XC2287M HOT

Solution ASC

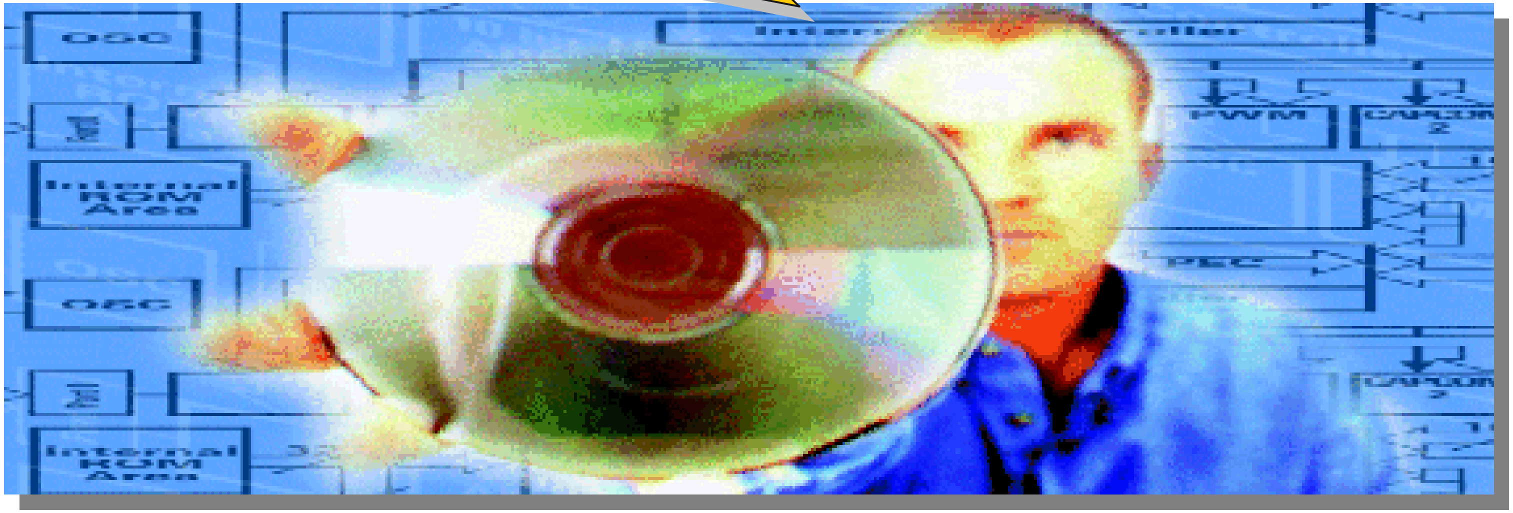
Uses a timer triggered LED to toggle with ASC Interrupt

Device: XC2287M-104F80
Compiler: Tasking Viper 2.4r1
Code Generator: DAvE 2.1

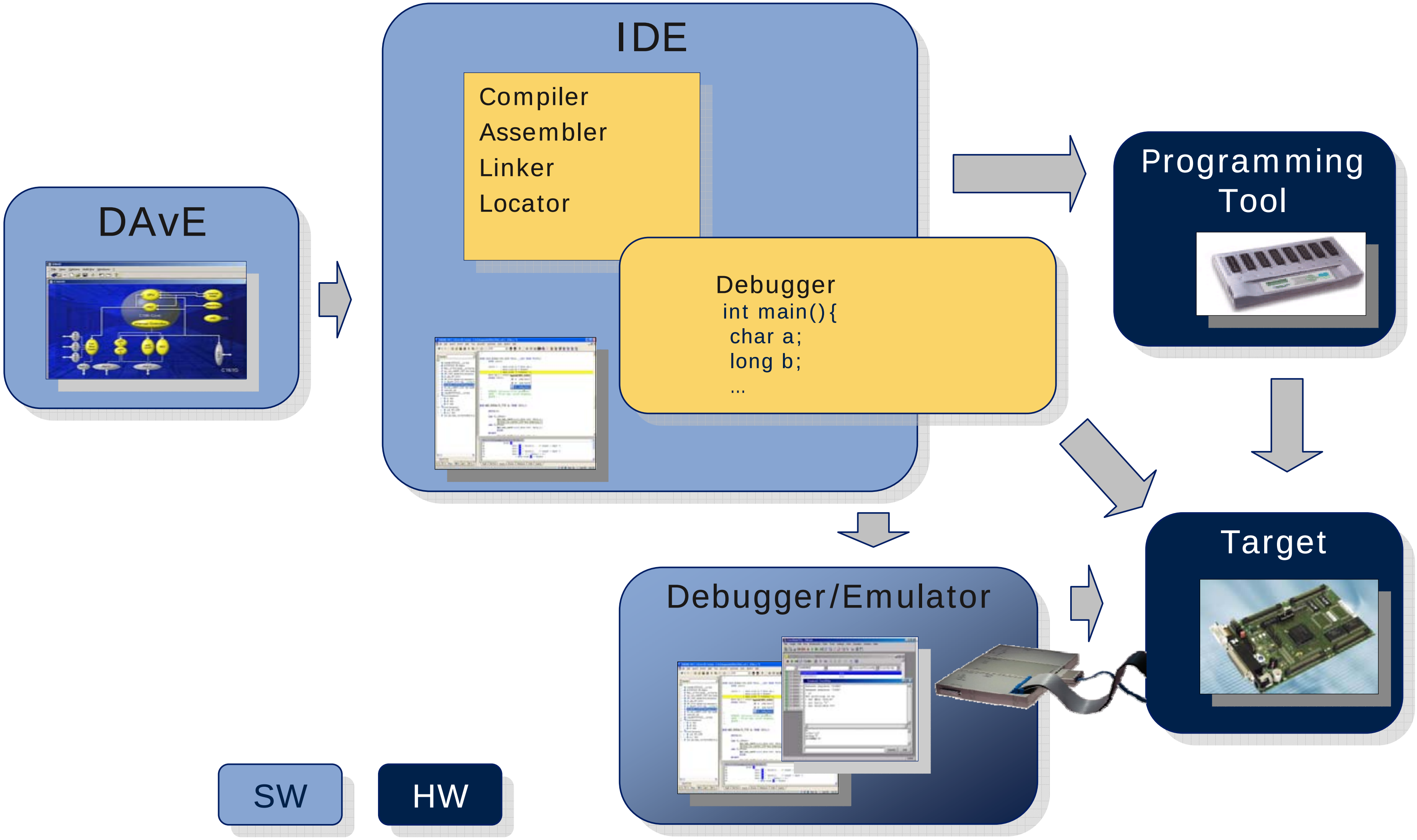
Never stop thinking

Hello World with ASC

Let’s get started now!



XC2287M HOT Exercise ASC
Interaction of Development Tools



DAvE

IDE

Compiler
Assembler
Linker
Locator

Debugger

```
int main(){
 char a;
 long b;
 ...
```

Programming Tool

Target

Debugger/Emulator

SW

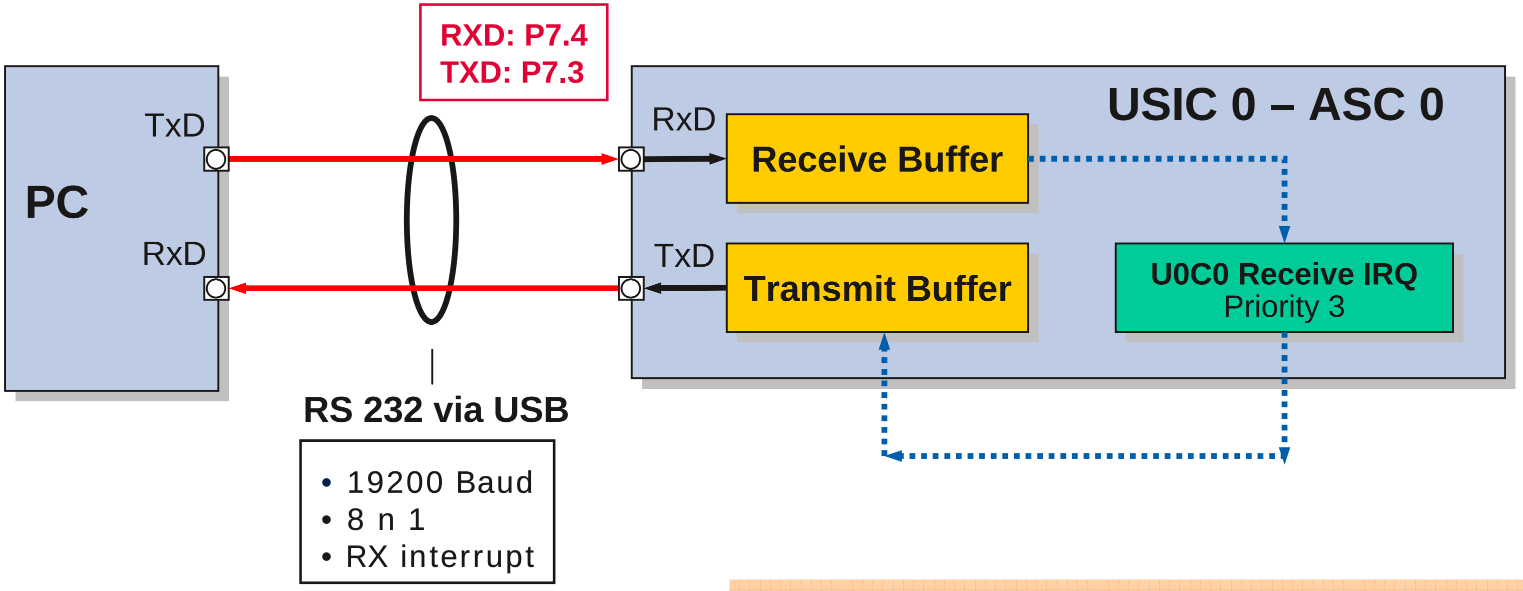
HW

HOT Exercise ASC
Hello World with ASC

In this exercise you will:

- Configure the XC2287M with DAvE
- Configure USIC 0 Channel 0 as a UART
- Receive a character from a PC and generate a receive interrupt
- Transmit the received data back to the PC
- Toggle one of the LED’s on the board on receipt of every character

HOT Exercise ASC
Block Diagram



RXD: P7.4
TXD: P7.3

PC

TxD

RxD

USIC 0 – ASC 0

RxD

Receive Buffer

TxD

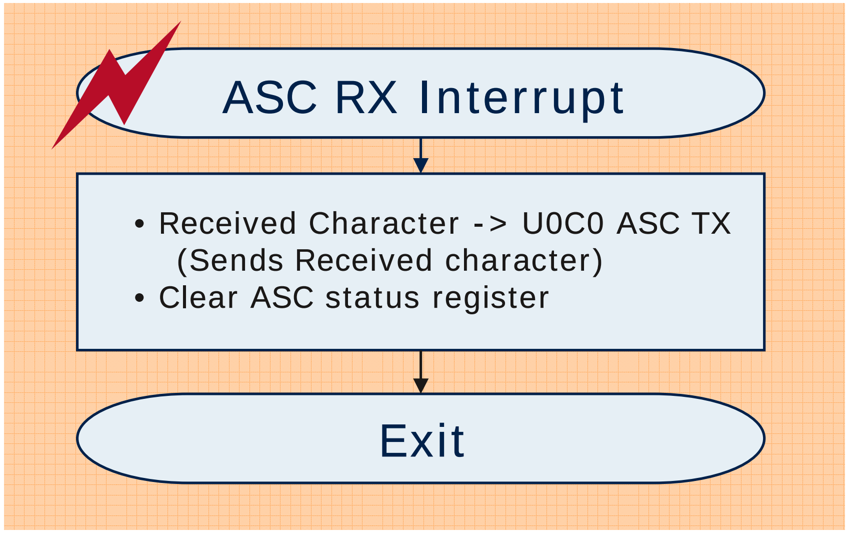
Transmit Buffer

U0C0 Receive IRQ
Priority 3

RS 232 via USB

- 19200 Baud
- 8 n 1
- RX interrupt

······▶ = Transfer by Interrupt handler



ASC RX Interrupt

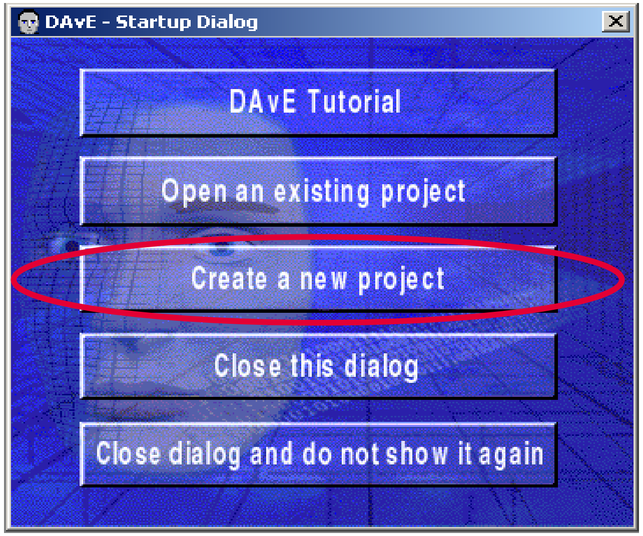
- Received Character -> U0C0 ASC TX (Sends Received character)
- Clear ASC status register

Exit

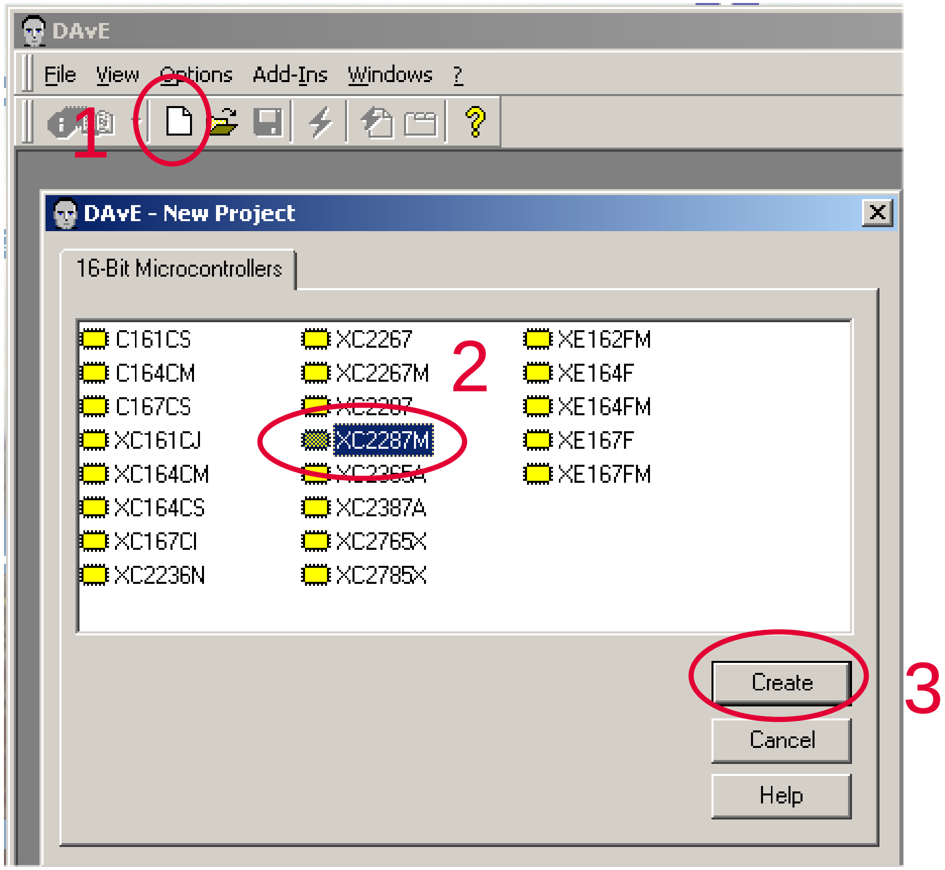
HOT Exercise ASC
Start DAvE

- **Start DAvE**
  - □ Click on the DAvE

- **Create a new project** (Startup Dialog pop up automatically)
  - □ Click on ‘Create a new project’ or select File -> New
  - □ Select microcontroller: ‘XC2287M’



