

XC2287M HOT

CAPCOM_2 Solution

LED's toggling with CAPCOM2

Device: XC2287M-104F

Compiler: Tasking Viper 2.4r1

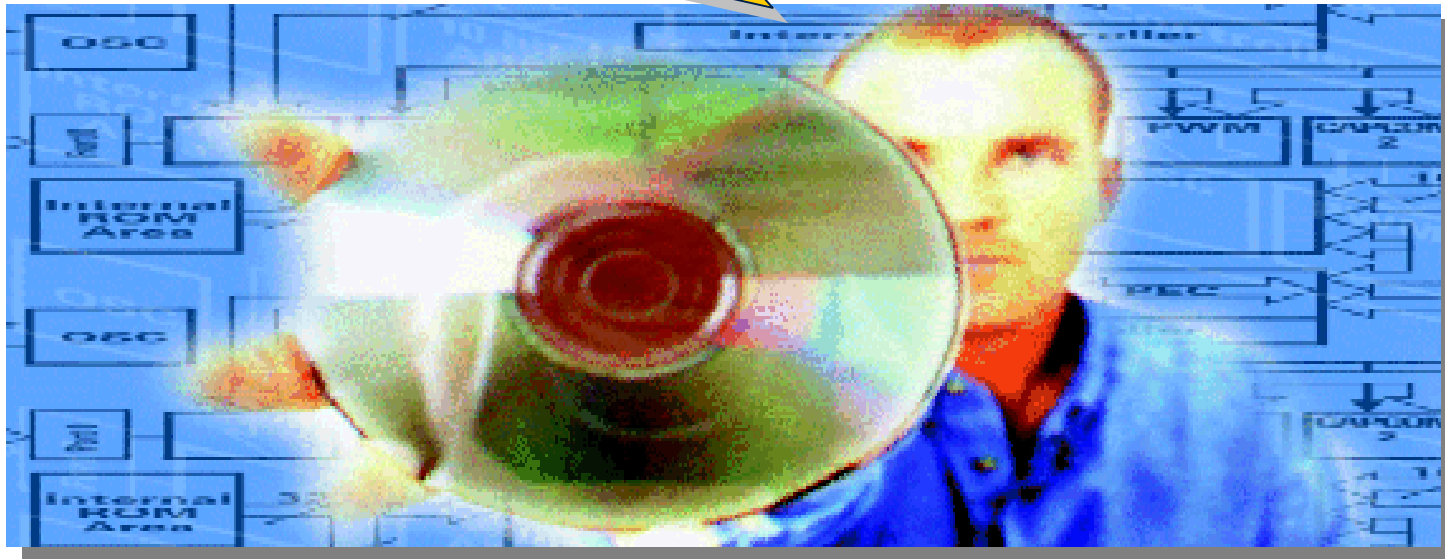
Code Generator: DAvE 2.1



Never stop thinking

Pin/LED's toggling using CAPCOM2

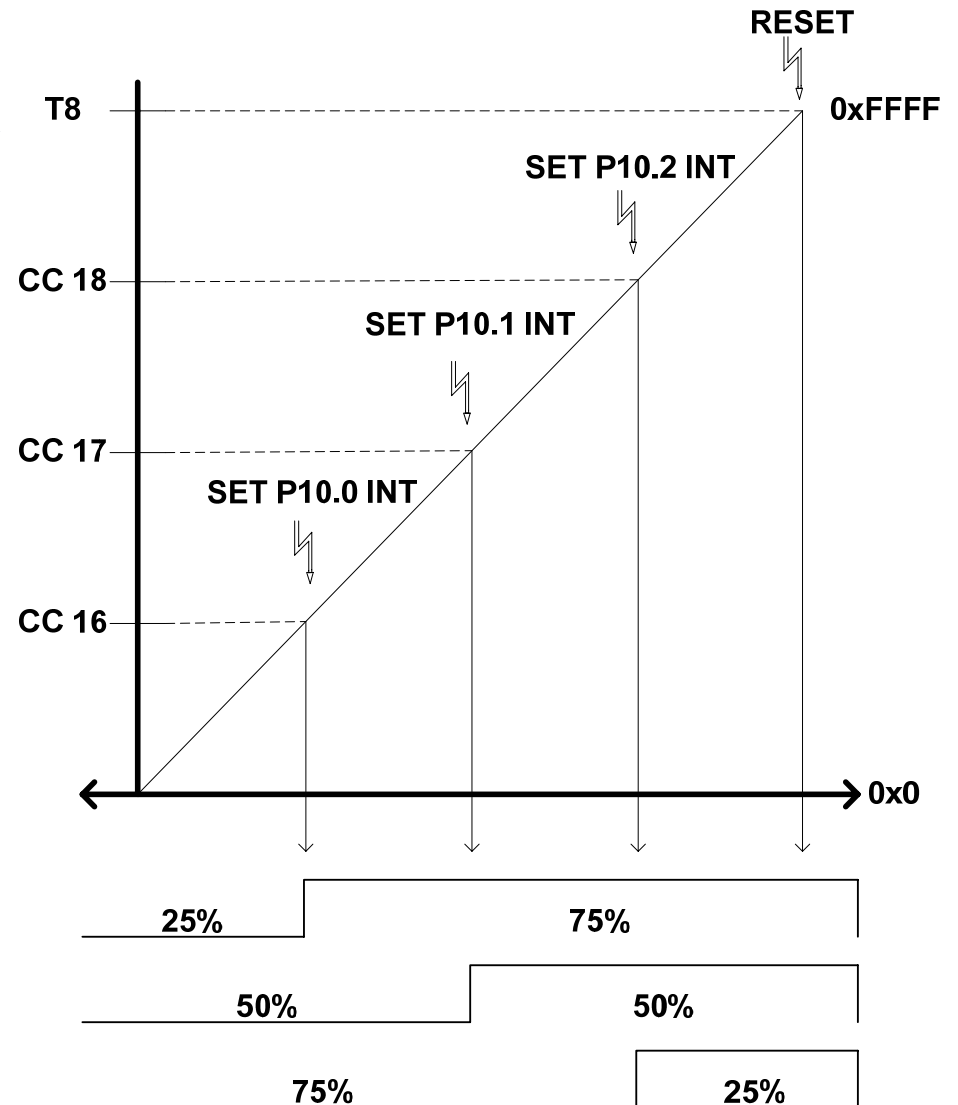
Let's get started now!



CAPCOM_2

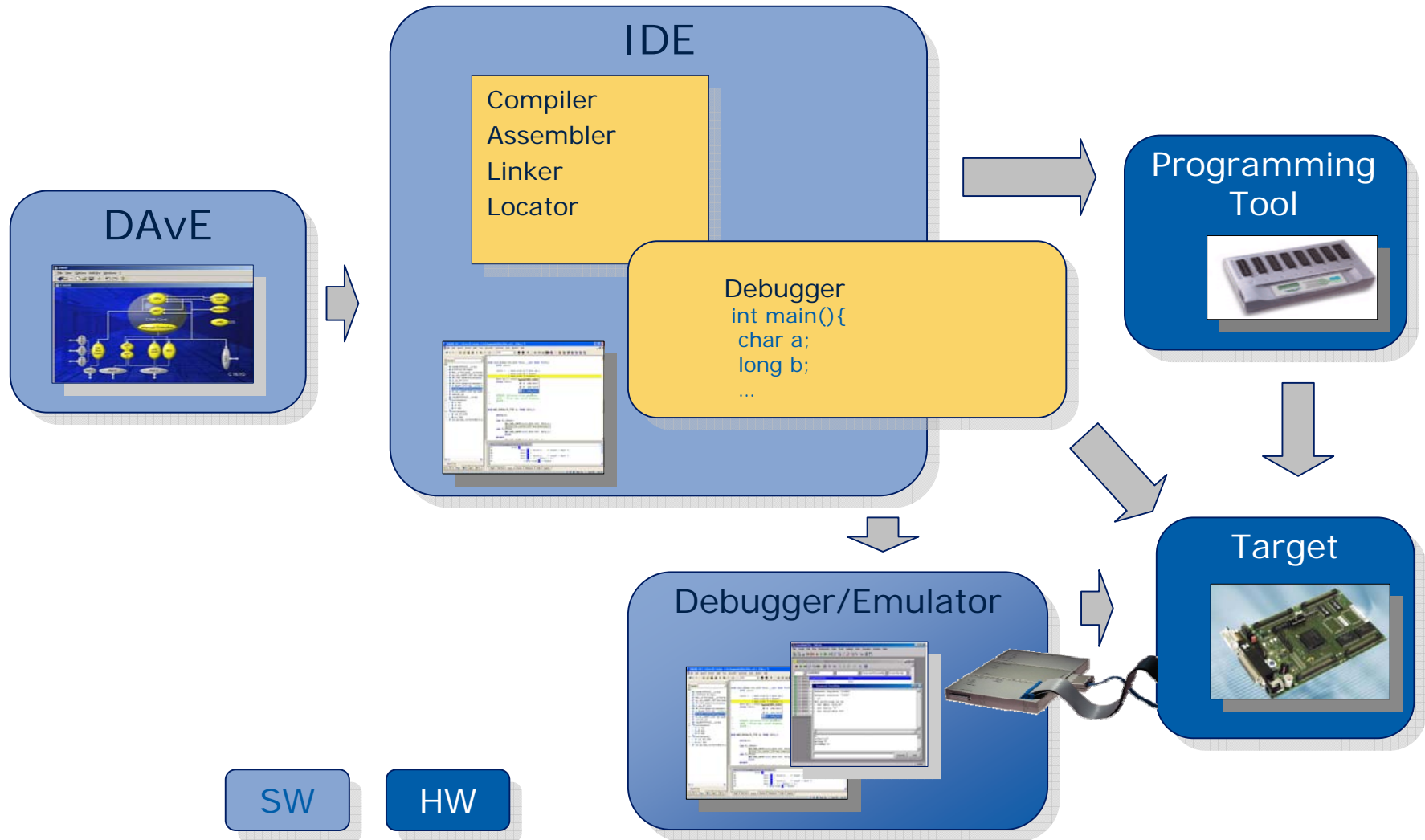
Objective

- Setup Prescaler and Timer T8 of GPT12
- Use Capture/Compare (CC16,CC17,CC18) interrupt to toggle port pins P10.0, P10.1, P10.2
- Using Timer8 interrupt to Reset the LED Pins



CAPCOM_2

Interaction of Development Tools



■ DAVe

- General Project Settings
- Setup Timer T8
- Port Settings
- Generate Code (Peripheral Initialization; C Code)

■ Compiler

- Import your DAVe Project
- Include Peripheral Initialization
- Add Application C Code (Interrupt routine)
- Compile Code

■ Debugger

- Download/Flash Compiled Code
- Execute/Debug Code

■ Start DAvE

- Click on the



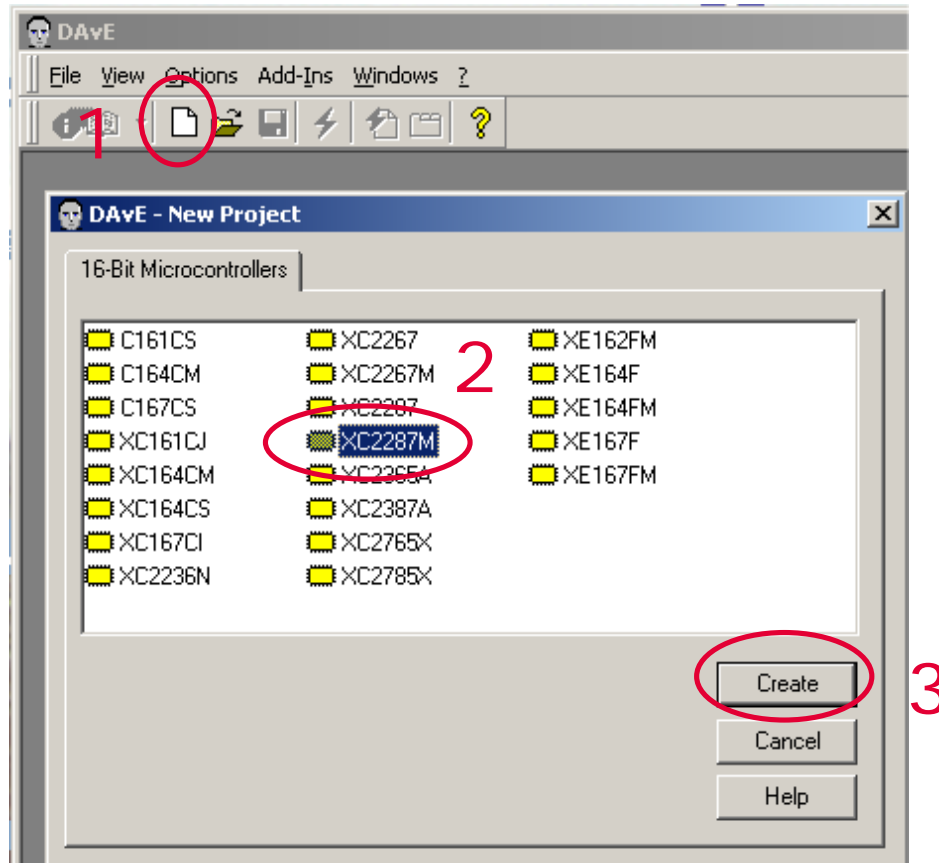
■ Create a new project (Startup Dialog pop up automatically)

- Click on 'Create a new project' or select File -> New
- Select microcontroller: 'XC2287M'



CAPCOM_2 - DAvE Configurations

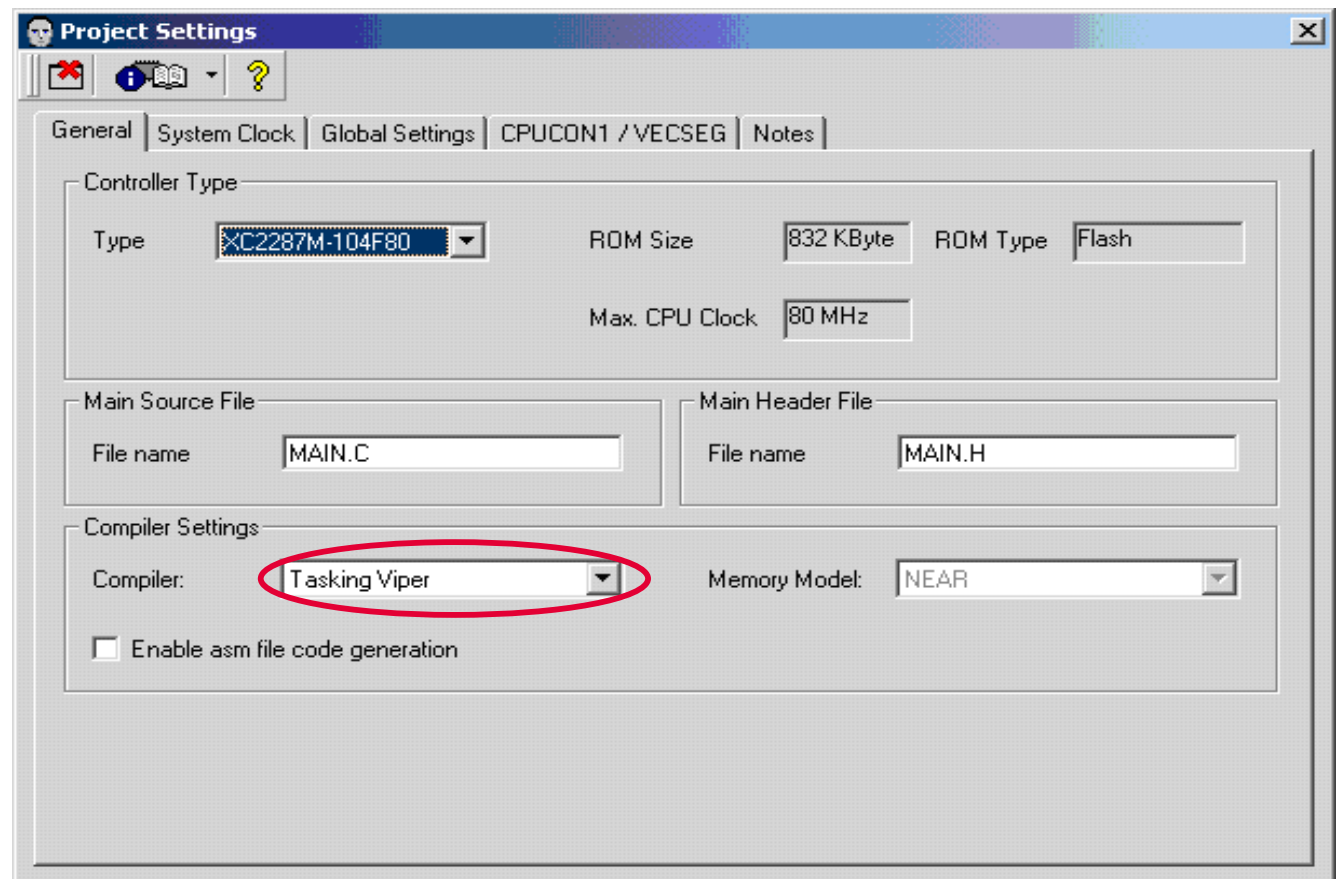
Select Device



CAPCOM_2 - DAVe Configurations

Project Settings

- Project Settings
- Close the window



CAPCOM_2 - DAVe Configurations

Save DAVe Project



■ Save your DAVe project



□ Path:

