

QuickStart Instructions

phyCORE[®] - i.MX31 Rapid Development Kit for Windows Embedded CE

**Using Microsoft Visual Studio 2005
Software Development Tool Chain**

[Note: The PHYTEC phyCORE[®]-i.MX31 Rapid Development Kit CD includes the electronic version of the English phyCORE[®]-i.MX31 Hardware Manual](#)

Edition: April 2009

In this manual copyrighted products are not explicitly indicated. The absence of the trademark (™) and copyright (©) symbols does not imply that a product is not protected. Additionally, registered patents and trademarks are similarly not expressly indicated in this manual.

The information in this document has been carefully checked and is believed to be entirely reliable. However, PHYTEC Messtechnik GmbH assumes no responsibility for any inaccuracies. PHYTEC Messtechnik GmbH neither gives any guarantee nor accepts any liability whatsoever for consequential damages resulting from the use of this manual or its associated product. PHYTEC Messtechnik GmbH reserves the right to alter the information contained herein without prior notification and accepts no responsibility for any damages which might result.






Additionally, PHYTEC Messtechnik GmbH offers no guarantee nor accepts any liability for damages arising from the improper usage or improper installation of the hardware or software. PHYTEC Messtechnik GmbH further reserves the right to alter the layout and/or design of the hardware without prior notification and accepts no liability for doing so.

© Copyright 2009 PHYTEC Messtechnik GmbH, 55129 Mainz.

Rights - including those of translation, reprint, broadcast, photomechanical or similar reproduction and storage or processing in computer systems, in whole or in part - are reserved. No reproduction may be made without the explicit written consent from PHYTEC Messtechnik GmbH.

	EUROPE	NORTH AMERICA
Address:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 55129 Mainz GERMANY	PHYTEC America LLC 203 Parfitt Way SW, Suite G100 Bainbridge Island, WA 98110 USA
Ordering Information:	+49 (800) 0749832 order@phytec.de	1 (800) 278-9913 sales@phytec.com
Technical Support:	+49 (6131) 9221-31 support@phytec.de	1 (800) 278-9913 support@phytec.com
Fax:	+49 (6131) 9221-33	1 (206) 780-9135
Web Site:	http://www.phytec.de	http://www.phytec.com

1st Edition: March 2009

1	Introduction	1	
1.1	Rapid Development Kit Documentation	1	
1.2	Professional Support Packages Available	1	5 min
1.3	Overview of this QuickStart Instruction	2	
1.4	Conventions used in this QuickStart	2	
1.5	Kit Contents	4	
2	Software Installation	1	
2.1	Installation Sequence	1	
2.2	Installing the phyCORE [®] -i.MX31 Rapid Development Kit CD	2	
2.3	Installing Visual Studio 2005 Development Tool Chain	3	4-5 h
2.4	Installing Windows Embedded CE 6.0 Platform Builder	5	
2.5	Installing Service Packs and WinCE 6.0 Platform Builder R2	6	
2.6	Installing Windows Embedded CE 6.0 Monthly Updates (QFEs)	6	
2.7	Installing Microsoft Active Sync 4.5	6	
2.8	Installing the phyCORE [®] -i.MX31 Binary BSP	7	
2.9	Installing phyCORE [®] -i.MX31 SDK	8	
3	Getting Started	9	
3.1	Boot Windows CE on the phyCORE [®] -i.MX31	9	
3.2	Establishing an Active Sync Connection	10	
3.3	Downloading Example Code with Active Sync	11	30 min
3.4	Downloading Example Code with Visual Studio 2005	13	
4	Getting More Involved	16	
4.1	Creating a New Project	16	
4.2	Modifying the Source Code	20	
4.3	Building the Project	22	40 min
5	Debugging	23	
5.1	Starting the Debugger	23	
5.2	Visual Studio 2005 Debug Features	25	
5.3	Using the Visual Studio 2005 Debug Features	26	50 min
5.4	Running, Stopping and Resetting	26	
5.5	Changing Target Settings for Release	27	
6	Building a Customized Image	28	
6.1	Building an Image	28	
Appendix A	Installing the Bootloader	30	
Appendix B	Installing the WinCE Binary Image	34	

1 Introduction

This chapter provides an overview of the kit documentation, technical support, QuickStart Instructions, document conventions, and kit contents.



5 min

Please refer to the [phyCORE[®]-i.MX31 Hardware Manual](#) for specific information on such board-level features as [jumper configuration](#), [memory mapping](#) and [pin layout](#). Selecting the links on the electronic version of this document leads to the respective section of the phyCORE[®]-i.MX31 Hardware Manual.

1.1 Rapid Development Kit Documentation

This Rapid Development Kit (RDK) includes the following electronic documentation on the enclosed phyCORE[®]-i.MX31 Rapid Development Kit CD:

- the PHYTEC [phyCORE[®]-i.MX31 Hardware Manual](#)
- controller [User's Manuals and Data Sheets](#)
- this QuickStart Instruction with general Rapid Development Kit description, software installation advice and an example program, enabling quick out-of-the box start-up of the phyCORE[®]-i.MX31 in conjunction with the Microsoft Visual Studio 2005 software development tool chain

1.2 Professional Support Packages Available

This Kit comes with free installation support. If you do have any questions concerning installation and setup, you are welcome to contact our support department.

For more in-depth questions, we offer a variety of custom tailored packages with different support options (e-mail, phone, direct contact to the developer) and different reaction times.

Please contact our sales team to discuss the appropriate support option if professional support beyond installation and setup is important to you.

1.3 Overview of this QuickStart Instruction

This QuickStart Instruction gives a general "Rapid Development Kit" description, as well as software installation advice and one example program enabling quick out-of-the box start-up of the phyCORE[®]-i.MX31 in conjunction with Microsoft's Visual Studio 2005. It is structured as follows:

1. The **Introduction** section provides an overview of the kit documentation, technical support, QuickStart Instructions, document conventions, and kit contents.
2. The **Software Installation** section guides you through all steps required for the phyCORE[®]-i.MX31 WinCE kit setup.
3. In the **Getting Started** section you will learn how to download user code to the target device from a host-PC using Visual Studio 2005 with Active Sync.
4. The **Getting More Involved** section provides step-by-step instructions on how to create and build a new project and generate and download output files to the phyCORE[®]-i.MX31 using Visual Studio 2005 with Active Sync.
5. The **Debugging** section demonstrates simple debug functions using Visual Studio 2005 debug environment.
6. In the **Building an OS Image** section you will learn how to add your own application to a WinCE project, create a new image and download this image on the phyCORE[®]-i.MX31.

1.4 Conventions used in this QuickStart

The following is a list of the typographical conventions used in this book:

Bold	Commands or other text that should be typed literally by the user
<i>Bold Italic</i>	File and directory names and paths
<i>Italic</i>	Field and window names or titles, menu items, and other terms that correspond to items on the PC desktop and Windows
<u>Underline Blue</u>	Hyperlinks to documents, webpage URLs, or FTPs
Bold Red	Command line entries
Courier New	Source code listings and examples

Pay special attention to notes set apart from the text with the following icons:



At this point you might leave the sequential path of this QuickStart.



This warning will help you avoid problems.



Useful supplementary information about the topic.



Estimated completion time for the following section.



Completion of an important part of this QuickStart.



Helpful information for troubleshooting.

1.5 Kit Contents

The following PHYTEC hardware components are included in the phyCORE[®]-i.MX37 Windows Embedded CE Rapid Development Kit (part number KPCM-037-WinCE) and are necessary for completing the instructions in this QuickStart instruction:

- phyCORE[®]-i.MX31 (PCM-037)
- phyCORE[®]-i.MX31 Carrier Board (PCM-970)
- phyCORE[®]-i.MX31 Mapper Board (PMA-001)
- Bare PCB Expansion Board (PCM-977)¹
- Sharp LCD screen
- AC adapter supplying 5 VDC /3.2A adapter, center positive
- Cross-over Ethernet cable
- Serial cable (RS-232)
- USB Standard A to mini-B cable
- SOM extraction tool
- Hard copy schematics
- phyCORE[®]-i.MX31 Rapid Development Kit CD
- Windows Embedded CE 6.0 180-Day Evaluation Kit

¹ The Bare PCB Expansion Board is not demonstrated in this QuickStart instruction.

2 Software Installation

In this chapter you will install all of the software tools necessary to begin development with Windows Embedded CE.



