the Project Documentation file. On the left hand side a browser window shows the generated files. You can take a look at the generated code or just find out in which file DAvE put the included functions.
  - In general:
    - if the included function is a macro it is included in the '.h' file
    - if the included function is a function it is included in the '.c' file

Exercise MEM_1 - Start Tasking and Create New Work Space

- Start Tasking EDE for TriCore :

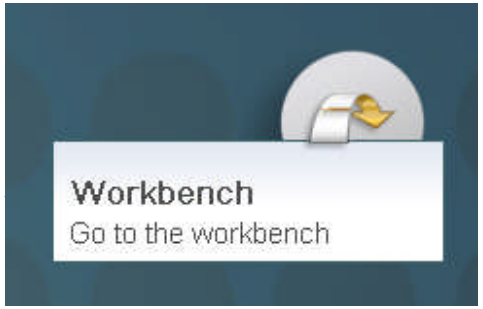


  - Browse to the directory where the DAvE generated files are saved.



Exercise MEM_1
- Create New Work Space

- *Click 'workbench*

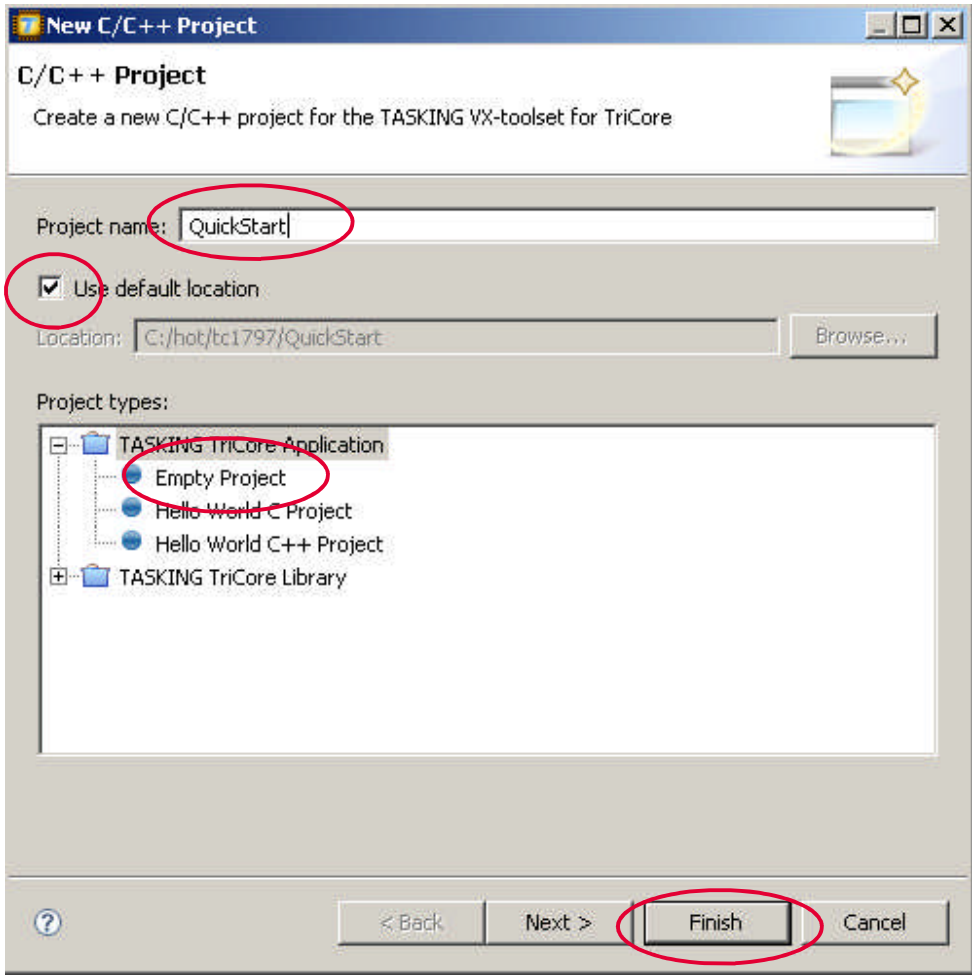


- From the **File** menu select **New** –> **TASKING VX-toolset for TriCore C/C++ Project**.

- *The New C/C++ Project wizard appears.*

Exercise MEM_1 - Create New Project

- In the **Project name** field, type **MEM_1** as the name of your new project.
- Select **Empty Project**. This creates a project without a C source file containing the function main() .
- Leave the **Use default location** checked to use the default location for your new project.
- Click **Finish** to finish the wizard and to create the project.
- Selecting **Finish** instead of **Next**, leaving the additional settings as defined by the defaults.