C:\IFX_HOT\XC2287M\Examples\CAPCOM_2

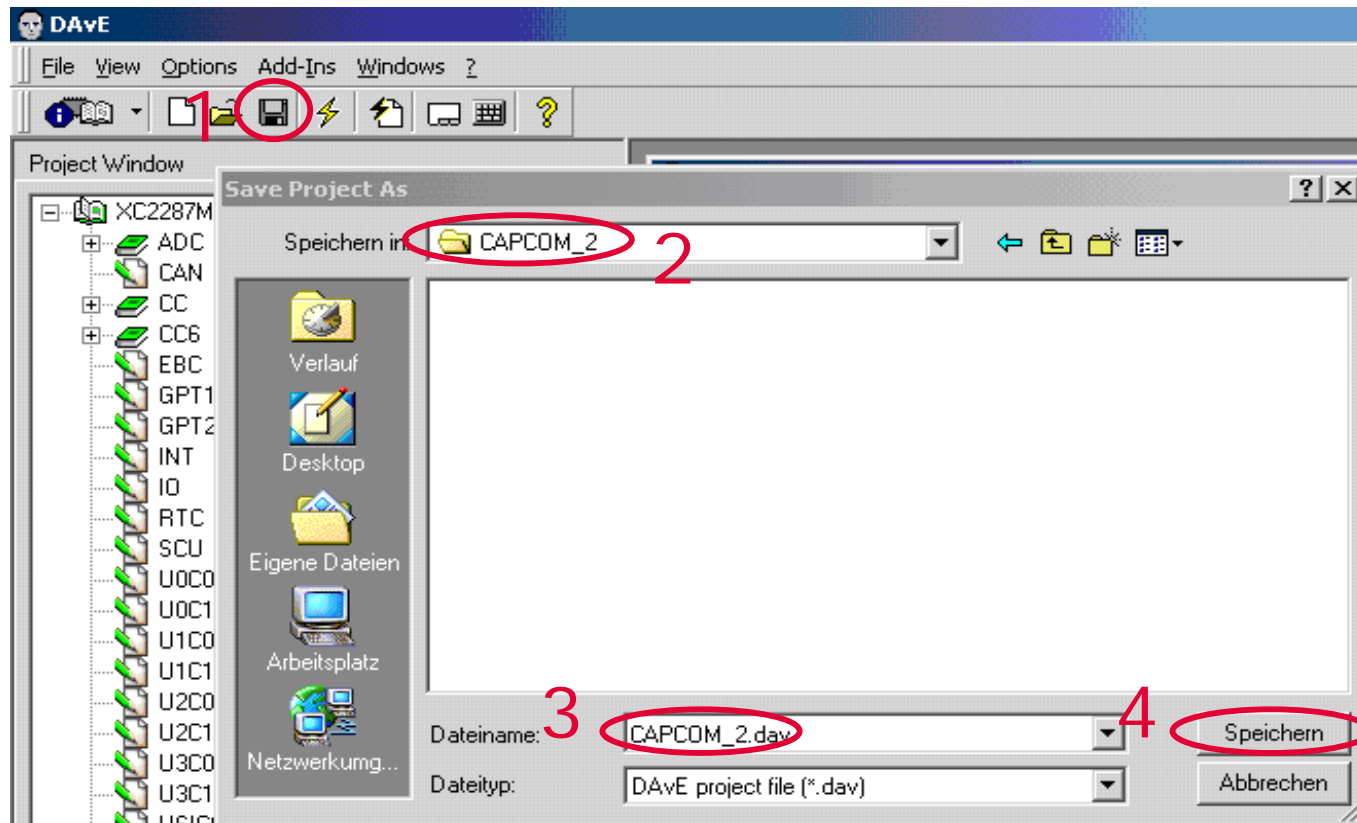
□ Project name:

CAPCOM_2\CAPCOM_2.dav

CAPCOM_2 - DAVe Configurations

Save DAVe Project

■ Save your DAVe Project File



■ Timer Settings

- Enable the module (enable clock path)
- Set Timer8 Configurations
- Configure interrupt routine
- Include timer setup function

- Close CAPCOM_2 setup window



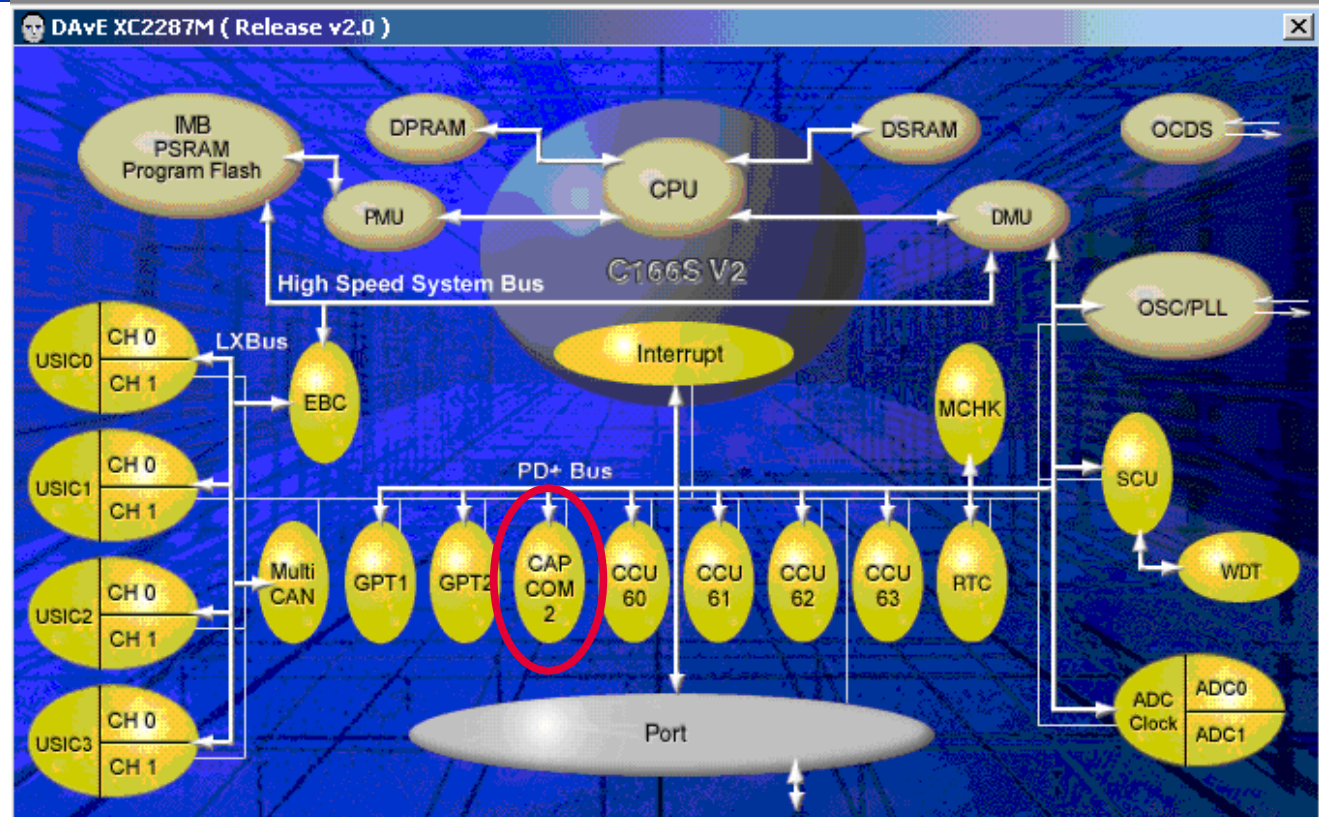
CAPCOM_2 - DAVe Configurations

Capcom_2 Settings

■ Timer Settings

□ CAPCOM2 :