4-5 hr

2.1 Installation Sequence

It is important to install the software tools in the following sequence:

1. phyCORE[®]-i.MX31 Rapid Development Kit CD
2. Visual Studio 2005
3. Windows Embedded CE 6.0 Platform Builder
4. Visual Studio 2005 Team Suite Service Pack 1
5. Windows Embedded CE 6.0 Platform Builder Service Pack 1
6. Windows Embedded CE 6.0 R2
7. Windows Embedded CE 6.0 QFE's
8. ActiveSync 4.5
9. phyCORE[®]-i.MX31 Binary BSP
10. phyCORE[®]-i.MX31 SDK

2.2 Installing the phyCORE[®]-i.MX31 Rapid Development Kit CD

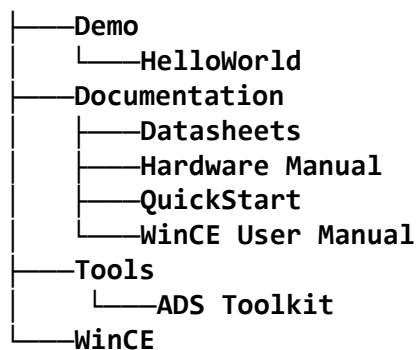
The Rapid Development Kit CD will auto install kit documentation, datasheets, and demo files for the phyCORE[®]-i.MX31. When you insert the Rapid Development Kit CD into the CD-ROM drive of your host-PC, the CD should automatically launch a setup program. Otherwise the setup program (setup.exe) can be manually executed from the root folder of the CD.



All path and file statements within this QuickStart Instruction are based on the assumption that you keep C:\PHYTEC as the default path when installing the Rapid Development Kit CD to your hard drive.

1. Insert the phyCORE-i.MX31 Rapid Development Kit CD.
2. On the *Welcome to the phyCORE-i.MX31 Rapid Development Kit Setup Wizard* window, select **Next** to continue.
3. Keep *C:\PHYTEC* as the installation path and select **Next** to continue.
4. On the *Ready to Install* window, select **Install**.
5. After the installation process is complete, select **Finish**.

The Rapid Development Kit CD installation path is *C:\PHYTEC\phyCORE-i.MX31* and this directory will have the following structure:



2.3 Installing Visual Studio 2005 Development Tool Chain

Microsoft's Software development tools for the i.MX31 architecture aid every level of developer from the professional applications engineer to the student just learning about embedded software development.

Visual Studio 2005 is a new version of Microsoft's popular IDE that also supports embedded devices. The Platform Builder is now integrated in Visual Studio 2005, so it is possible to customize the WinCE Image and write user applications with just one tool. Visual Studio 2005 combines project management, source code editing, program debugging, and target programming in a single, powerful environment. This QuickStart provides an overview of the most commonly used VS 2005 features including:

- Project management, device setup, and tool configuration
- Editor facilities for creating, modifying, and correcting programs
- Target debugging
- Building a new image

The VS 2005 editor offers many standard and advanced software editing features like:

- **Automatic completion** of code statements
- **Auto listing** of object members
- Detailed **Syntax Highlighting**. Colors are used in printed output.
- **Automatic indent** and **tab blocks** of code

The Visual Studio 2005 Tool chain is available from your local Microsoft dealer. There is also an evaluation version available which is used in this QuickStart. You can use it to develop standard WinCE applications, MFC WinCE applications and .NET Applications for WinCE.



Even if you already have installed another version of Visual Studio, we recommend installing Visual Studio 2005 when working with this QuickStart.

To install Visual Studio 2005 on your desktop PC insert the DVD labelled **Visual Studio 2005 Professional Edition** included in this kit in your DVD Rom drive. The setup should launch automatically. If not, open a Windows file explorer, go to your DVD drive, change to the directory vs and start *setup.exe* manually.

The following screen appears:



Click on **Install Visual Studio 2005** and follow the instructions of the setup program.

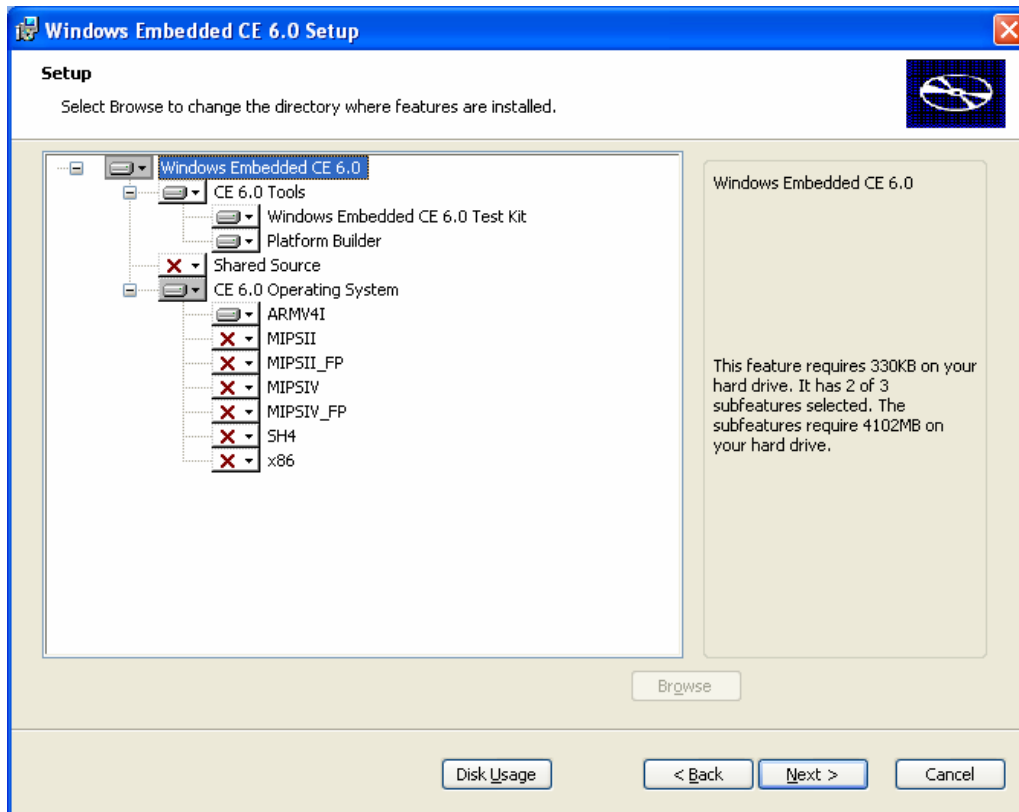


*You will need activation keys for both Visual Studio and Platform Builder to perform this installation. You can request these keys from Microsoft. Please read the inside of the **Windows Embedded CE 6.0 R2 DVD package** for information.*

2.4 Installing Windows Embedded CE 6.0 Platform Builder

To install the Platform Builder Plug-in for Visual Studio 2005, insert the DVD labeled **Windows Embedded CE6.0** in your DVD drive. The setup should start automatically. If not, open the windows file explorer, go to your DVD drive and start *setup.exe* manually.

When the following dialog appears, be sure to select **Platform Builder** and under *CE6.0 Operating System* the files for **ARMV4I**.



2.5 Installing Service Packs and WinCE 6.0 Platform Builder R2

When the Visual Studio installation has finished, you must install Service Packs for both Visual Studio 2005 and Windows Embedded CE 6.0. Additionally, you must install Windows Embedded CE 6.0 R2.

Insert the DVD labelled **Windows Embedded CE6.0 R2** in your DVD drive. The installation will begin automatically.

1. Install Microsoft Visual Studio 2005 Team Suite Service Pack 1
2. Install Windows Embedded CE 6.0 Platform Builder Service Pack 1
3. Install Windows Embedded CE 6.0 R2

2.6 Installing Windows Embedded CE 6.0 Monthly Updates (QFEs)

After installing the Visual Studio service pack, you must install updates for Platform Builder. Install the monthly updates starting with the rollup package through the end of 2008 from the following site. Be sure to install the packages for **ARM4I**, and to install the packages starting with the oldest one.

<http://msdn.microsoft.com/en-us/embedded/aa731256.aspx>

2.7 Installing Microsoft Active Sync 4.5

To be able to upload the software you are developing to your i.MX31 device you need Microsoft Active Sync 4.5. When your PC is connected to your i.MX31 device via a USB cable this software allows you to create a direct connection to your device.

You can download **Microsoft Active Sync 4.5** from this link:

<http://www.microsoft.com/windowsmobile/activesync/activesync45.msp>

To download the software, follow the instructions given on the webpage and save the downloaded file to your hard drive. Select the folder you saved the software to and double-click *setup.exe*.

Chose a folder and click **Next** and in the following dialog click **Install** to start the installation process for Microsoft Active Sync 4.5. This may take a few minutes.



You have successfully installed the software for the WinCE-i.MX31-Kit. You can find the programs you will need to develop own applications for the target on your host system. All necessary configurations were done by the setup program.

2.8 Installing the phyCORE[®]-i.MX31 Binary BSP

Adeneo is the official partner of PHYTEC America for all Windows Embedded CE developments for the phyCORE[®]-i.MX31.