Exercise MEM_1
- Set Project Options

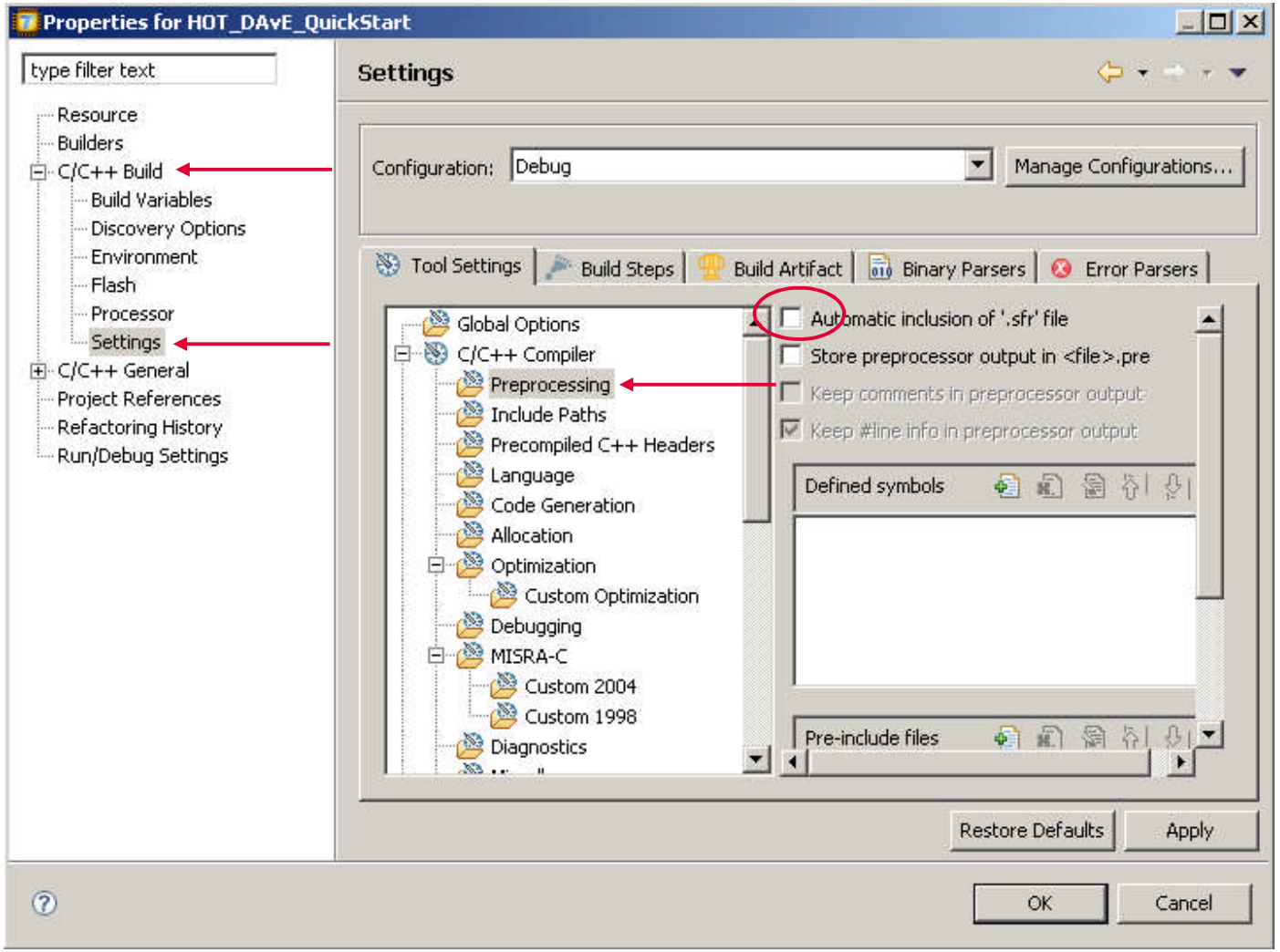
- From the **Project** menu, select **Properties**. Alternatively, you can click the button. 