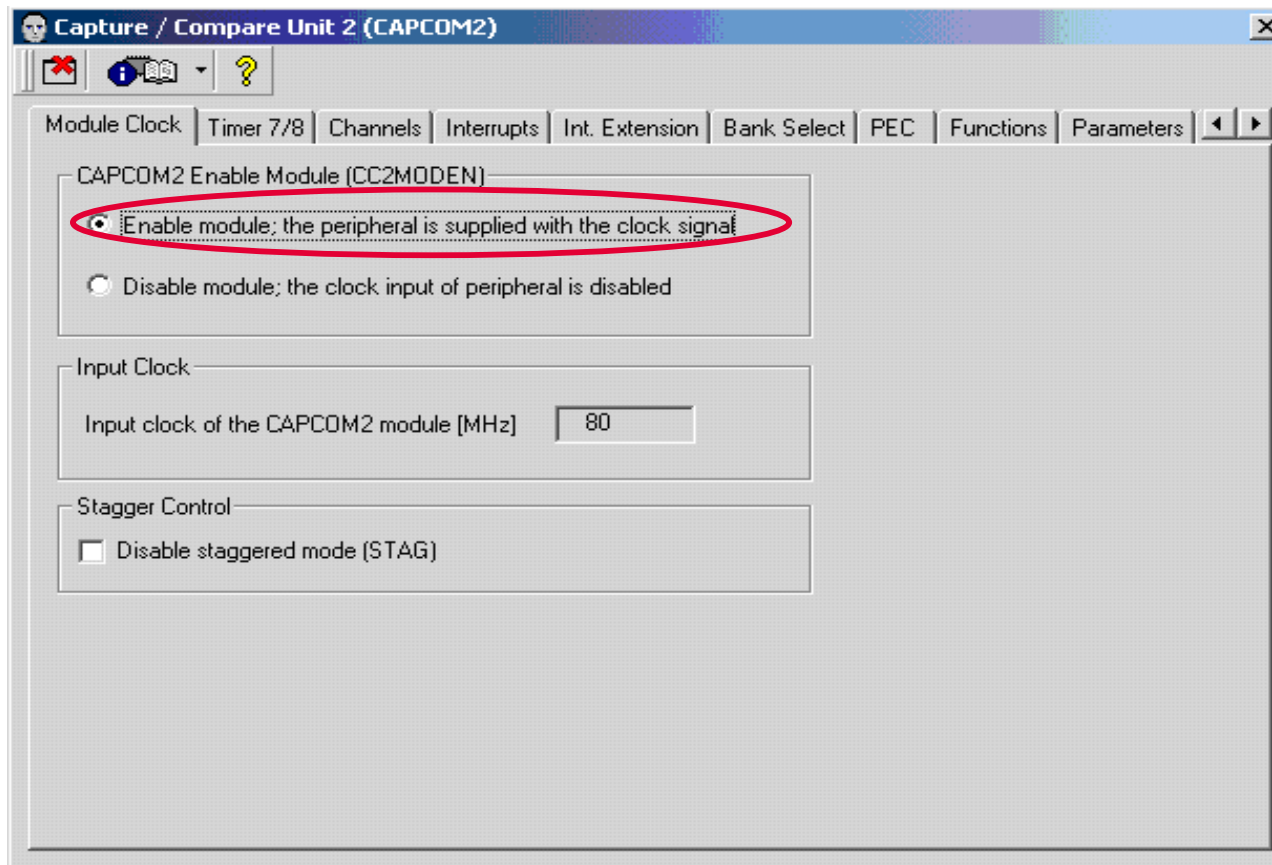
→ Click on



CAPCOM_2 - DAVe Configurations

Capcom_2 Settings

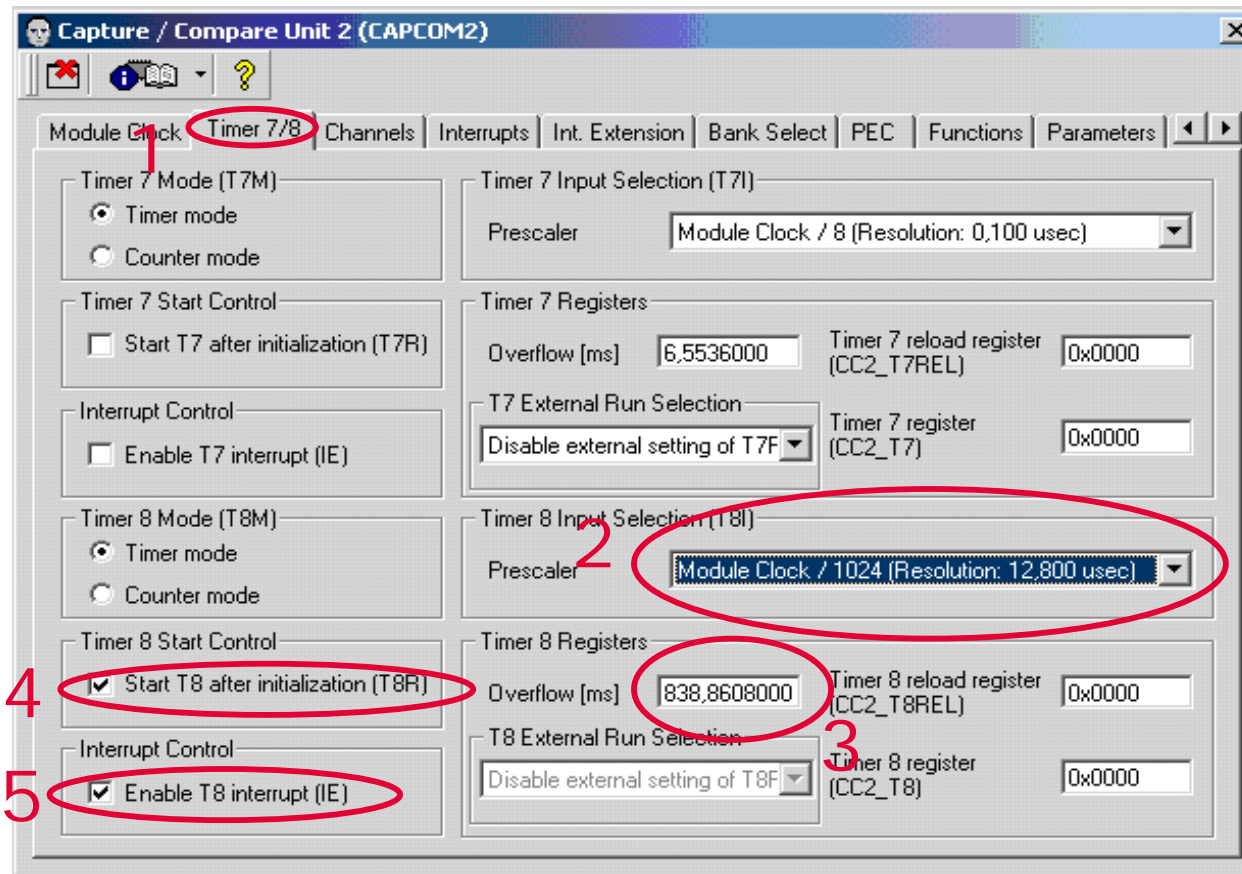
■ Module Clock settings



CAPCOM_2 - DAVe Configurations

Capcom_2 Settings

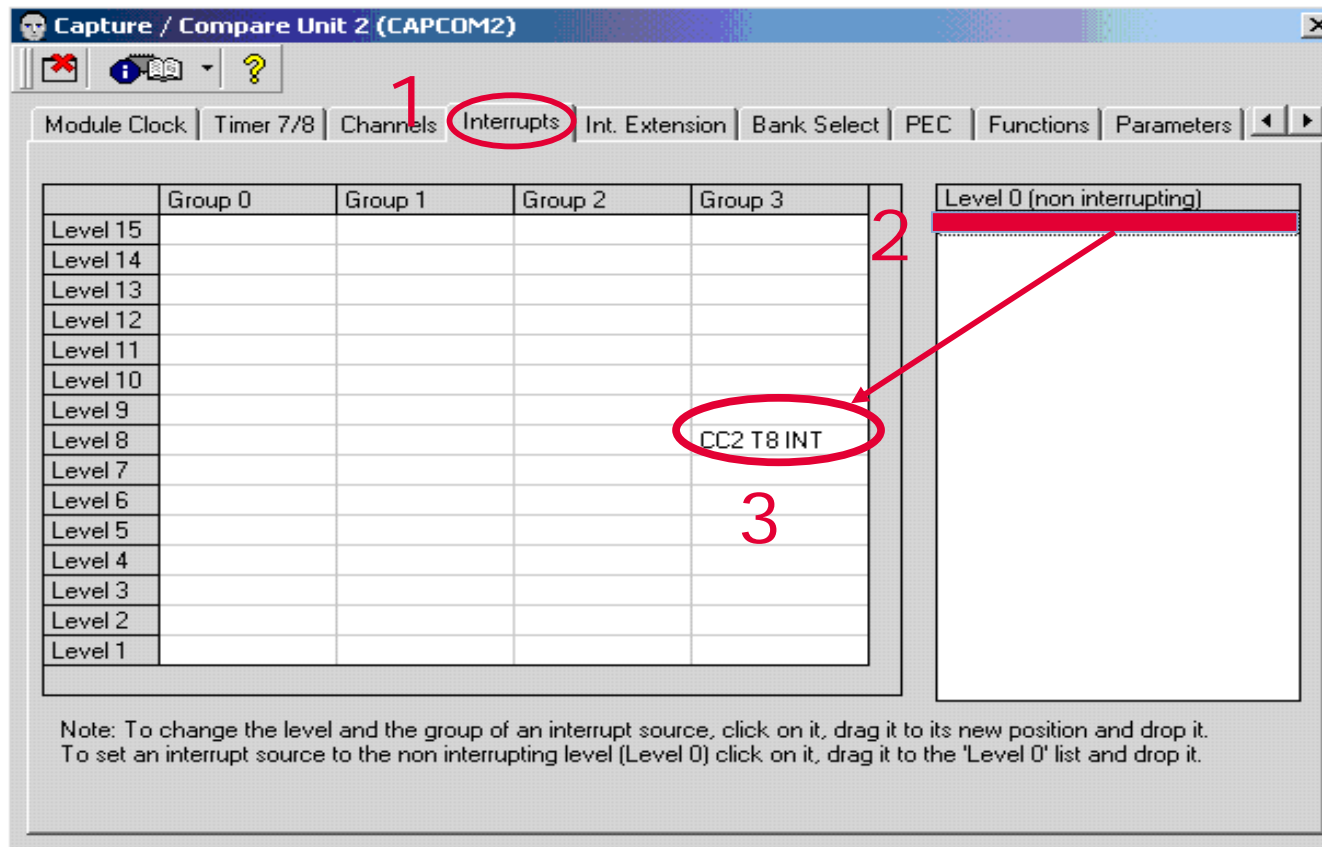
■ T8 settings



CAPCOM_2 - DAVe Configurations

CAPCOM_2 settings

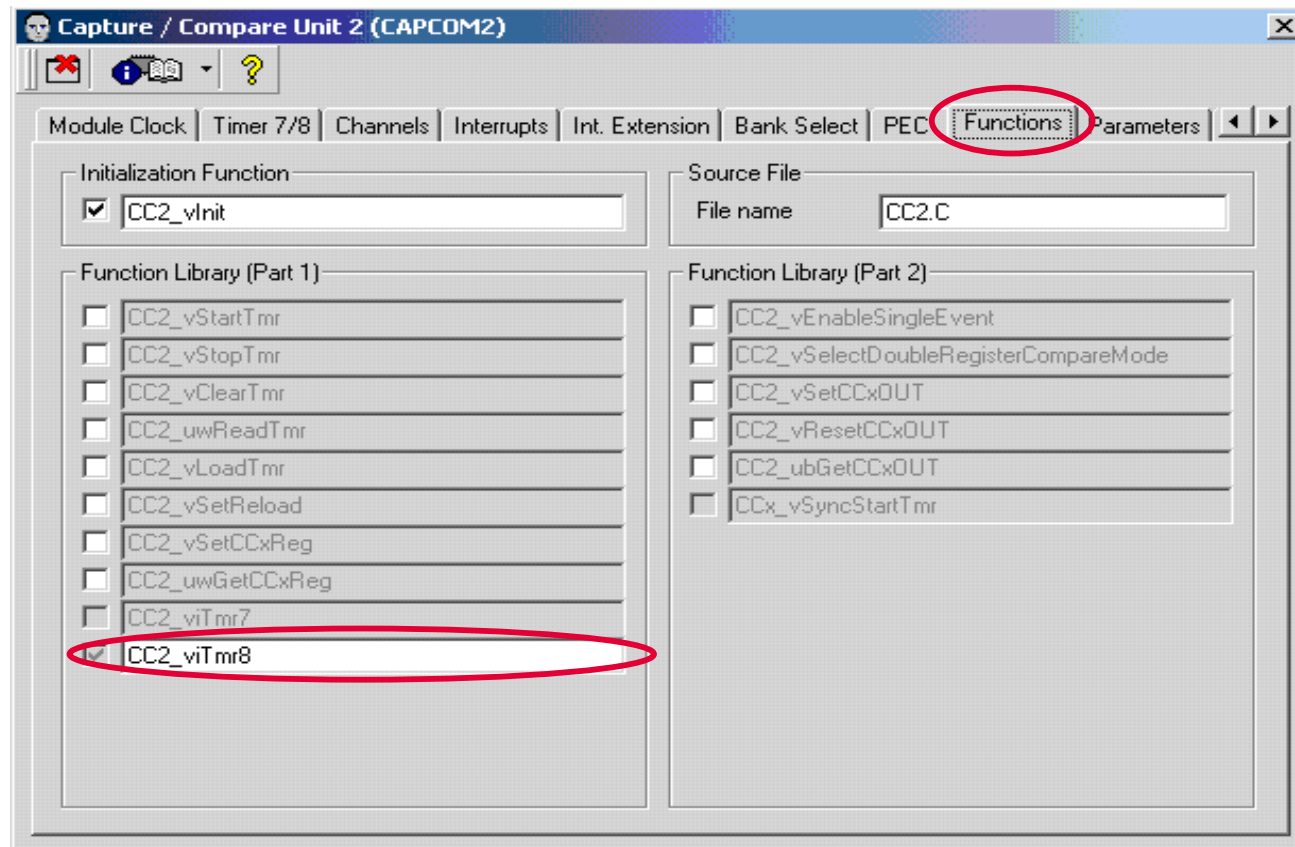
■ Interrupts settings



CAPCOM_2 - DAVe Configurations

Capcom2 Settings

■ Functions settings



■ Channel Settings

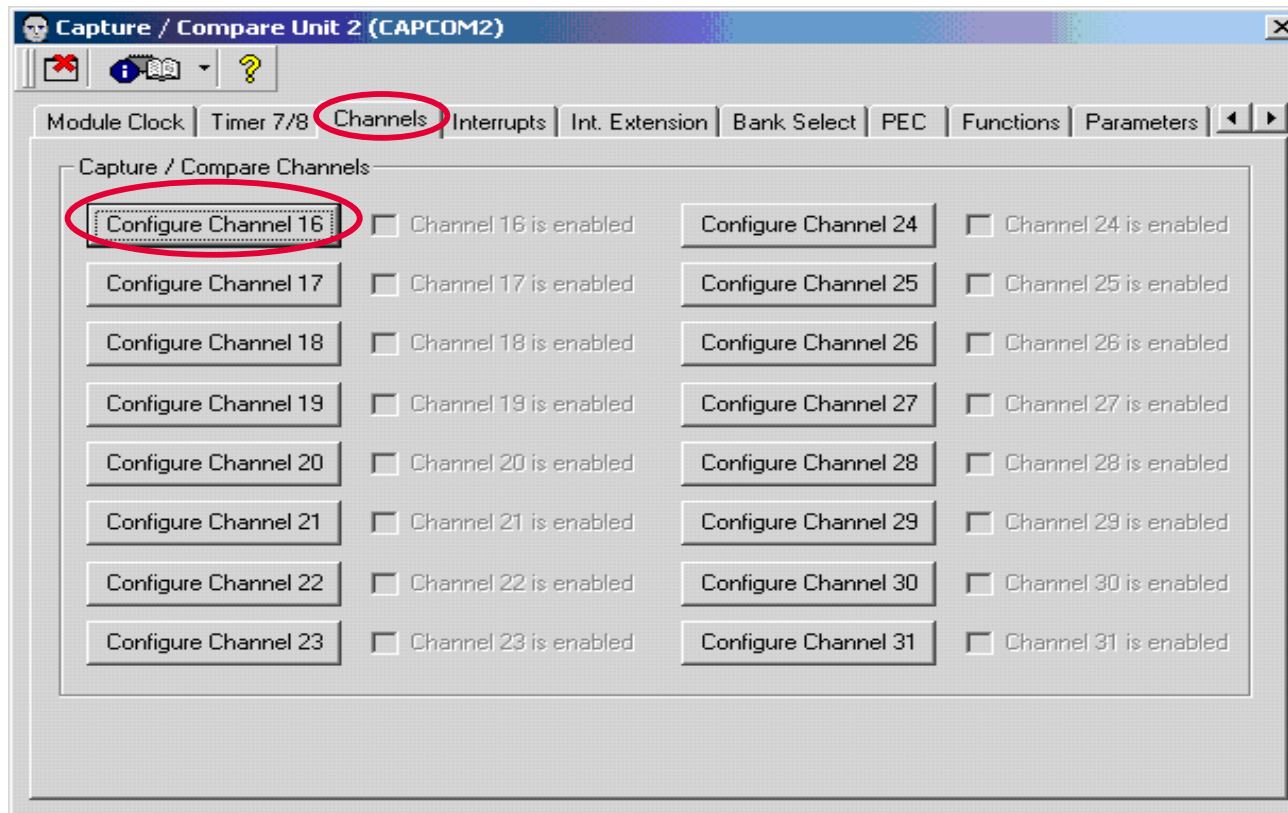
- Select 3 Channels
- Set Mode Selection for Capture/Compare
- Set the Control Register
- Configure interrupt routine
- Set Capture/Compare Registers

- Close Channel setup window

CAPCOM_2 - DAVe Configurations

Channel settings

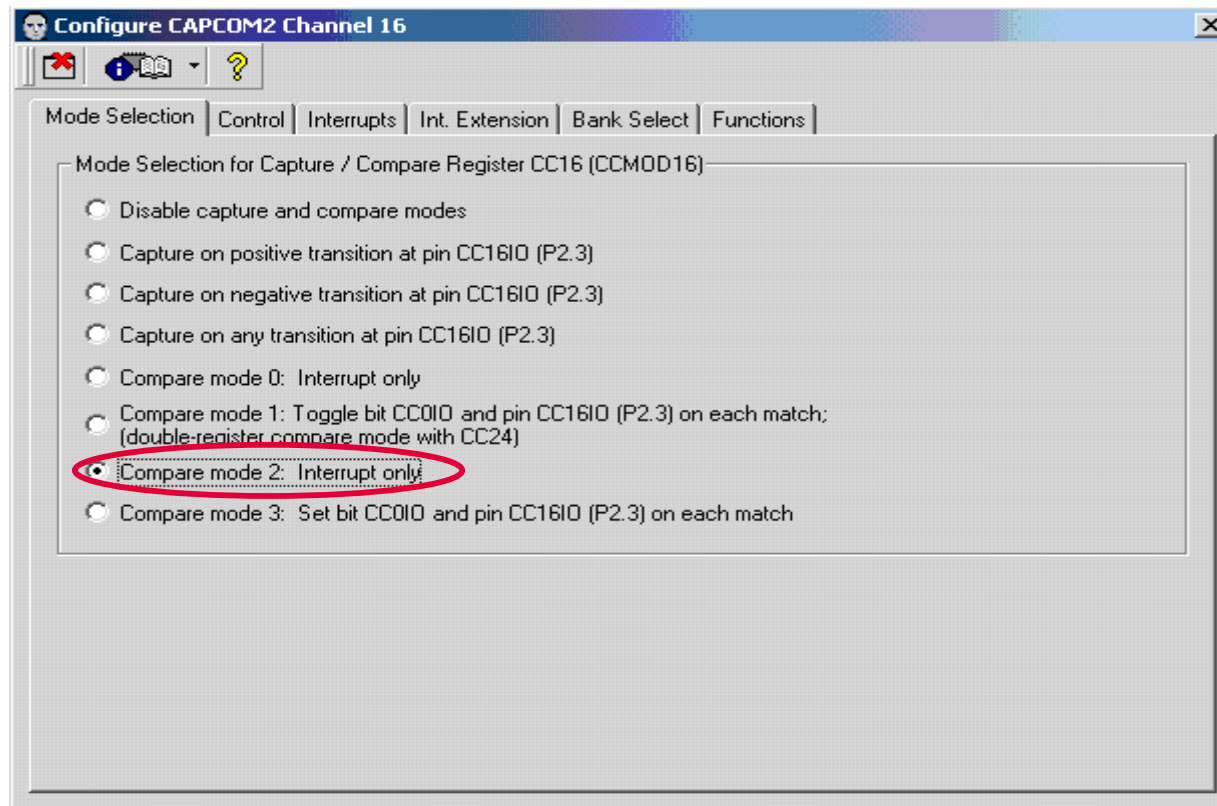
■ Choose Channel 16



CAPCOM_2 - DAVe Configurations

Channel settings

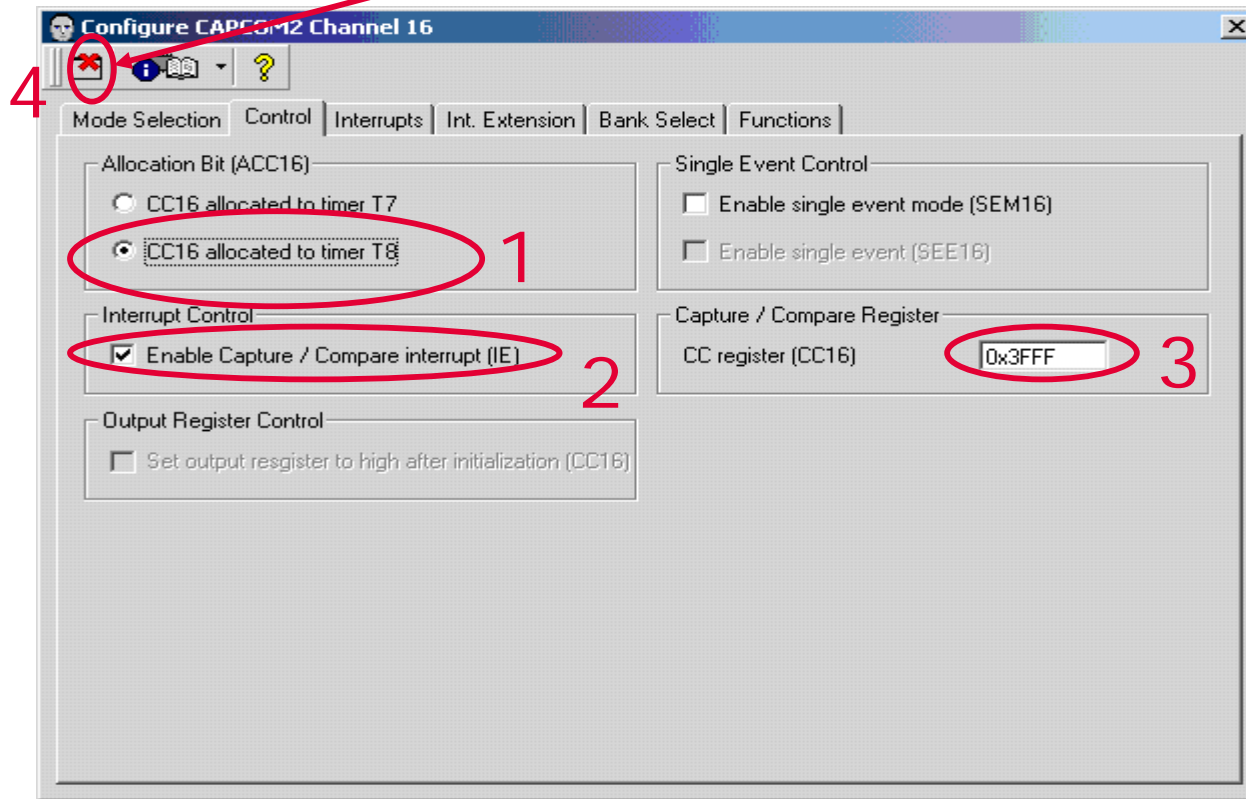
■ Configure Channel 16



CAPCOM_2 - DAVe Configurations

Channel settings

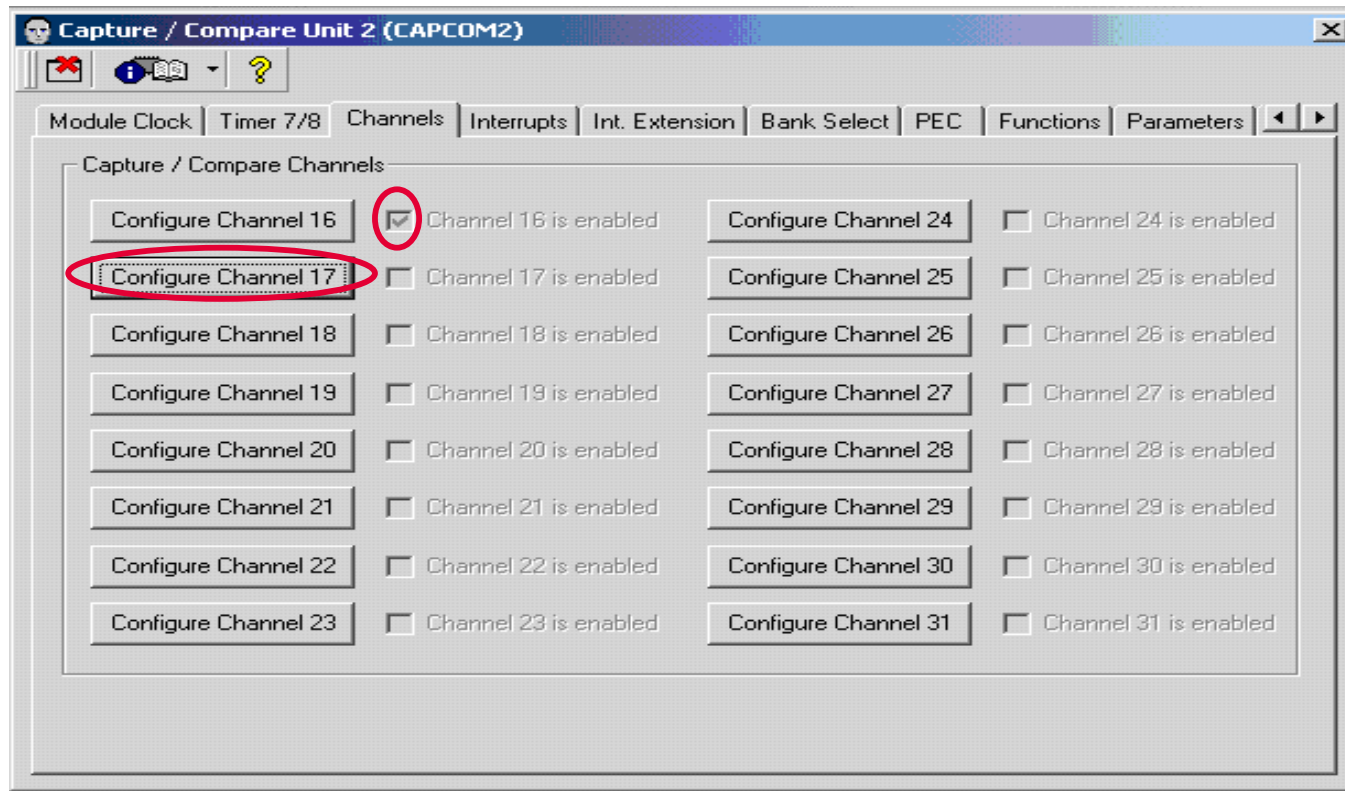
- Configure Control Register, close Configure Channel window