Download the most recent phyCORE[®]-i.MX31 Binary BSP and Demo Image release from Adeneo FTP:

[DOWNLOAD HERE](#)



*Be sure to download the most current BSP. At the time this QuickStart was created, version AD8.2.0A2 (**iMX31_AD8.2.0A2_Binary.zip** and **iMX31_AD8.2.0A2_DemoImage.zip**) was the most current BSP release from Adeneo.*

- Copy the zip file **iMX31_AD8.2.0A2_Binary.zip** to:
C:\PHYTEC\phyCORE-i.MX31\WinCE
- Copy the zip file **iMX31_AD8.2.0A2_DemoImage.zip** to:
C:\PHYTEC\phyCORE-i.MX31\WinCE
- Unzip the packages
- Navigate to:
C:\PHYTEC\phyCORE-i.MX31\WinCE \ iMX31_AD8.2.0A2_Binary\BSP
- Open **iMX_Phytec_BIN_Setup.msi**
- Click **Next** at the welcome page
- Click **Next** to agree to User Licensing Agreement
- Install the BSP at the default location **C:\WINCE600\PLATFORM** and click **Next**
- Click **Next** to confirm installation
- Click **Close** after successful installation

2.9 Installing phyCORE[®]-i.MX31 SDK

The i.MX31 SDK for WinCE is needed for writing target-oriented applications. It will integrate in the Visual Studio IDE, offering a new target device for code generation. Also the SDK includes all functionality that is included in the image that runs on the i.MX31. Therefore using the SDK allows the compiler to check if all functionality that is needed in the application will be present in the image. The absence of specific features is therefore recognized at “compile-time” not at “runtime”.

- navigate to *C: PHYTEC\phyCORE-i.MX31\WinCE\iMX31_AD8.2.0A2_Binary*
- open *iMX31_SDK.msi*
- Click **Next** at the welcome page
- Select **Accept** to agree to User Licensing Agreement and click **Next**
- Enter *Customer Information* and click **Next**
- Select **Complete** install and click **Next**
- Install the BSP at the default and click **Next**
- Click **Next** to begin the installation
- Click **Finish** after successful installation



You have successfully installed the software for the WinCE-i.MX31-Kit. You can find the programs you will need to develop own applications for the target on your host system. All necessary configurations were done by the setup program.

3 Getting Started

In this Chapter you will learn how to download user code to the target device from a host-PC using Visual Studio 2005 and Active Sync.



30 min

3.1 Boot Windows CE on the phyCORE[®]-i.MX31

Connect the AC adapter with the power supply connector X30 (5V) on your board.



The power connector should have 5 VDC inside, and outside should be ground.



If Windows CE doesn't start within 1 minute, you may have a kit without Windows CE pre-installed. Please see 0 before continuing further.

When Windows CE has loaded, it will show a screen to calibrate the touch panel. Click on the cross on the screen to calibrate your touch panel. When calibration is done you will see the WinCE screen.

3.2 Establishing an Active Sync Connection

The i.MX31 device should be powered and connected to the PC as described in section 3.1.

ActiveSync should already be running after the installation in section 2.7.

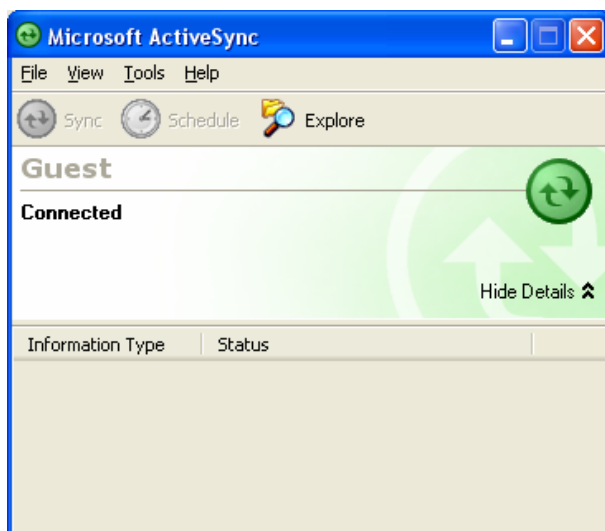
Connect the USB A/mini-B cable with the USB OTG connector (X16) on the target to a free USB port on your host.



Be sure to use the USB A/mini-B included in this RDK.

Establishing the connection might take a few seconds. Once the connection has been established Active Sync will show a screen with status information.

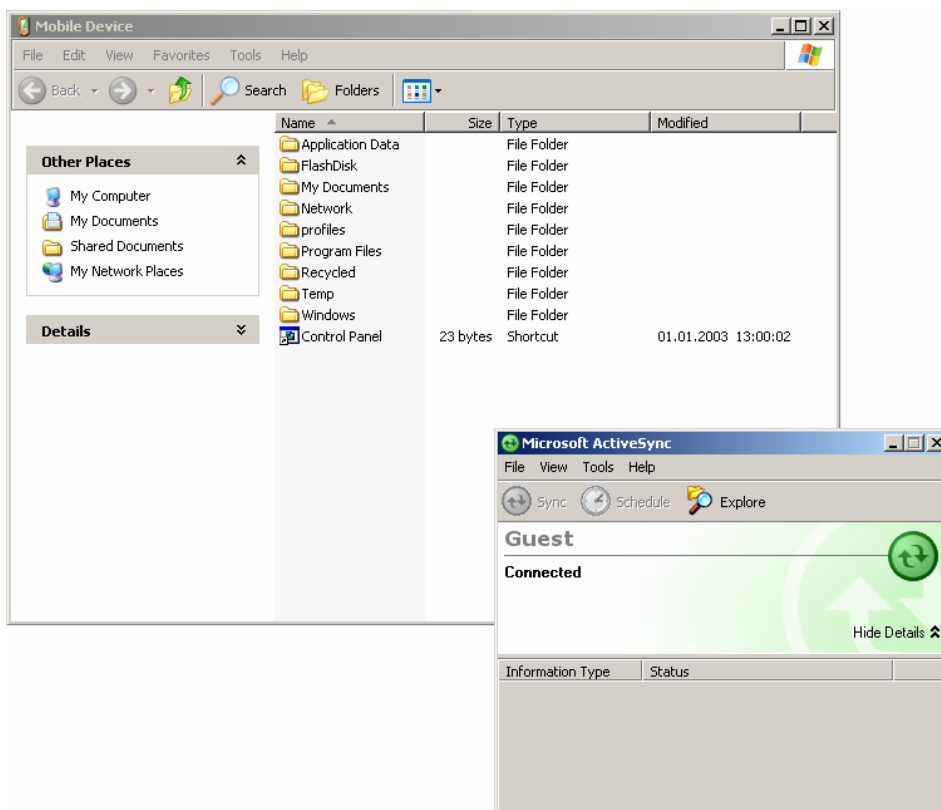
Your PC is now connected to your i.MX31 device.



3.3 Downloading Example Code with Active Sync

To download the example code provided on the phyCORE[®]-i.MX31 Rapid Development Kit CD, you need to establish a connection to your i.MX31 device through Active Sync as described in the previous sections.

- Click on *Tools* in the menu bar of Active Sync and select *Explore Device*. An Explorer window called *Mobile Device* will open that will show you the folder structure of the Windows CE installation on your i.MX31 device.



- Open a file manager and navigate to:

C:\PHYTEC\phyCORE-i.MX31\Demo\HelloWorld\HelloWorld\iMX31 SDK (ARMV4I)\Debug

Select ***HelloWorld.exe*** and copy it to the *Mobile Device* window in the folder ***Program Files*** using drag and drop.

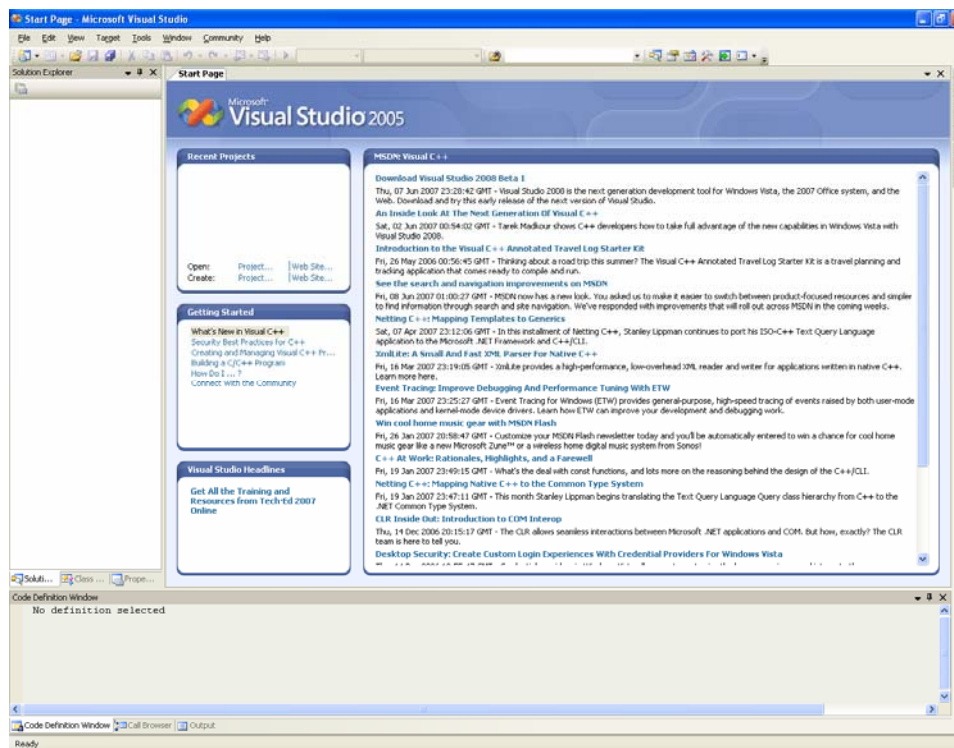
The program has now been copied to your i.MX31 device. You will be able to access it by double-clicking the **My Device** icon on your i.MX31 device. This opens a window that will show you the file structure of your device. You should be able to find ***HelloWorld.exe*** at the location you copied it to. Double-click it to execute the program. A window will open on your i.MX31 device that will print a **Hello World** message.