- ***Properties for hello_dave*** *dialog box appears*

Exercise MEM_1
- Set Project Options

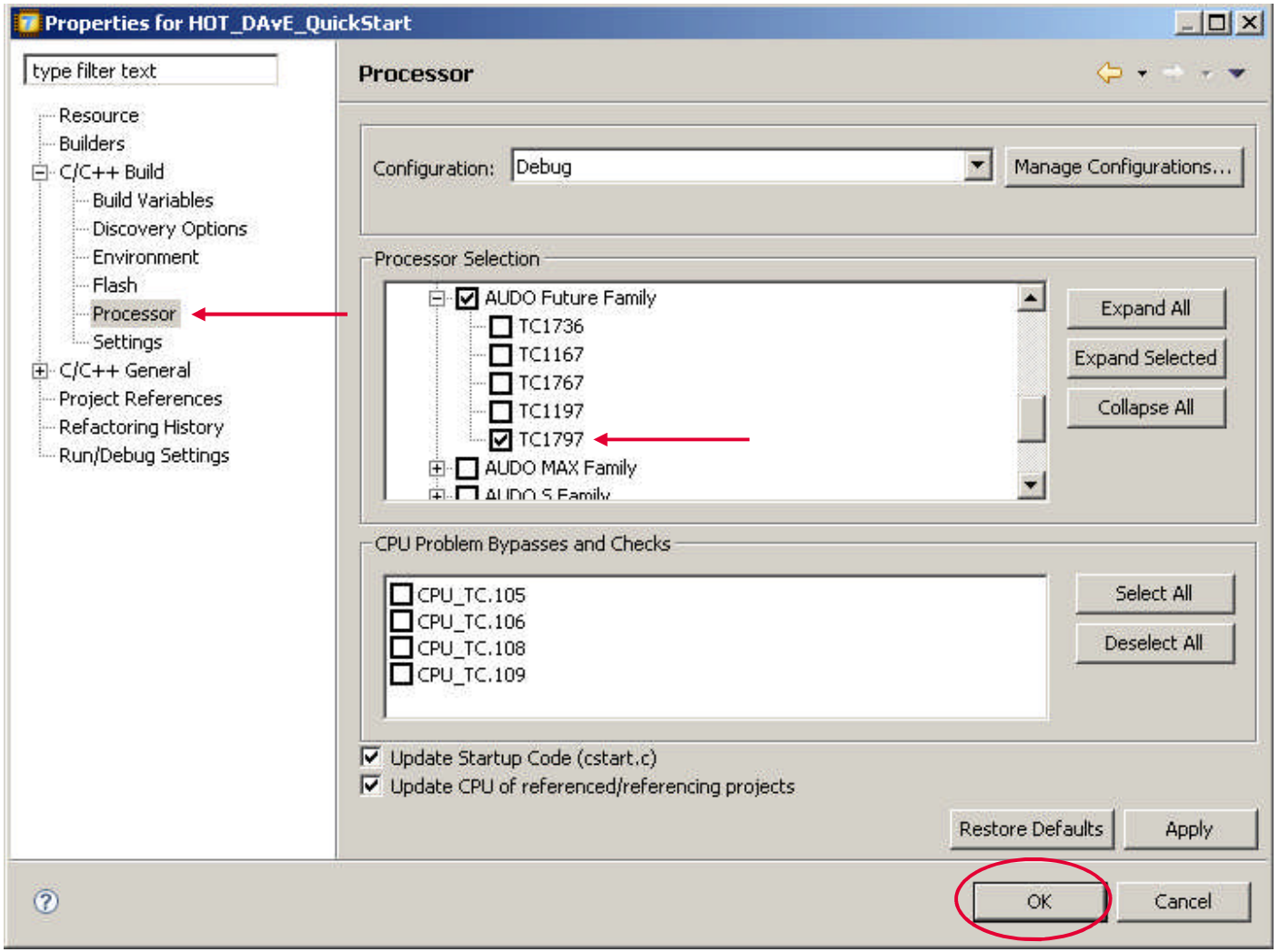
- In the left pane, if not selected expand **C/C++ Build**, select **Settings**
- In the **C/C++ Compiler -> Preprocessing** disable **Automatic inclusion of `.sfr file**.

(This is necessary when the project uses DAvE generated code.)



Exercise MEM_1
- Set Project Options

- In the left pane select **Processor** to access the TriCore **Processor Selection**
- In **Processor Selection**, if not selected, expand **AUDO Future Family**, enable **TC1797** checkbox.
- Click **OK**