CAPCOM_2 - DAVe Configurations

Channel settings

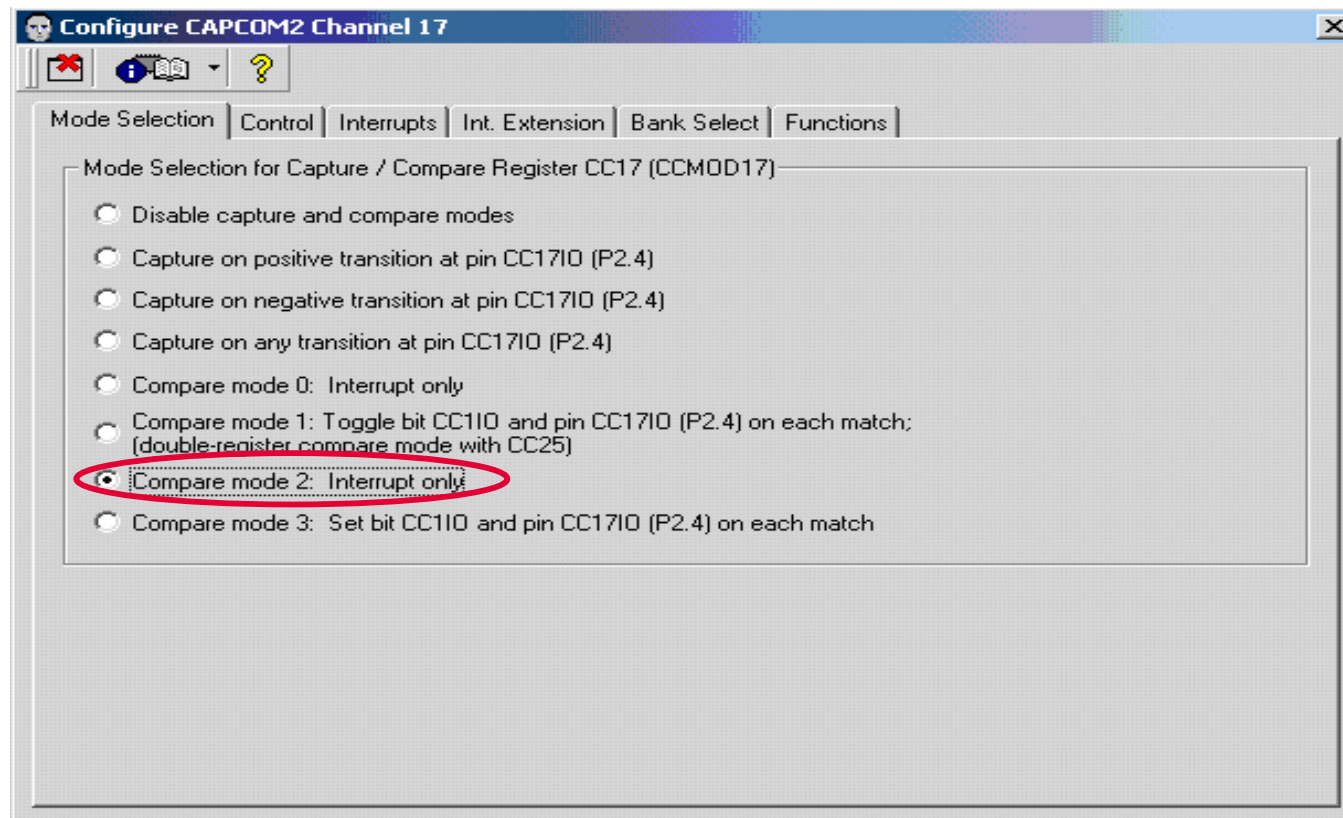
■ Choose Channel 17



CAPCOM_2 - DAVe Configurations

Channel settings

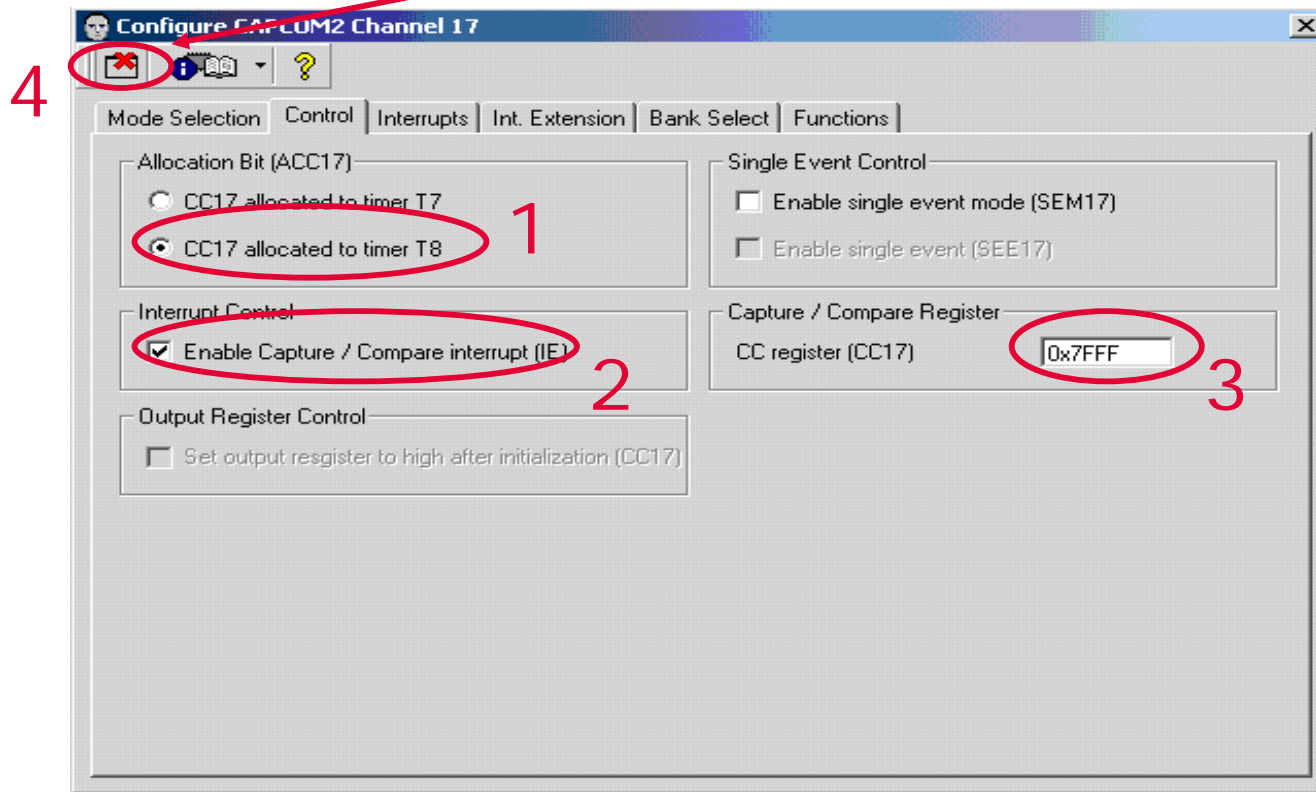
■ Configure Channel 17



CAPCOM_2 - DAVe Configurations

Channel settings

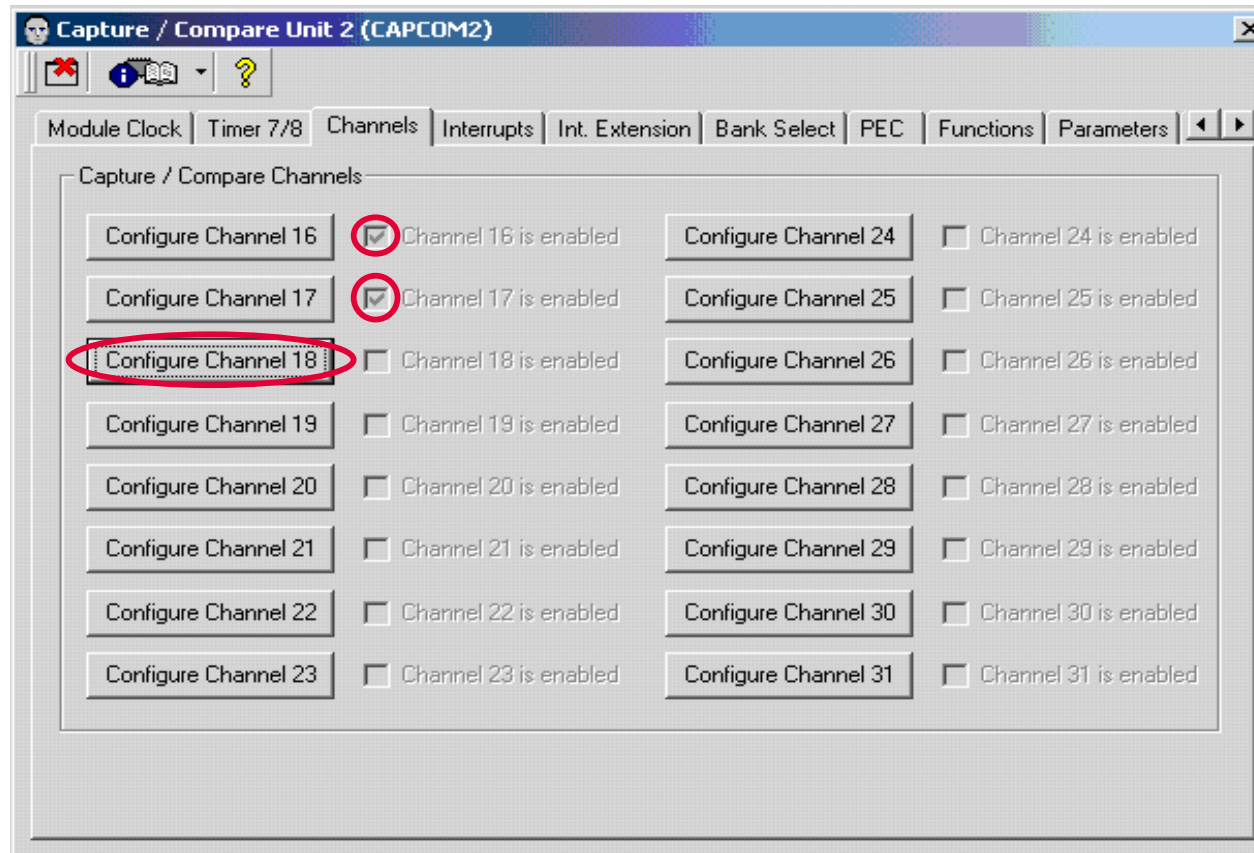
- Configure Control Register, close Configure Channel window



CAPCOM_2 - DAVe Configurations

Channel settings

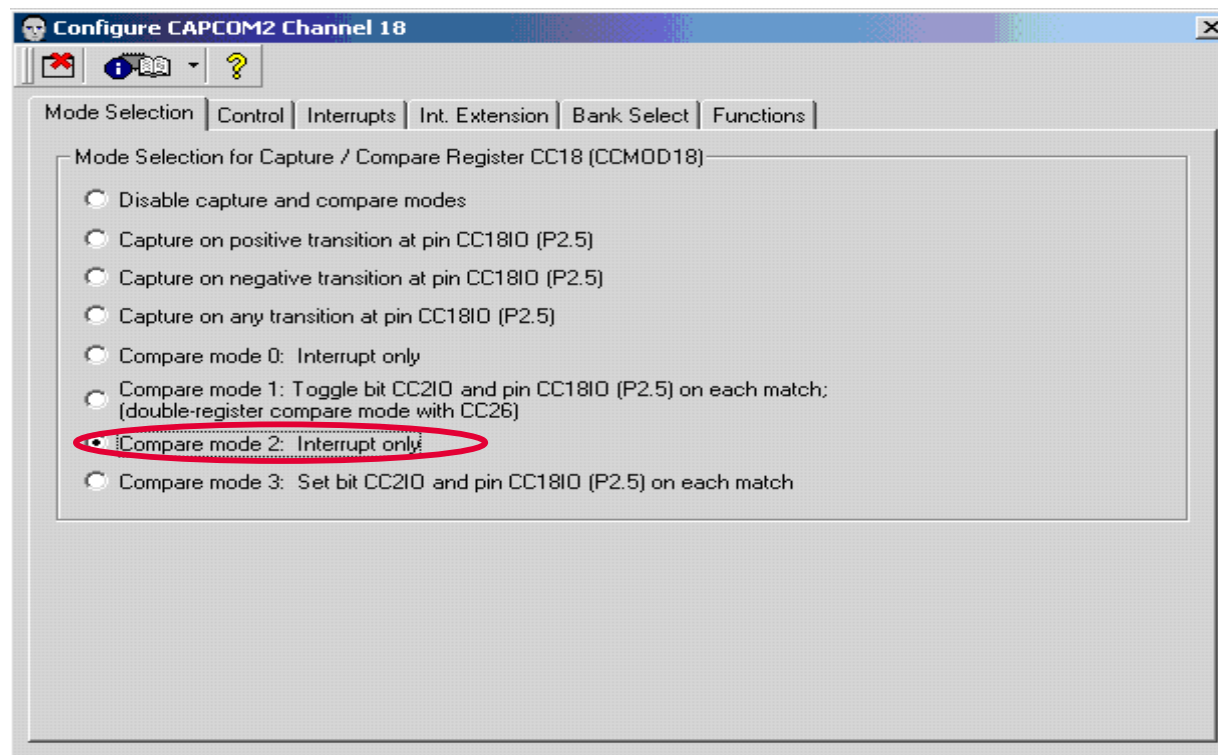
■ Choose Channel 18



CAPCOM_2 - DAVe Configurations

Channel settings

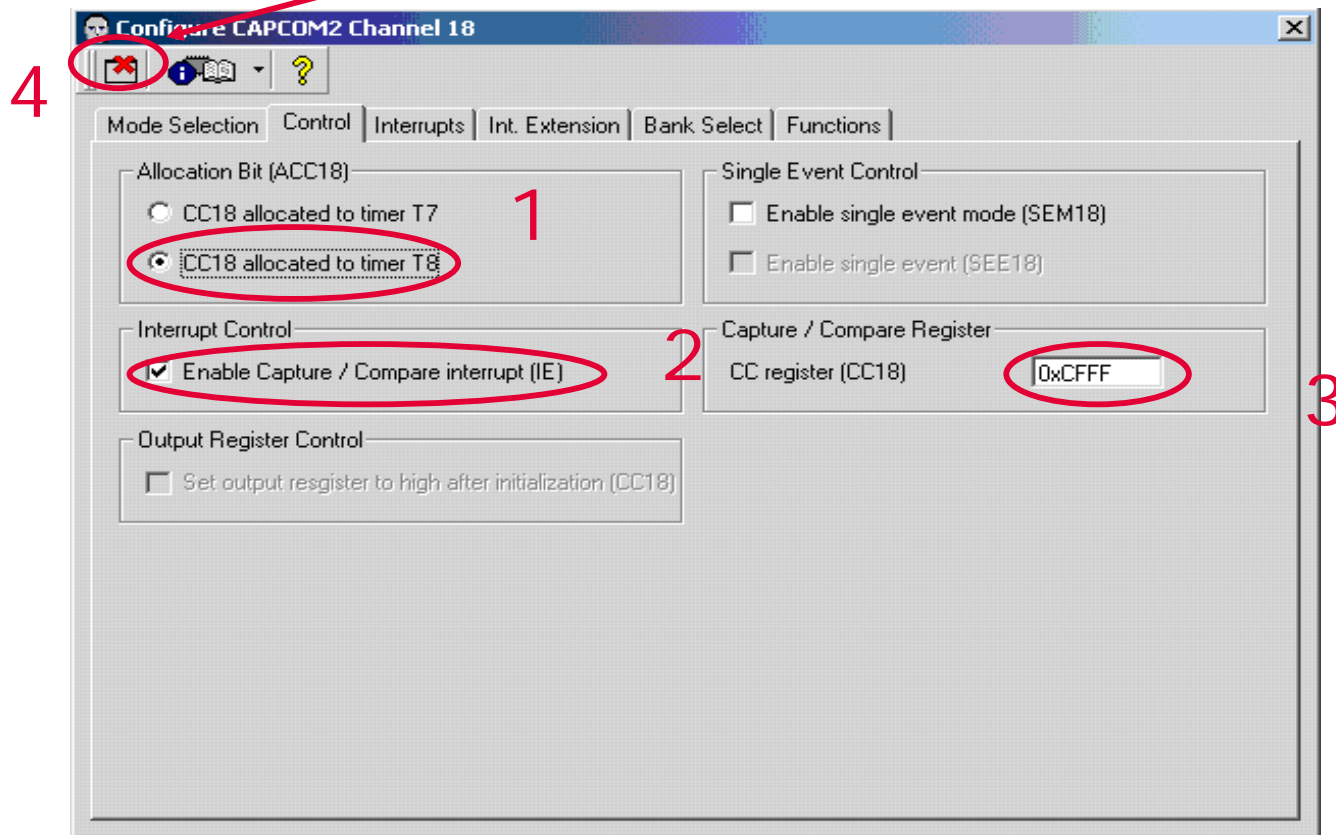
■ Configure Channel 18



CAPCOM_2 - DAVe Configurations

Channel settings

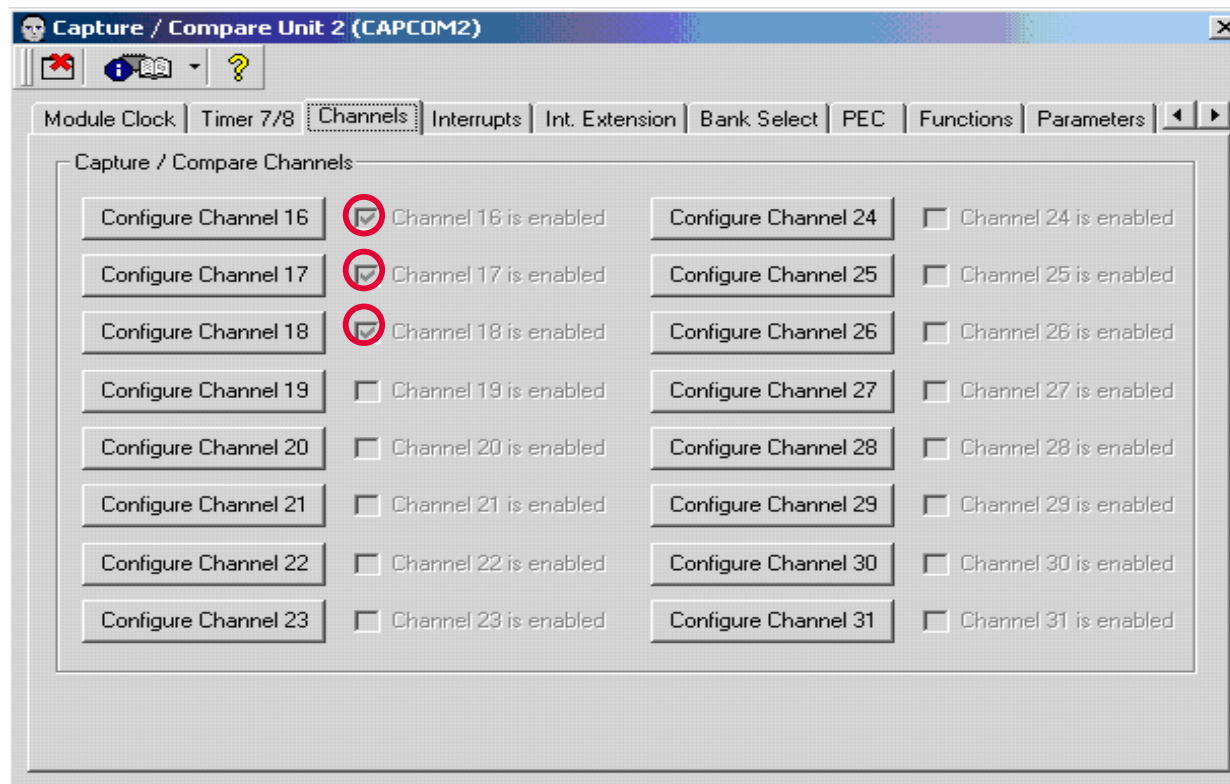
- Configure Control Register, close Configure Channel window