3.4 Downloading Example Code with Visual Studio 2005

PHYTEC provides you with example code to learn about downloading code onto your i.MX31 device. This code has already been copied to your local hard drive by the setup.

- Start the tool chain by selecting Microsoft Visual Studio 2005 from the programs group:
Start / Programs / Microsoft Visual Studio 2005.

After starting Visual Studio 2005 the window shown below will appear. Here you can create projects, edit files, configure tools, assemble, link and start the debugger. Close all projects that might be open by selecting *Close Project* from the *Project* menu.

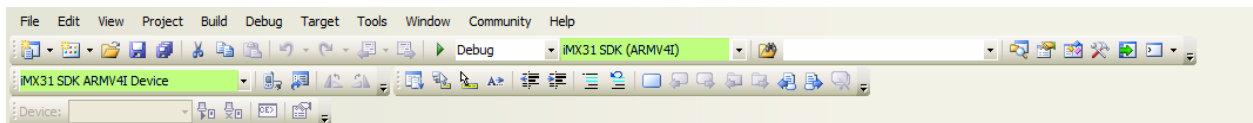


The HelloWorld example downloads a program to your device that, when executed, prints the text **Hello World** on the display of your phyCORE[®] Development Board i.MX31.


- To open the HelloWorld project select *File / Open / Project / Solution* from the Visual Studio 2005 menu bar.
- Browse to the folder:

C:\PHYTEC\ phyCORE-i.MX31\Demo\HelloWorld

- Select the *HelloWorld.sin* project.
- Click **Open**.
- In the *Solution Platform* pull-down menu make sure that the **i.MX31 SDK (ARMV4I)** configuration is selected. Also make sure that **i.MX31 SDK ARMV4I Device** is selected in the *Target device* menu.



If these menus are not shown in your Visual Studio IDE, move your mouse to an empty space on the toolbar, press the right mouse button and select from the pop up menu the menus Target and Device.

- Build the target by either selecting the Build icon  on the build toolbar or in the main menu bar select *Build / Build HelloWorld*.
- If any source file of the project contains any errors, they will be shown in the *Output Window - Build* tab. Use the editor to correct the error(s) in the source code, save the file and repeat the build.
- If there are no errors, the code can be downloaded to your i.MX31 device. For this to work you need to have established an Active Sync connection between your PC and your i.MX31 device as described in section 2.6. When the connection is established select *Build/Deploy HelloWorld* from the main menu bar.
- The individual steps of the download procedure can be viewed in the *Output Window - Build* tab.
- Wait until the download is complete.
- The newly created application was now deployed on your i.MX31 device. To start it select *My Device* on the target platform and go to the folder *Program Files / Helloworld* and double tap on *Helloworld*
- A window will open on your i.MX31 device that will print a **Hello World** message.
- Click on the **Close** icon in the window to close the application.



You have successfully passed the Getting Started part of this QuickStart. In this chapter you learned how to open a project, build an executable and transfer and run this executable on the target hardware.

4 Getting More Involved

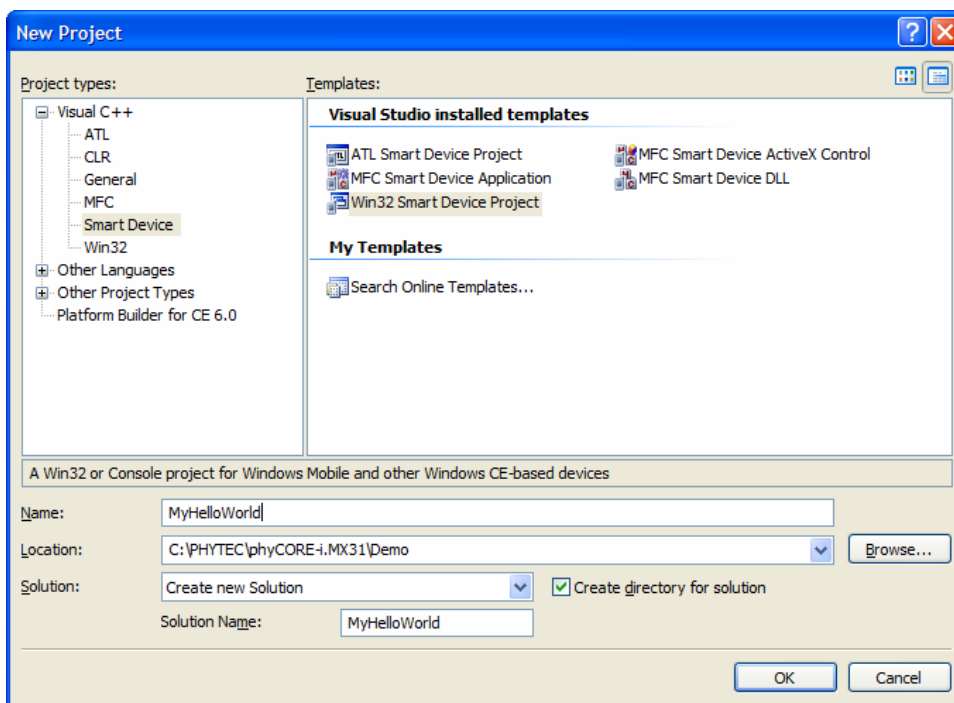
In this chapter you will learn how to configure the Visual Studio 2005 IDE (Integrated Development Environment), modify the source code from our example, create a new project and build and download a machine-readable output file to the target hardware.