HOT Exercise ASC
Select Device

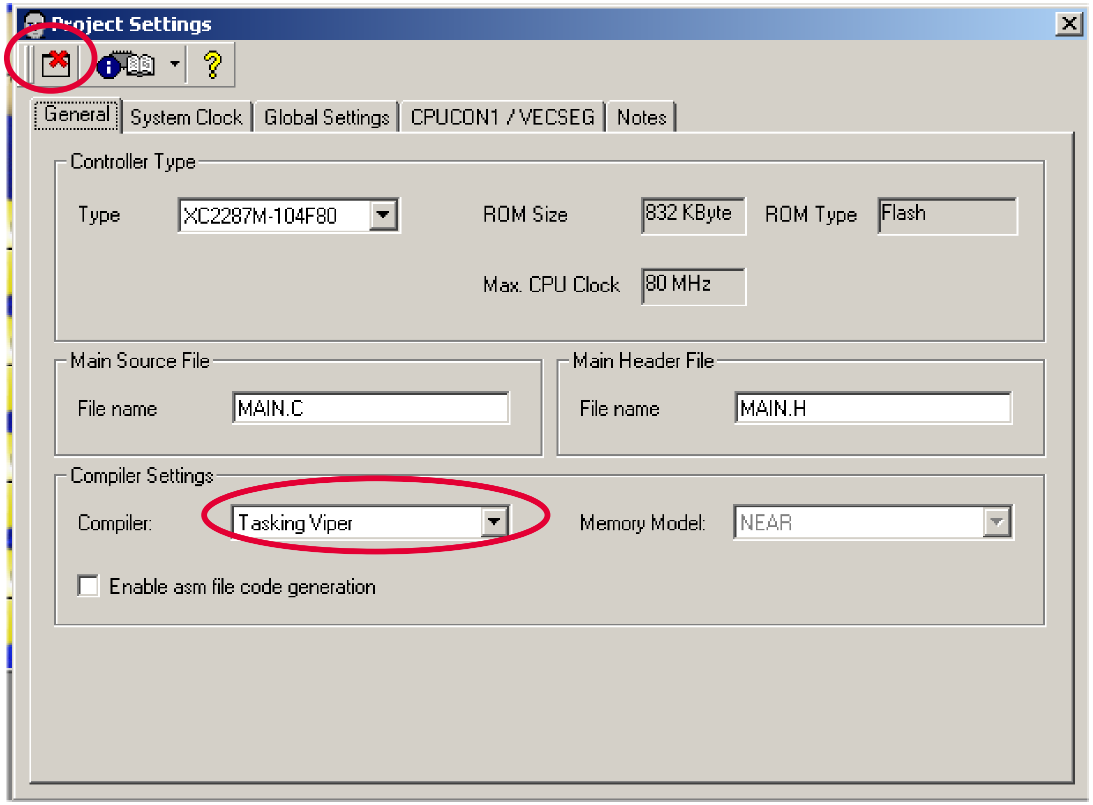


C161CS
C164CM
C167CS
XC161CJ
XC164CM
XC164CS
XC167CI
XC2236N
XC2267
XC2267M
XC2287
XC2287M
XC2365A
XC2387A
XC2765X
XC2785X
XE162FM
XE164F
XE164FM
XE167F
XE167FM

HOT Exercise ASC - DAvE Configurations Project Settings

- Project Settings

- Close the window



Controller Type

Type: XC2287M-104F80

ROM Size: 832 KByte

ROM Type: Flash

Max. CPU Clock: 80 MHz

Main Source File

File name: MAIN.C

Main Header File

File name: MAIN.H

Compiler Settings

Compiler: Tasking Viper

Memory Model: NEAR

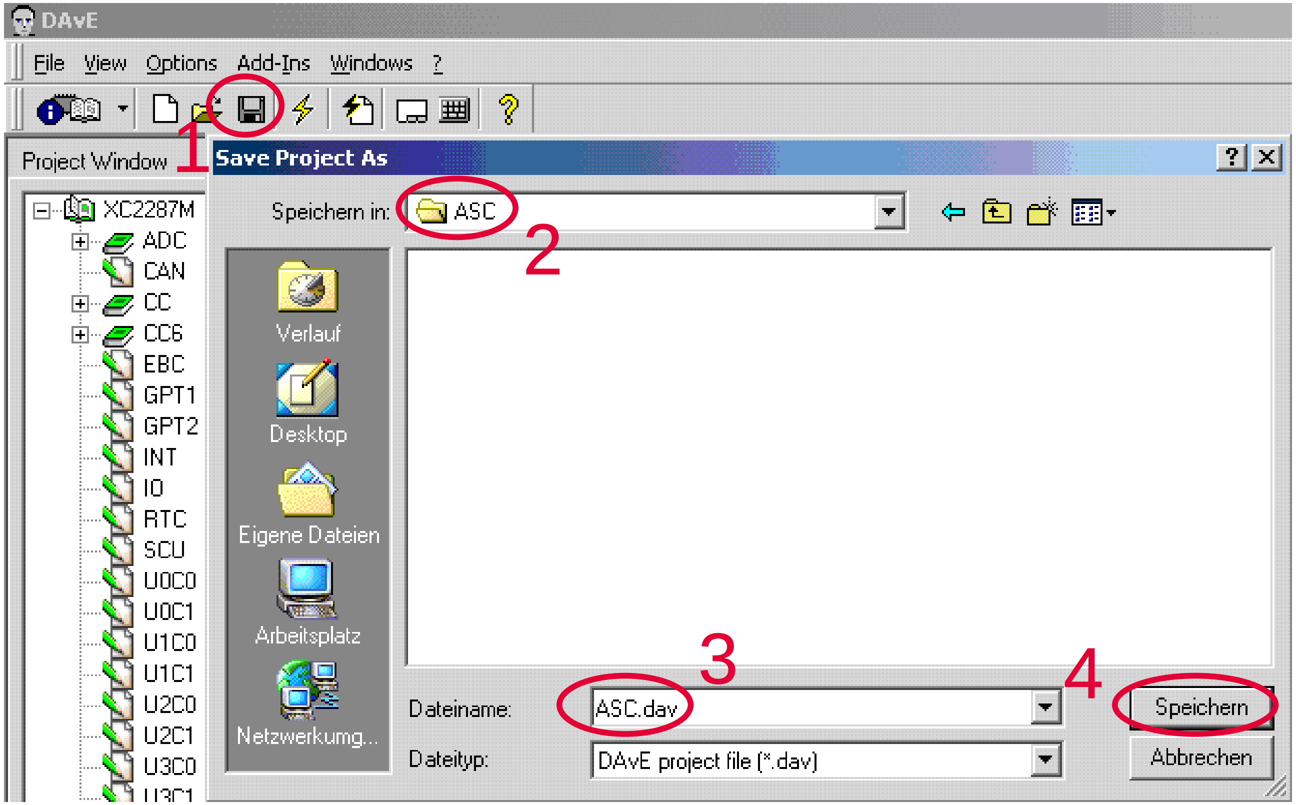
□ Enable asm file code generation

HOT Exercise ASC - DAvE Configurations
Save DAvE Project

- Save your DAvE project 💾
  - Path:
    C:\IFX_HOT\XC2287M\Examples\ASC
  - Project name:
    ASC\ASC.dav

HOT Exercise ASC - DAvE Configurations
Save DAvE Project

- Save your DAvE Project File

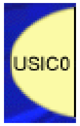


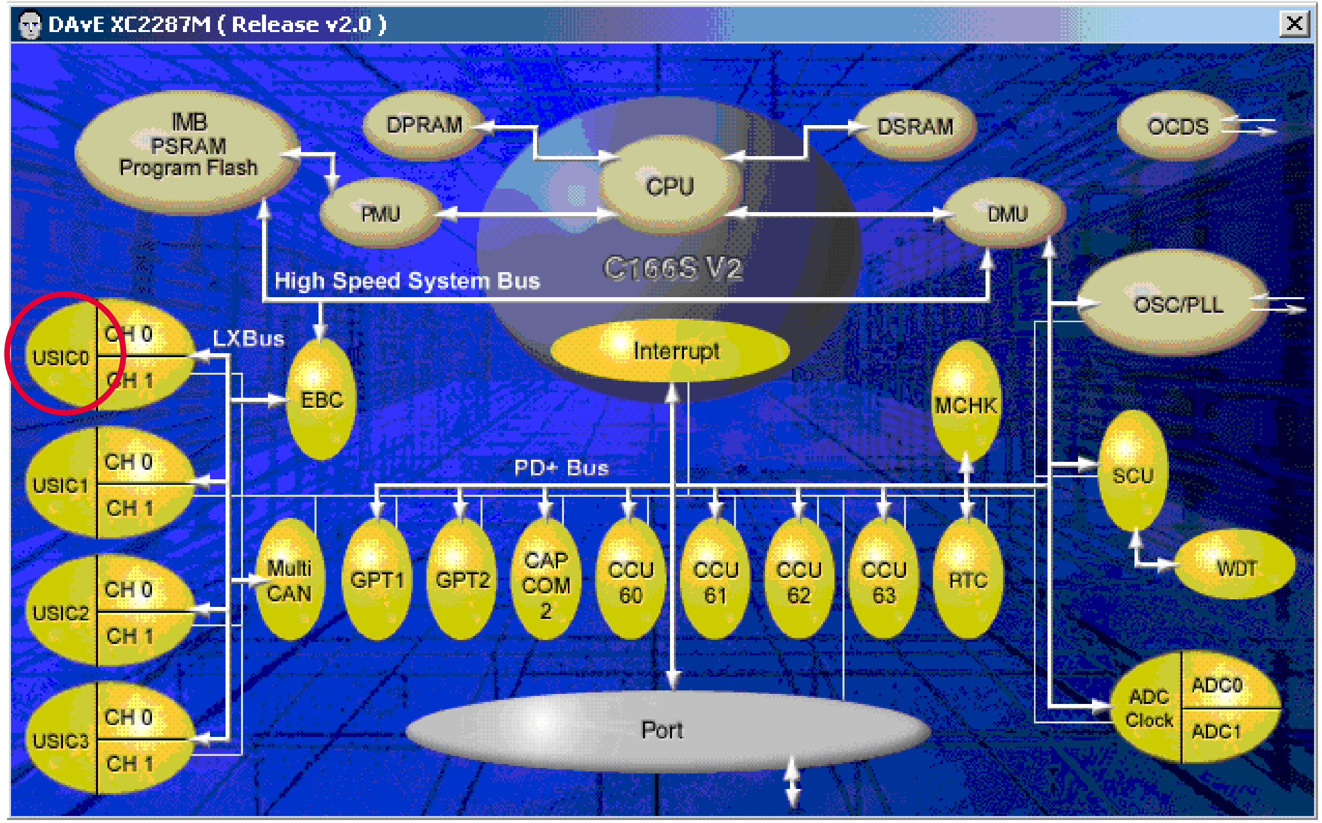
Speichern in: ASC

Dateiname: ASC.dav

Dateityp: DAvE project file (*.dav)

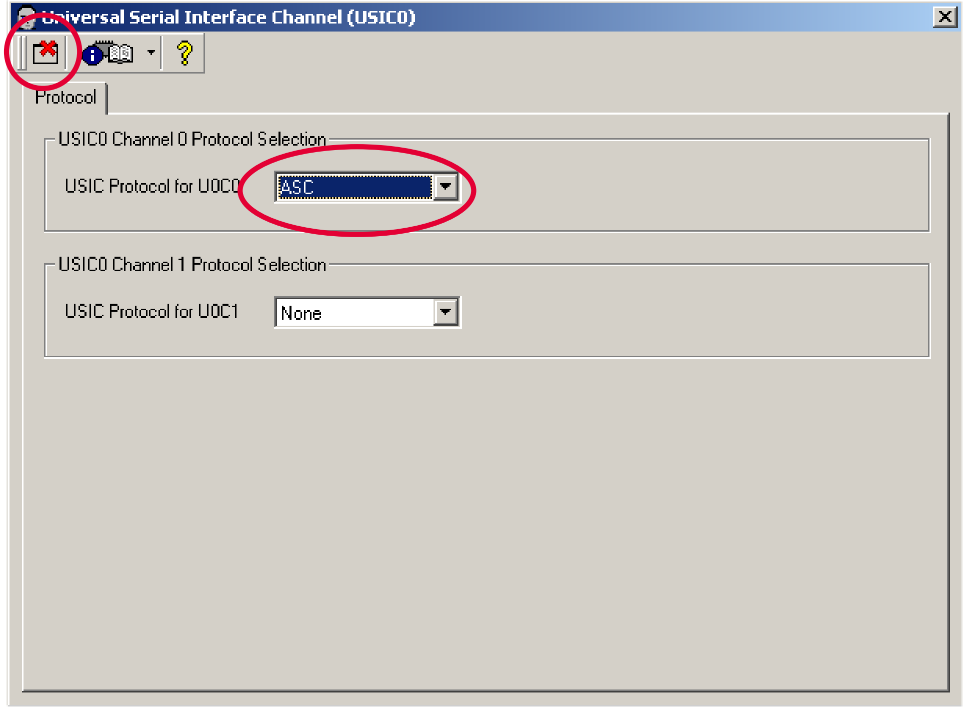
HOT Exercise ASC - DAvE Configurations
ASC settings

- **XC2287M**
  - **USIC0 :**
    - Click on the 



HOT Exercise ASC - DAvE Configurations
ASC Settings (cont.)

- Configure 'Protocol'
  - Select ASC for USIC0 (U0C0)
  - Close the windows by pressing ☒



Universal Serial Interface Channel (USIC0)

