

**QuickStart Instructions**

**WinCE-Kit**

**phyCORE<sup>®</sup>-PXA270**

**Using embedded Visual C++ 4.0  
Software Development Tool Chain**

Note: The PHYTEC WinCE-Xscale-Disc includes the electronic version of the English phyCORE<sup>®</sup>-PXA270 Hardware Manual

**Edition: November 2006**

A product of a PHYTEC Technology Holding company

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 2006 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 <a href="mailto:order@phytec.de">order@phytec.de</a>	1 (800) 278-9913 <a href="mailto:sales@phytec.com">sales@phytec.com</a>
Technical Support:	+49 (6131) 9221-31 <a href="mailto:support@phytec.de">support@phytec.de</a>	1 (800) 278-9913 <a href="mailto:support@phytec.com">support@phytec.com</a>
Fax:	+49 (6131) 9221-33	1 (206) 780-9135
Web Site:	<a href="http://www.phytec.de">http://www.phytec.de</a>	<a href="http://www.phytec.com">http://www.phytec.com</a>

2<sup>nd</sup> Edition: November 2006

<b>1</b>	<b>Introduction .....</b>	<b>1</b>	
1.1	Rapid Development Kit Documentation .....	1	<b>5 min</b>
1.2	Professional Support Packages available .....	1	
1.3	Overview of this QuickStart Instruction.....	2	
1.4	Conventions used in this QuickStart .....	3	
1.5	System Requirements .....	4	
1.6	The PHYTEC phyCORE®-PXA270 .....	5	
1.7	Microsoft eMbedded Visual C++ 4.0 Development Tool Chain	8	
<b>2</b>	<b>Getting Started .....</b>	<b>10</b>	<b>60 min</b>
2.1	Preparing the phyCORE®-PXA270 to interface to a host-PC....	10	
2.2	Installing eMbedded Visual C++ 4.0.....	16	
2.3	Installing the WinCE PXA270 SDK.....	20	
2.4	Installing Microsoft Active Sync 4.1.....	21	
2.5	Establishing an Active Sync Connection .....	23	
2.6	Downloading Example Code with Active Sync.....	25	
2.7	Downloading Example Code with eMbedded Visual C++.....	27	
<b>3</b>	<b>Getting More Involved.....</b>	<b>33</b>	<b>30 min</b>
3.1	Creating a New Project.....	33	
3.2	Modifying the Source Code.....	36	
3.3	Building the Project.....	37	
<b>4</b>	<b>Debugging.....</b>	<b>38</b>	
4.1	Starting the Debugger.....	38	<b>40 min</b>
4.2	eMbedded Visual C++ 4.0 Debug Features .....	40	
4.3	Using the eMbedded Visual C++ 4.0 Debug Features.....	41	
4.4	Running, Stopping and Resetting .....	42	
4.5	Changing Target Settings for the "Executable Version" .....	43	
<b>5</b>	<b>Summary .....</b>	<b>44</b>	



## 1 Introduction

**5 min**

*This QuickStart provides general information on the PHYTEC phyCORE<sup>®</sup>-PXA270 Single Board Computer (SBC). It gives you also an overview of Microsoft's Embedded Visual C++ software development tool chain and instructions on how to run example programs on the phyCORE<sup>®</sup>-PXA270, mounted on the PHYTEC phyCORE<sup>®</sup> Development Board PXA270, in conjunction with a remote shell and eMbedded Visual C++.*

Please refer to the [phyCORE<sup>®</sup>-PXA270 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<sup>®</sup>-PXA270 Hardware Manual.

### 1.1 Rapid Development Kit Documentation

This "Rapid Development Kit" (RDK) includes the following electronic documentation on the enclosed "PHYTEC WinCE-Xscale-Disc ":

- the PHYTEC [phyCORE<sup>®</sup>-PXA270 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<sup>®</sup>-PXA270 in conjunction with the Microsoft eMbedded Visual C++ 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®-PXA270 in conjunction with Microsoft's eMbedded Visual C++. It is structured as follows:

- 1) The "Getting Started" section uses the example *HelloWorld* to demonstrate the download of user code to the target device using eMbedded Visual C++ with ActiveSync.
- 2) The "Getting More Involved" section provides step-by-step instructions on how to modify the example, create and build a new project and generate and download output files to the phyCORE®-PXA270 using the Microsoft tools.
- 3) The "Debugging" section demonstrates simple debug functions using eMbedded Visual C++ debug environment.

In addition to the dedicated data for this Rapid Development Kit, the PHYTEC WinCE Kit CD-ROM contains supplemental information on embedded microcontroller design and development.

## 1.4 Conventions used in this QuickStart

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

*Italic* Used for file and directory names, program and command names, command-line options, menu items, URLs, and other terms that correspond to the terms on your desktop.

**Bold** Used in examples to show commands or other text that should be typed literally by the user.

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



At this part you might leave the path of this QuickStart.



This is a warning. It helps you to avoid annoying problems.



You can find useful supplementary information about the topic.



At the beginning of each chapter you can find information of the time to pass the following chapter.



You have successfully passed an important part of this QuickStart.



You can find information to solve problems.