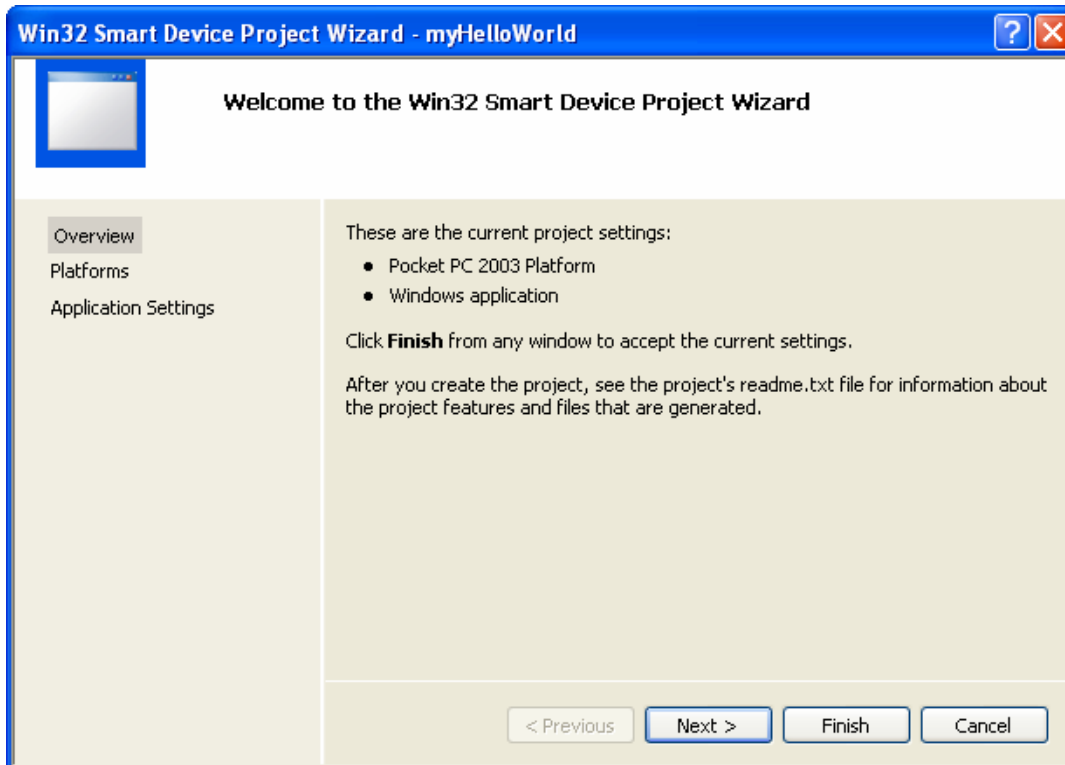
30 min

4.1 Creating a New Project

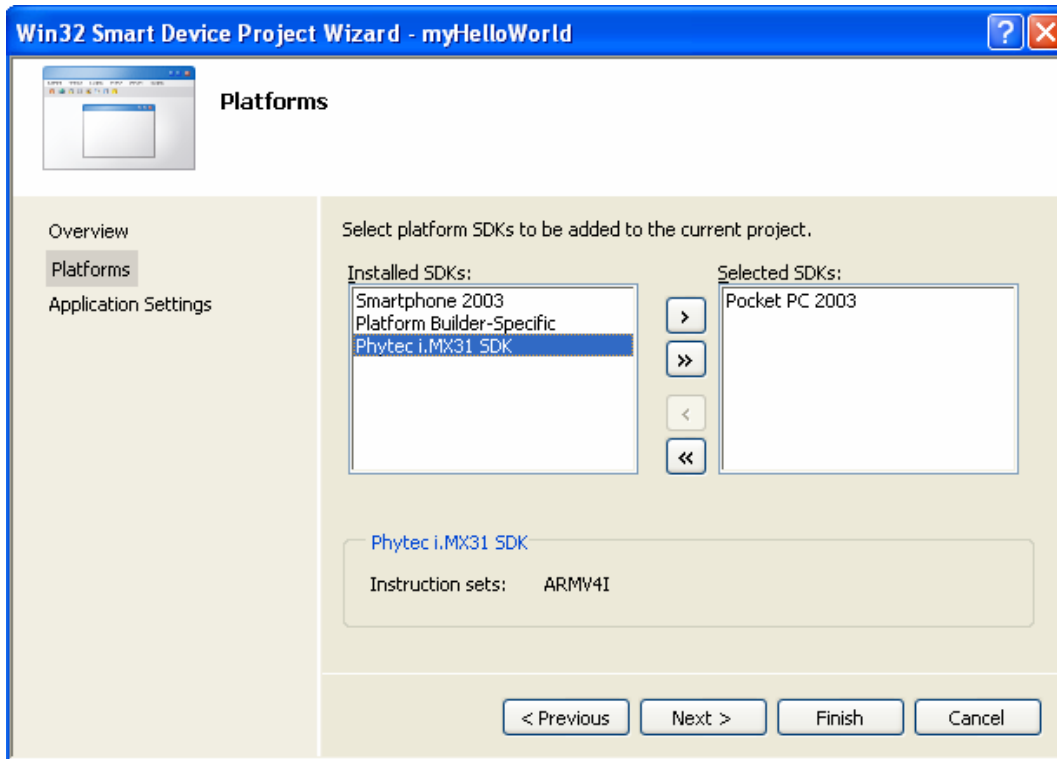
- To create a new project file select *File / New / Project* from the Visual Studio 2005 menu. This opens a dialog that will ask about the kind of project you wish to create.



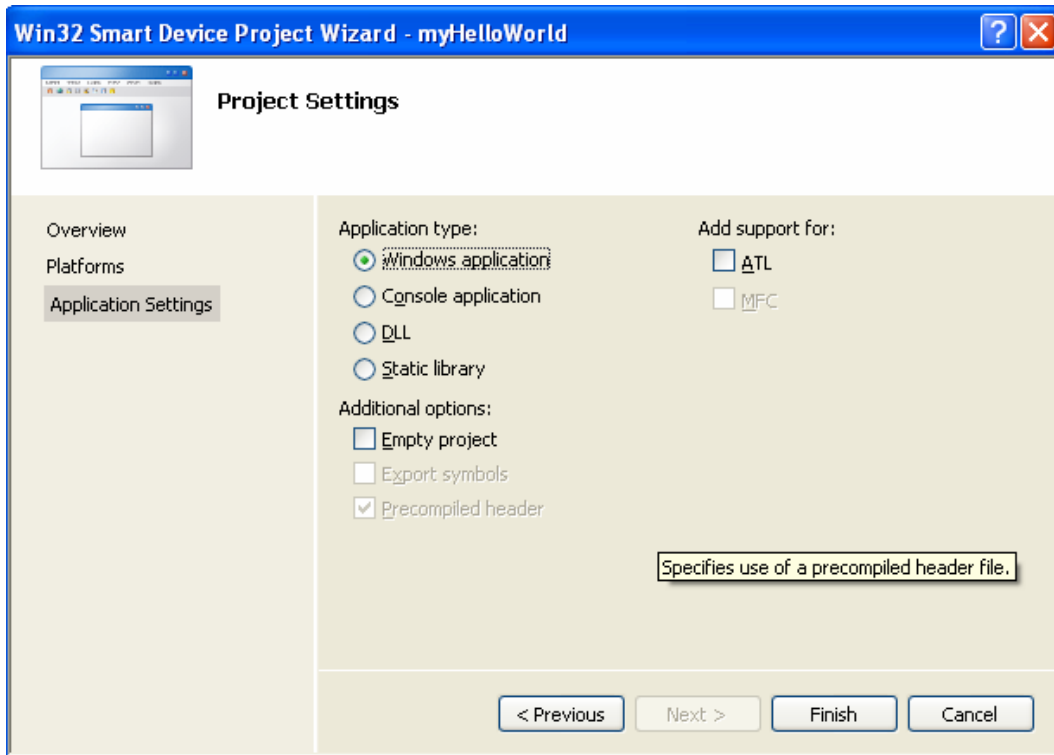
- Open the *Visual C++* tree and select **Smart Device**.
- In the *Templates* field select **Win32 Smart Device Project**.
- In the text field *Name*, enter the name of the project you wish to create, e.g. **myHelloWorld**.
- In the Location Field, accept the default or save
- Click **OK**.



- Click **Next**.



- Click on the entry **i.MX31 SDK** in the field *Installed SDKs:* and press the > button. The **i.MX31 SDK** will appear in the *Selected SDKs* field.
- Select **Pocket PC 2003** in the field *Selected SDKs* and press the < button. This removes the **Pocket PC 2003** SDK from the list of selected SDKs.
- Press **Next**

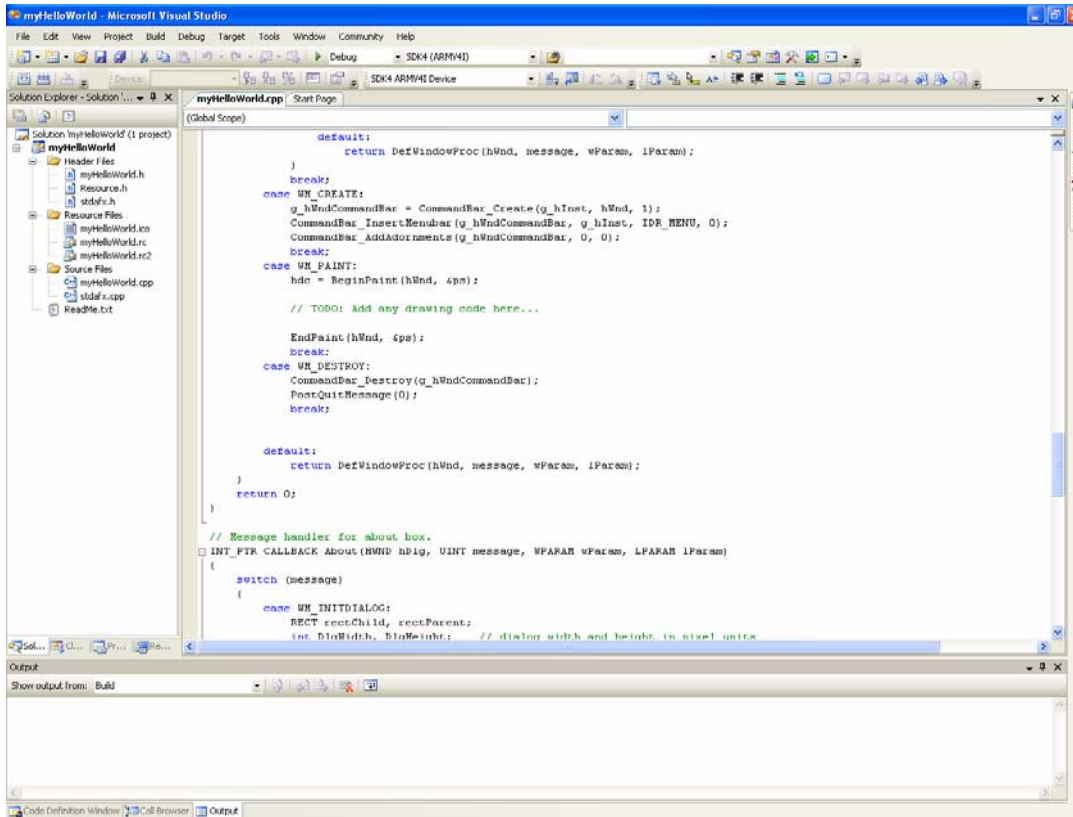


- Don't change the settings in this window, then press **Finish**.

At this point you have created a new project for the i.MX31 board. The next step is to modify the C source file before building your project. This includes compiling, linking, locating and creating the executable.