Protocol

USIC0 Channel 0 Protocol Selection

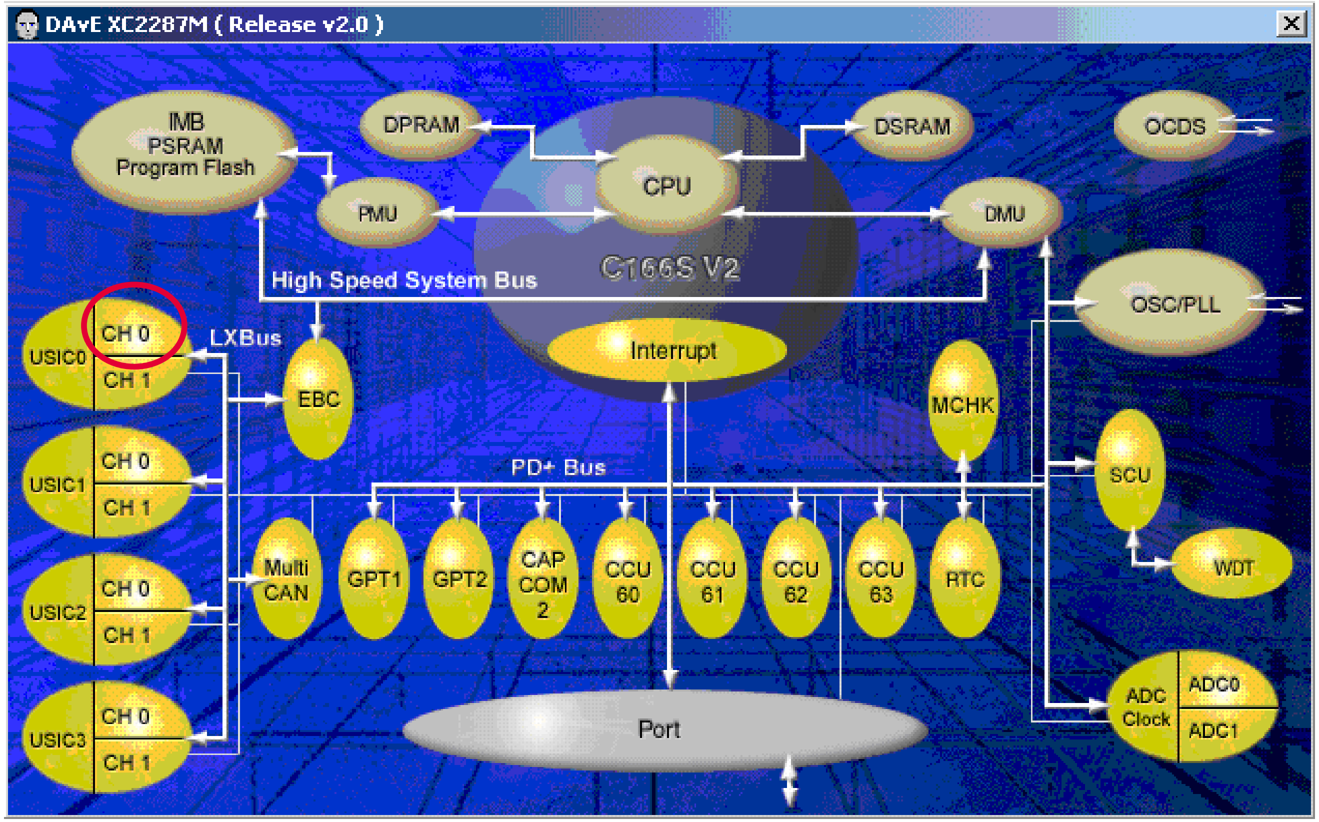
USIC Protocol for U0C0: ASC

USIC0 Channel 1 Protocol Selection

USIC Protocol for U0C1: None

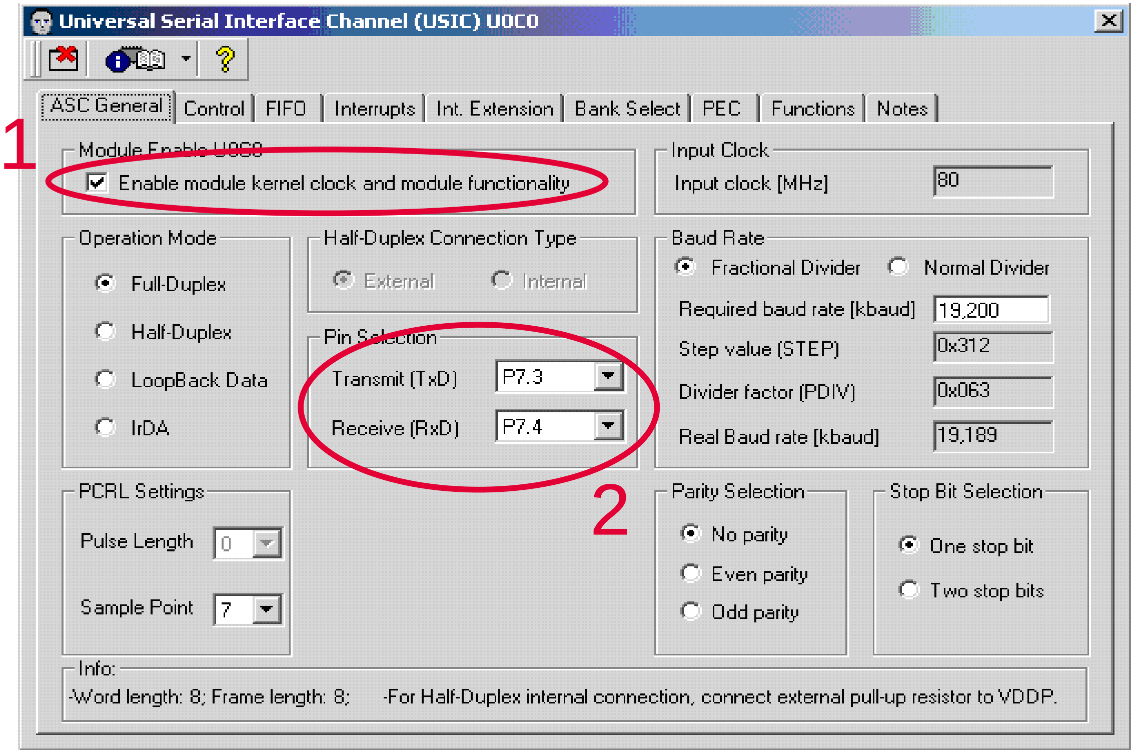
HOT Exercise ASC - DAvE Configurations
ASC settings (cont.)

- **XC2287M**
  - **USIC0, CH0 :**
    - Click on the



HOT Exercise ASC - DAvE Configurations
ASC Settings (cont.)

- Configure ASC – General
  - □ Enable module
  - □ Pin selection – Transmit (TxD) P7.3, Receive (RxD) P7.4
  - □ Others- default



Universal Serial Interface Channel (USIC) U0C0

Module Enable U0C0
☑ Enable module kernel clock and module functionality

Input Clock
Input clock [MHz]: 80

Operation Mode
● Full-Duplex
○ Half-Duplex
○ LoopBack Data
○ IrDA

Half-Duplex Connection Type
● External ○ Internal

Pin Selection
Transmit (TxD): P7.3
Receive (RxD): P7.4

Baud Rate
● Fractional Divider ○ Normal Divider
Required baud rate [kbaud]: 19,200
Step value (STEP): 0x312
Divider factor (PDIV): 0x063
Real Baud rate [kbaud]: 19,189

PCRL Settings
Pulse Length: 0
Sample Point: 7

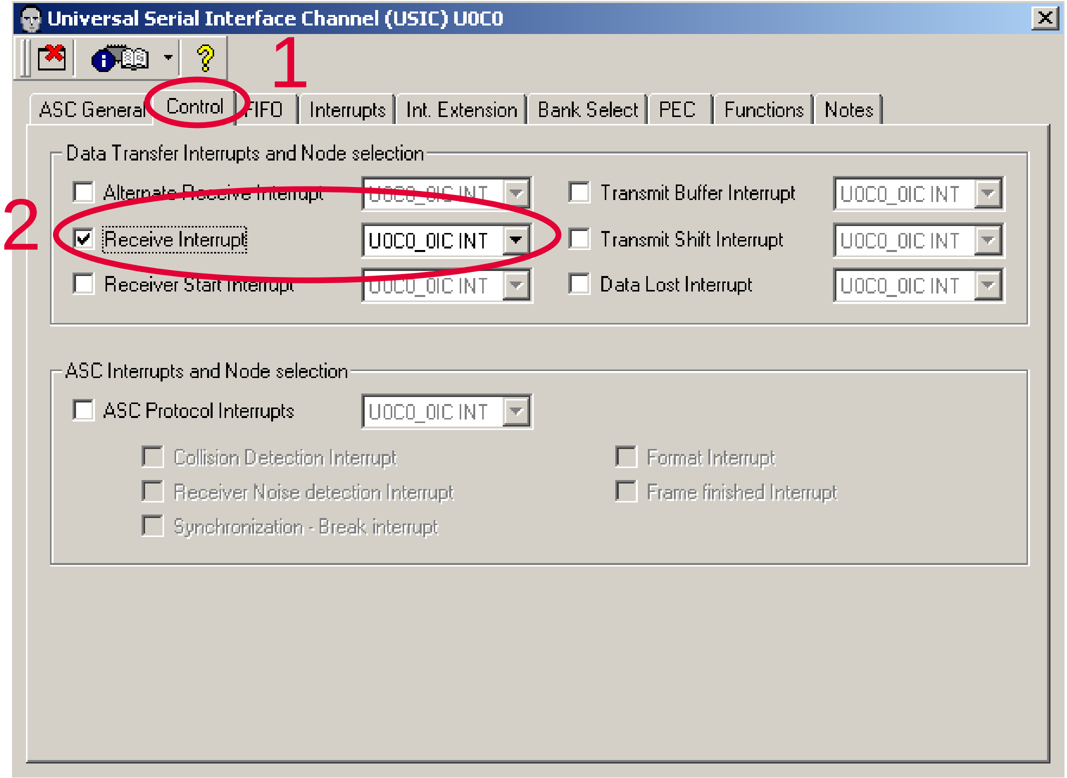
Parity Selection
● No parity
○ Even parity
○ Odd parity

Stop Bit Selection
● One stop bit
○ Two stop bits

Info:
-Word length: 8; Frame length: 8; -For Half-Duplex internal connection, connect external pull-up resistor to VDDP.

HOT Exercise ASC - DAvE Configurations ASC Settings (cont.)

- Configure ASC – Control
  - Click on 'Receive Interrupt'



Universal Serial Interface Channel (USIC) U0C0

1

ASC General | Control | FIFO | Interrupts | Int. Extension | Bank Select | PEC | Functions | Notes

Data Transfer Interrupts and Node selection

2

☐ Alternate Receive Interrupt	U0C0_0IC INT	☐ Transmit Buffer Interrupt	U0C0_0IC INT
☑ Receive Interrupt	U0C0_0IC INT	☐ Transmit Shift Interrupt	U0C0_0IC INT
☐ Receiver Start Interrupt	U0C0_0IC INT	☐ Data Lost Interrupt	U0C0_0IC INT

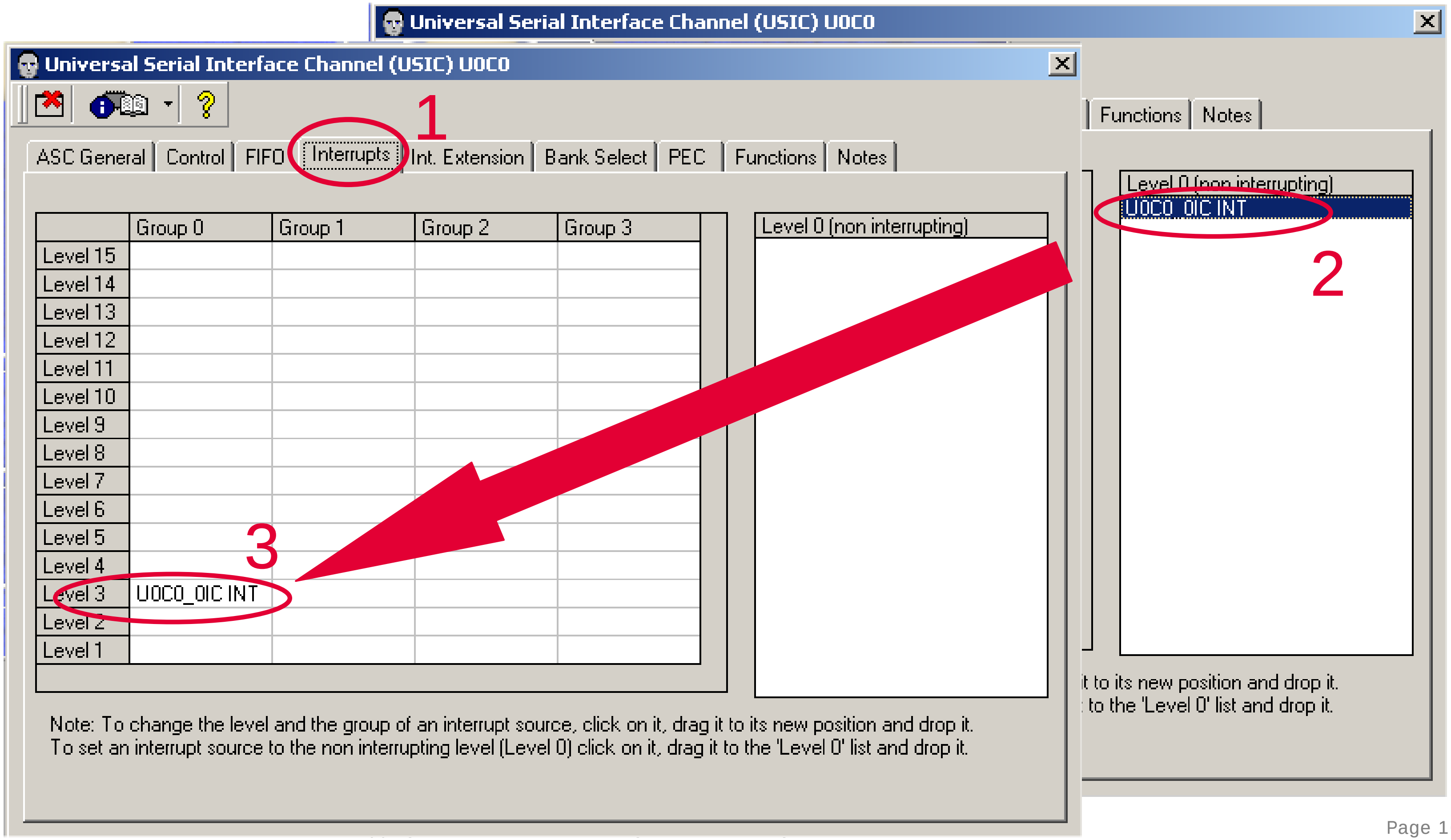
ASC Interrupts and Node selection

☐ ASC Protocol Interrupts U0C0_0IC INT

☐ Collision Detection Interrupt	☐ Format Interrupt
☐ Receiver Noise detection Interrupt	☐ Frame finished Interrupt
☐ Synchronization - Break interrupt	

HOT Exercise ASC - DAvE Configurations ASC Settings (cont.)

- Configure ASC – Interrupts
  - Drag 'U0C0 OIC INT' from Level 0 to Level 3, Group 0



Universal Serial Interface Channel (USIC) U0C0

ASC General | Control | FIFO | Interrupts | Int. Extension | Bank Select | PEC | Functions | Notes

	Group 0	Group 1	Group 2	Group 3
Level 15				
Level 14				
Level 13				
Level 12				
Level 11				
Level 10				
Level 9				
Level 8				
Level 7				
Level 6				
Level 5				
Level 4				
Level 3	U0C0_OIC INT			
Level 2				
Level 1				

Level 0 (non interrupting)

Note: To change the level and the group of an interrupt source, click on it, drag it to its new position and drop it.
To set an interrupt source to the non interrupting level (Level 0) click on it, drag it to the 'Level 0' list and drop it.

Universal Serial Interface Channel (USIC) U0C0

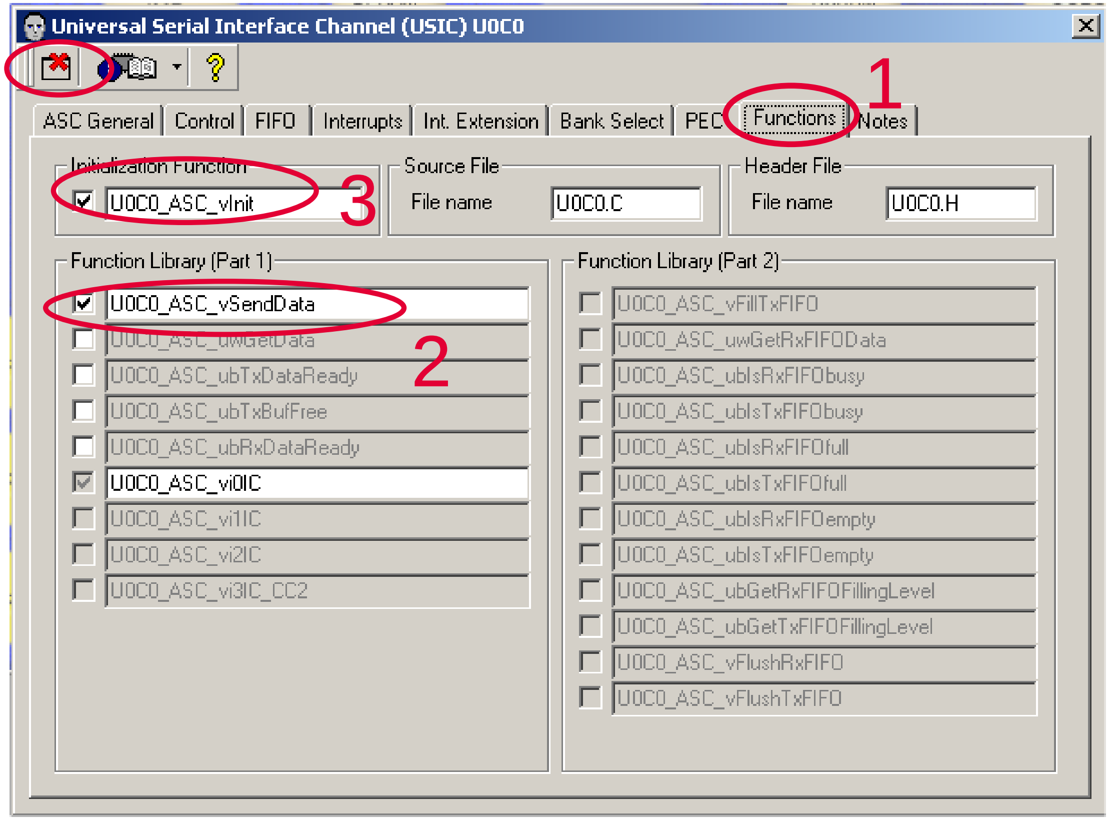
Functions | Notes

Level 0 (non interrupting)
U0C0_OIC INT

t to its new position and drop it.
to the 'Level 0' list and drop it.

HOT Exercise ASC - DAvE Configurations ASC Settings (cont.)