## 1.5 System Requirements

The use of this "Rapid Development Kit" requires:

- the PHYTEC phyCORE®-PXA270
- the phyCORE® Development Board PXA270 with the included DB-9 serial cable and AC adapter supplying 12 VDC /min. 1A
- eMbedded Visual C++ 4.0
- the PHYTEC WinCE-Xscale-Disc for PXA270 WinCE
- an IBM-compatible host-PC (586 or higher running with Windows 2000 or XP)

For more information and example updates, please refer to the following sources:

The logo for PHYTEC, featuring the word "PHYTEC" in a bold, stylized, metallic-looking font with a 3D effect.

<http://www.phytec.de>  
[support@phytec.de](mailto:support@phytec.de)



## 1.6 The PHYTEC phyCORE<sup>®</sup>-PXA270

The phyCORE<sup>®</sup>-PXA270 represents an affordable yet highly functional Single Board Computer (SBC) solution in sub-miniature dimensions (70 x 57 mm). The universal design enables its insertion in a wide range of embedded applications.

All controller signals and ports extend from the controller to high-density (0.635 mm) Molex pin header connectors aligning two sides of the board, allowing it to be plugged like a "big chip" into a target application.

The standard module runs at a maximum of 520 MHz internal clock speed and offers 64 MByte (up to 128 MByte) SDRAM and 32 MByte (up to 64 MByte) Flash on-board for DATA and CODE storage.

The module communicates by means of one RS-232 transceiver, a CAN bus interface, and an SMSC LAN91C111 10/100BaseT Ethernet controller which enables the implementation of the module in embedded Internet applications. The phyCORE<sup>®</sup>-PXA270 operates within a temperature range of -40°C to +85°C.

**The phyCORE®-PXA270 offers the following features:**

- subminiature Single Board Computer (70 x 57 mm) achieved through modern SMD technology
- populated with the Intel PXA270 microcontroller (BGA-360 packaging)
- improved interference safety achieved through multi-layer PCB technology and dedicated Ground pins
- controller signals and ports extend to two 160-pin high-density (0.635 mm) Molex connectors aligning two sides of the board, enabling it to be plugged like a "big chip" into target application
- max. 520 MHz clock frequency
- 32 MByte (up to 64 MByte) on-board Flash<sup>1</sup>
- 64 MByte (up to 256 MByte) RAM on-board
- one CAN transceiver
- RS-232 transceiver
- SMSC 91C111 Ethernet controller with configuration EEPROM
- USB Client Interface and Host Interface
- 2 \* PCMCIA / Compact Flash
- 2 \* MMC card and Memory Stick
- LCD controller
- Synchronous Serial Protocol ports (SSP and NSSP)
- I<sup>2</sup>C Interface
- 4 \* PWM
- RTC
- General purpose I/O pins (GPIO)

---

<sup>1</sup> : Please contact PHYTEC for more information about additional module configurations.

---

The Development Board PCM-990 is a universal carrier board for start-up and programming of the PHYTEC phyCORE-PXA270 Single Board Computer module with two high density SMT (160 pins each, 0.63 mm pitch) pin header connectors.

The Development Board is fully equipped with all mechanical and electrical components necessary for the speedy and secure insertion and subsequent programming of the high-density PHYTEC phyCORE module. These components include TFT display connection, touch screen interface, Ethernet transformer and connector, DB-9, USB and power socket connectors.

### **phyCORE® Development Board PXA270 Technical Highlights**

- Molex connectors for mating with phyCORE-PXA270 SBC module
- Connector for 8.4" Sharp TFT display with 640\*480 pixel resolution, incl. adjustable inverter for background illumination
- AC97 controller WM9712L with touch and sound interface (Line-IN, Line-OUT, MIC-IN and 0.4 watts speaker)
- Two DB-9 sockets for RS-232 interface from FF-UART and BT-UART, 2\*5 pin header for IR-UART
- One DB-9 plug for CAN interface, optical isolation
- RJ-45 Ethernet connector with transformer for 100/10 MBit/s
- USB Host socket for PXA270 USB host interface
- USB client socket for PXA270 USB client interface
- Power supply for unregulated input voltage from 12 V (optional 24V). It supplies regulated +3.3 V for the phyCORE-PXA270. Additional +5 V is generated for the IDE and CF card and LCD/inverter circuitry.
- Optional battery charger circuitry for NiMH batteries
- Power management for MMC, CF, IDE and LCD display
- Two user programmable LEDs
- Optional IDE interface for two 2.5" hard disks (Master / Slave)
- Interface for Matrix keyboards
- JTAG interface
- AC97 expansion port
- Reset and GPIO push button

## 1.7 Microsoft eMbedded Visual C++ 4.0 Development Tool Chain

Microsoft's Software development tools for the PXA270 Architecture aid every level of developer from the professional applications engineer to the student just learning about embedded software development.

EMbedded Visual C++, a free version of Microsoft's popular IDE, intended for embedded devices, 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 eVC++ features including:

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

The eVC++ editor offers many standard and advanced software editing features like:

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

The eMbedded Visual C++ Tool chain is for free. You can use it to develop standard WinCE applications or MFC WinCE applications. For developing .NET Applications for WinCE you will need Visual .NET 2005 which can be purchased from Microsoft. Because eVC++ is for free, this RDK only works with

evC++. If you already have installed another version of eVC++ or Visual Studio .NET, we recommend installing eVC++ 4.0 when working with this Quickstart.