4.2 Modifying the Source Code

- Open the *myHelloWorld* tree and Source Files. Double-click on the **myHelloWorld.cpp** file to open the file in the source code editor.



- Locate the following code section at line 163:

```
case WM_PAINT:
    hdc = BeginPaint(hWnd, &ps);
```
- If line numbers are not displayed, go to *Tools / Options*. Under the *Text Editor* tree, select *C/C++* and check the *Line numbers* option under *Display options*.


To make the HelloWorld program print the text **Hello World!** 5 times instead of just one, replace this code section with the following:

```
case WM_PAINT:
    RECT rt;
    hdc = BeginPaint(hWnd, &ps);
    GetClientRect(hWnd, &rt);
    int i;
    for(i = 1; i <= 5 ; i++)
    {
        rt.top = rt.top + 30;
        DrawText(hdc, _T("Hello World!"), 12, &rt,
            DT_SINGLELINE | DT_VCENTER | DT_CENTER);
    }
    EndPaint(hWnd, &ps);
    break;
```

- Save the modified file by choosing *File / Save* or by clicking the *save icon* .

4.3 Building the Project

You are now ready to run the compiler and linker.

- Build the target by either selecting the *Build* icon  on the build toolbar or in the main menu bar select *Build / Build myHelloWorld*.
- If any source file of the project contains any errors, they will be shown in the *Output Window - Build* tab. Use the editor to correct the error(s) in the source code, save the file and repeat the build.
- If there are no errors, the code can be downloaded to your i.MX31 device. For this to work you need to have established an Active Sync connection between your PC and your i.MX31 device as described in section 2.6. When the connection is established select *Build / Deploy HelloWorld* from the main menu bar.
- The individual steps of the download procedure can be viewed in the *Output Window - Build* tab.
- Wait until the download is complete.
- The newly created application was now deployed on your i.MX31 device. To start it select *My Device* on the target platform and go to the folder *Program Files / myHelloWorld* and double click on *myHelloWorld*

A window will open on your i.MX31 device that will print 5 **Hello World** messages.



You have now modified source code, recompiled the code, created a downloadable file, and successfully executed this modified code.



You can find demo code for EEPROM, GPT, I2C, PCMIC, PWM, Serial, and SPI here: C:\PHYTEC\phyCORE-i.MX31\WinCE\iMX31_AD8.2.0A2_Binary\Documentation\Sample_Code

5 Debugging

This Debugging section provides a basic introduction to the debug functions included in Visual Studio 2005. The most important features are described by using an existing example. For a more detailed description of the debugging features, please refer to the appropriate manuals provided by Microsoft.



40 min

5.1 Starting the Debugger

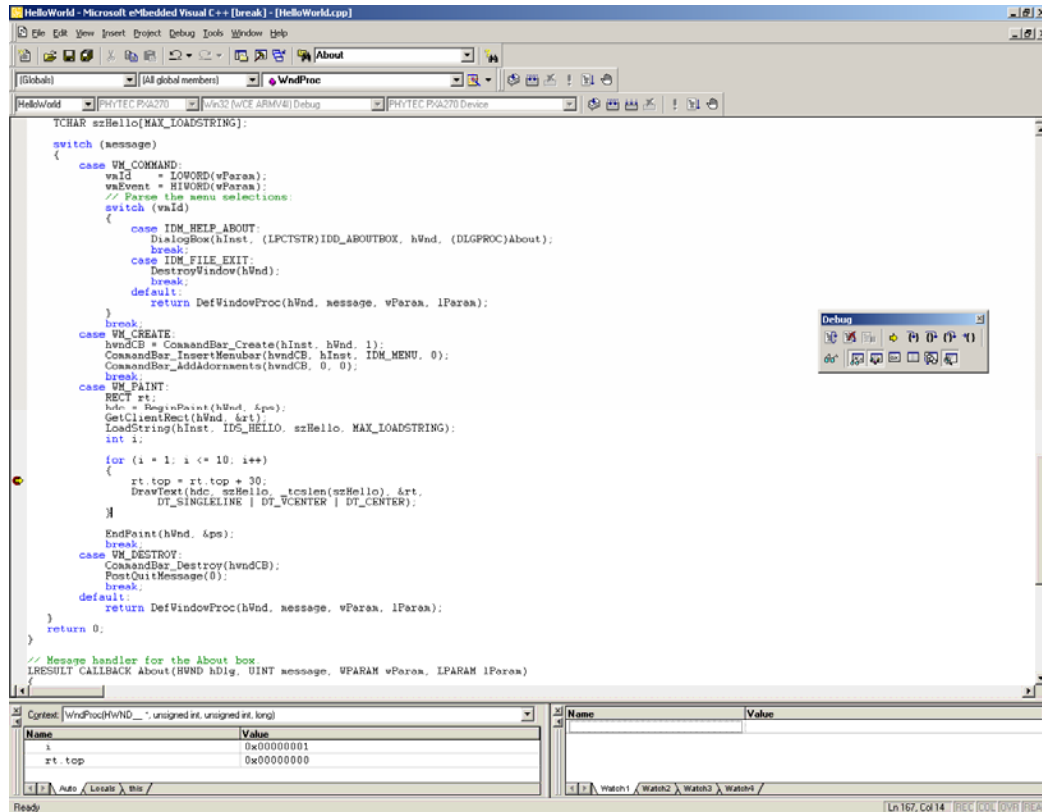
- Make sure that **Debug** is selected in the *Solution Configurations* pull-down menu.
- Open up the *myHelloWorld* tree and *Source Files*. Double-click on the *myHelloWorld.cpp* file to open the file in the source code editor.
- Locate the following code section of code on line 170 and click on the line

```
rt.top = rt.top + 30;
```

- Select *Debug / Toggle Breakpoint* on the main menu bar or press F9 to set a breakpoint here. The red marker on the left-hand side of the selected line indicates the breakpoint.
- To start debugging with Visual Studio 2005, select *Debug / Start Debugging* or press F5.
- A progress bar will indicate the download process of the debug program.

If a problem occurs during data transfer, an error message will be displayed. If this should occur, make sure you have established a connection between your PC and your i.MX31 device as described in section 3.2.

If data transfer was successful, a screen similar to the one shown below will appear. The debug toolbar is displayed. In the lower part of the debug screen you will see the Variables and Watch windows.



You may need to open, resize and/or move some windows to make your screen look similar to the screen capture. You can open inactive windows by choosing the desired window from the Debug / Windows pull-down menu.

The debugger will run up to the code section you have marked with a breakpoint and stop automatically. Notice the yellow arrow pointing to that breakpoint.

You can click on *Debug / Toggle Breakpoint* or press F9 again to remove the breakpoint you have set.

5.2 Visual Studio 2005 Debug Features

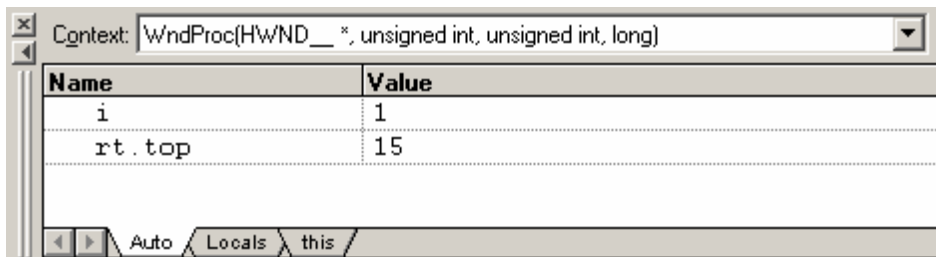
- The Debugger window toolbar gives access to the following debug commands: Restart, Stop Debugging, Break Execution, Show Next Statement, Step Into, Step Over, Step Out and Run to Cursor.



- The first button on the debugger toolbar is the *Start Debugging* button. This starts or continuous debugging.
- The *Break All* halts execution and the editor shows the current source code line.
- The *Stop Debugging* button stops the debugger and will lead you back to your project screen.
- The *Restart* button restarts the execution of the program from the beginning.
- The *Show Next Statement button* allows you to go through your program code line by code line.
- The first button allowing exact control of the program execution is the *Step Into* button. The *Step Into* command performs the execution of the command line to which the *Current-Statement Arrow* points. This can be a C command line or a single assembler line, depending on the current display mode. If the command line is a function call, *Step Into* jumps to the C function or subroutine, enabling you to explore the code contained in the accessed subroutine.
- The *Step Over* button is next on the debugger toolbar. The *Step Over* command executes the command line, to which the *Current-Statement Arrow* points. This can be a C command line or a single assembler line, depending on the current display mode. If the command line is a function call, the function will be executed without single stepping into the function.
- The next button is the *Step Out* button. *Step Out* is used to exit a function you are currently in. *Step Out* is very useful if you find yourself in a function you are not interested in and need to return quickly to your intended function.
- The *Hex* button toggles the display style of watch variables between hexadecimal and decimal.
- The last button pops up a menu where you can choose different windows useful for debugging, like watch windows, process information, memory information and calling stack.