- Configure ASC – Functions
  - Click on 'U0C0_ASC_vInit'
  - Click on 'U0C0_ASC_vSendData'
  - Click on ☒



Universal Serial Interface Channel (USIC) U0C0

ASC General | Control | FIFO | Interrupts | Int. Extension | Bank Select | PEC | Functions | Notes

Initialization Function

☑ U0C0_ASC_vInit

Source File

File name: U0C0.C

Header File

File name: U0C0.H

Function Library (Part 1)

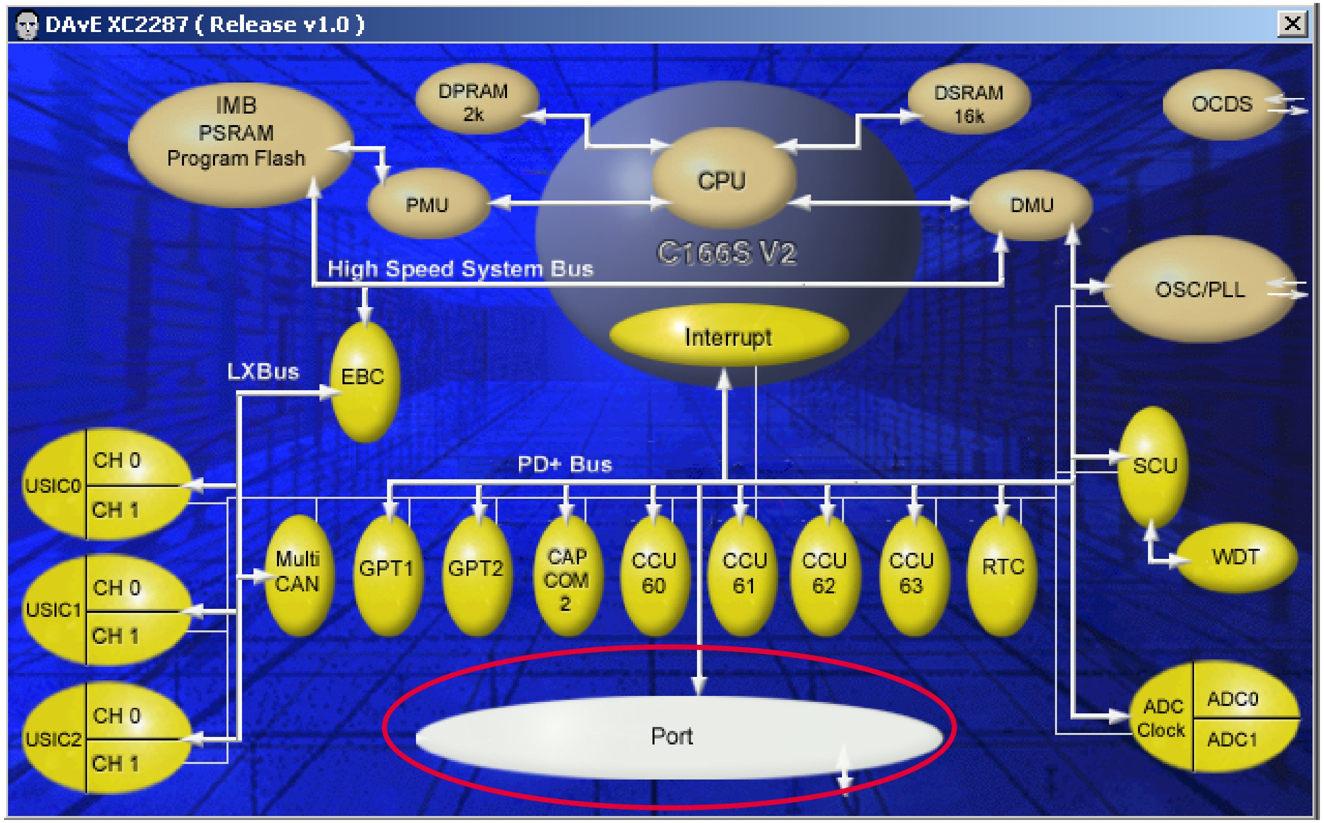
☑ U0C0_ASC_vSendData
☐ U0C0_ASC_uwGetData
☐ U0C0_ASC_ubTxDataReady
☐ U0C0_ASC_ubTxBufFree
☐ U0C0_ASC_ubRxDataReady
☑ U0C0_ASC_vi0IC
☐ U0C0_ASC_vi1IC
☐ U0C0_ASC_vi2IC
☐ U0C0_ASC_vi3IC_CC2

Function Library (Part 2)

☐ U0C0_ASC_vFillTxFIFO
☐ U0C0_ASC_uwGetRxFIFOData
☐ U0C0_ASC_ubIsRxFIFObusy
☐ U0C0_ASC_ubIsTxFIFObusy
☐ U0C0_ASC_ubIsRxFIFOfull
☐ U0C0_ASC_ubIsTxFIFOfull
☐ U0C0_ASC_ubIsRxFIFOempty
☐ U0C0_ASC_ubIsTxFIFOempty
☐ U0C0_ASC_ubGetRxFIFOFillingLevel
☐ U0C0_ASC_ubGetTxFIFOFillingLevel
☐ U0C0_ASC_vFlushRxFIFO
☐ U0C0_ASC_vFlushTxFIFO

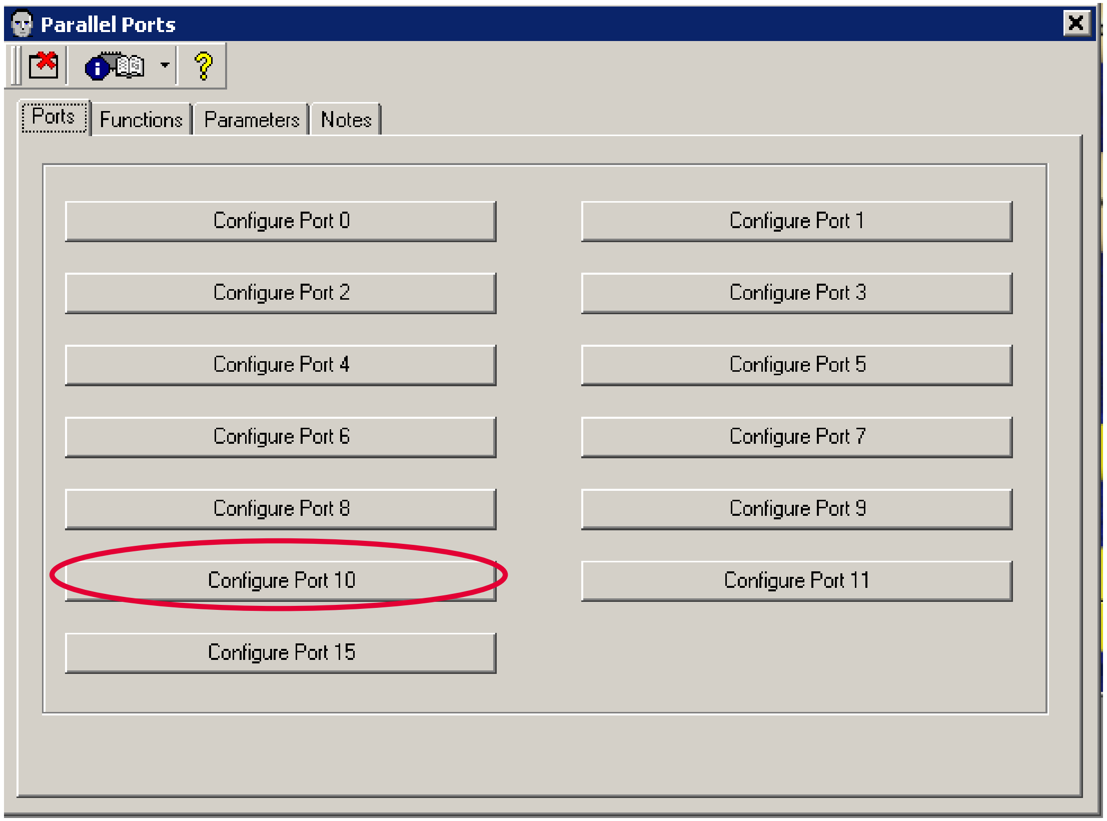
HOT Exercise ASC - DAvE Configurations
Port settings

- **XC2287M**
  - **Port:**
    - Click on the



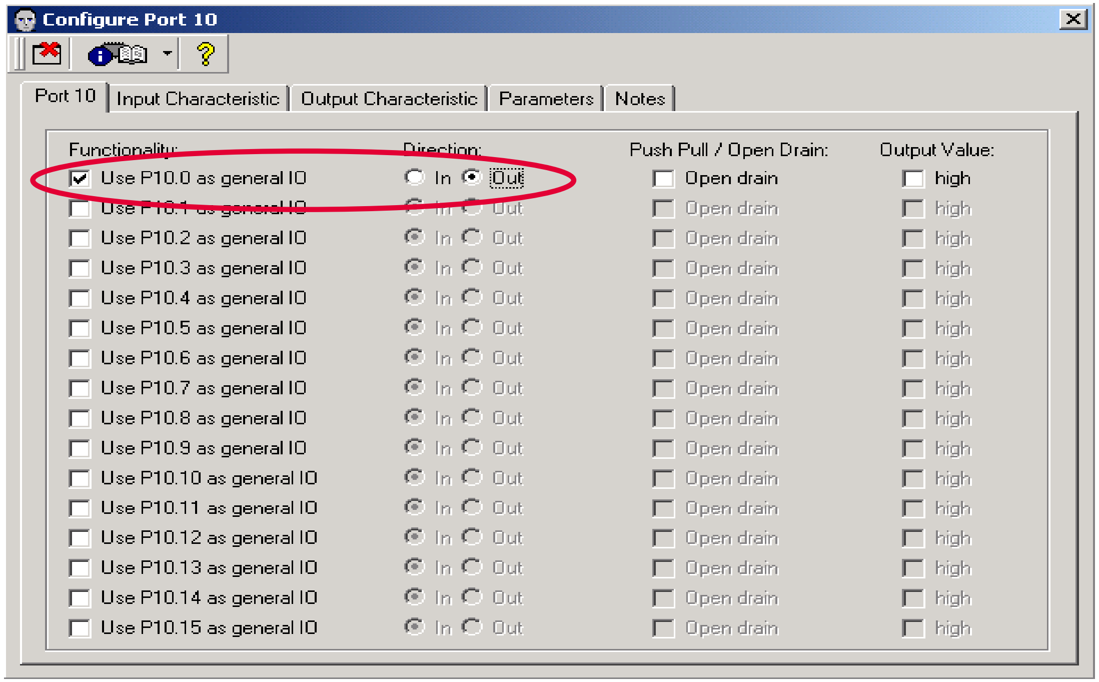
HOT Exercise ASC - DAvE Configurations
Port settings

- **Parallel Ports**
  - **Ports:**
    - Configure Port 10



HOT Exercise ASC - DAvE Configurations
Port settings

- **Configure Port 10**
  - **Port 10:**
    - Use P10.0 as general IO
    - Set Direction to Out
    - Close the window

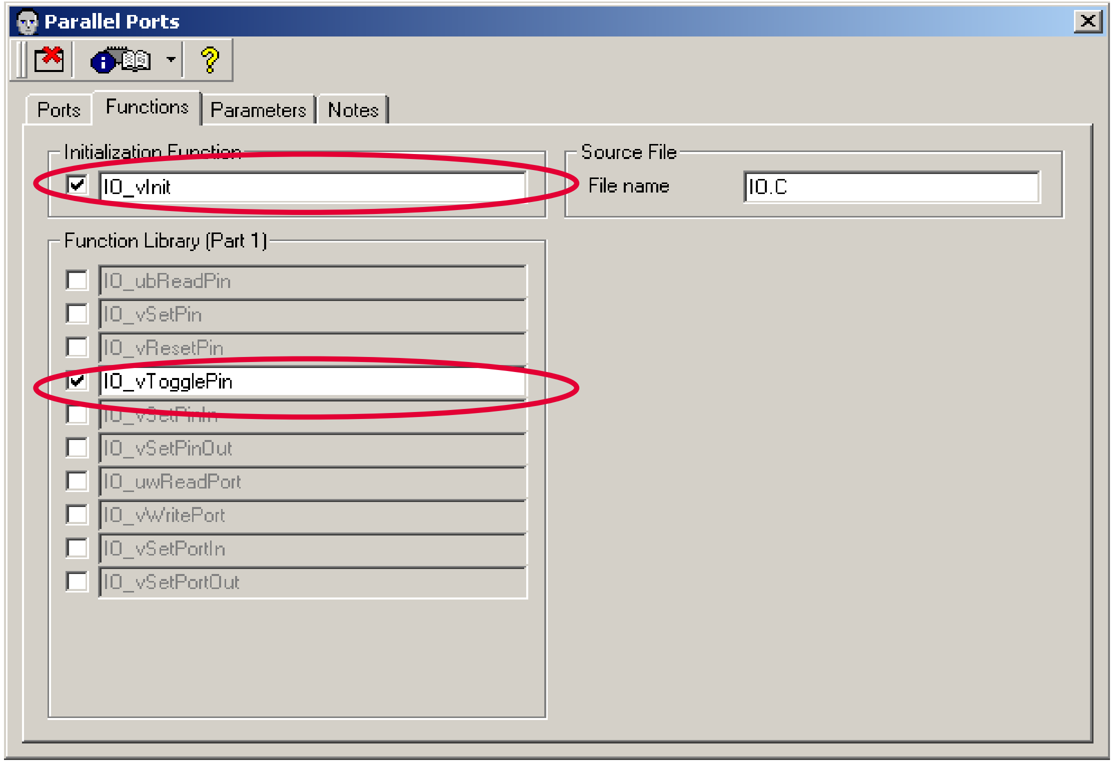


Configure Port 10

Functionality:	Direction:	Push Pull / Open Drain:	Output Value:
☑ Use P10.0 as general IO	○ In ● Out	□ Open drain	□ high
□ Use P10.1 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.2 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.3 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.4 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.5 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.6 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.7 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.8 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.9 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.10 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.11 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.12 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.13 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.14 as general IO	● In ○ Out	□ Open drain	□ high
□ Use P10.15 as general IO	● In ○ Out	□ Open drain	□ high

HOT Exercise ASC - DAvE Configurations
Port settings

- **Parallel Ports**
  - **Functions:**
    - Include ‘IO_vInit’
    - Include ‘IO_vTogglePin’



Initialization Function

☑ IO_vInit

Source File

File name: IO.C

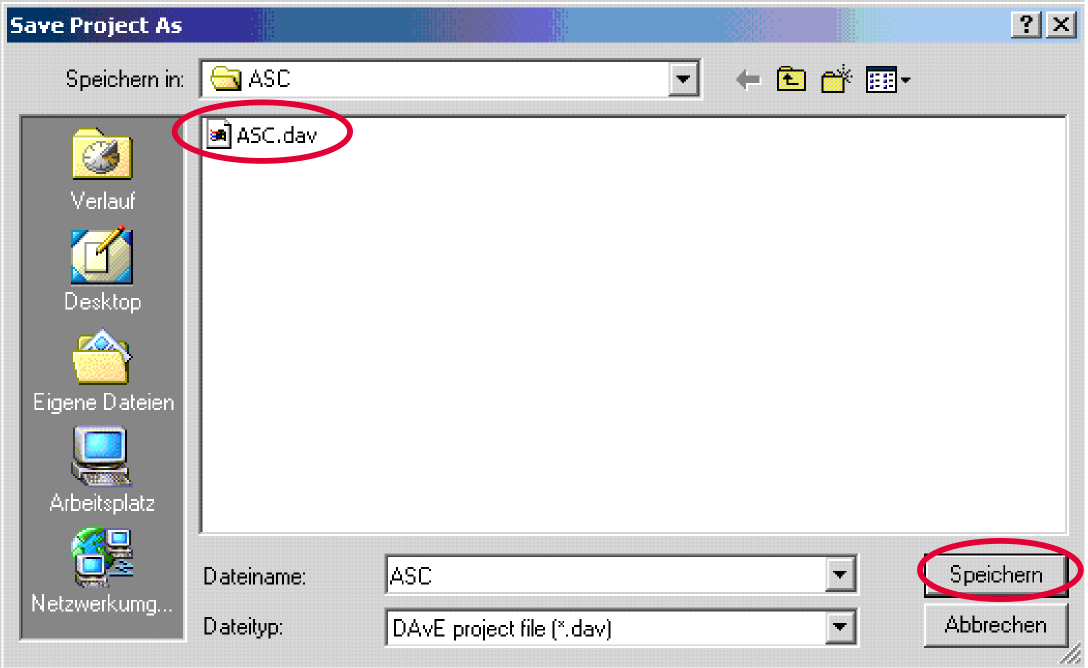
Function Library (Part 1)