## 2 Getting Started

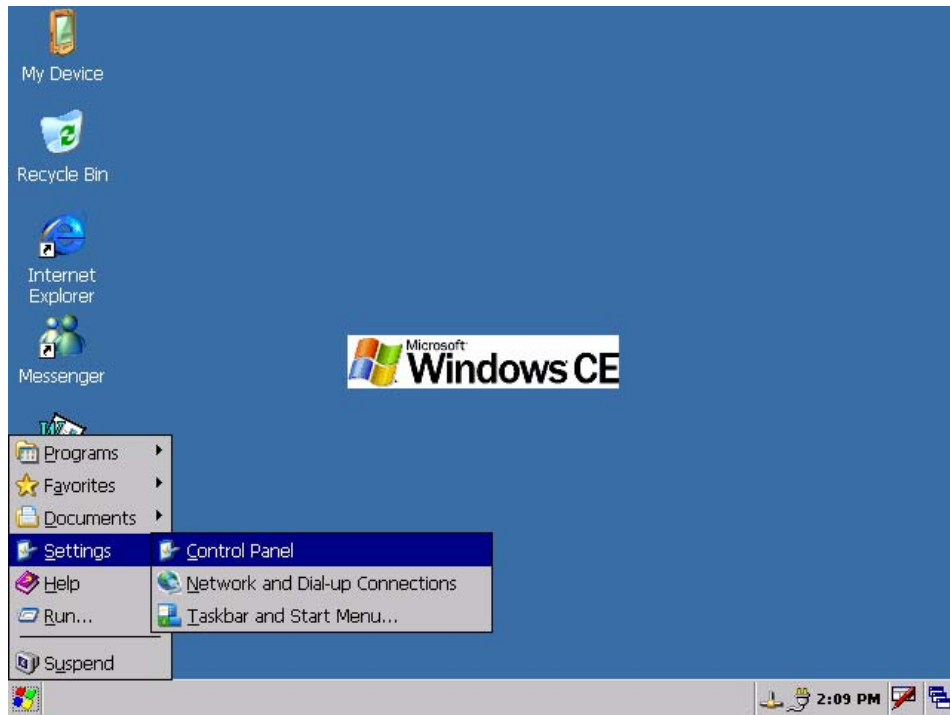
**60 min**

*In this chapter you will install Rapid Development Kit software and prepare the phyCORE®-PXA270, mounted on the Development Board, to interface to a host-PC. You will also learn how to download an example application from a host-PC using eMbedded Visual C++.*

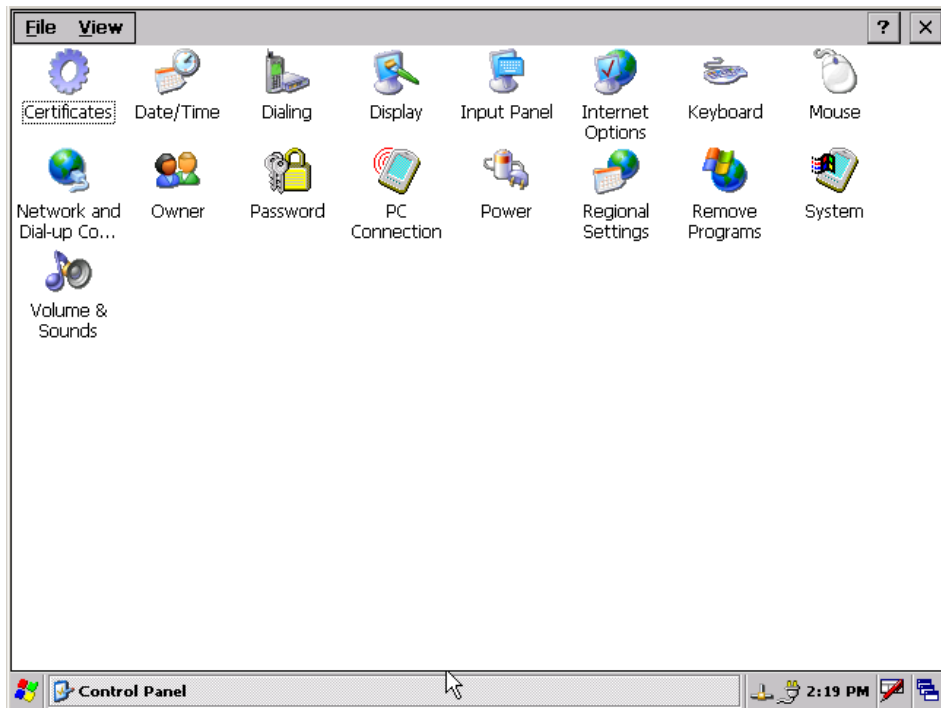
### 2.1 Preparing the phyCORE®-PXA270 to interface to a host-PC

*To be able to interface your PXA270 with your PC you must first setup the correct network settings on your PXA270. For downloading software to the PXA270 with ActiveSync we will use the serial interface in this Quickstart. This chapter explains how to configure WinCE to connect to your host PC.*

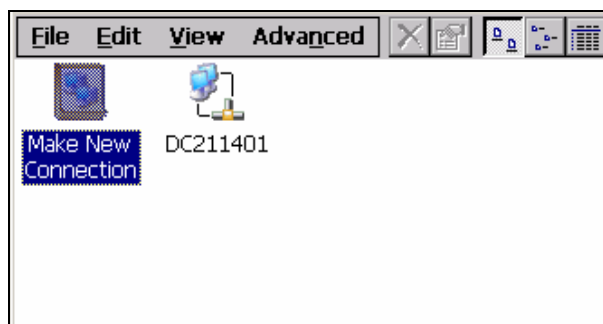
Connect your serial cable to your PC and to your PXA270 device in the DB-9 socket labelled “FF-UART”. Connect the AC adapter to your device to start up Windows CE. When Windows CE has loaded click on “Start” in the task bar of your PXA270 and select “Settings”->”Control Panel”.