5.3 Using the Visual Studio 2005 Debug Features

- The *Variables window* – *Auto* tab automatically shows the value of the local variable **i**. You can change the number base from decimal to hexadecimal by right clicking on the variable and selecting Hexadecimal Display.



- Click *Step Over* several times and watch the value of **i** count up.
- As you can see in the source code, the *for{}* loop will end if **i** becomes equal to 5. To leave the *for{}* loop, change the value of **i** by selecting the value, changing it to **10** and pressing **<Enter>**. Now repeat clicking on *Step Over* until you leave the loop.
- Remove the breakpoint by clicking into the source code line that has the breakpoint and press F9 or the *Insert / Remove Breakpoint* icon.
- Click in the source code, at **return 0**; and choose *Run to Cursor* from the *debug* toolbar. Your program will be executed until it reaches this line.

5.4 Running, Stopping and Resetting

- To run your program without stopping at any time, delete all breakpoints by clicking on the *Insert / Remove Breakpoint* icon.
- Click the *Go*  button.
- **Hello World!** will be displayed on the screen of your i.MX31 device.
- You can use the *Stop Debugging*  button to stop program execution at any time.

5.5 Changing Target Settings for Release

After successfully debugging the program, next change the project and the target settings in order to create an executable file that can then be downloaded to and executed out of the memory on the phyCORE[®]-i.MX31.

- Make sure the program execution is stopped.
- Exit the current debug session by selecting *Stop Debugging*.
- In the *Solution Configurations* pull down menu select **Release**.
- Build and execute your project as described in 3.3.

You can now watch your final example execute and **Hello World!** will be displayed on the screen of your i.MX31 device.



You have successfully finished the debug section of this QuickStart. You learned how to set breakpoints, run your program to a specific line of code and inspect variables.

6 Building a Customized Image

In this section you will learn how to add your own application to a WinCE project and build a customized OS Image using Platform builder.



50 min

6.1 Building an Image

- Open Visual Studio 2005, if not already open.
- select *File / Open / Project/Solution* from the Visual Studio 2005 menu bar.
- Browse to:

C:\PHYTEC\phyCORE-i.MX31\WinCE\iMX31_AD8.2.0A2_Binary

- Copy the *OSDesigns* directory to:

C:\WINCE600

- Browse to:

C:\WINCE600\OSDesigns\iMX31_Phytec_Proj3BIN

- Select the Microsoft Visual Studio Solution: *iMX31_Phytec_Proj3bin.sln*

When the project is loaded you will see the project in the *Solution Explorer* Tab in the left window.

- Open the Tree:

iMX31_Phytec_Proj3 ->Parameter Files ->iMX31_Phytec_BIN:ARMV4I(Active)

- Double click on *project.bib*.
- Under the **Files** section add the following line:

```
HelloWorld.exe "C:\PHYTEC\phyCORE-
i.MX31\Demo\HelloWorld\HelloWorld\Phytec i.MX31 SDK (ARMV4I)\Debug"
NK
```

This will add the executable *HelloWorld.exe* to the Windows directory of the target. The path tells the platform builder where to find the executable that should be added to the image and the parameter NK specifies the name of the memory region where it should be added to.

- Select *Build / Build Solution* from the Visual Studio 2005 menu bar. The build process will take about 20 minutes, depending on your host platform hardware.



You won't always need to do a complete build process that takes this long. If you just want to add a file to the image like in the example before, you can do this by selecting `Build\Copy Files to Release dir`. This copies all necessary files to the Release directory. Then you can build the image with `Build\Make Run-Time image`. This takes only about 5 minutes. Please refer to the Visual Studio 2005/Platformbuilder documentation for more information.

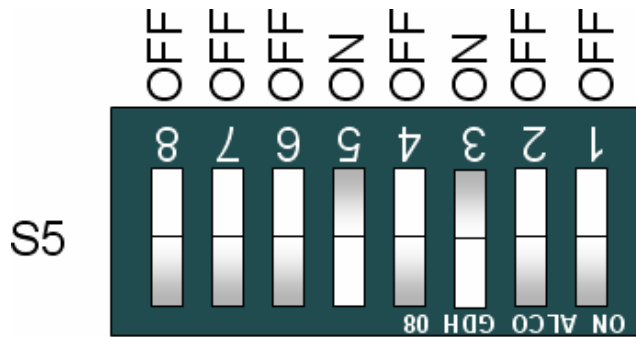
Appendix A Installing the Bootloader

This section provides instructions for loading eboot to the phyCORE-i.MX31 NOR Flash using the ADSToolkit program.



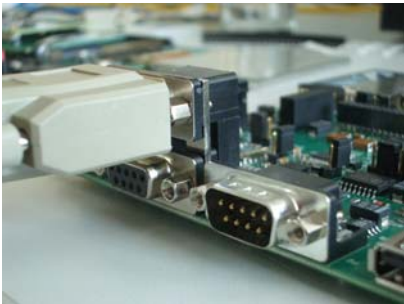
20 min

1. Configure the switch S5 to boot with UART



Boot Mode/CLK-Sel.

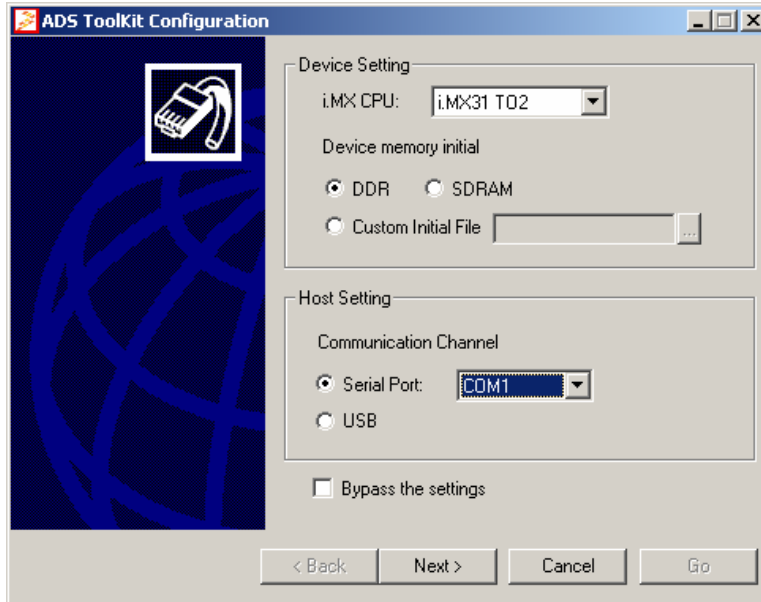
2. Connect the UART 3 (connector P1 - TOP) to the computer



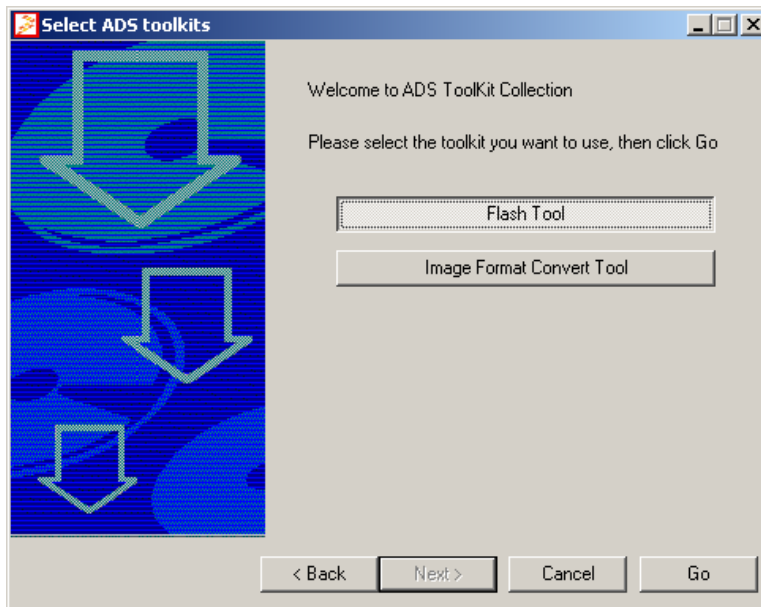
3. Power up the baseboard by connecting the power adapter to X26
4. Start the ADSToolkit program *ADSToolkit_std.exe* from:

C:\PHYTEC\phyCORE-i.MX31\Tools\ADS Toolkit

5. Choose **i.MX31_TO2** for *i.MX CPU*
6. Choose the correct *Communication Channel* depending on your setup

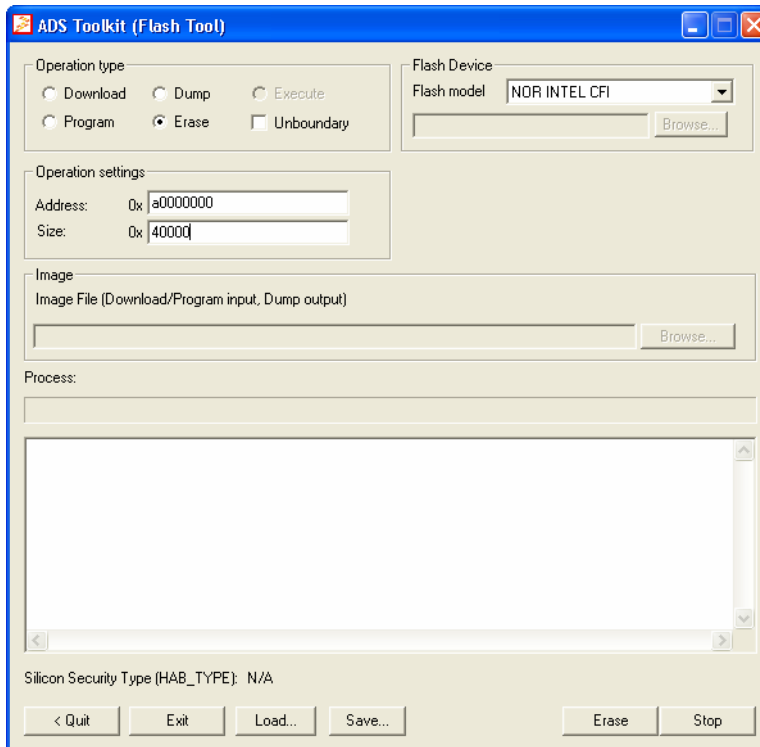


7. Click **Next**
8. On the next window select **Flash Tool**



9. Click **Go** to run the selected tool.

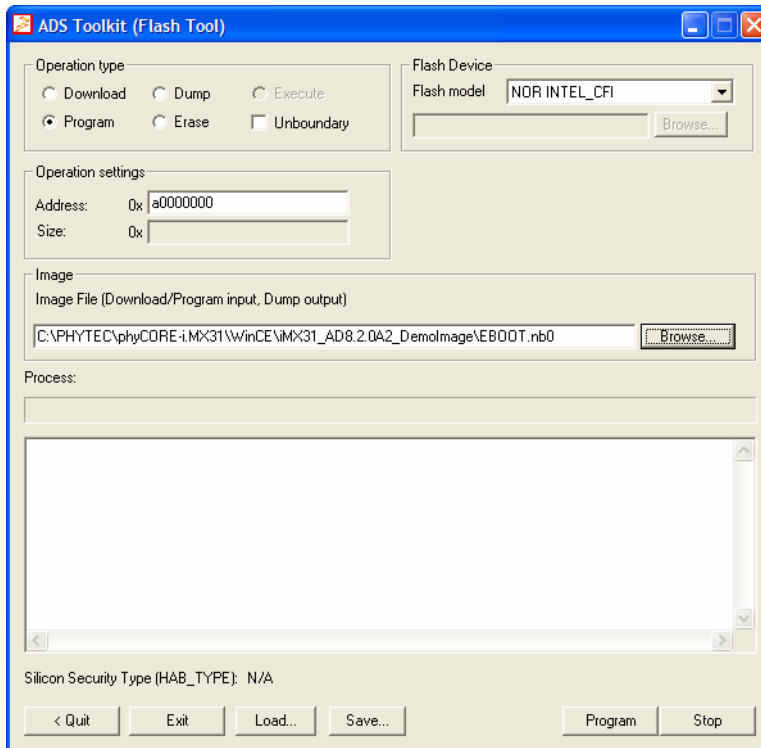
10. Select the *Operation Type* **Erase**
11. Choose **NOR INTEL_CFI** for *Flash Device*
12. Set the *Address* to **a0000000**
13. Set the *Size* to **40000**



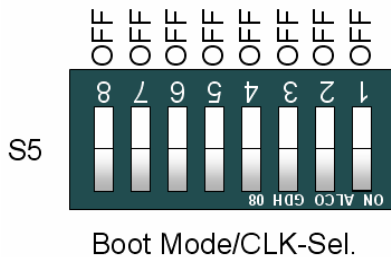
14. Click **Erase**
15. When finished, select the *Operation type* **Program**
16. With the *Browse* button navigate to:

C:\PHYTEC\phyCORE-i.MX31\phyCORE-i.MX31\WinCE\iMX31_AD8.2.0A2_DemoImage

17. Select ***eboot.nb0***



18. Click on **Program**
19. When finished, remove power from the baseboard
20. Configure the switch S5 to boot with NOR Flash



*You have now installed the WinCE bootloader, **eboot.nb0**.*

Appendix B Installing the WinCE Binary Image

This section provides instructions for loading a Windows Embedded CE Image to the phyCORE-i.MX31 NOR Flash using Visual Studio.



40 min



*To download an NK.bin, the phyCORE[®]-i.MX31 must have **eboot.nb0** pre-installed. See Appendix A.*

1. Start a Hyperterminal session with the following COM settings:

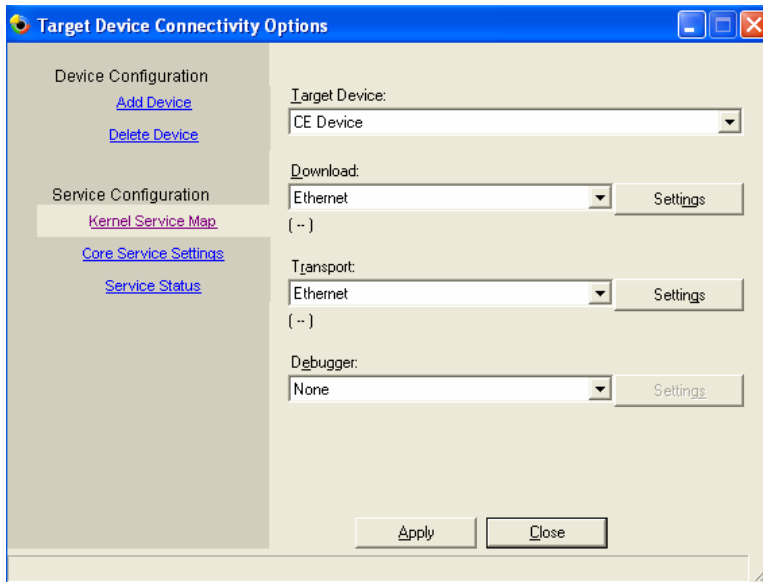
115200
8 bit
No parity
1 bit stop
No flow control

2. Connect the UART 3 (connector P1 – bottom DB-9) to the computer.

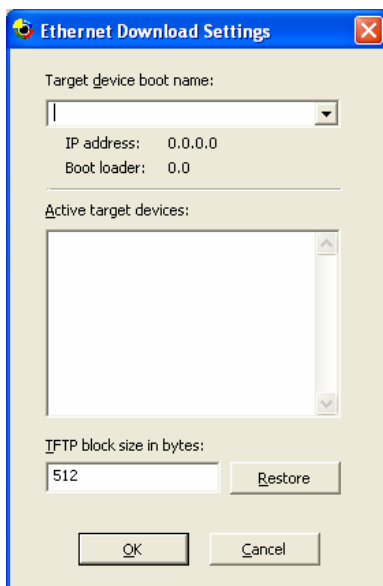


Be sure you are connected to the bottom DB-9. ADSToolkit requires a connection to UART 1 (top DB-9), the Eboot messages will be output on UART 3 (bottom DB-9).

3. Power up the baseboard.
4. You will now see some output messages from the bootloader in the HyperTerminal window. Press space to stop the bootloader.
5. Connect the cross-over Ethernet cable with the connector X27 on the target your host PC.
6. In the HyperTerminal window you can press **0** to change the IP address of the device and **1** to change the network mask. Change the default settings so it will fit to the network settings on your PC.
7. Press **3** to disable DHCP.
8. Press **5** to change the Autoboot entry to NK from NOR
9. Press **S** to save the configuration.
10. Start Microsoft Visual Studio 2005
11. Select *Target / Connectivity Options* from the Visual Studio 2005 menu bar.
12. Select **Ethernet** for the Download and Transport settings from the drop down box.



13. Click on the *Settings* button on the right of the *Download* drop-down box. A window will appear. Leave this window open.
14. Change to the Hyperterminal window and press **d** on the keyboard to start the download of the image. You will see some output messages followed by a **BOOTME** message.
15. Go back to Visual Studio. In the window *Ethernet download settings* a new entry should appear in the Active target devices list. Select the device and close the window.



16. Select *Target / Attach Device* from the Visual Studio 2005 menu bar.
17. In the following screen, select a run-time image dialog, navigate to:
`C:\PHYTEC\phyCORE-i.MX31\WinCE\iMX31_AD8.2.0A2_DemoImage`
18. Select *nk.bin*
19. When the download is finished, the image has been downloaded in RAM. Now go to the HyperTerminal window, the bootloader should state a message to program the image now in flash. Press **Y** to confirm this. The flash program process will take approximately 20 minutes to complete.
20. When programming the flash is completed, the bootloader will state **spin forever**. Reset the board by disconnecting power from X26 and reconnecting. WinCE should boot now.



*You have now installed the WinCE image, **nk.bin**.*