☐ IO_ubReadPin
☐ IO_vSetPin
☐ IO_vResetPin
☑ IO_vTogglePin
☐ IO_vSetPinIn
☐ IO_vSetPinOut
☐ IO_uwReadPort
☐ IO_vWritePort
☐ IO_vSetPortIn
☐ IO_vSetPortOut

HOT Exercise ASC - DAvE Configurations
Save DAvE Project

- Save your DAvE Project File
  - □ Go to **File → Save (or Save As)** or click on 💾
  - □ Filename entered previously:

    “c:****IFX_HOT\XC2287M\Examples\ASC\ASC.dav**”**



Save Project As

Speichern in: ASC

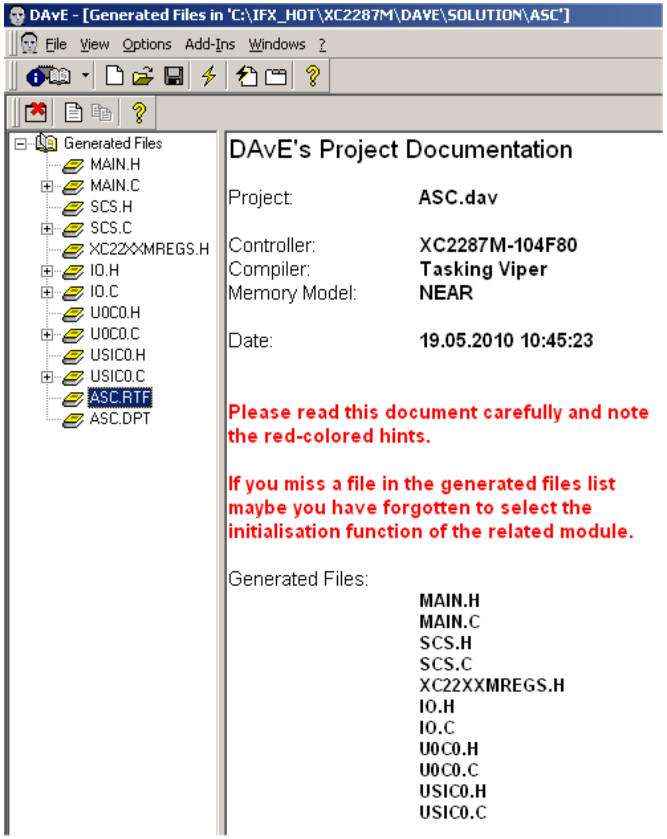
ASC.dav

Dateiname: ASC

Dateityp: DAvE project file (*.dav)

HOT Exercise ASC - DAvE Configurations Code Generation

- Let DAvE Generate Code for You
  - □ Go to **File → generate Code** or click on ⚡
  - □ DAvE generated code files are
    - ¬ MAIN.C, MAIN.H
    - ¬ U0C0.C, U0C0.H
    - ¬ USIC0.C, USIC0.H
    - ¬ IO.C, IO.H
    - ¬ SCS.C, SCS.H
    - ¬ XC22XXREGS.H



DAvE's Project Documentation

Project:	ASC.dav
Controller:	XC2287M-104F80
Compiler:	Tasking Viper
Memory Model:	NEAR
Date:	19.05.2010 10:45:23

Please read this document carefully and note the red-colored hints.

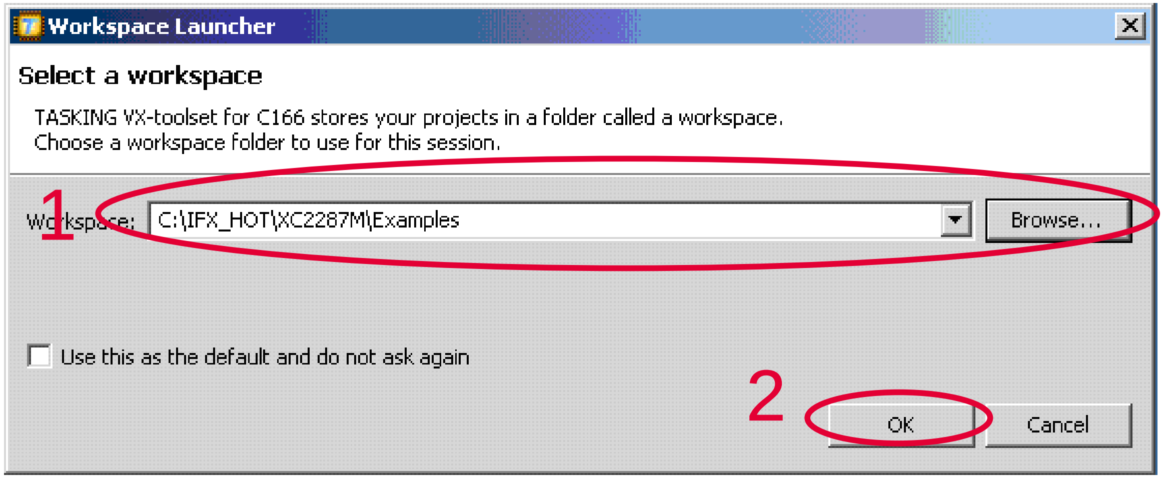
If you miss a file in the generated files list maybe you have forgotten to select the initialisation function of the related module.

Generated Files:

MAIN.H
MAIN.C
SCS.H
SCS.C
XC22XXREGS.H
IO.H
IO.C
U0C0.H
U0C0.C
USIC0.H
USIC0.C

HOT Exercise ASC – Tasking VX Toolset

- Create New Work Space
  - □ Click on VX V2.2r1
  - □ Filename: **“c:\IFX_HOT\XC2287M\Examples”**
  - □ Click ‘OK’


Workspace Launcher

Select a workspace

TASKING VX-toolset for C166 stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

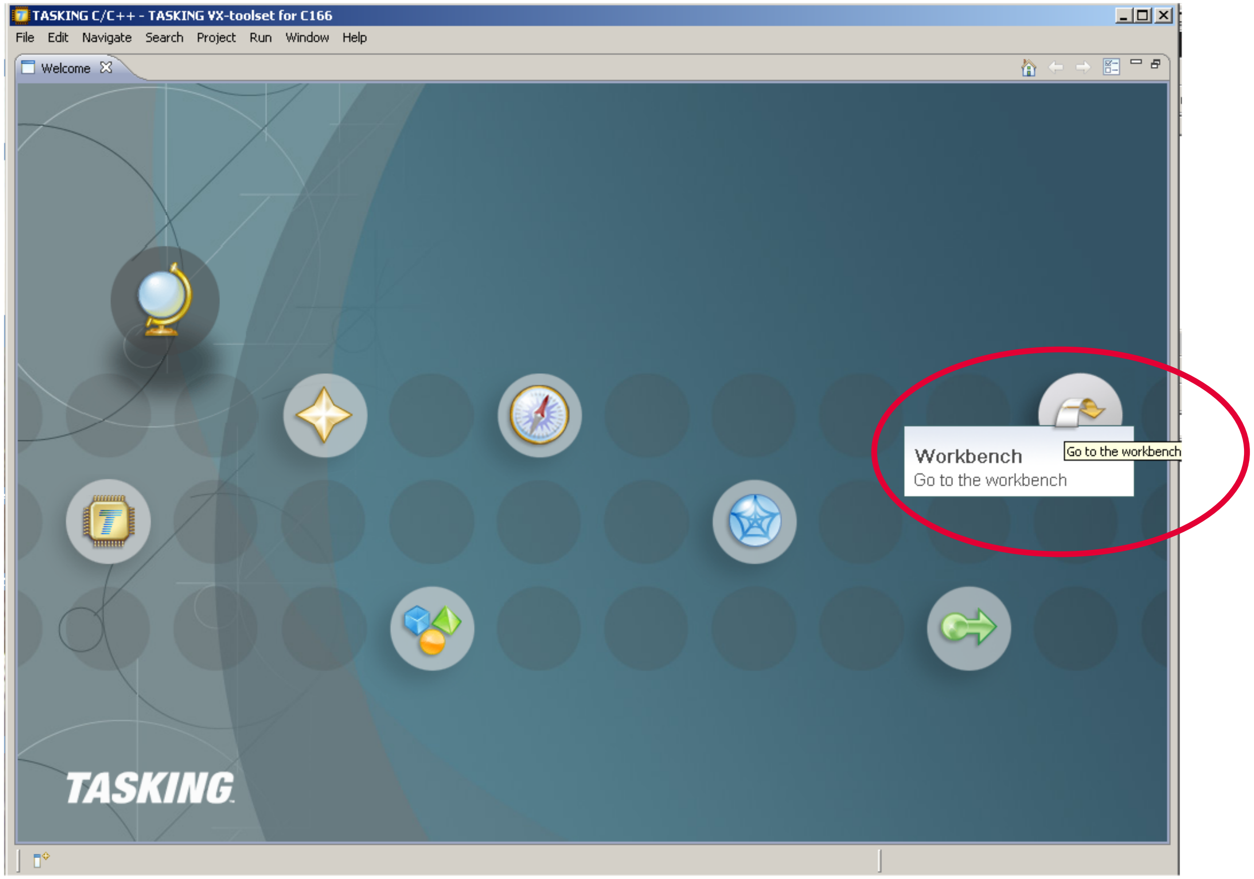
1 Workspace: C:\IFX_HOT\XC2287M\Examples

□ Use this as the default and do not ask again

2

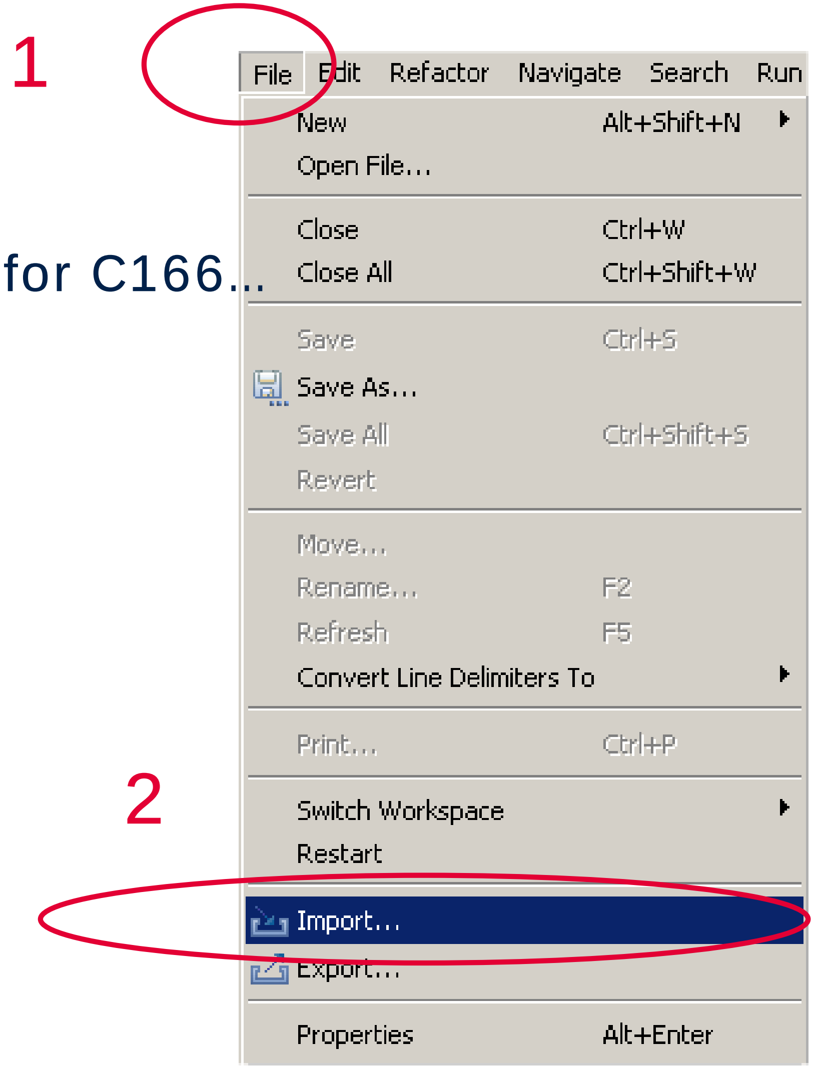
HOT Exercise ASC – Tasking VX Toolset

- Create New Project
  - □ Click on Workbench


TASKING C/C++ - TASKING VX-toolset for C166
File Edit Navigate Search Project Run Window Help
Welcome
Workbench
Go to the workbench
Go to the workbench
TASKING

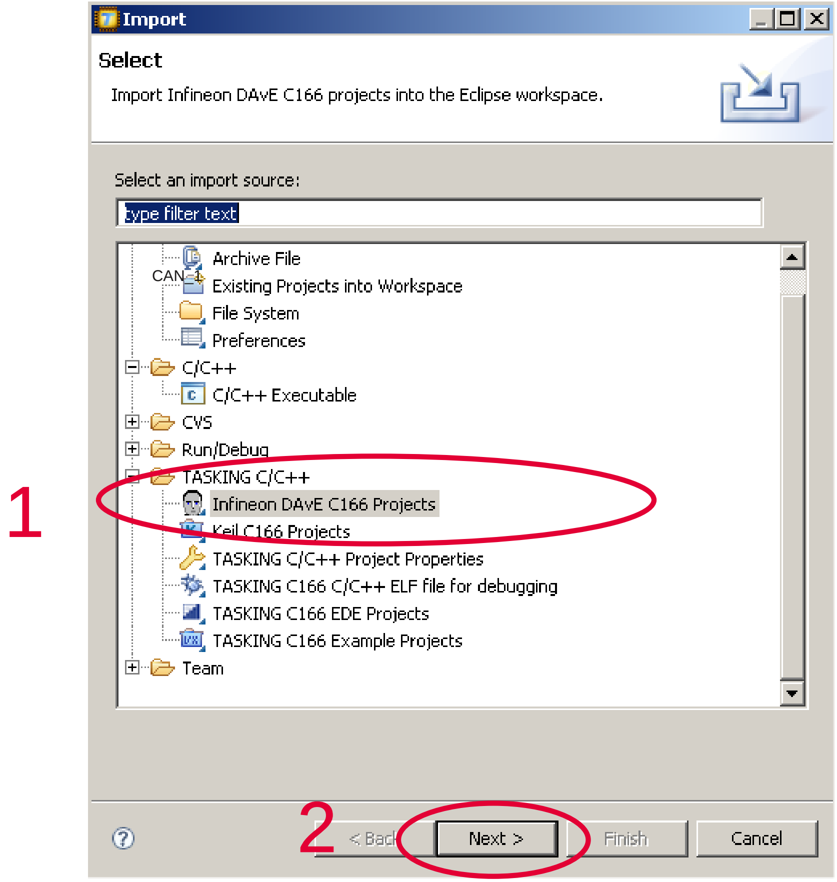
HOT Exercise ASC – Tasking VX Toolset

- Import DAVE Project
  - □ Click on File -> Import
  - □ Select Tasking VX-toolset for C166...
  - □ Click ‘OK’



HOT Exercise ASC – Tasking VX Toolset

- Import DAVE Project
  - □ Click `Infineon DAvE C166 Project´
  - □ Click ‘Next’



Import

Select

Import Infineon DAvE C166 projects into the Eclipse workspace.