This opens Windows CE's control panel, which allows you to administrate several aspects of the operating system like display, sound, mouse or network settings.

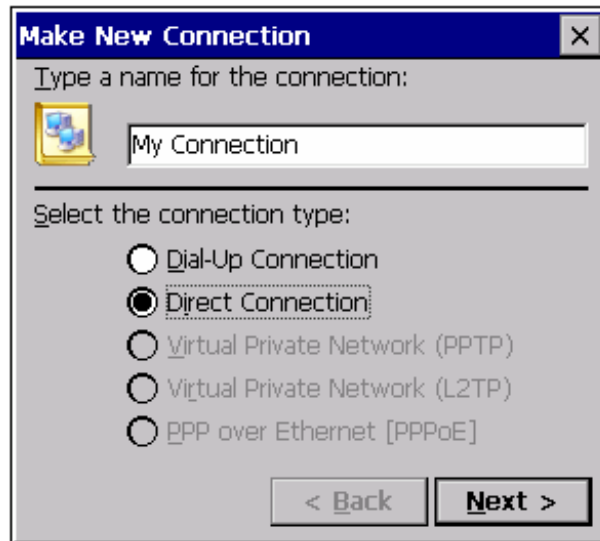


Double-click on the *Network and Dial-up Connections* icon. This will open a window that shows your existing network connections. Double-click on the *Make New Connection* icon.



The following dialog asks you to type a name for your connection and to select the type. Please enter a name, select *Direct Connection* and click *Next*.

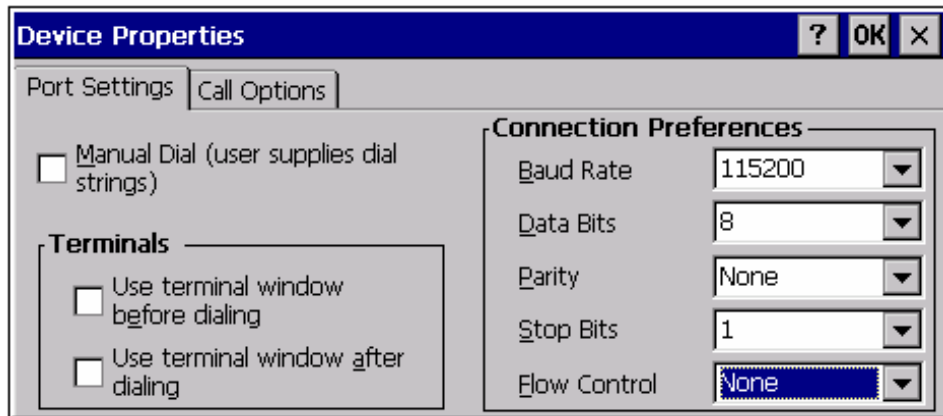




The next dialog asks you to select a device. Select *Serial cable on COM1* and click on *Configure....*

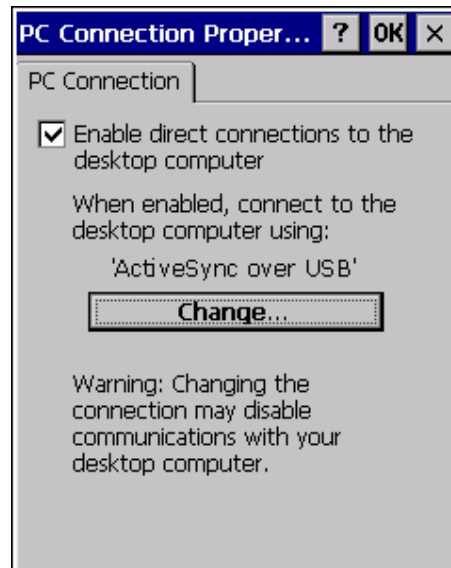


This will open the *Device Properties* dialog in which you can setup your device.



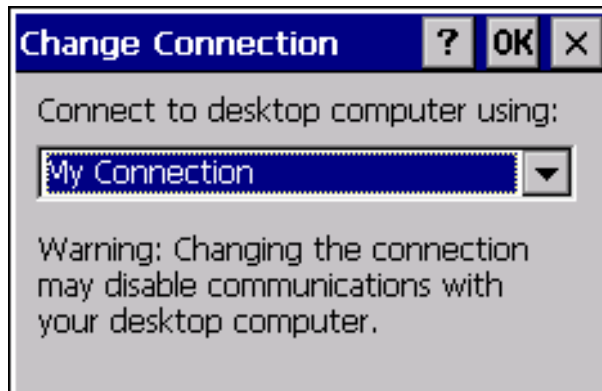
Leave all settings at default except for *Baud Rate* and *Flow Control*. Set the Baud Rate to 115200 and the Flow Control to *None*. Click on *OK* in the upper right corner. This will close the *Device Properties* dialog and you will return to the *Device* dialog. Click on *Finish* to create your new network connection. Close the network connection window.

The control panel is shown again. Double-click on the *PC Connection* icon. The *PC Connection Properties* dialog will open up.