CAPCOM_2 - DAVe Configurations

Channel settings

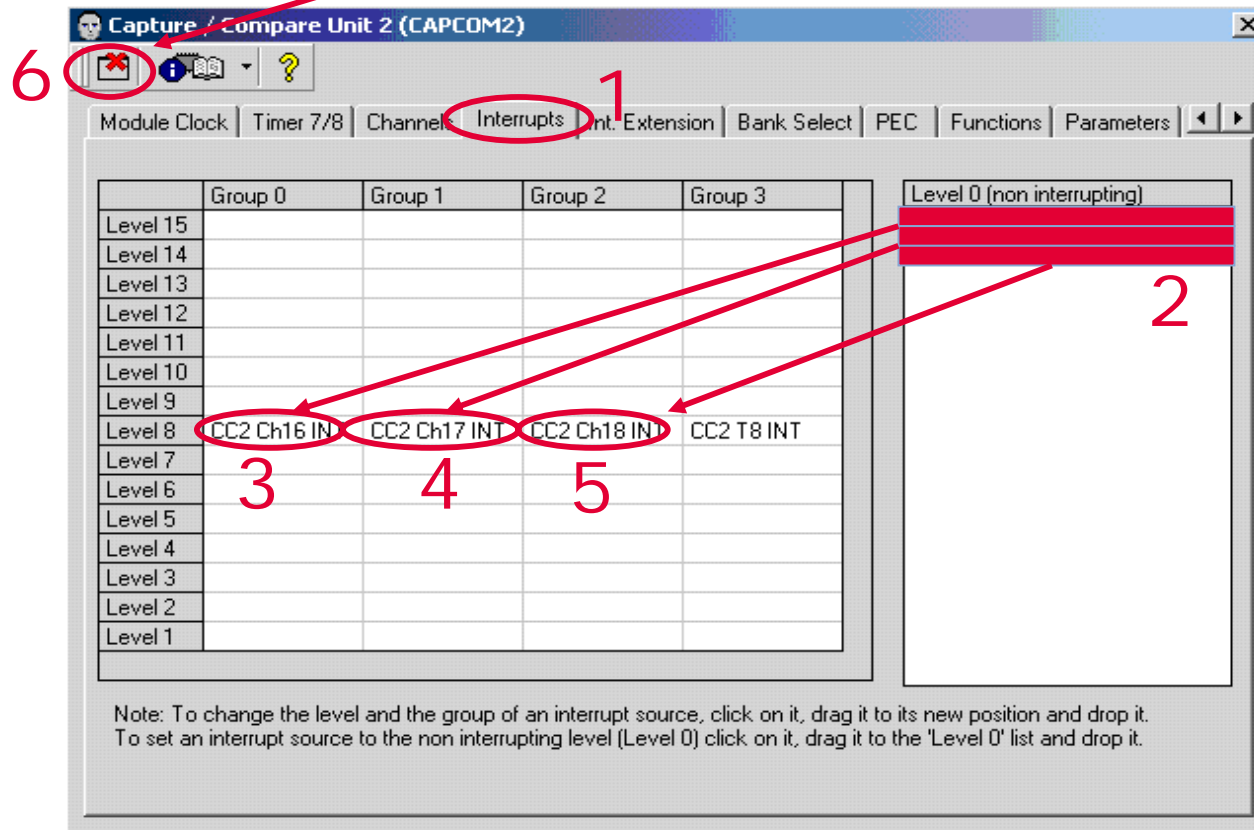
- Channel 16, 17, 18 are selected




CAPCOM_2 - DAVe Configurations

Channel settings

- Interrupts settings, close Interrupt Capture/Compare window



■ Port Settings

- LED's connected to P10.0, P10.1 and P10.2
- Use P10.0, P10.1 and P10.2 as general outputs
- Include port setup function
- Include port SetPin and ResetPin function
- Close port setup window 

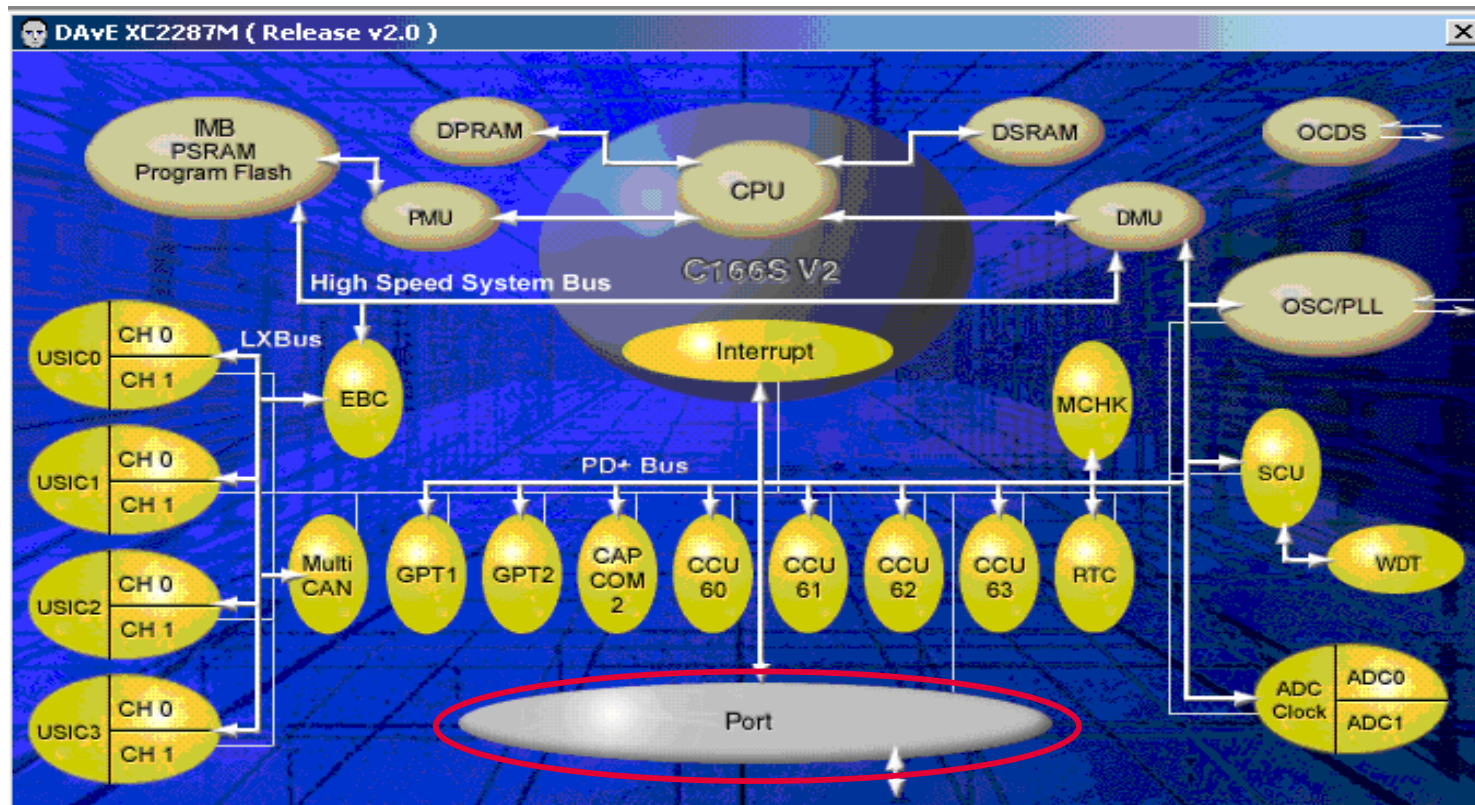
CAPCOM_2 - DAVe Configurations

Port settings

■ Port Settings

□ Port:

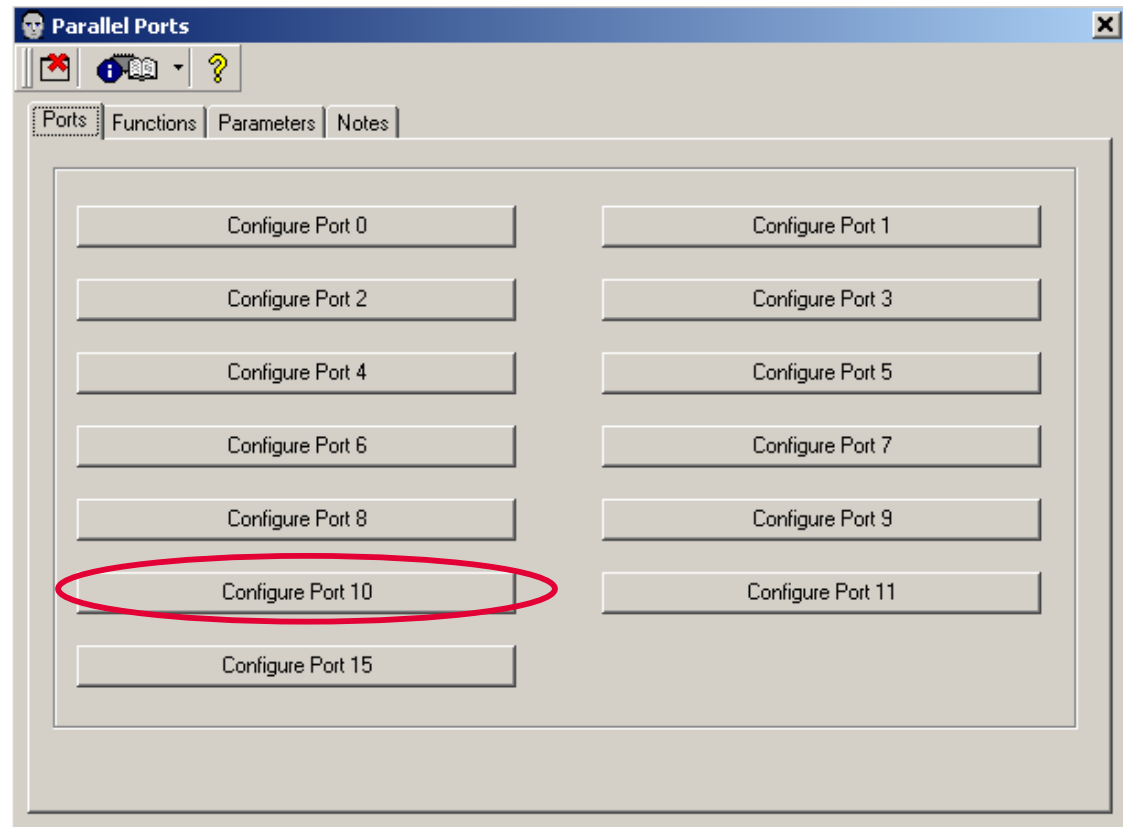
→ Click on



■ Port Settings

□ Ports:

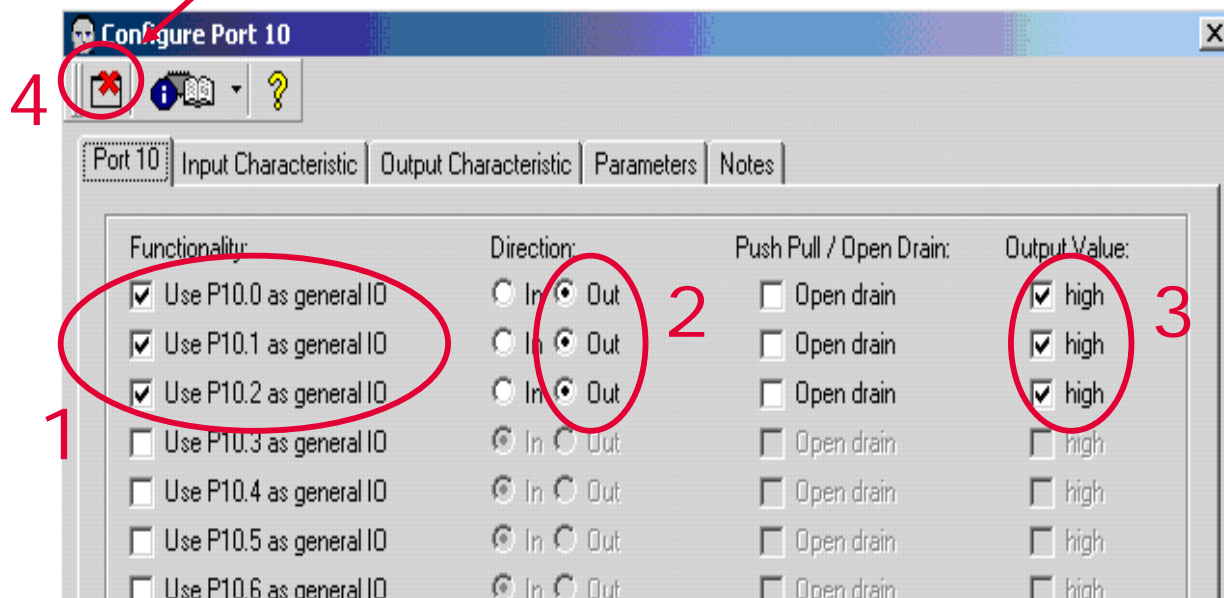
- Configure Port 10



■ Configure Port 10

□ Port 10:

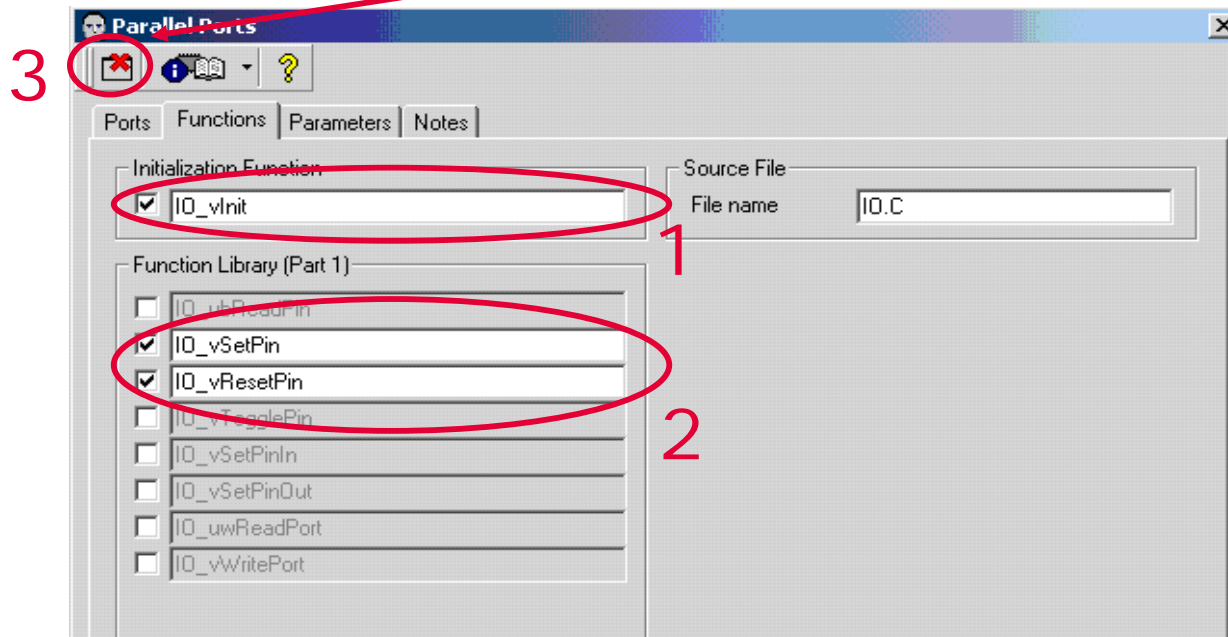
- Use P10.0, P10.1 and P10.2 as general IO
- Set Direction to Out, Set Output Value to high
- Close the window



■ Parallel Ports

□ Functions:

- Include 'IO_vInit'
- Include 'IO_vSetPin', 'IO_vResetPin', close the window



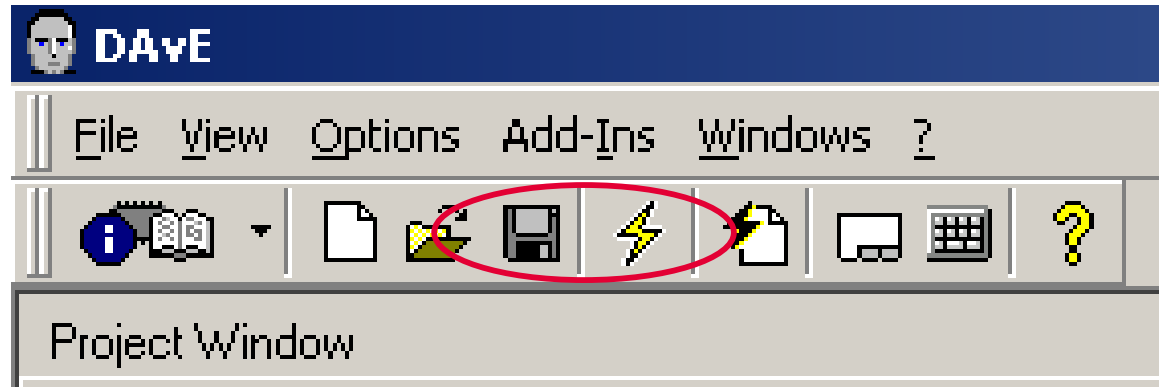
CAPCOM_2 - DAVe Configurations

Save DAVe Project and Generate Code

- Save your DAVe Project File



- Generated Code



CAPCOM_2 - DAVe Configurations

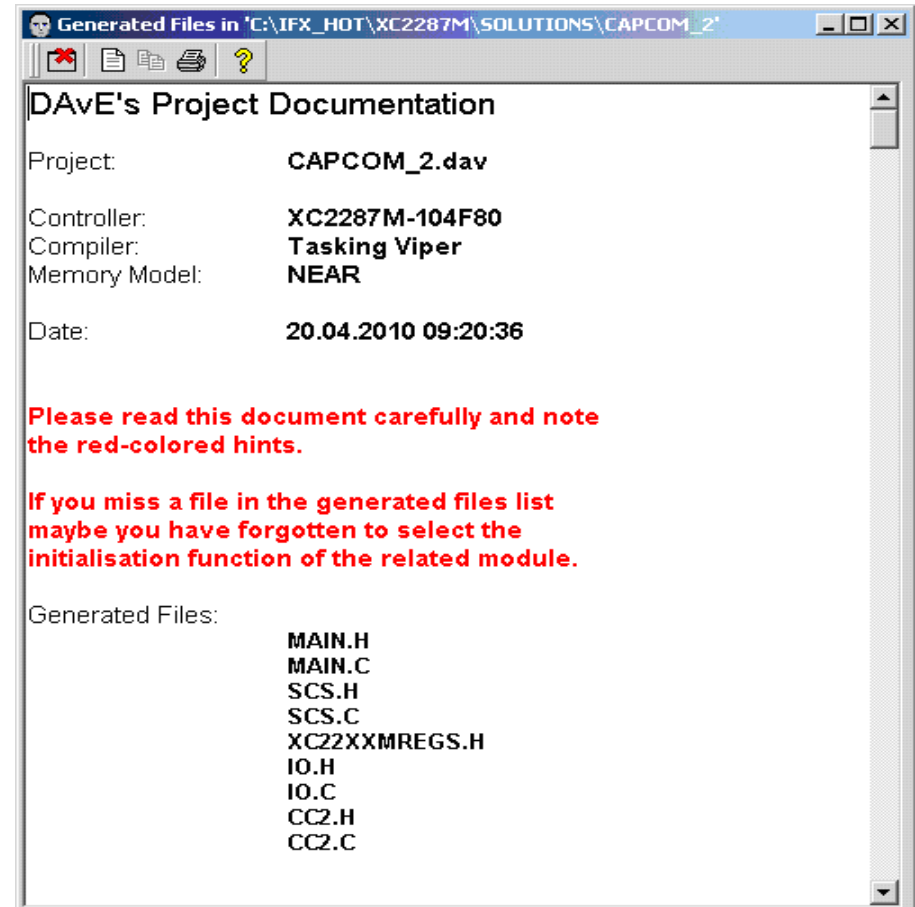
Generated Code

■ DAVe generated code files are

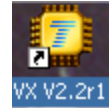
- 'MAIN.c', 'MAIN.h'
- 'SCS.c', 'SCS.h'
- 'IO.c', 'IO.h'
- 'CC2.c', 'CC2.h'
- 'XC22xxMREGS.h'

□ 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

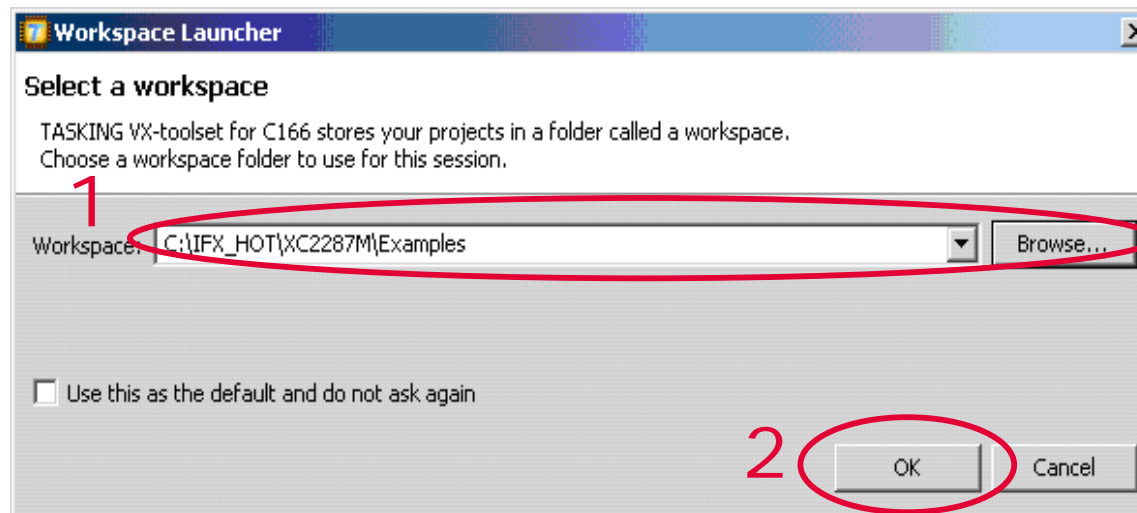


- Start Tasking Viper Tool Set



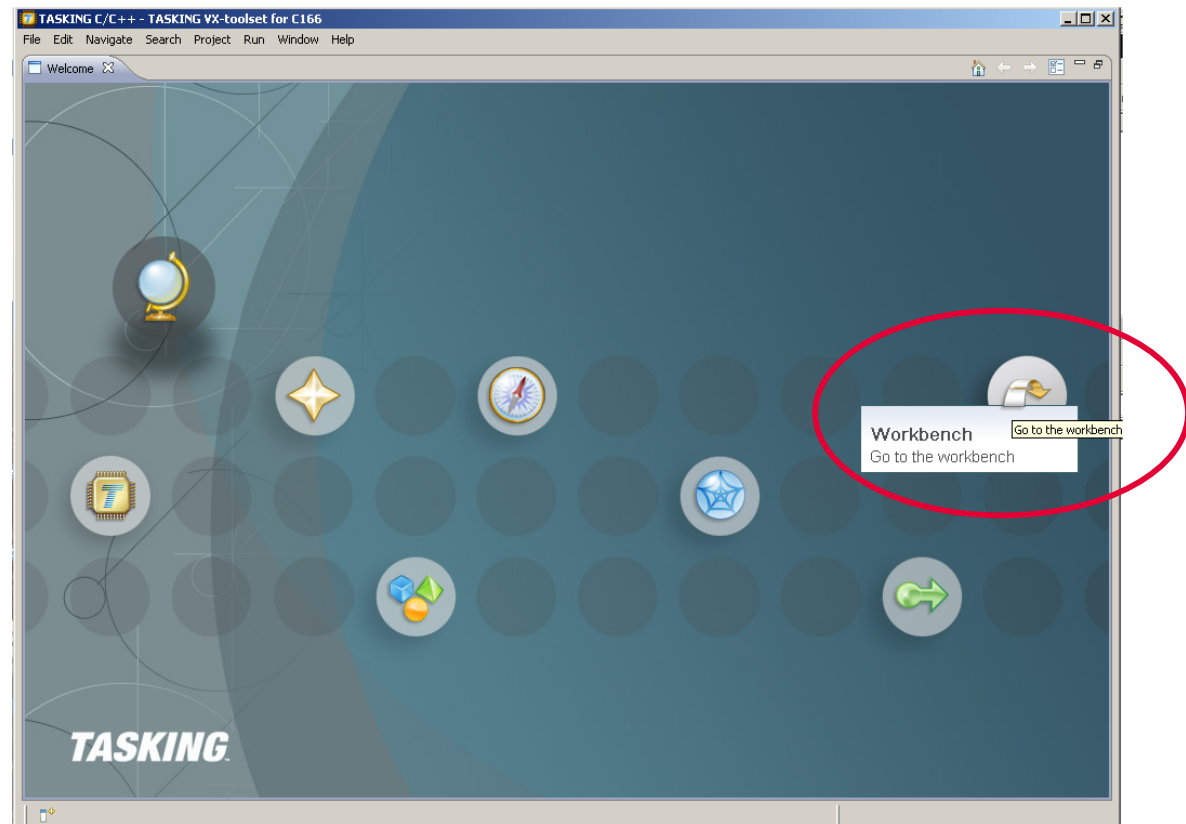
- Open Project Work Space

- Browse to “c:\IFX_HOT\XC2287M\Examples”
- Click ‘OK’



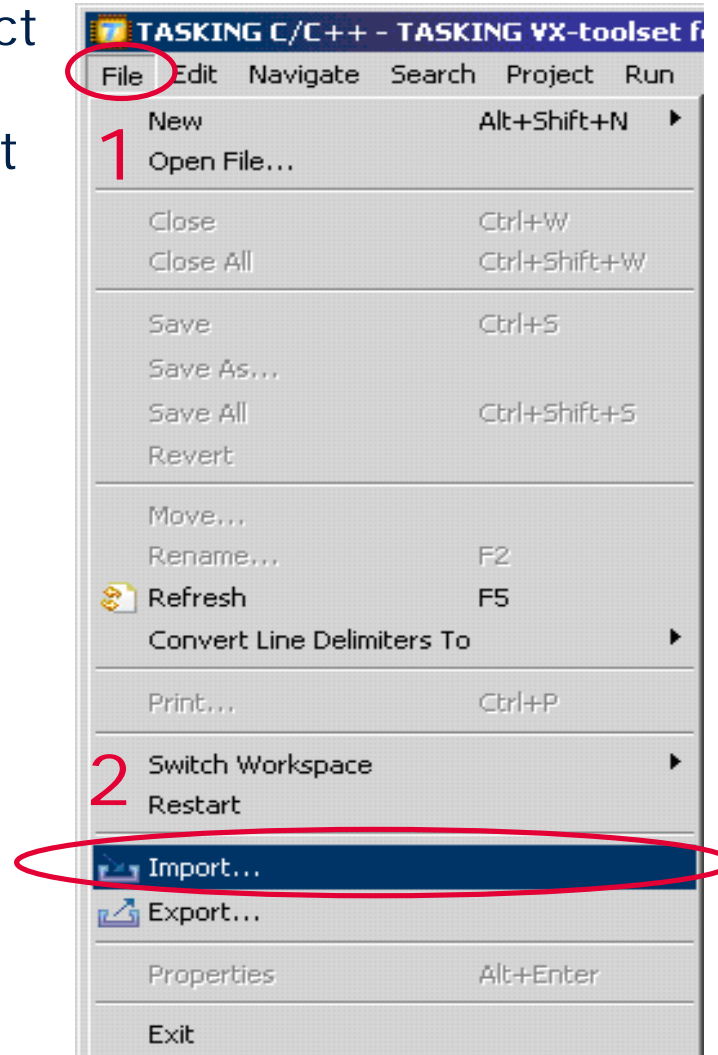
- Create New Project

- Click on Workbench (if not already there...)