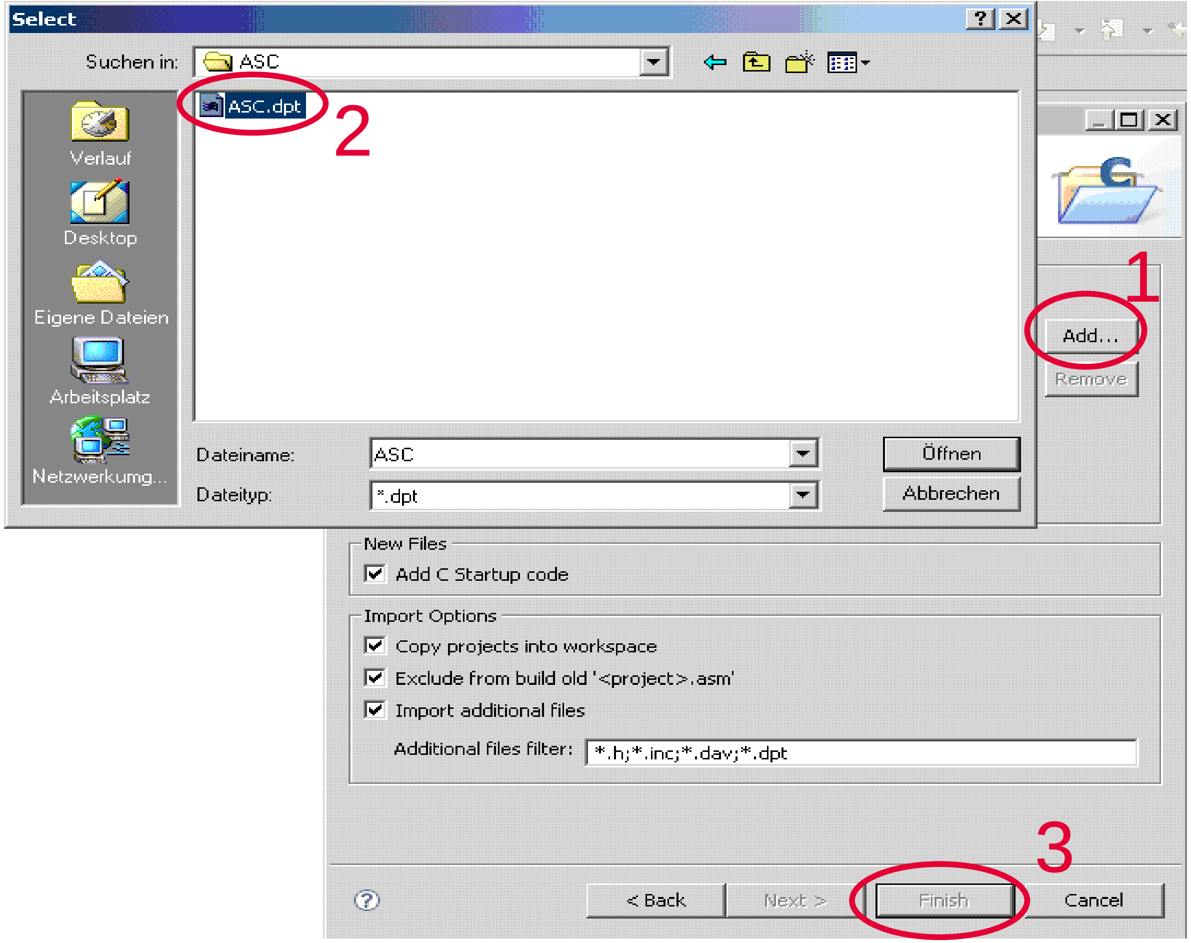
Select an import source:

type filter text

Archive File
CAN_1
Existing Projects into Workspace
File System
Preferences
C/C++
C/C++ Executable
CVS
Run/Debug
TASKING C/C++
Infineon DAvE C166 Projects
Keil C166 Projects
TASKING C/C++ Project Properties
TASKING C166 C/C++ ELF file for debugging
TASKING C166 EDE Projects
TASKING C166 Example Projects
Team
1
2
< Back
Next >
Finish
Cancel

HOT Exercise ASC – Tasking VX Toolset

- Import DAvE Project
  - □ Add Dave Project ‘ASC´
  - □ Click `Finish´

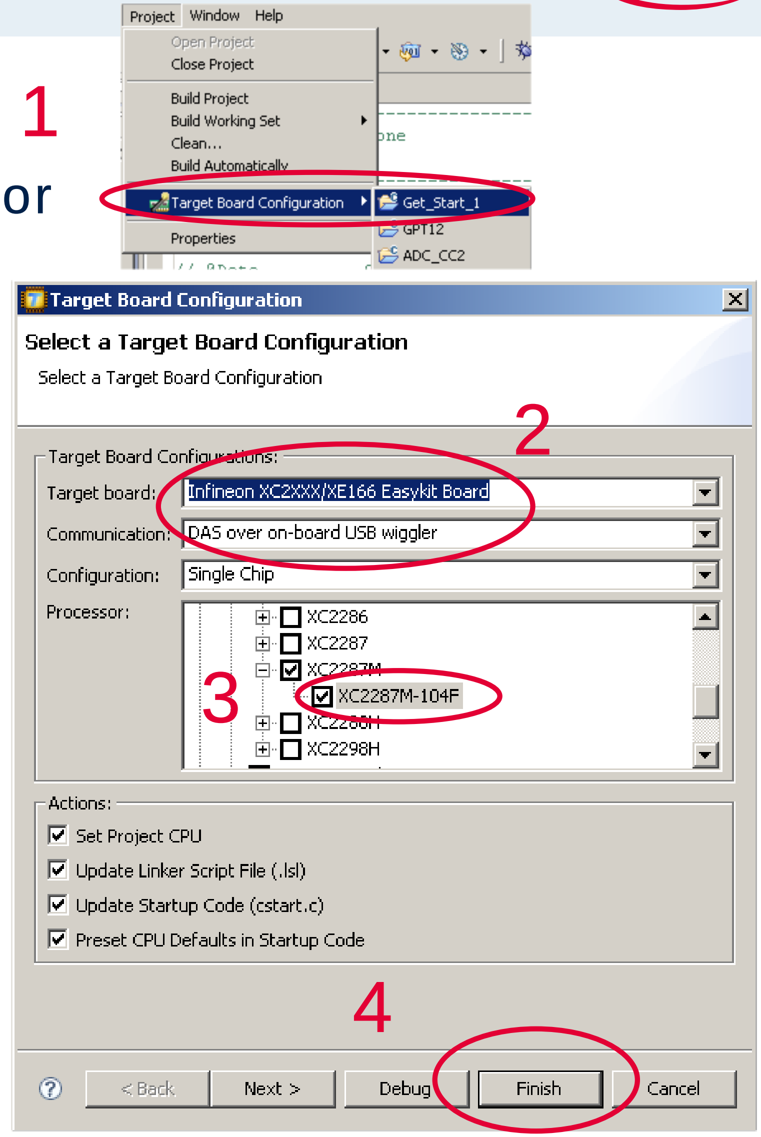


New Files
☑ Add C Startup code

Import Options
☑ Copy projects into workspace
☑ Exclude from build old '<project>.asm'
☑ Import additional files
Additional files filter: *.h;*.inc;*.dav;*.dpt

HOT Exercise ASC – Tasking VX Toolset

- Configure Target Board
  - □ Select the project in the navigator
  - □ Select ‘Project/Target Board Configuration’
  - □ Select ‘Infineon XC2000/XE166 Easykit Board’
  - □ Choose `XC2287M-104F´
  - □ Click `Finish´



Target Board Configuration

Select a Target Board Configuration

Select a Target Board Configuration

Target Board Configurations:

Target board: Infineon XC2XXX/XE166 Easykit Board

Communication: DAS over on-board USB wiggler

Configuration: Single Chip

Processor:
- □ XC2286
- □ XC2287
- ☑ XC2287M
  - ☑ XC2287M-104F
- □ XC2286H
- □ XC2298H

Actions:
- ☑ Set Project CPU
- ☑ Update Linker Script File (.lsl)
- ☑ Update Startup Code (cstart.c)
- ☑ Preset CPU Defaults in Startup Code

HOT Exercise ASC – Tasking VX Toolset

- 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.

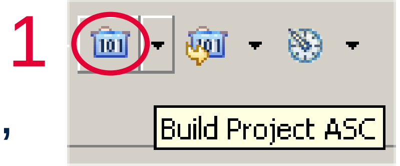
    The code you really have to add looks like this:

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

BlinkLED();

// USER CODE END
}
```

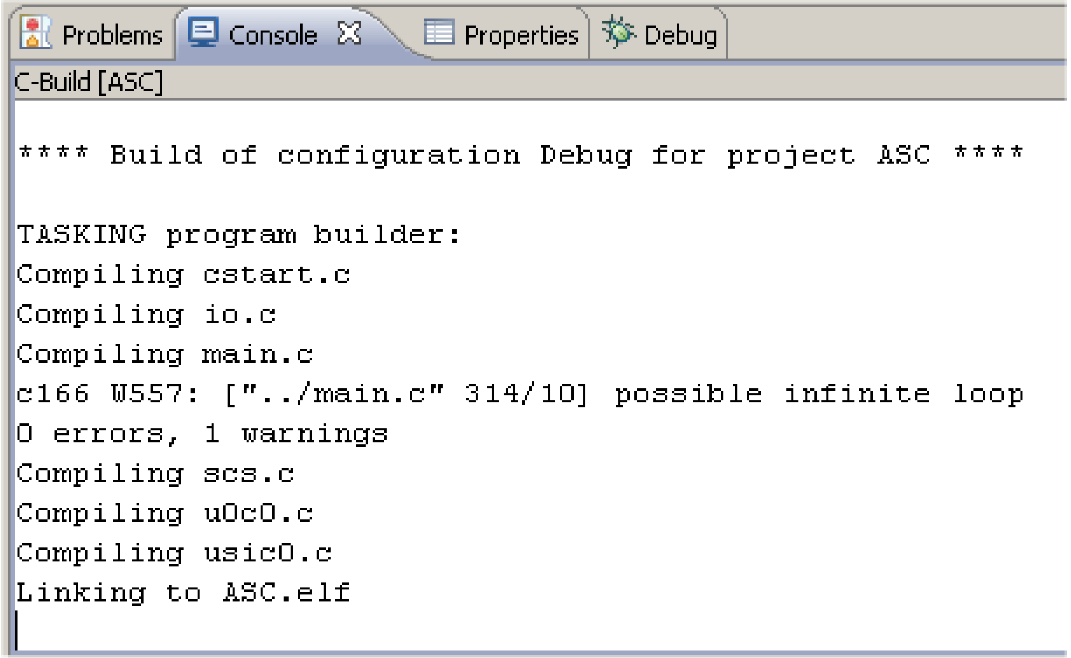
HOT Exercise ASC – Tasking VX Toolset
Add User Code – U0C0.C (ISR)

```
_interrupt(U0C0_0INT)  void U0C0_ASC_vi0IC(void)
{
  // USER CODE BEGIN (ASC0IC,2)
  // USER CODE END
  if (U0C0_PSR & 0x4000)
  {
    // USER CODE BEGIN (ASC0IC,4)
    IO_vTogglePin(IO_P10_0);
    U0C0_ASC_vSendData (U0C0_RBUF);
    // USER CODE END
    U0C0_PSCR   |= 0x4000;          // clear PSR_RIF
  }
  // USER CODE BEGIN (ASC0IC,15)
  // USER CODE END

} //  End of function U0C0_ASC_vi0IC
```

HOT Exercise ASC – Tasking VX Toolset
Build Project

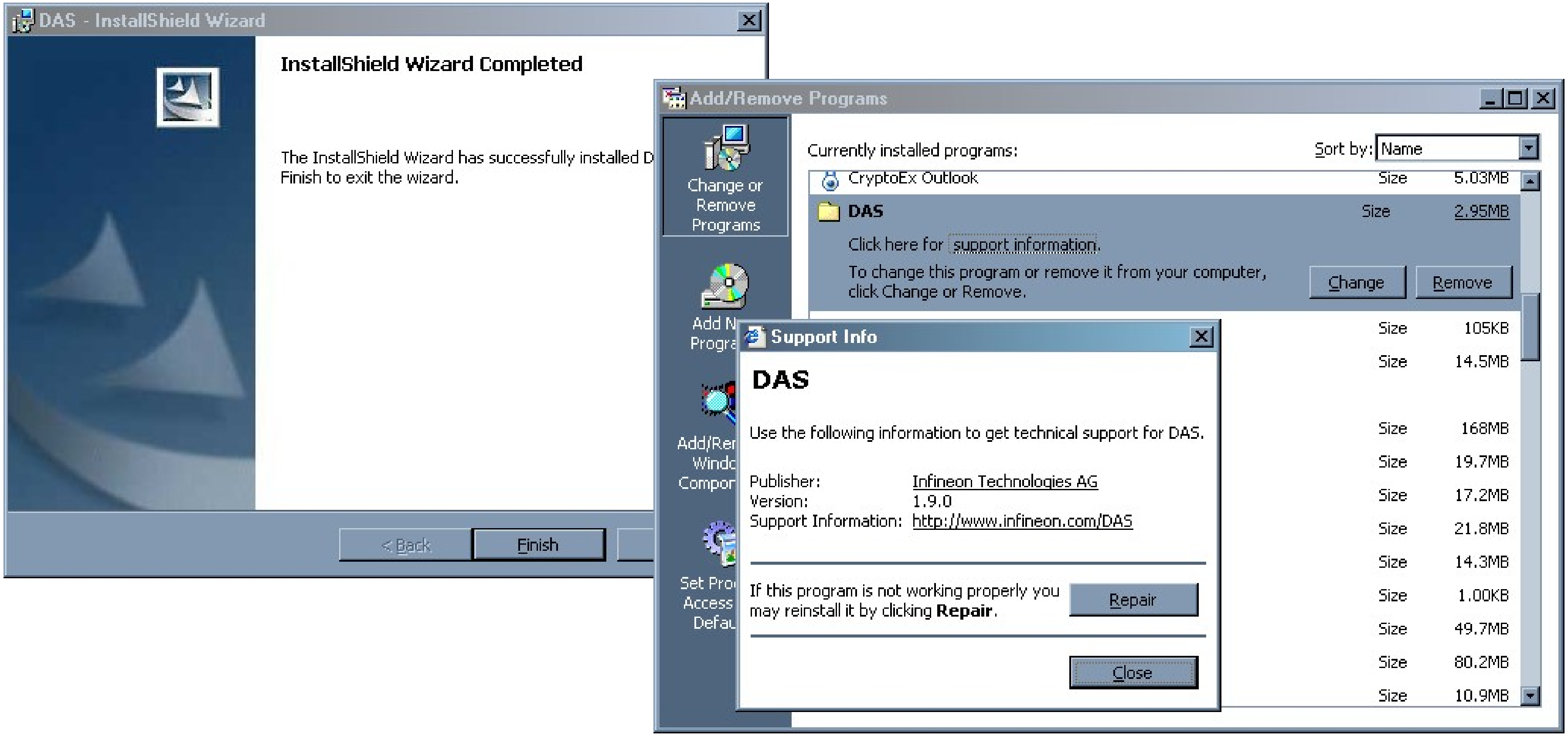


- Click on ‘Build Project ASC’