Make sure the checkbox next to *Enable direct connections to the desktop computer* is checked. Below this the connection that is

currently being used is listed. Click on *Change...* to change this connection to the one you just created.

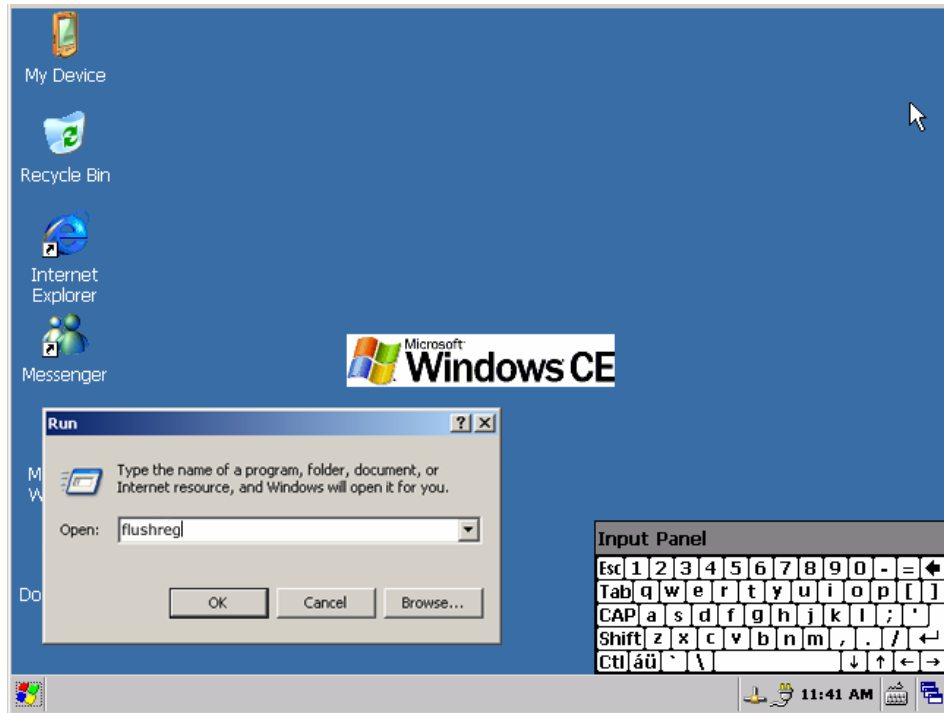


A dialog box will open that will display a drop-down box that lists all of your network connections. Choose the one you just created and click on *OK* in the upper right corner. This will return you to the *PC Connection Properties* dialog. Close it by clicking on *OK* in the upper right corner.

Your PXA270 is now configured to interface to your PC and you can close the control panel.

To save these settings to the registry of your device so that you don't have to set them again after turning your device off you need to call the program "flushreg".

Click on *Start* in the task bar of your PXA270 and select *Run...* Open the virtual keyboard by clicking on the icon on the right corner of the taskbar. Set the focus back on the *Run* dialog and type the program name *flushreg* into the dialog box using the virtual keyboard.



In the following dialog click “Flush” to save the settings to your registry. Close the dialog by clicking on “Done”.

## 2.2 Installing eMbedded Visual C++ 4.0

The eMbedded Visual C++ 4.0 development environment provides you with everything you need to write your Windows CE applications. You can download the software from the Microsoft website:

### eMbedded Visual C++ 4.0

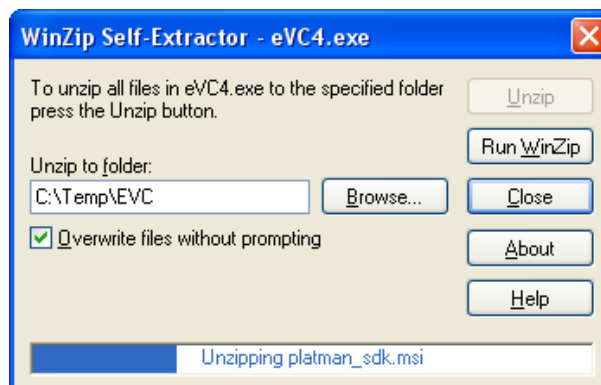
<http://www.microsoft.com/downloads/details.aspx?FamilyID=1dacdb3d-50d1-41b2-a107-fa75ae960856&DisplayLang=en>