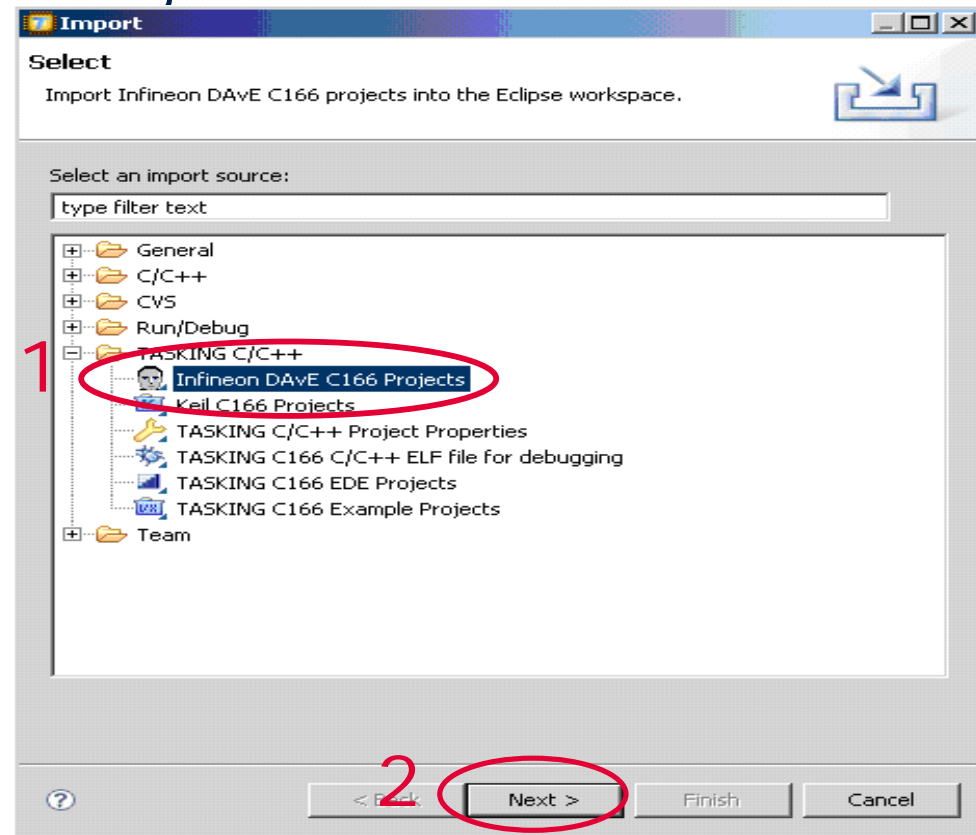


■ Import your DAVe Project

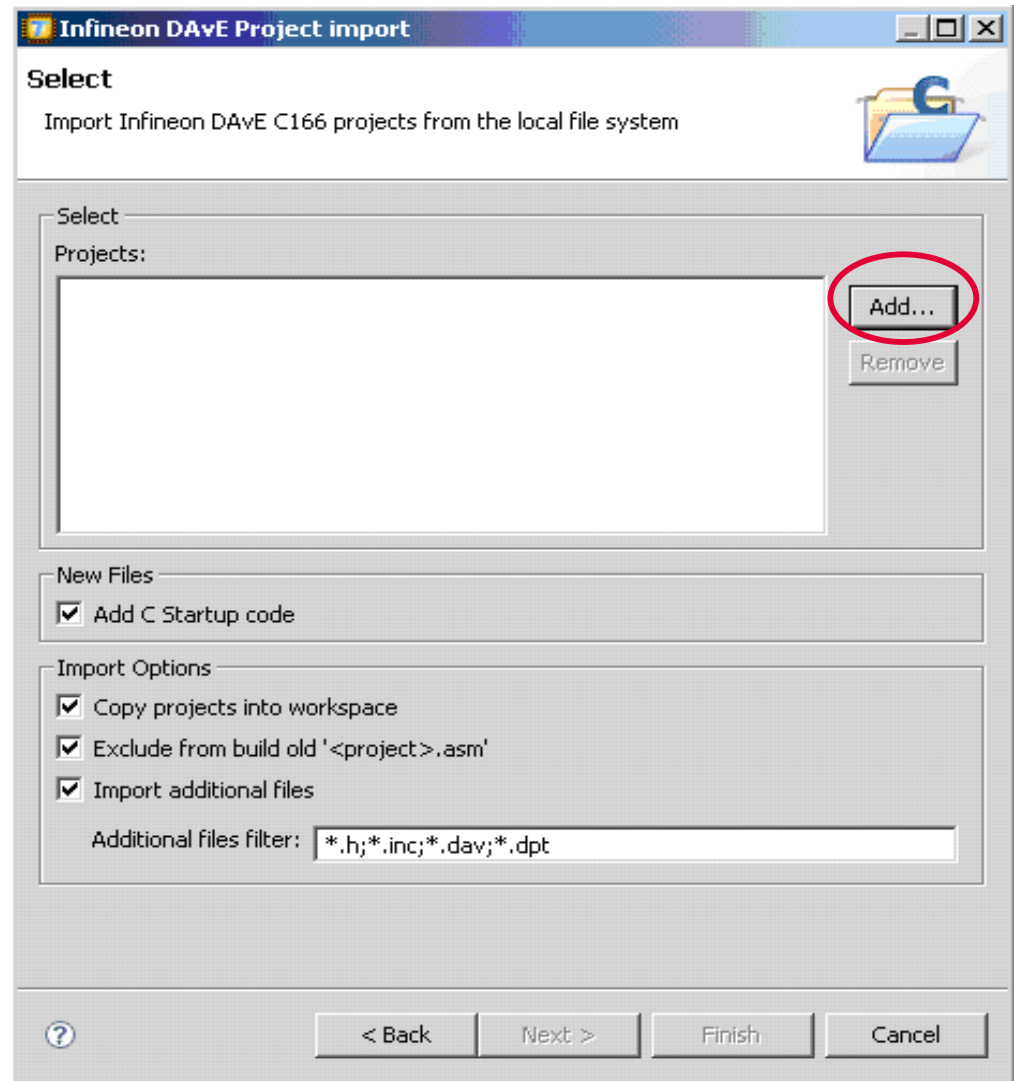
□ Click on File -> Import



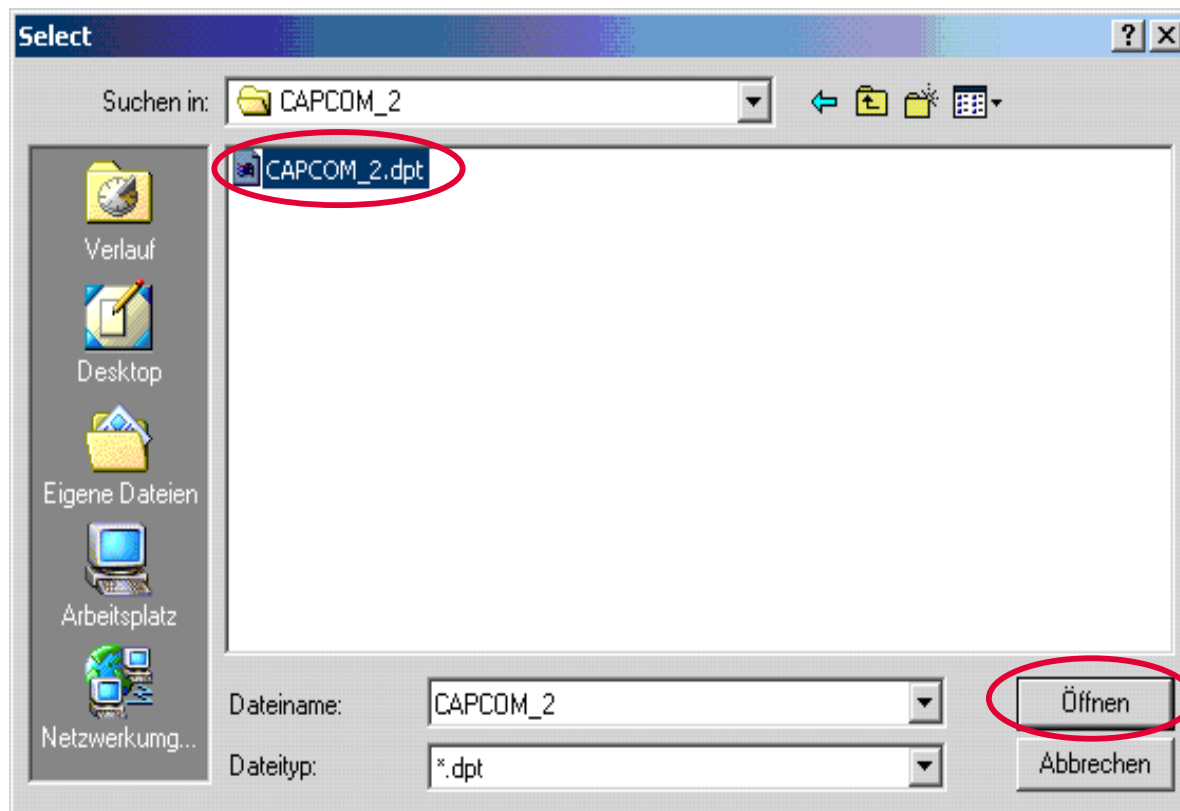
- Select your DAVe Project
 - Click on Tasking C/C++
 - Infineon DAVe C166 Projects
 - Click 'Next'



■ Add your DAvE Project file

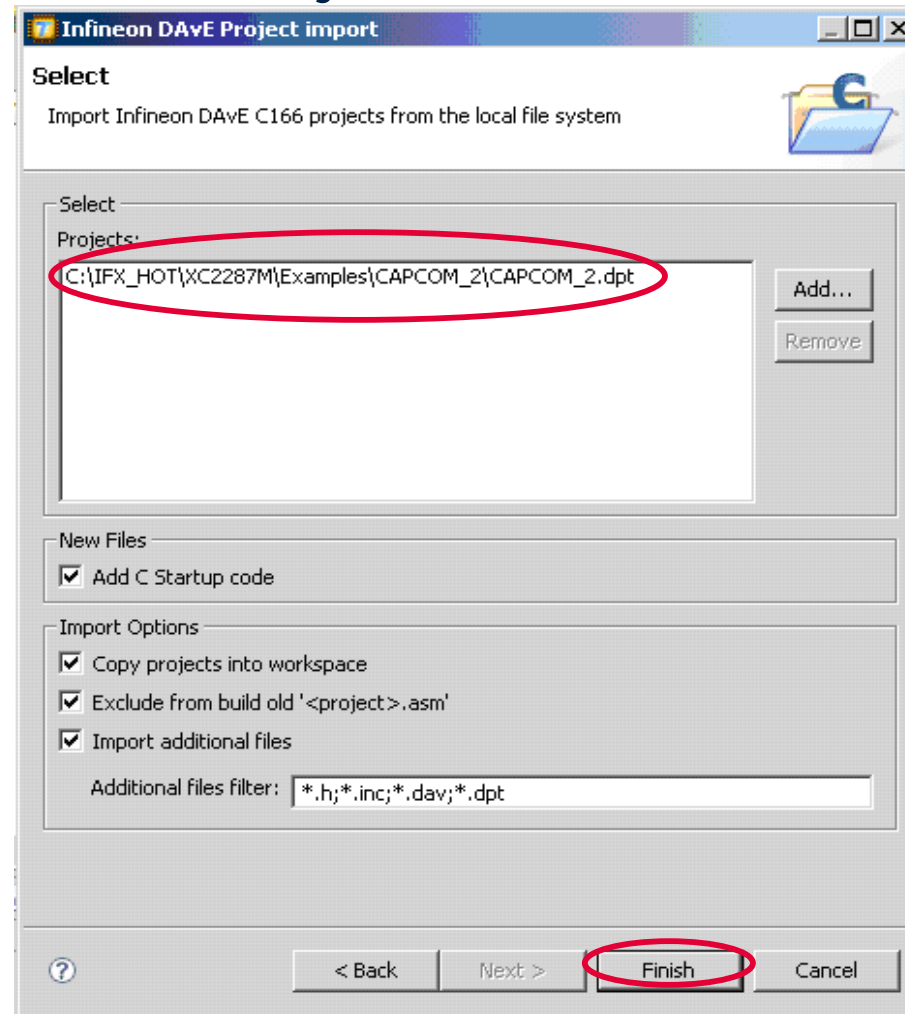


■ Open your DAvE Project




■ Import NOW your DAVe Project file

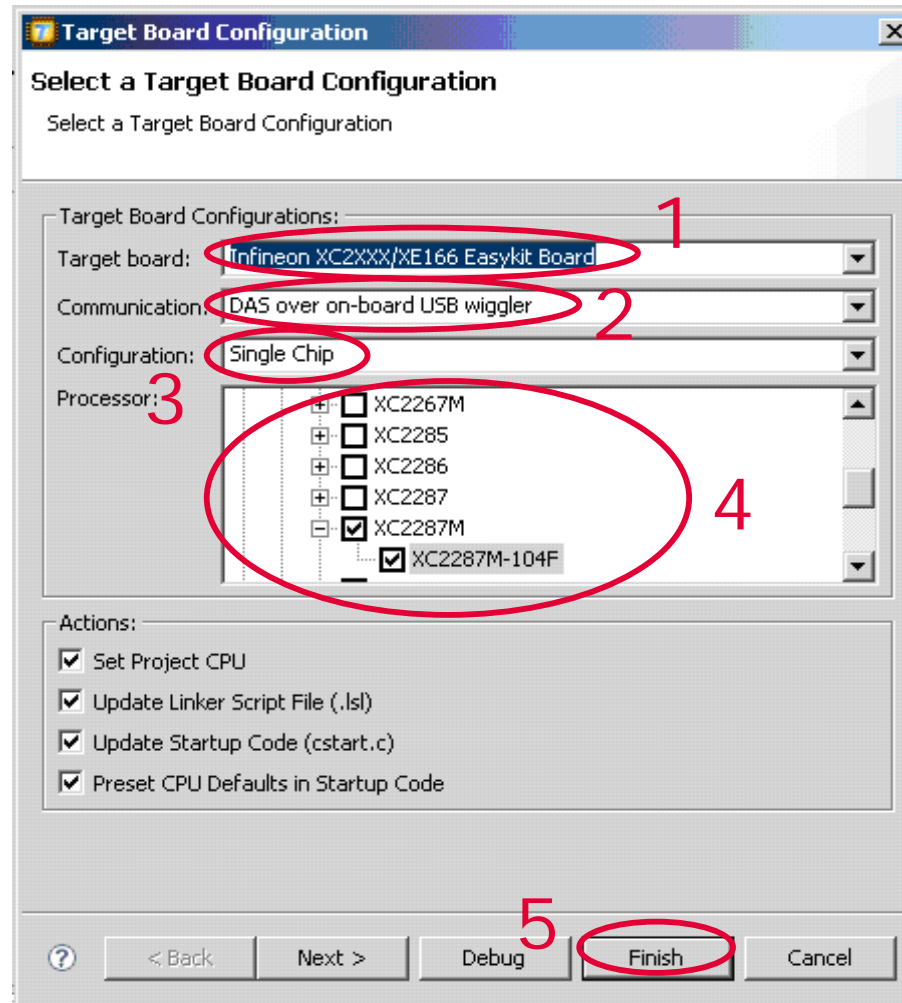
□ Click 'Finish'



■ Set Target Board Configuration for Project CAPCOM_2

- Click on 
- Select 'Infineon XC2XXX/XE166 Easy Kit Board'
- Select 'DAS over on-board USB wiggler'
- Select 'Single Chip'
- Select 'XC2287M-104F'
- Click 'Finish'

■ Set Target Board Configuration for Project CAPCOM_2



■ Software Hint

- DAvE doesn't change code that is inserted in the 'USER CODE' sections if you let DAvE regenerate the code.

Therefore, **whenever adding code to the generated code, write it into a 'USER CODE' section.**

The code you really have to add looks like this:

```
while(1)
{
// USER CODE BEGIN (Main,4)
```

```
BlinkLED();
```

```
// USER CODE END
}
```

CAPCOM_2 – Tasking VX Toolset

Edit File 'CC2.C'



- In the ISR function 'void CC2_viTmr8(void)'
(almost at the end)

```
// USER CODE BEGIN (Tmr8,1)

// USER CODE END

_interrupt(CC2_TSINT) void CC2_viTmr8(void)
{
    // USER CODE BEGIN (Tmr8,2)
    IO_vSetPin(IO_P10_0);
    IO_vSetPin(IO_P10_1);
    IO_vSetPin(IO_P10_2);
    // USER CODE END
} // End of function CC2_viTmr8
```

CAPCOM_2 – Tasking VX Toolset

Edit File 'CC2.C'



- In the ISR function 'void CC2_viCC16(void)' (almost at the end)

```
// USER CODE BEGIN (CC16,1)

// USER CODE END

_interrupt(CC2_CC16INT) void CC2_viCC16(void)
{
    // USER CODE BEGIN (CC16,2)
    IO_vResetPin(IO_P10_0);
    // USER CODE END
} // End of function CC2_viCC16
```

CAPCOM_2 – Tasking VX Toolset

Edit File 'CC2.C'



- In the ISR function 'void CC2_viCC17(void)'
(almost at the end)

```
// USER CODE BEGIN (CC17,1)

// USER CODE END

_interrupt(CC2_CC17INT) void CC2_viCC17(void)
{
    // USER CODE BEGIN (CC17,2)
    IO vResetPin(IO_P10_1);
    // USER CODE END
} // End of function CC2_viCC17
```

CAPCOM_2 – Tasking VX Toolset

Edit File 'CC2.C'



- In the ISR function 'void CC2_viCC18(void)' (almost at the end)

```
// USER CODE BEGIN (CC18,1)

// USER CODE END

_interrupt(CC2_CC18INT) void CC2_viCC18(void)
{
    // USER CODE BEGIN (CC18,2)
    IO vResetPin(IO_P10_2);
    // USER CODE END
} // End of function CC2_viCC18
```

CAPCOM_2 – Tasking VX Toolset Build Project

- Click on 'Build Project Capcom_2'

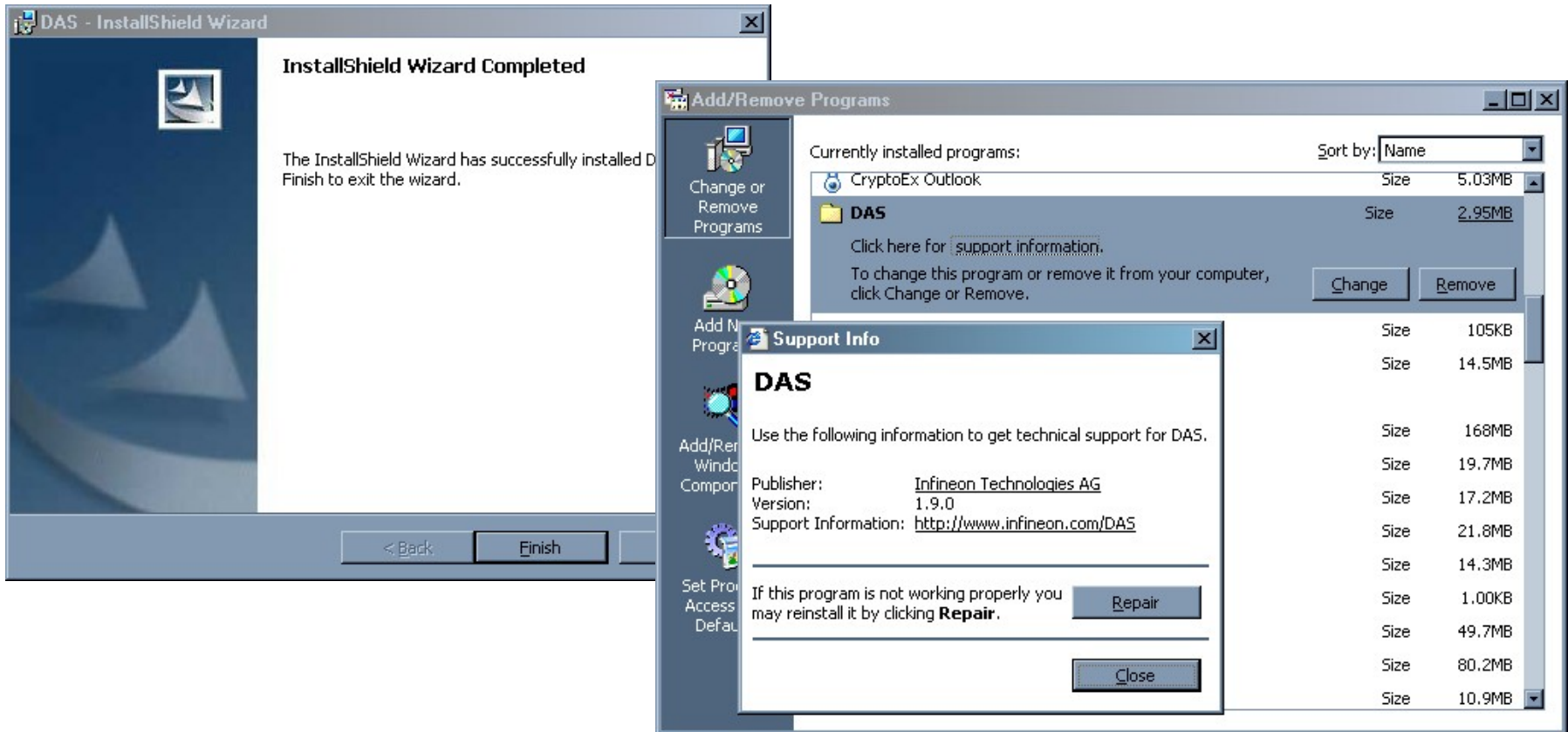


```
Problems Console Properties
C-Build [Capcom_2_LED_dimmen]
Compiling SC5.C
c166 W533: ["..\XC22xxMREGS.h" 1712
list
0 errors, 1 warnings
Linking to Capcom_2_LED_dimmen.elf

Time consumed: 10594 ms
**** End of build ****
```