```
**** Build of configuration Debug for project ASC ****

TASKING program builder:
Compiling cstart.c
Compiling io.c
Compiling main.c
c166 W557: ["../main.c" 314/10] possible infinite loop
0 errors, 1 warnings
Compiling scs.c
Compiling u0c0.c
Compiling usic0.c
Linking to ASC.elf
```

HOT Exercise ASC - Device Access Server
Check for the latest DAS version



DAS - InstallShield Wizard

InstallShield Wizard Completed

The InstallShield Wizard has successfully installed D
Finish to exit the wizard.

Add/Remove Programs

Currently installed programs: Sort by: Name

CryptoEx Outlook	Size	5.03MB
DAS	Size	2.95MB
	Size	105KB
	Size	14.5MB
	Size	168MB
	Size	19.7MB
	Size	17.2MB
	Size	21.8MB
	Size	14.3MB
	Size	1.00KB
	Size	49.7MB
	Size	80.2MB
	Size	10.9MB

Click here for support information.

To change this program or remove it from your computer, click Change or Remove.

Support Info

DAS

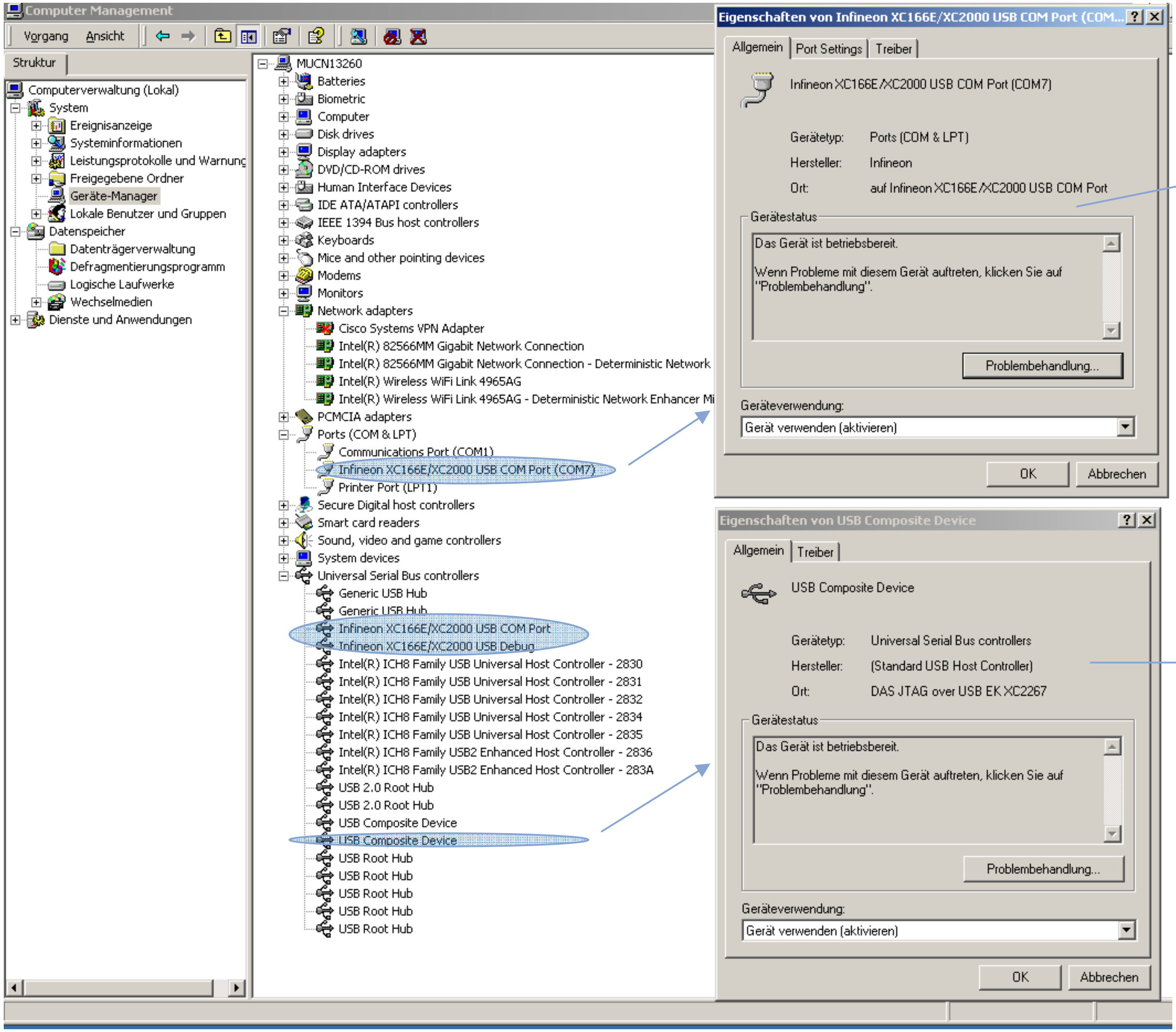
Use the following information to get technical support for DAS.

Publisher: Infineon Technologies AG
Version: 1.9.0
Support Information: http://www.infineon.com/DAS

If this program is not working properly you may reinstall it by clicking **Repair**.

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

HOT Exercise ASC - Device Access Server
1.) Checking USB connections



Computerverwaltung (Lokal)
- System
  - Ereignisanzeige
  - Systeminformationen
  - Leistungsprotokolle und Warnung
  - Freigegebene Ordner
  - Geräte-Manager
  - Lokale Benutzer und Gruppen
- Datenspeicher
  - Datenträgerverwaltung
  - Defragmentierungsprogramm
  - Logische Laufwerke
  - Wechselmedien
- Dienste und Anwendungen

MUCN13260
- Batteries
- Biometric
- Computer
- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Human Interface Devices
- IDE ATA/ATAPI controllers
- IEEE 1394 Bus host controllers
- Keyboards
- Mice and other pointing devices
- Modems
- Monitors
- Network adapters
  - Cisco Systems VPN Adapter
  - Intel(R) 82566MM Gigabit Network Connection
  - Intel(R) 82566MM Gigabit Network Connection - Deterministic Network
  - Intel(R) Wireless WiFi Link 4965AG
  - Intel(R) Wireless WiFi Link 4965AG - Deterministic Network Enhancer Mi
- PCMCIA adapters
- Ports (COM & LPT)
  - Communications Port (COM1)
  - Infineon XC166E/XC2000 USB COM Port (COM7)
  - Printer Port (LPT1)
- Secure Digital host controllers
- Smart card readers
- Sound, video and game controllers
- System devices
- Universal Serial Bus controllers
  - Generic USB Hub
  - Generic USB Hub
  - Infineon XC166E/XC2000 USB COM Port
  - Infineon XC166E/XC2000 USB Debug
  - Intel(R) ICH8 Family USB Universal Host Controller - 2830
  - Intel(R) ICH8 Family USB Universal Host Controller - 2831
  - Intel(R) ICH8 Family USB Universal Host Controller - 2832
  - Intel(R) ICH8 Family USB Universal Host Controller - 2834
  - Intel(R) ICH8 Family USB Universal Host Controller - 2835
  - Intel(R) ICH8 Family USB2 Enhanced Host Controller - 2836
  - Intel(R) ICH8 Family USB2 Enhanced Host Controller - 283A
  - USB 2.0 Root Hub
  - USB 2.0 Root Hub
  - USB Composite Device
  - USB Composite Device
  - USB Root Hub
  - USB Root Hub
  - USB Root Hub
  - USB Root Hub
  - USB Root Hub

Eigenschaften von Infineon XC166E/XC2000 USB COM Port (COM...

Infineon XC166E/XC2000 USB COM Port (COM7)

Gerätetyp: Ports (COM & LPT)

Hersteller: Infineon

Ort: auf Infineon XC166E/XC2000 USB COM Port

Gerätestatus

Das Gerät ist betriebsbereit.

Wenn Probleme mit diesem Gerät auftreten, klicken Sie auf "Problembehandlung".

Geräteverwendung: Gerät verwenden (aktivieren)

Eigenschaften von USB Composite Device

USB Composite Device

Gerätetyp: Universal Serial Bus controllers

Hersteller: (Standard USB Host Controller)

Ort: DAS JTAG over USB EK XC2267

Gerätestatus

Das Gerät ist betriebsbereit.

Wenn Probleme mit diesem Gerät auftreten, klicken Sie auf "Problembehandlung".

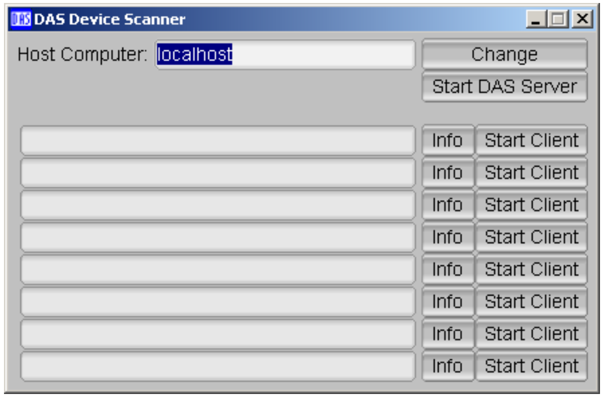
Geräteverwendung: Gerät verwenden (aktivieren)

This gets identified only when COM port is used
- Via the USB interface on the Easykit with FTDI chip

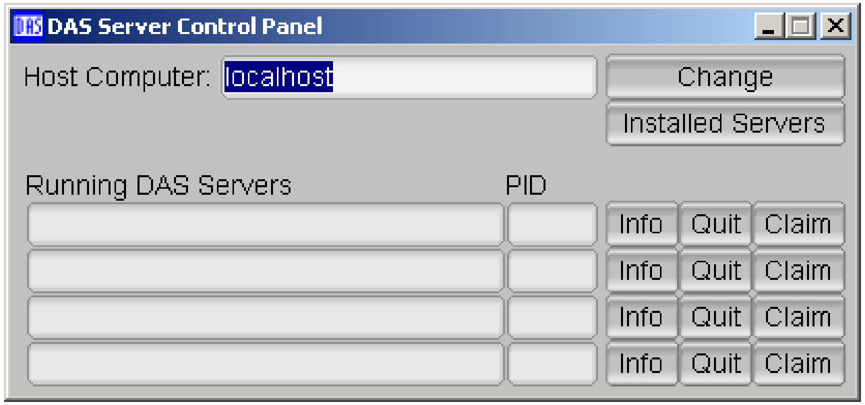
The DAS JTAG composite device gets identified
- When miniWiggler is connected
- When USB Wiggler Box is connected
- Via the USB interface on the Easykits with FTDI chip

HOT Exercise ASC - Device Access Server
2.) Check DAS status

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

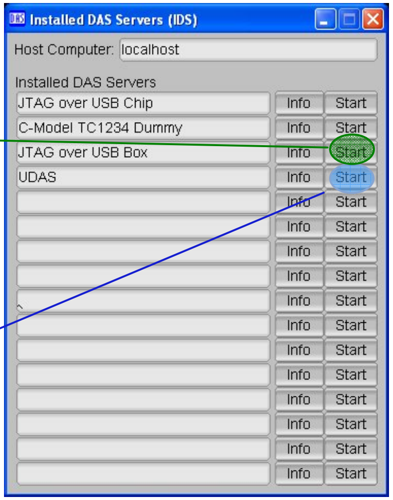


Host Computer: localhost



Host Computer: localhost

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



Host Computer: localhost

JTAG over USB Chip

C-Model TC1234 Dummy

JTAG over USB Box

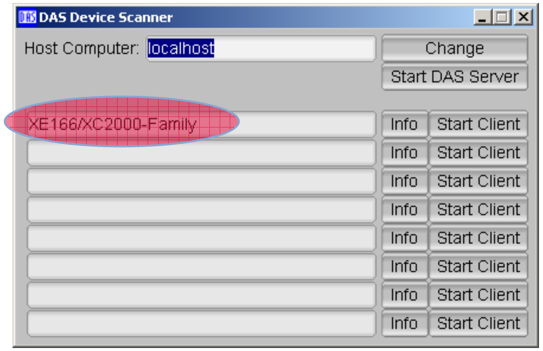
UDAS

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“

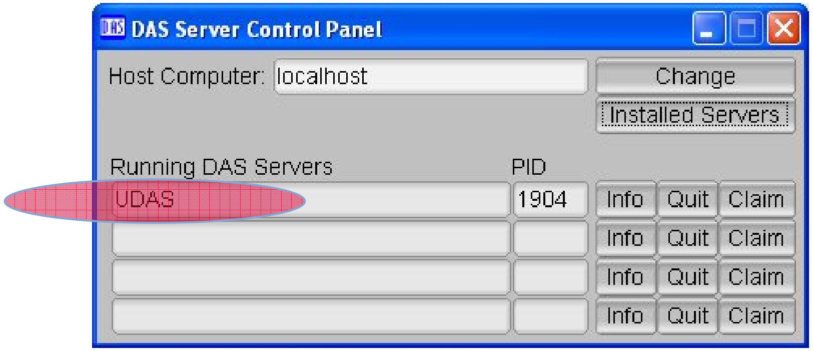
HOT Exercise ASC - Device Access Server
3.) Starting the servers manually

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


DAS Device Scanner

Host Computer: localhost

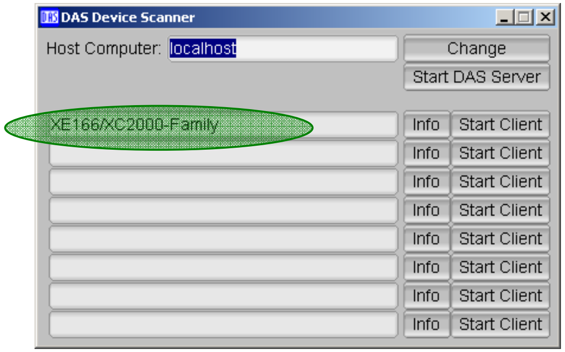
XE166/XC2000-Family


DAS Server Control Panel

Host Computer: localhost

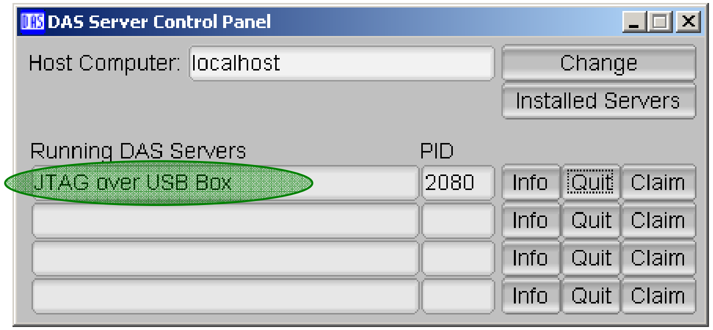
Running DAS Servers	PID
UDAS	1904

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


DAS Device Scanner

Host Computer: localhost

XE166/XC2000-Family

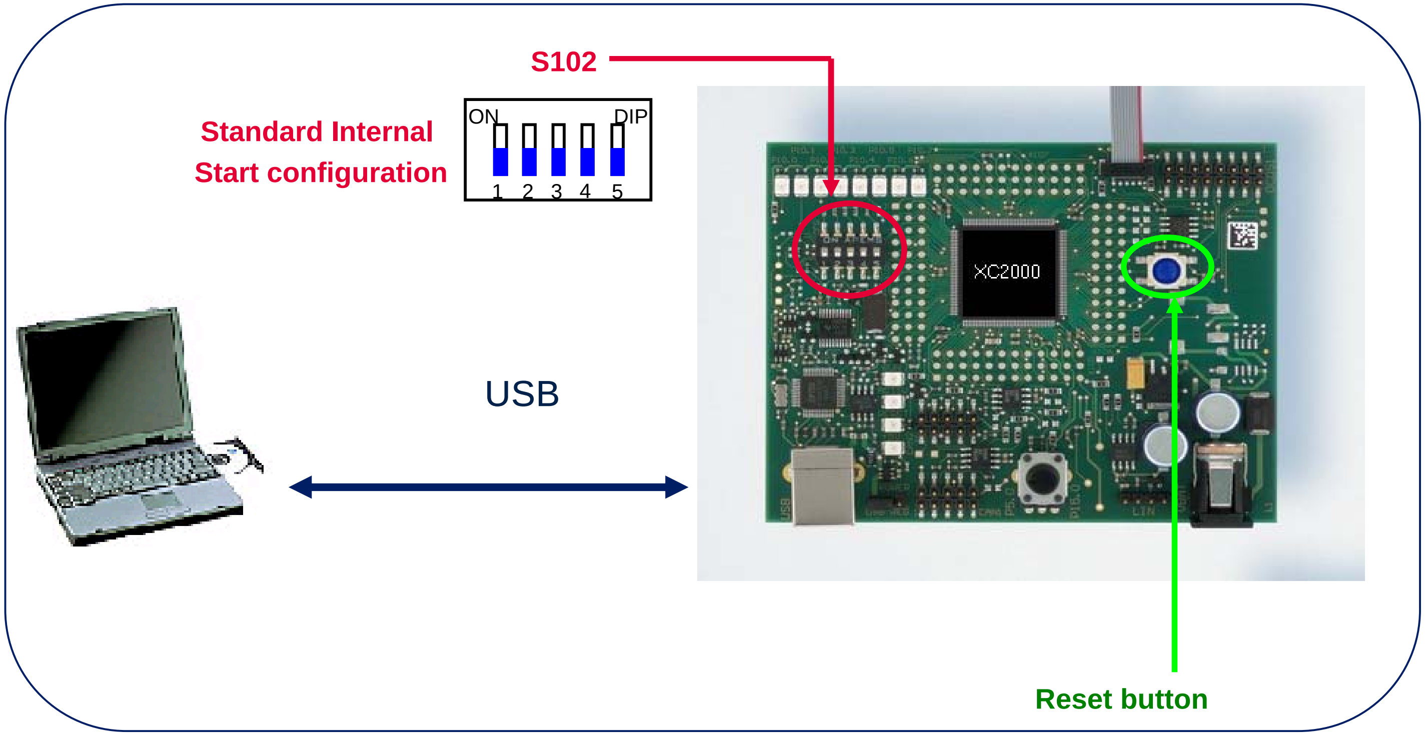

DAS Server Control Panel

Host Computer: localhost

Running DAS Servers	PID
JTAG over USB Box	2080

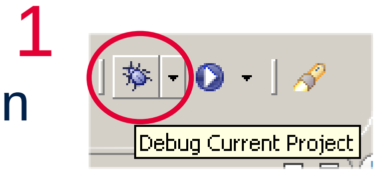
HOT Exercise ASC – Tasking VX Toolset Connect XC2287M Easy Kit

- Connect XC2287M Board to PC
- Modify The DIP Switch Settings, S102: **OFF-OFF-OFF-OFF-OFF** (Start from Internal Flash)
- Reset The Board (Press The Reset Button)

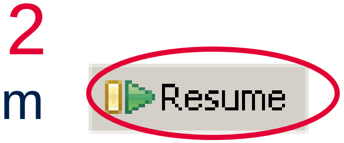


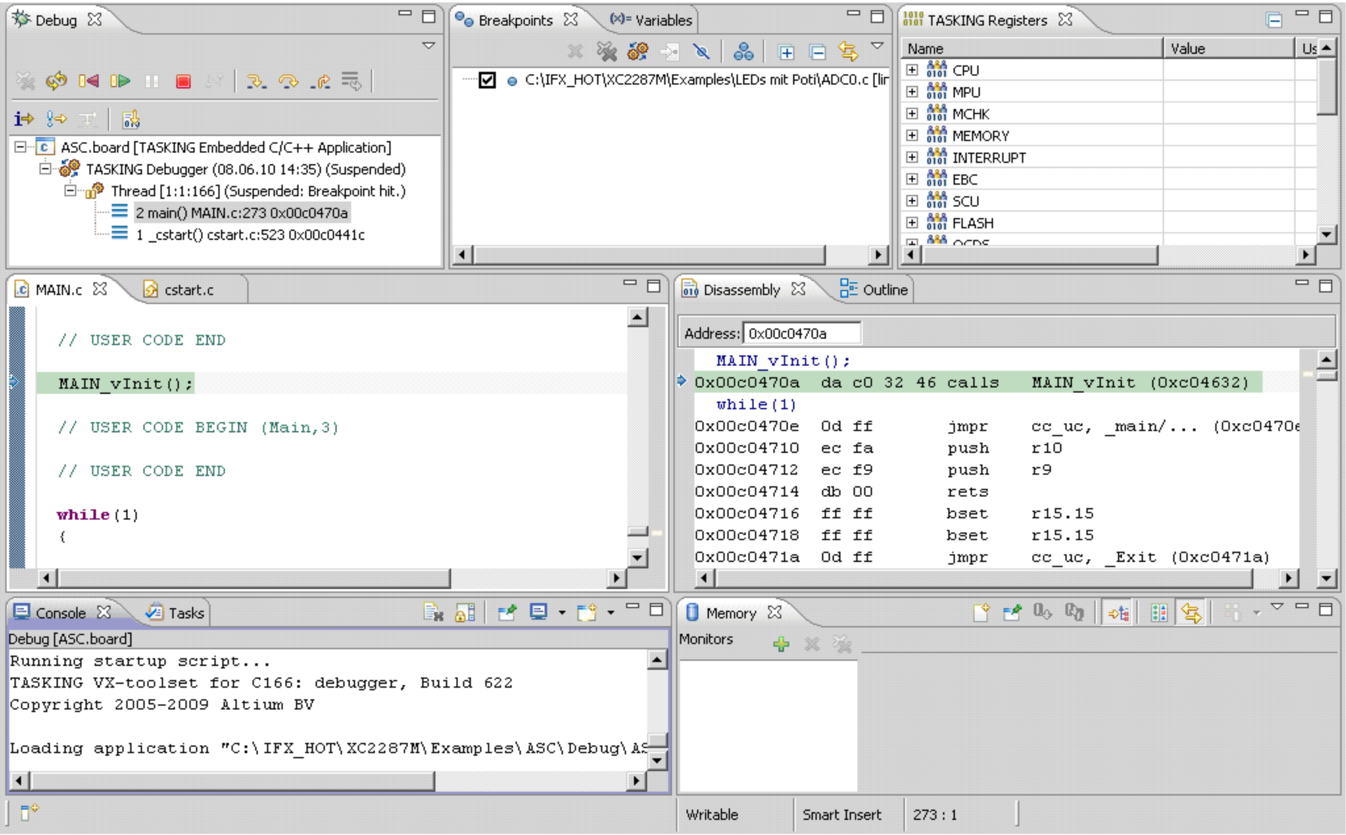
HOT Exercise ASC – Tasking VX Toolset
Run Debugger

- Click on



- Click on ‘Resume’ and start program