### Service Pack 4 for eMbedded Visual C++ 4.0

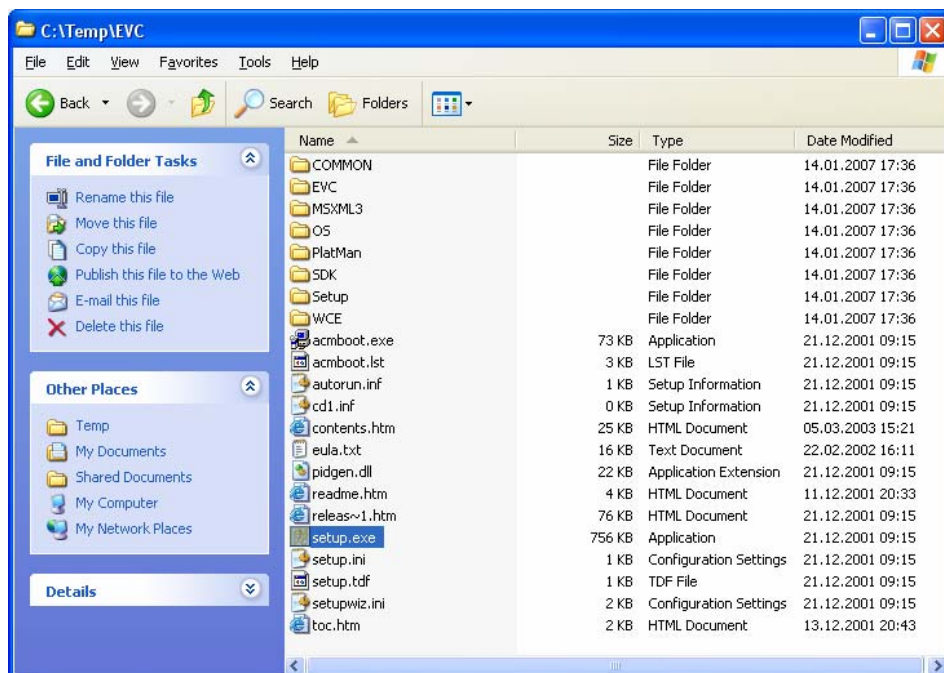
<http://www.microsoft.com/downloads/details.aspx?FamilyID=4a4ed1f4-91d3-4dbe-986e-a812984318e5&DisplayLang=en>

Download the files and save them to your hard drive. Open the folder you saved the files to and double-click the file eVC4.exe. This will

open a dialog that asks you to unzip the files needed to install eMbedded Visual C++ 4.0.

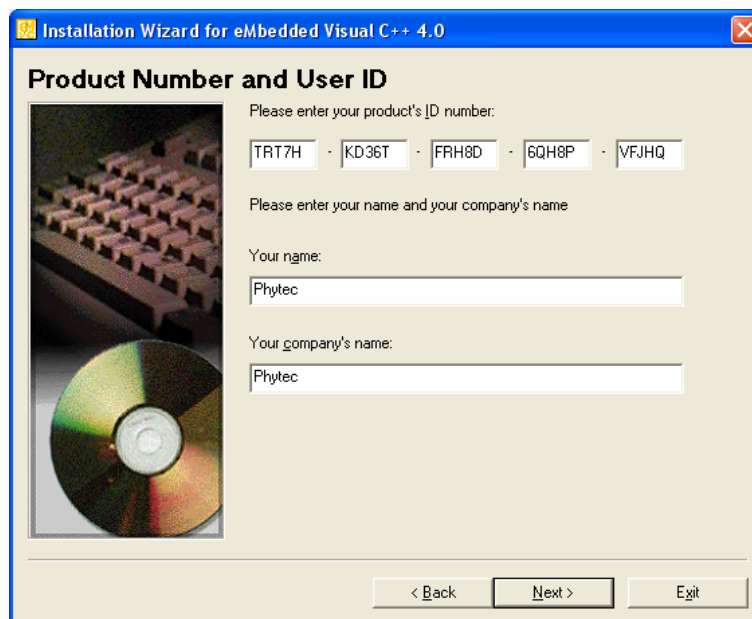


Choose a folder you wish to unzip the files to and click on *Unzip*. Unzipping may take several minutes. When it is complete click on *Close* and open the folder you unzipped the files to. Select *setup.exe* and double-click it.



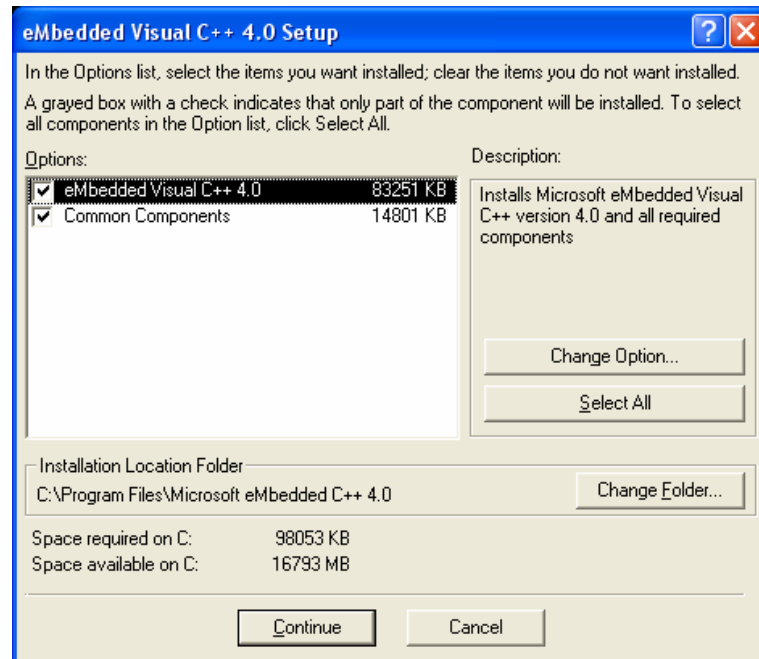
This will start the setup process for eMbedded Visual C++. You will have to accept the license agreement. After clicking *Next* enter your

product number and user name. You will find the product number (CD key) on the webpage you downloaded the software from.