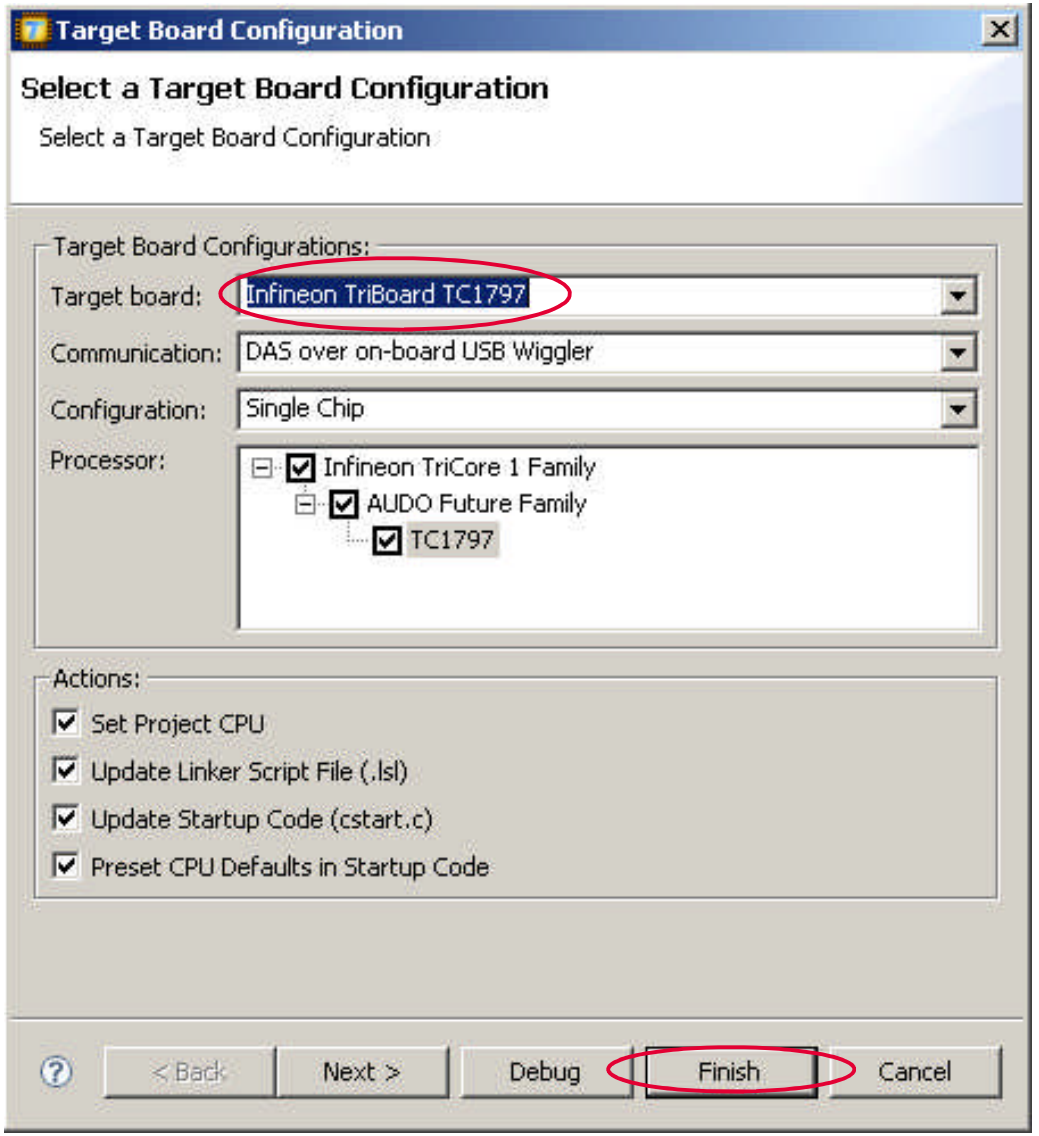
Exercise MEM_1
- Configure the Target

- From the **Project** menu select **Target Board Configuration** or click button. 

- ***Target Board Configuration*** *dialog box appears*

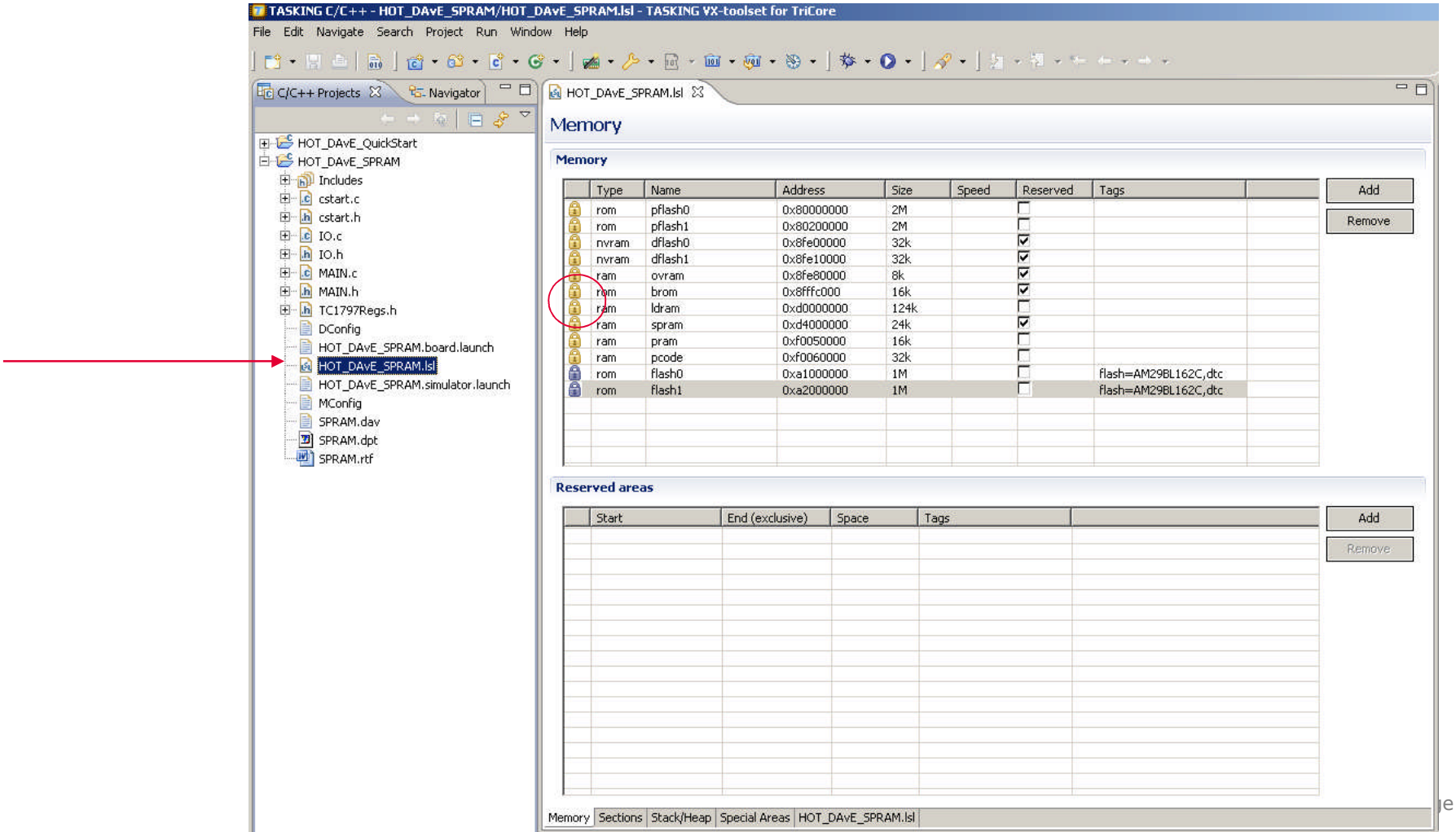
Exercise MEM_1
- Configure the Target

- In the **Target board** field, select TC1797
- Leave all other default settings as is.
- Click the **Finish** button to create the new board configuration **QuickStart.board.launch**.