```
ASC.board [TASKING Embedded C/C++ Application]
  TASKING Debugger (08.06.10 14:35) (Suspended)
    Thread [1:1:166] (Suspended: Breakpoint hit.)
      2 main() MAIN.c:273 0x00c0470a
      1 _cstart() cstart.c:523 0x00c0441c
```

☑ ● C:\IFX_HOT\XC2287M\Examples\LEDs mit Poti\ADC0.c [lir

Name	Value	Us
⊞ CPU		
⊞ MPU		
⊞ MCHK		
⊞ MEMORY		
⊞ INTERRUPT		
⊞ EBC		
⊞ SCU		
⊞ FLASH		
⊞ OCDS		

```
// USER CODE END

MAIN_vInit();

// USER CODE BEGIN (Main,3)

// USER CODE END

while(1)
{
```

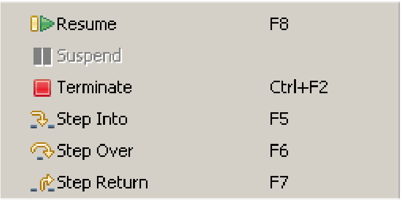
Address: 0x00c0470a

```
  MAIN_vInit();
0x00c0470a  da c0 32 46 calls   MAIN_vInit (0xc04632)
  while(1)
0x00c0470e  0d ff       jmpr    cc_uc, _main/... (0xc0470e
0x00c04710  ec fa       push    r10
0x00c04712  ec f9       push    r9
0x00c04714  db 00       rets
0x00c04716  ff ff       bset    r15.15
0x00c04718  ff ff       bset    r15.15
0x00c0471a  0d ff       jmpr    cc_uc, _Exit (0xc0471a)
```

Debug [ASC.board]

```
Running startup script...
TASKING VX-toolset for C166: debugger, Build 622
Copyright 2005-2009 Altium BV

Loading application "C:\IFX_HOT\XC2287M\Examples\ASC\Debug\AS
```

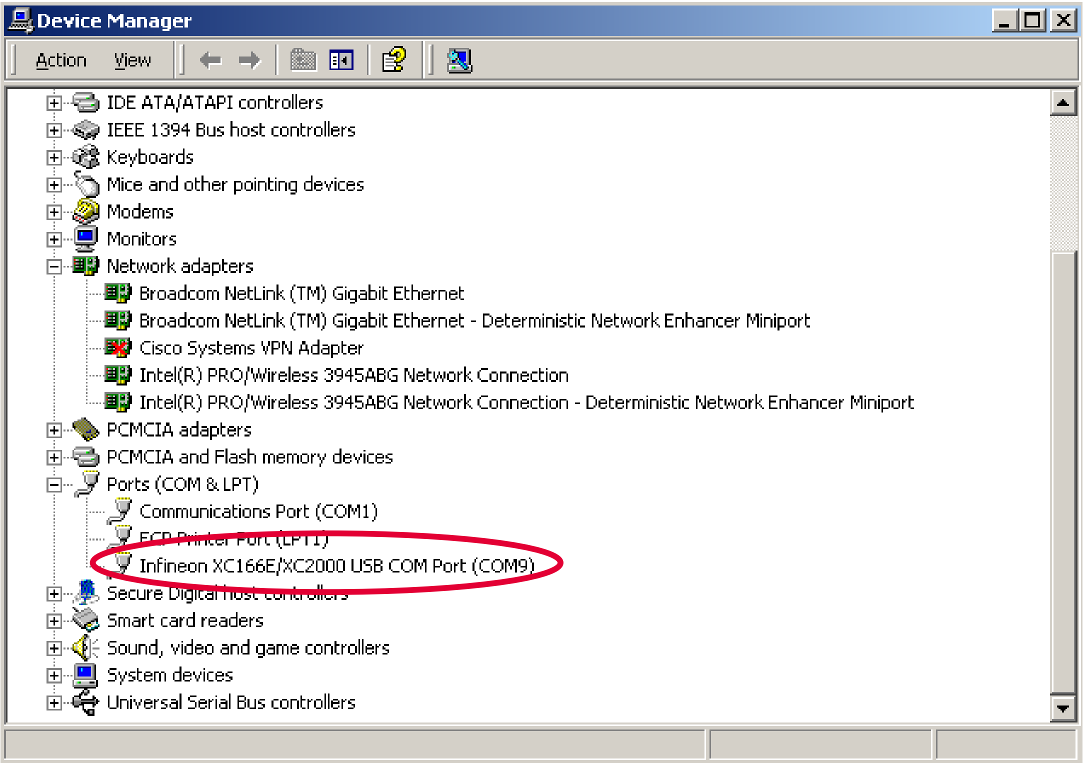
Writable | Smart Insert | 273 : 1



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

HOT Exercise ASC
Start HyperTerminal

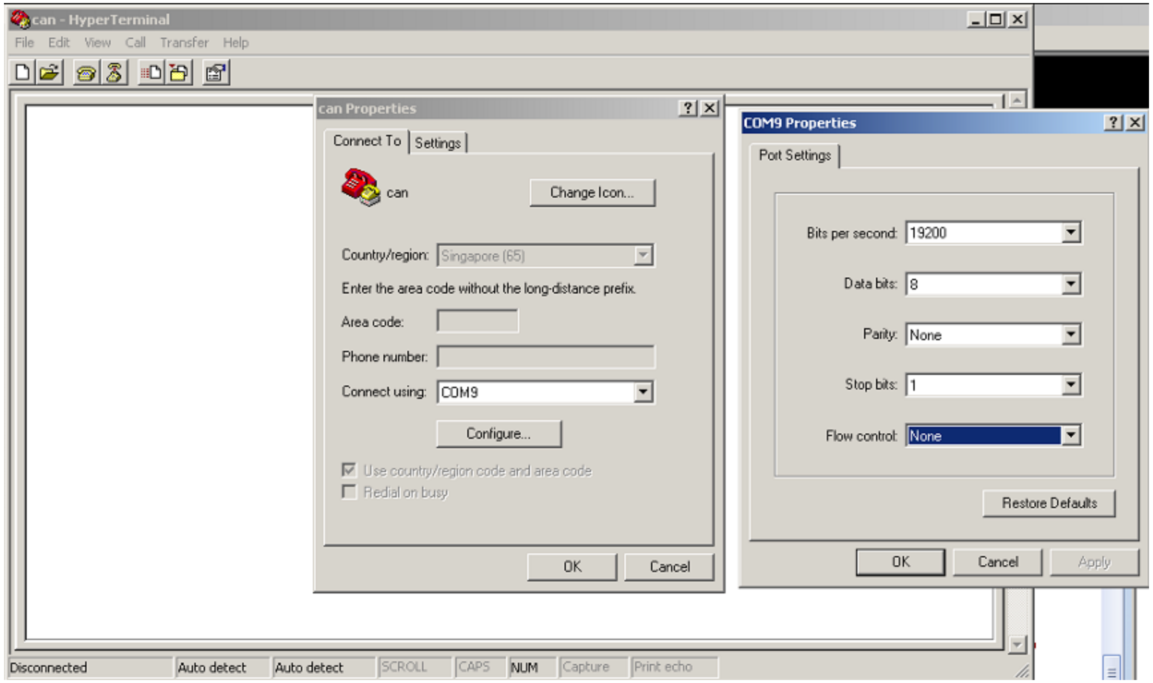
- With the FTDI chip an on board, USB interface can be used for UART. FTDI device will converts the USB protocol the ASC protocol. Both USB and UART can be used at the same time.
- Open Device Manger and check which COM port is activated for the FTDI chip



```
+ IDE ATA/ATAPI controllers
+ IEEE 1394 Bus host controllers
+ Keyboards
+ Mice and other pointing devices
+ Modems
+ Monitors
- Network adapters
    Broadcom NetLink (TM) Gigabit Ethernet
    Broadcom NetLink (TM) Gigabit Ethernet - Deterministic Network Enhancer Miniport
    Cisco Systems VPN Adapter
    Intel(R) PRO/Wireless 3945ABG Network Connection
    Intel(R) PRO/Wireless 3945ABG Network Connection - Deterministic Network Enhancer Miniport
+ PCMCIA adapters
+ PCMCIA and Flash memory devices
- Ports (COM & LPT)
    Communications Port (COM1)
    ECP Printer Port (LPT1)
    Infineon XC166E/XC2000 USB COM Port (COM9)
+ Secure Digital host controllers
+ Smart card readers
+ Sound, video and game controllers
+ System devices
+ Universal Serial Bus controllers
```

HOT Exercise ASC
Start HyperTerminal

1. Start->Programs->Accessories->Communications->HyperTerminal
2. Enter any name and click ‘OK’
3. Connect using: COMx (COM port activated for the FTDI chip)
4. Click ‘Configure’ to enter Port settings
5. Select 19200 baud, no Parity, 8 Data Bits and 1 Stop Bit
6. Click ‘OK’



Country/region: Singapore (65)

Enter the area code without the long-distance prefix.

Area code:

Phone number:

Connect using: COM9

☑ Use country/region code and area code

☐ Redial on busy

Bits per second: 19200

Data bits: 8

Parity: None

Stop bits: 1

Flow control: None

Disconnected | Auto detect | Auto detect | SCROLL | CAPS | NUM | Capture | Print echo

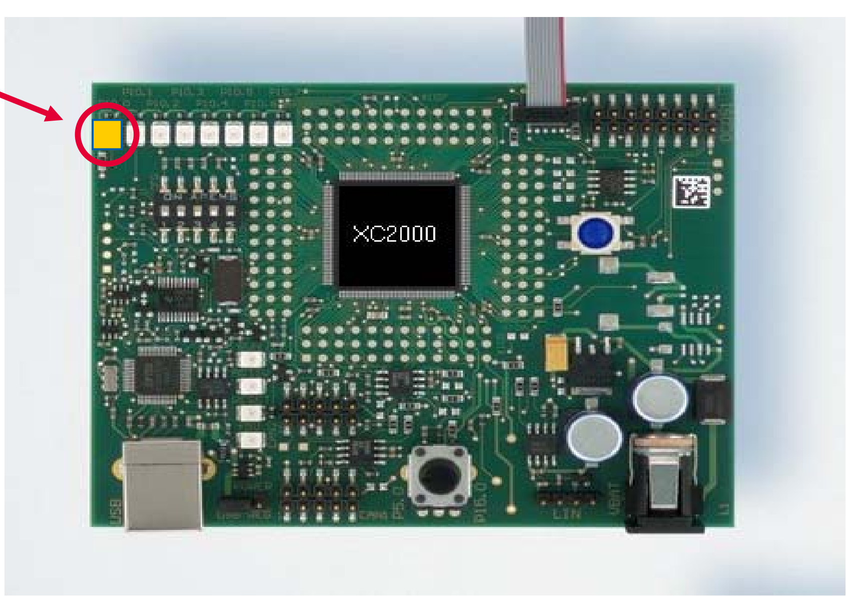
HOT Exercise ASC
Running the program

- **Start typing**
  - □ Enter ASCII characters in the HyperTerminal
  - □ The characters you enter are sent to the XC2287M and back to the Terminal Program so that you can read them on the screen
  - □ The characters are not sent directly from the keyboard to the screen!
    - ¬ Hold the reset button down to verify that the screen no longer displays the typed characters

HOT Exercise ASC
See Result

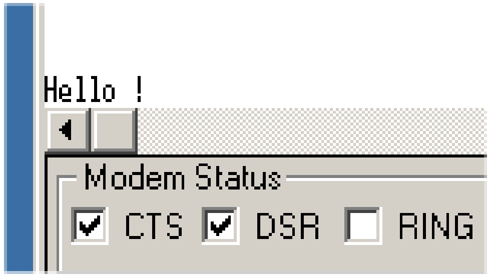
- The yellow LED will toggle when the ASC sending the data back

LED blinking



HOT Exercise ASC – Tasking VX Toolset
Run Debugger

- Verifications
  - □ Click on ‘Suspend’
  - □ Click on ‘Restart’
  - □ The LED no longer toggles when a key is pressed
  - □ Click on ‘Resume’
  - □ LED toggles again when a key is pressed



Hello !

Modem Status

☑ CTS ☑ DSR □ RING

infineon

Never stop thinking