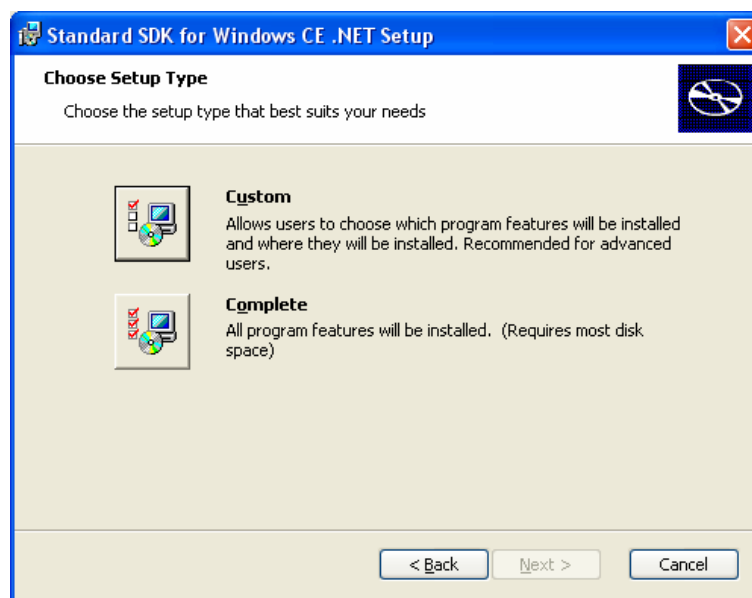


Click *Next* and ensure in the following dialog that both eMbedded Visual C++ and the Standard SDK for Windows CE.NET are selected.

After clicking *Next* again you will be able to choose the folder you wish to install eMbedded Visual C++ to. The installation wizard will first install Windows CE Platform Manager and after that the installation of eMbedded Visual C++ itself will start. A dialog will be shown that asks you to select the items you wish to install. Keep the standard options and click *Continue*.

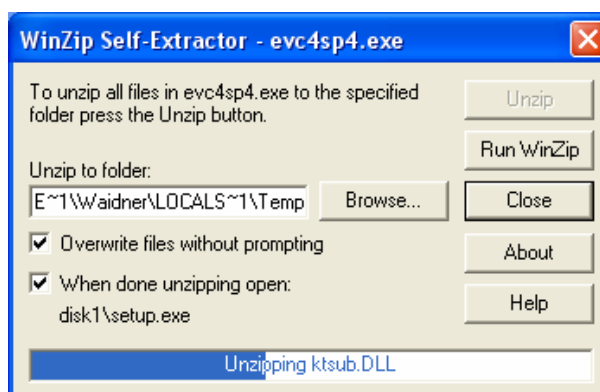


This will start the installation process that might take several minutes. After the installation is complete the installer will automatically start the installation of the Standard SDK for Windows CE.NET. After accepting the license agreement and entering your user name and organization you are asked to select a setup type.



Click on *Complete* and in the next dialog select the folder you wish to install the SDK to. Click *Next* and in the following dialog click *Install* to start the installation process for the SDK. This again may take a few minutes.

After completing the installation you should install Service Pack 4 for eMbedded Visual C++ next to update your development environment. Go to the folder you downloaded the Service Pack to and double-click *evc4sp4.exe*. This will open the self-extractor that will unzip the files needed to install the Service Pack.



After unzipping the installation process for the Service Pack will be started automatically. Accept the license agreement and click *Next* to start the installation. The installation may take a few minutes.

This finishes the setup of the eMbedded Visual C++ 4.0 development environment.

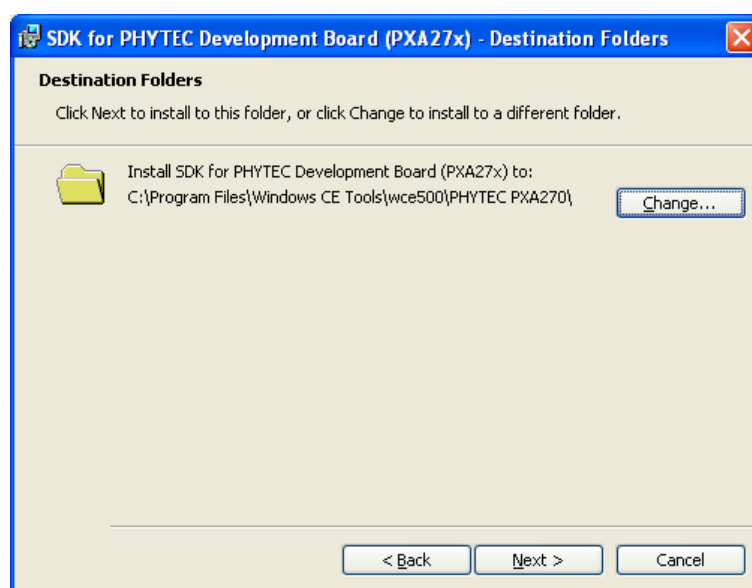
## 2.3 Installing the WinCE PXA270 SDK

The PXA270 SDK for WinCE is needed for writing target-oriented applications. It will integrate in the eMbedded Visual C++ 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 PXA270. 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”.



For proper installation of the SDK it is essential to first install eMbedded Visual C++ as described in 2.2.

Start the installation of the WinCE PXA270 SDK by double-clicking on PHYTECPXA270\_SDK.msi. This file is located on the included WinCE-Xscale-Disc in the folder “WinCE5.0\SDK”. After accepting the license agreement and entering your user name and organization you are asked to select a setup type. Click on *Complete* and in the next dialog select the folder you wish to install the SDK to.