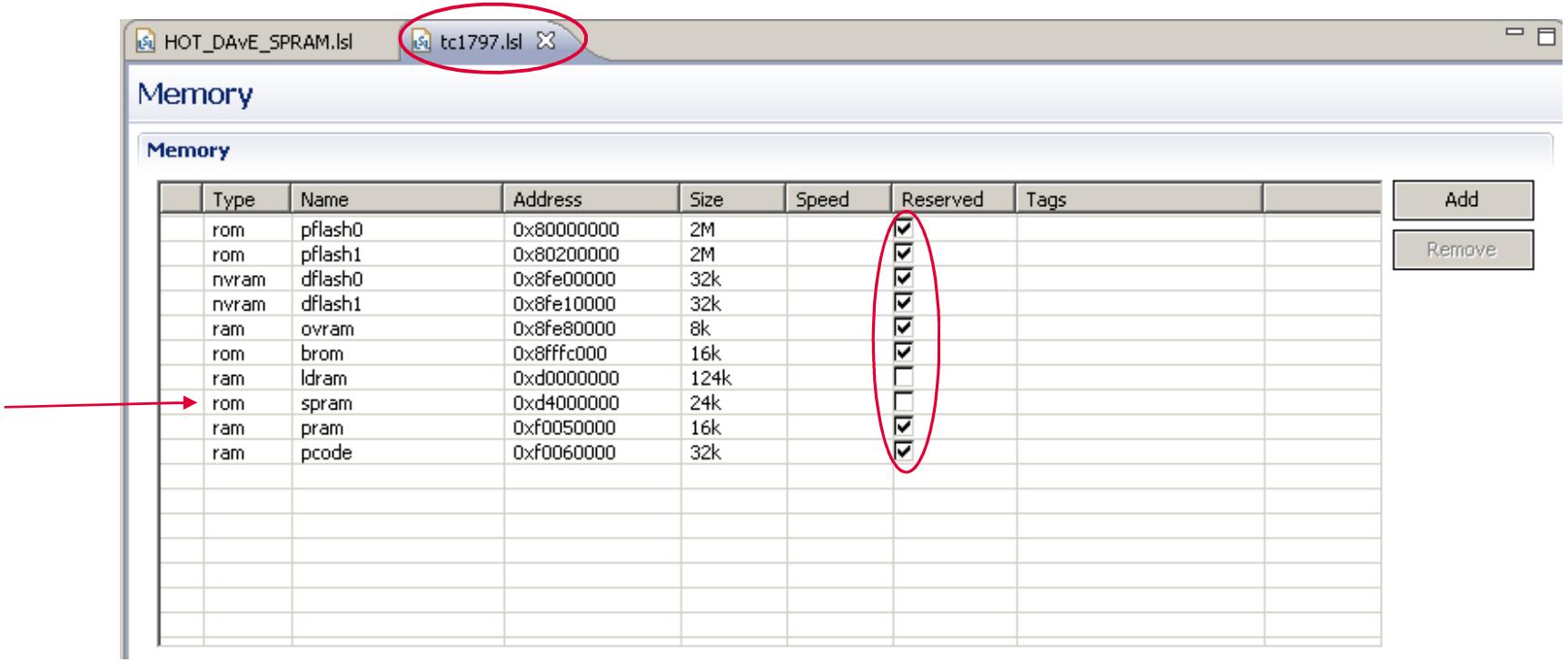
Exercise MEM_1 – Linker setting

- Linker Setting
  - Double click ‘HOT_DAvE_SPRAM.LSL’
  - Click on Yellow Lock and click ‘Yes’ to create ‘tc1797.lsl’