CAPCOM_2 - Device Access Server

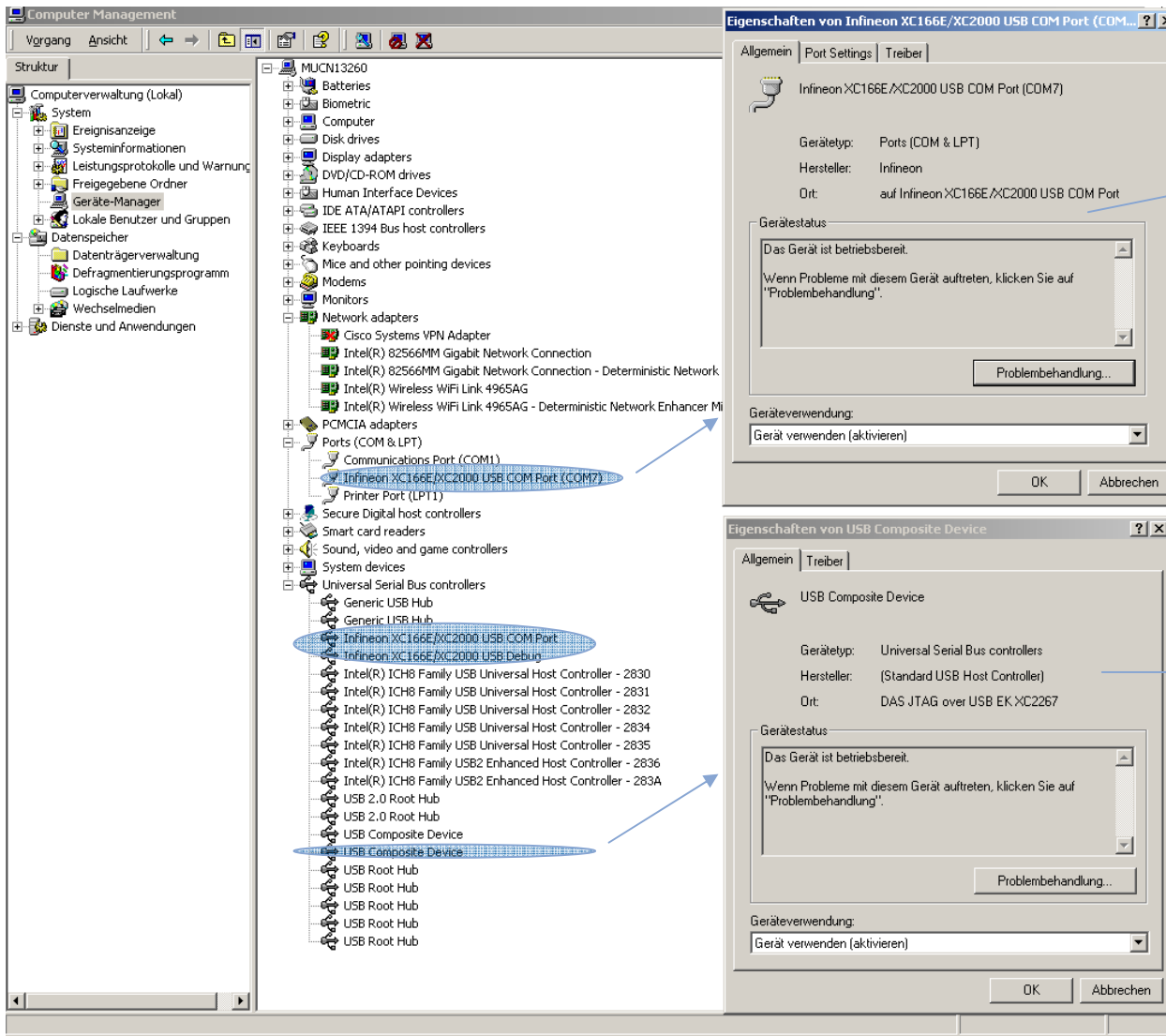
Check for the latest DAS version



Note: It is recommended to use the latest DAS version.
Download the latest version at www.infineon.com/DAS

CAPCOM_2 - Device Access Server

1.) Checking USB connections



This gets identified only when COM port is used

- Via the USB interface on the Easykit with FTDI chip

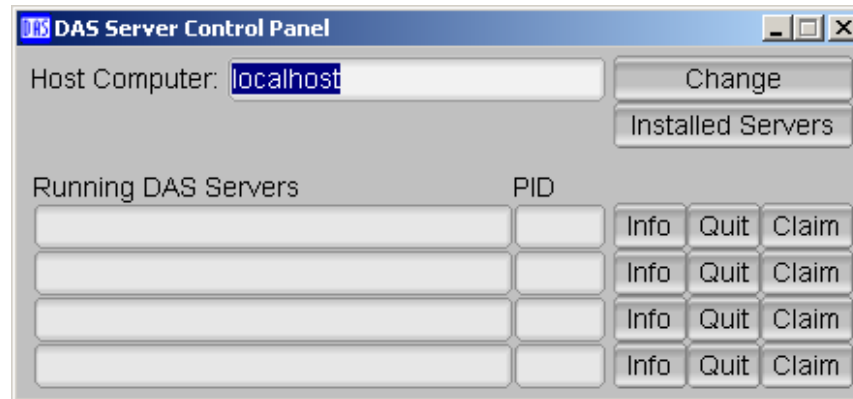
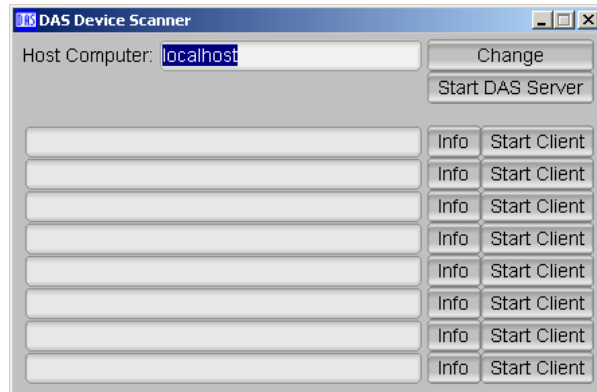
The DAS JTAG composite device gets identified

- When miniWiggler is connected
- When USB Wiggler Box is connected
- Via the USB interface on the Easykits with FTDI chip

CAPCOM_2 - Device Access Server

2.) Check DAS status

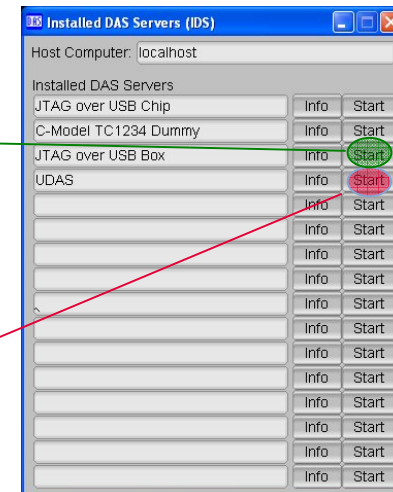
1. Start DAS device scanner
2. Start DAS Server Control panel



3. If DAS device scanner does not show any device, start the appropriate DAS server

Incase you are connected via the USB Wiggler box,
then start „JTAG over USB Box“

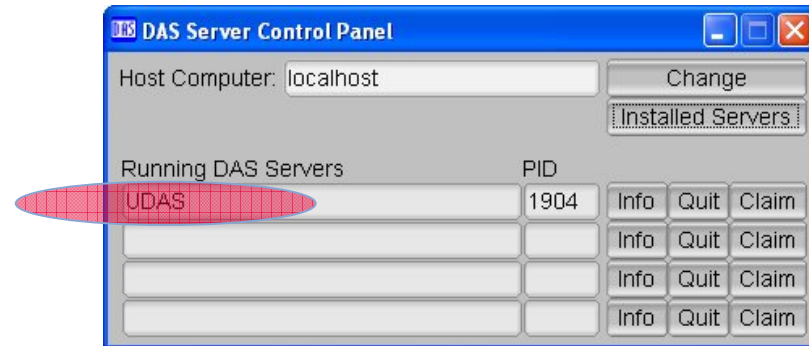
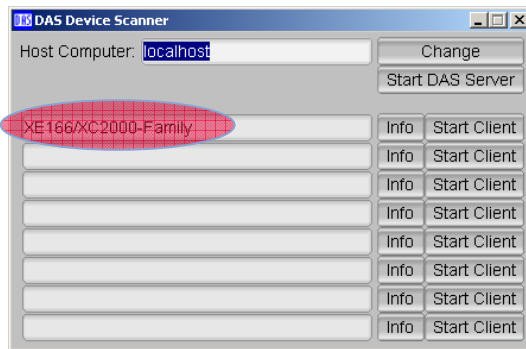
Incase you are connected via the FTDI chip or mini wiggler,
then start „UDAS“



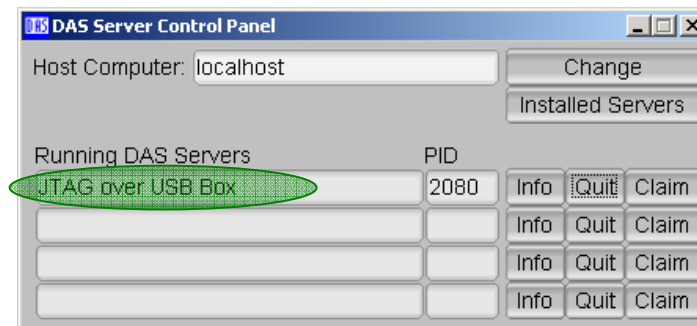
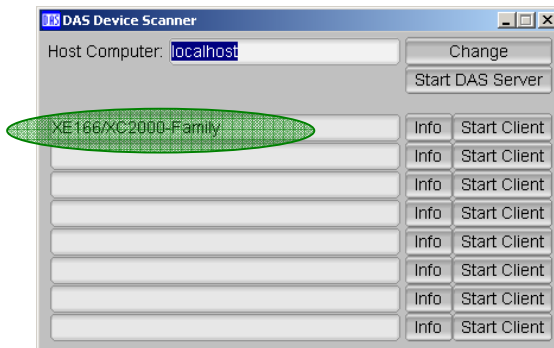
CAPCOM_2 - Device Access Server

3.) Starting the servers manually

4. In case „UDAS“ server is started and XC2000 easykit is connected via on-chip FTDI or via separate miniWiggler, following status changes could be noted

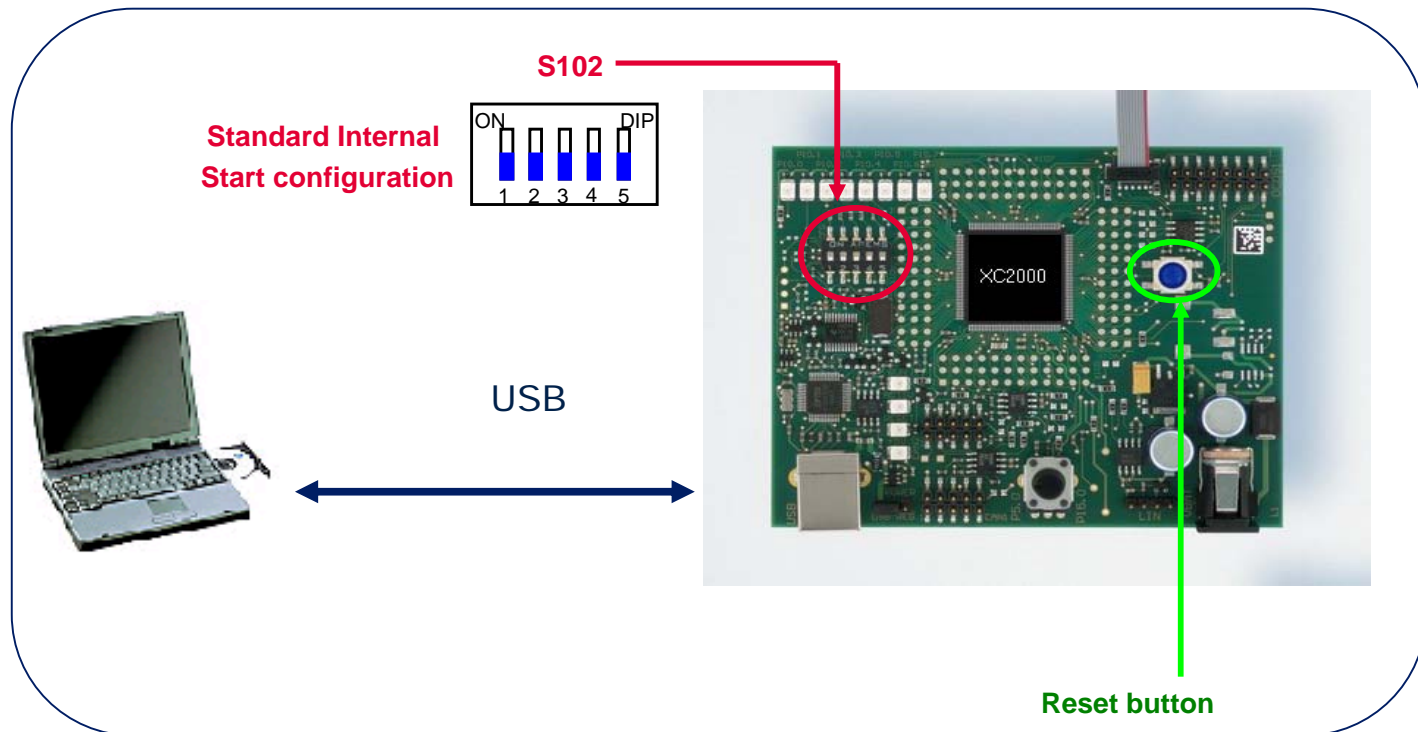


5. In case „JTAG over USB Box“ server is started and XC2000 starter kit is connected via Wiggler box, following status changes could be noted



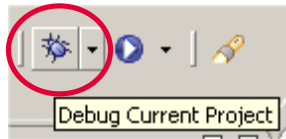
■ Internal start from Flash

- ❑ Connect XC2287M Board to PC
- ❑ Check the DIP switch settings, S102: **OFF-OFF-OFF-OFF-OFF**
- ❑ Reset the board (press the reset button)

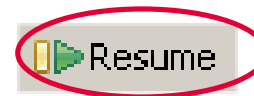


CAPCOM_2 – Tasking VX Toolset Run Debugger

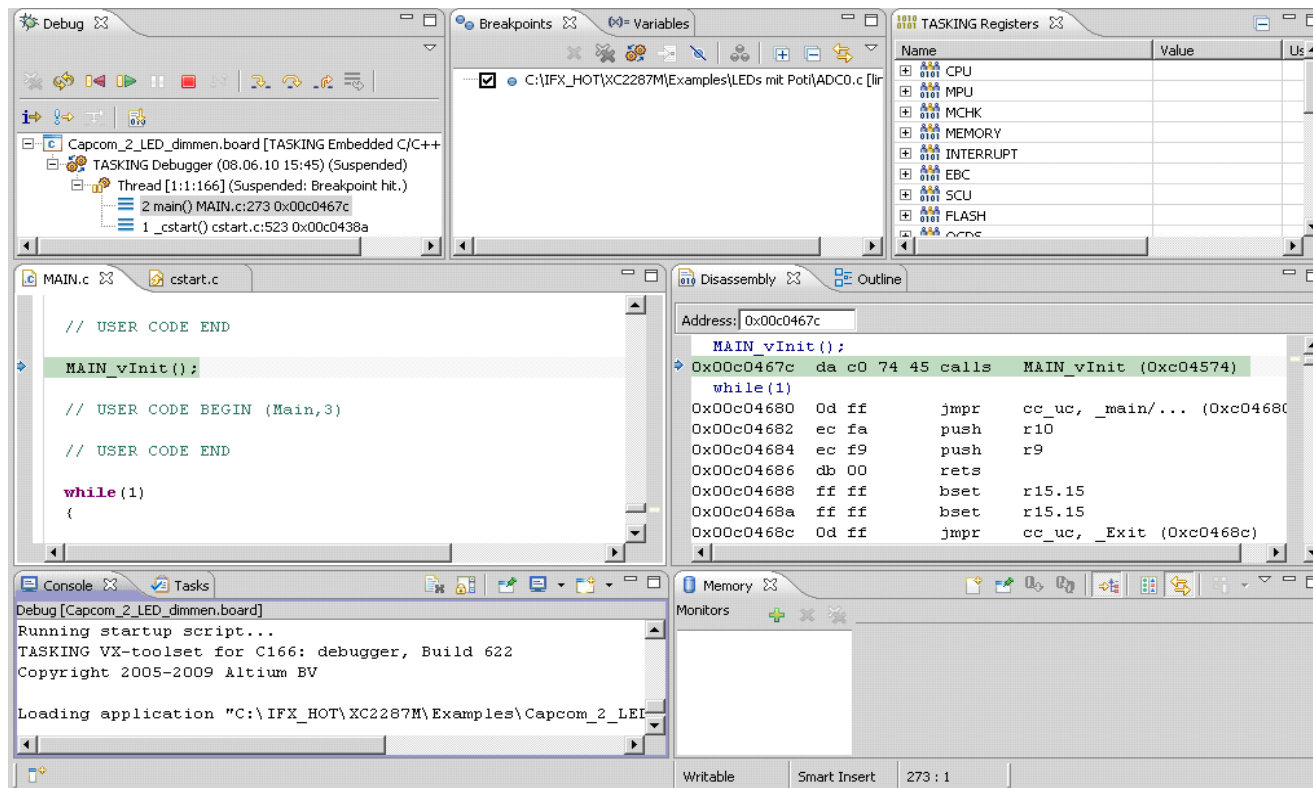
- 1
■ Click on



2



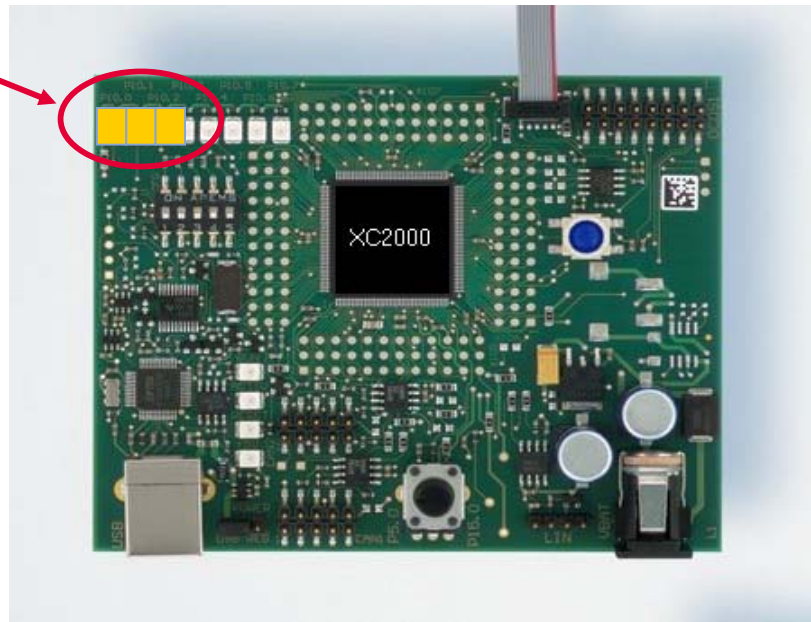
- Click on 'Resume' and start program




	Resume	F8
	Suspend	
	Terminate	Ctrl+F2
	Step Into	F5
	Step Over	F6
	Step Return	F7

- The yellow LED's will toggle

LED's blinking



- Verification: (stop the program)
 - Go to Tasking debugger
 - Click on 'Suspend' 
 - ⇒ LED is no more toggling; stays constant on or off



We commit.
We innovate.
We partner.
We create value.



Never stop thinking