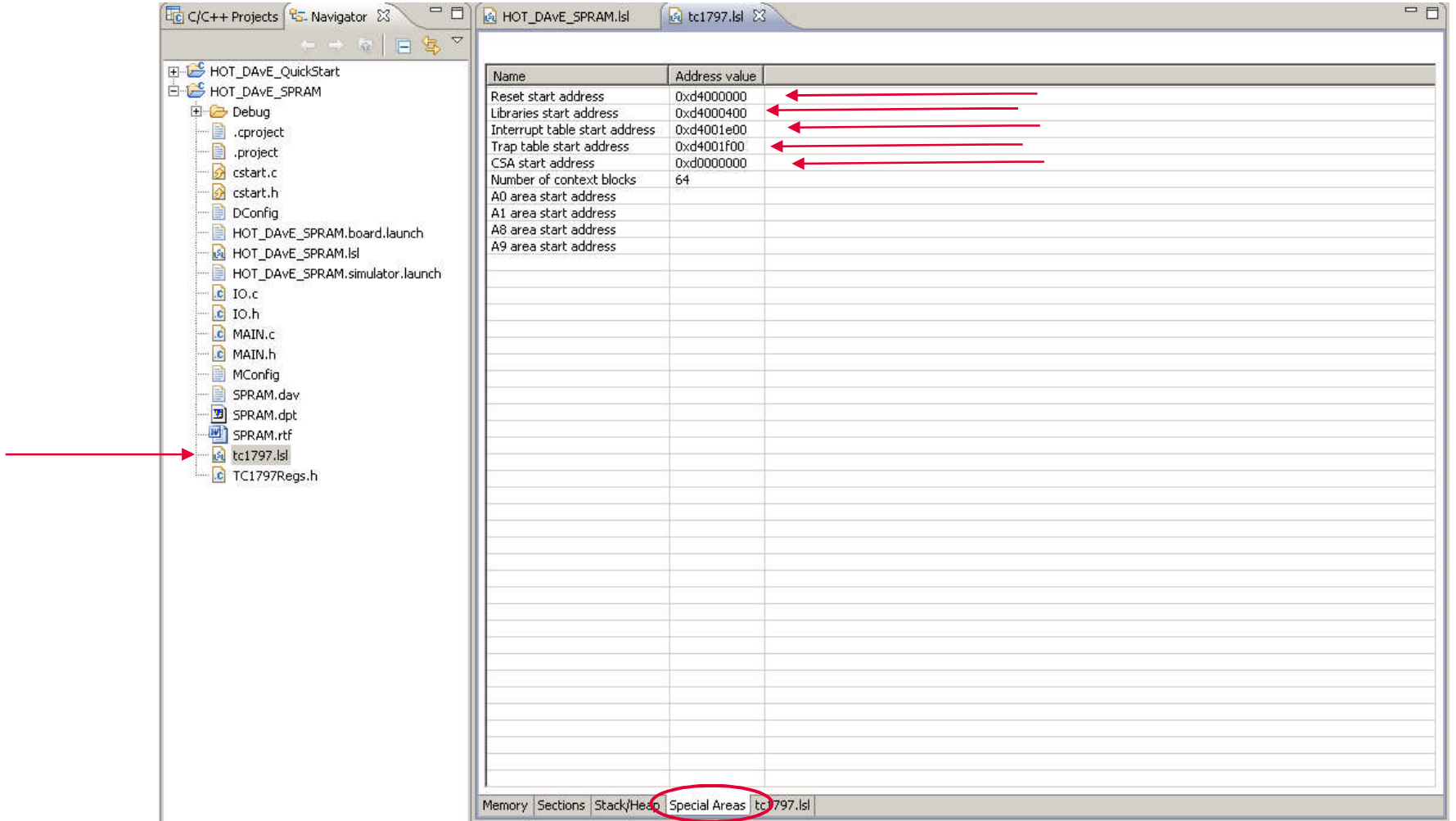
Exercise MEM_1 - Linker setting

- Linker Setting (cont.)
  - Double click 'tc1797.lsl'
    - Remove tick in Reserved box for SPRAM & LDRAM only, rest of memory are all reserved
    - Change SPRAM type: ROM



Exercise MEM_1 - Linker setting

- Linker → Special areas:
  - Reset start address: 0xD4000000.
  - Library start address: 0xD4000400.
  - Interrupt table start address: 0xD4001E00.
  - Trap table start address: 0xD4001F00.
  - CSA start address: 0xD0000000.



Exercise MEM_1 - Complete generated code

- On the following slides please find the code you have to add to the generated code. For reasons of orientation you will always find a few lines of already existing code as well. The code you really have to add looks like this:

```
while(1);
```

- **Note:**

  DAvE doesn’t change code that is inserted in the ‘USER CODE’ sections if you let DAvE regenerate code. Therefore, whenever adding code to the generated code, write it into a ‘USER CODE’ section.

  If you want to change DAvE generated code or add code outside these ‘USER CODE’ sections you have to modify your changes after each time you let DAvE regenerate code!

Exercise MEM_1 - Complete generated code Files 'main.c'

- Edit 'main.c' - function 'main' (almost at the end)

```
// USER CODE BEGIN (Main,7)
void delay(void);

 // endless loop
while(1)
        {
        // set led off
        IO_vSetPin(IO_P3_7);
        // wait
        delay();
        // set led on
        IO_vResetPin(IO_P3_7);
        // wait
        delay();
        }
 // USER CODE END
```

Exercise MEM_1 - Complete generated code Files ‘main.c’

- At the end:

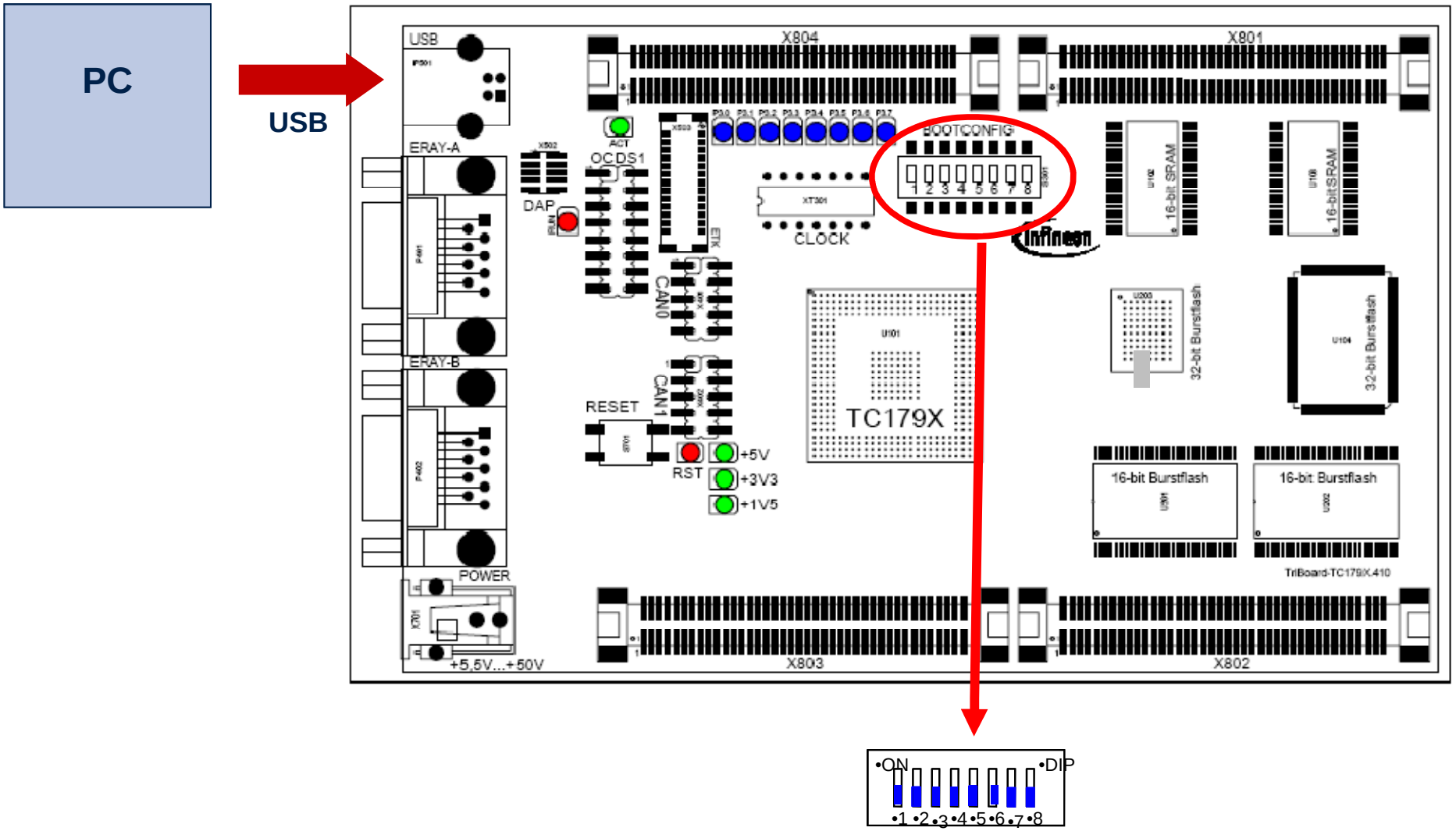
```
// USER CODE BEGIN (Main_General,10)
 void delay (void)
{
volatile int i,j,k;
for (i=0;i<5000;i++)
       for (j=0;j<200;j++)
         k = i+j;
}
// USER CODE END
```

Exercise MEM_1
Build the application

- Build the project
  - In the C/C++ Projects view, right-click on the project SPRAM and select **Build Project**. Alternatively, you can click the button. 
  - The Build process finishes successful.

Exercise MEM_1
Connect TriBoard

- Connect TriBoard to PC:



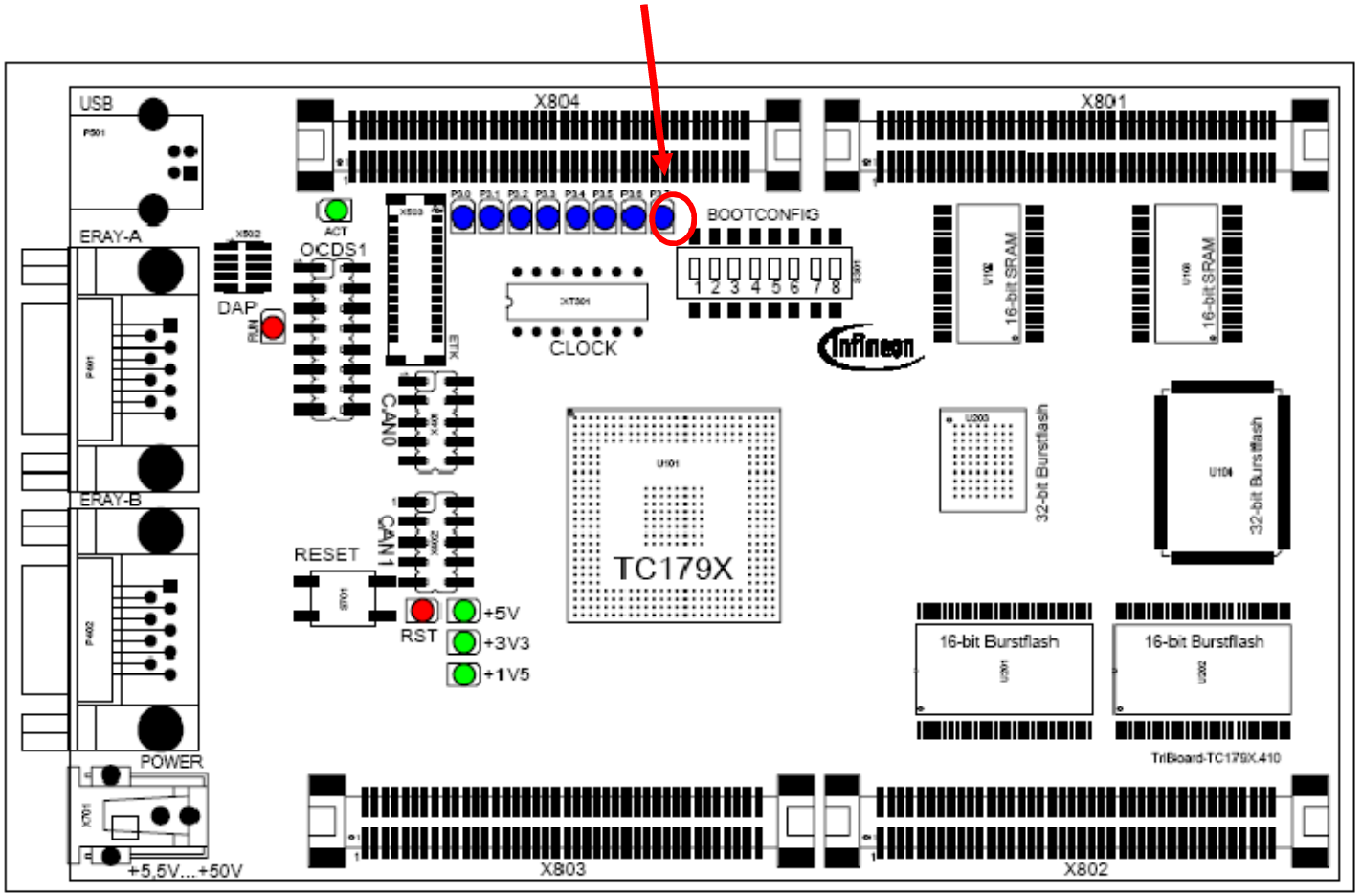
Configuration for [Internal Start from Flash] Boot mode

Exercise MEM_1
Run the application

- Press the blue reset button on the TriBoard.
  - From the Run menu select **Debug** or press **F11**. Alternatively you can click the button in the main toolbar. 
  - *The TASKING Debug perspective opens while the application has stopped before it enters main()*
  - In the Debug view, click on the (Resume) button. 

Exercise MEM_1 - Running the Program

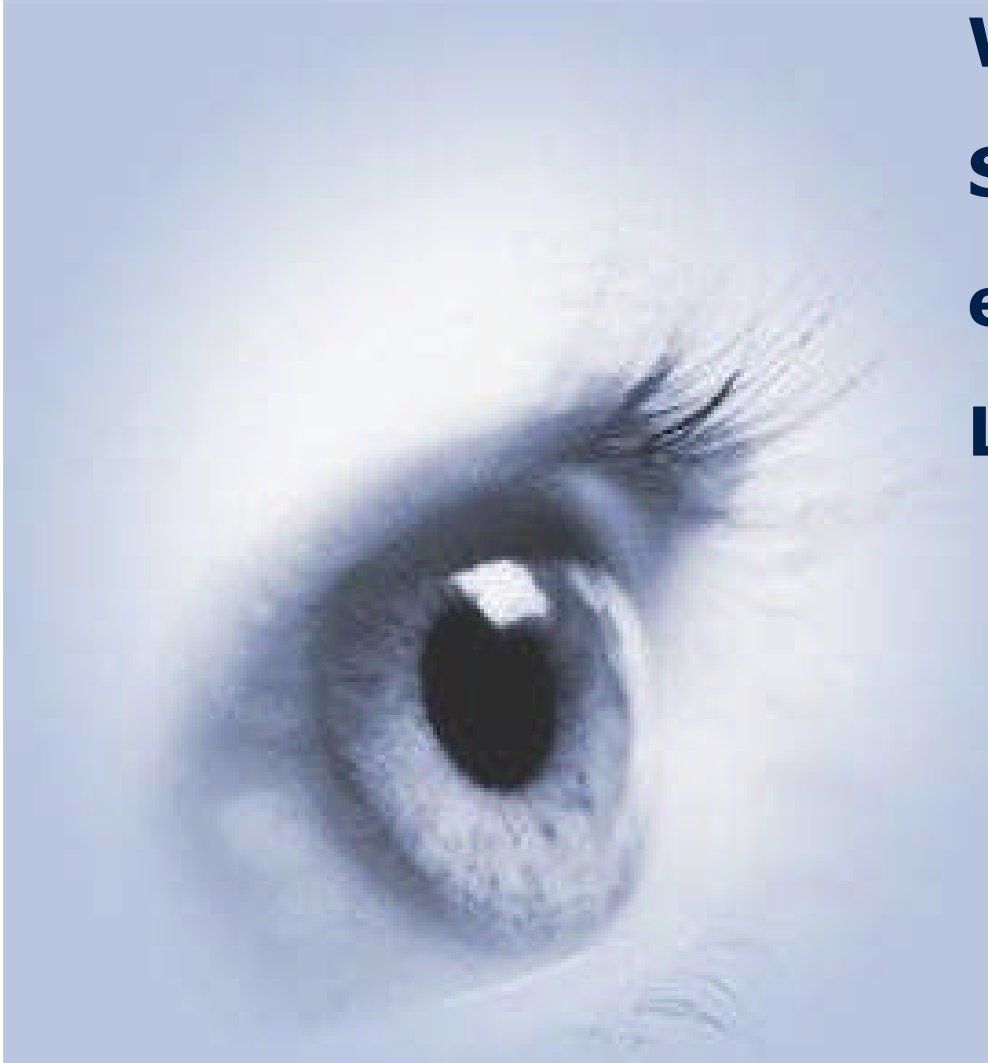
- Now, the LED on the TriBoard blinks very fast!!!



Exercise MEM_1 - End the Debug Session

- From the **Run** menu select **Terminate** or click on the **Terminate** button in the Debug view. 

- To remove the debug session from the Debug view, right-click on the debug session and select **Remove All Terminated** or click on the **Remove All Terminated Launches** button in the Debug view.



We create

Semiconductor Solutions,

enabling the Technology

Lifestyle of the Individual in

the 21st Century.

Never stop thinking.