Click *Next* and in the following dialog click *Install* to start the installation process for the SDK. This may take a few minutes.

## 2.4 Installing Microsoft Active Sync 4.1

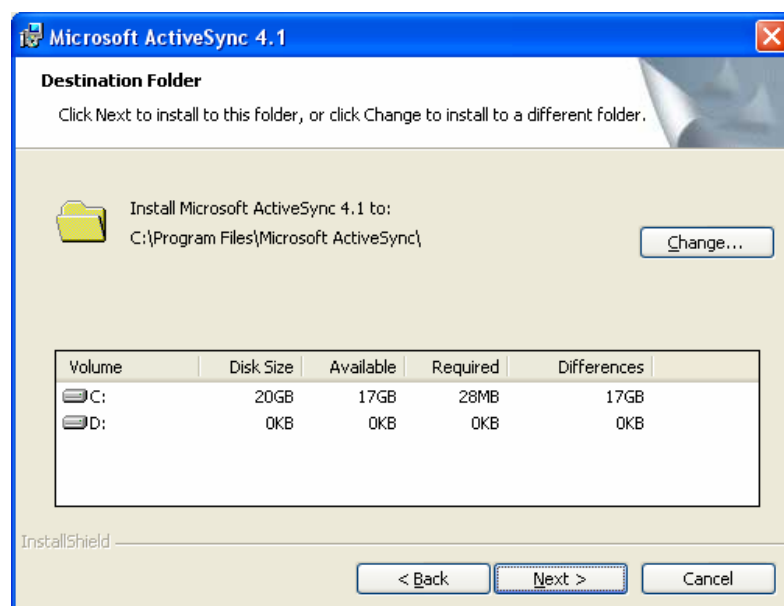
To be able to upload the software you are developing to your PXA270 device you need Microsoft Active Sync 4.1. When your PC is connected to your PXA270 device via a serial cable this software allows you to create a direct connection to your device.

You can download **Microsoft Active Sync 4.1** from the Microsoft website at

<http://www.microsoft.com/downloads/details.aspx?FamilyID=4c254e3f-79d5-4012-8793-d2d180a42dfa&DisplayLang=en>

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

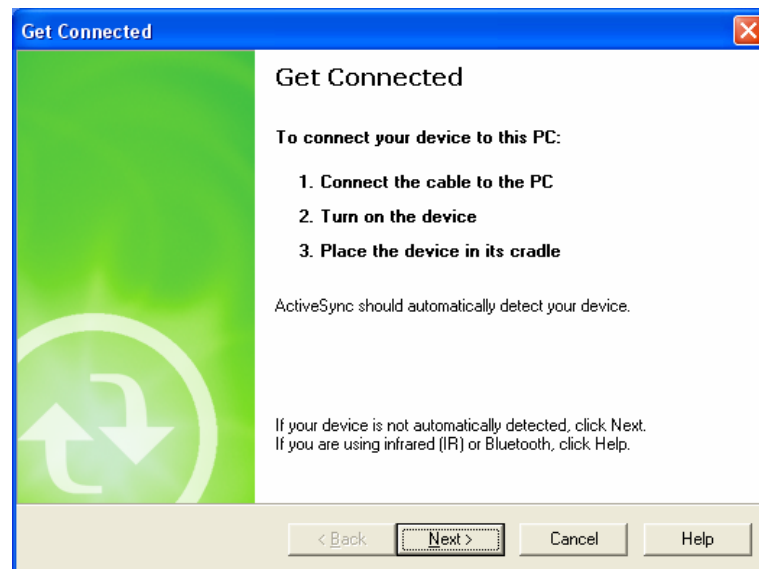
After accepting the license agreement and entering your user name and organization you are asked to select the folder you wish to install Active Sync 4.1 to.



Choose a folder and click *Next* and in the following dialog click *Install* to start the installation process for Microsoft Active Sync 4.1. This may take a few minutes.

## 2.5 Establishing an Active Sync Connection

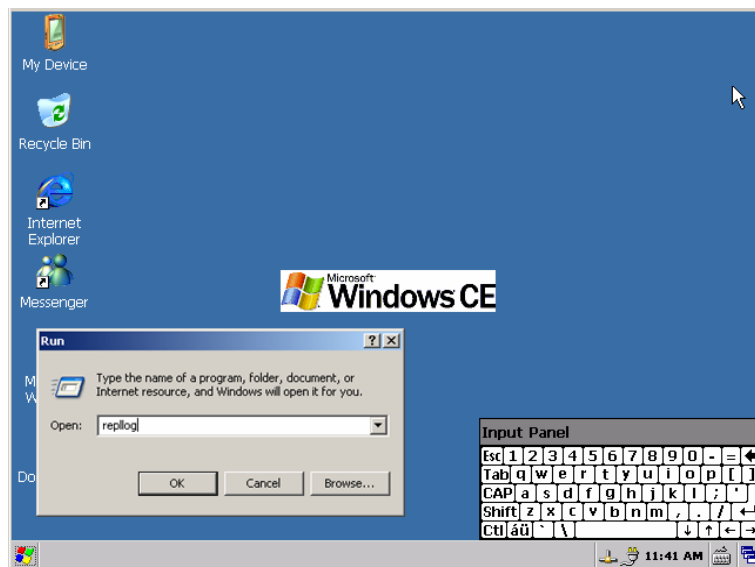
After the installation is complete, Active Sync will try to establish a connection to your PXA270 device.



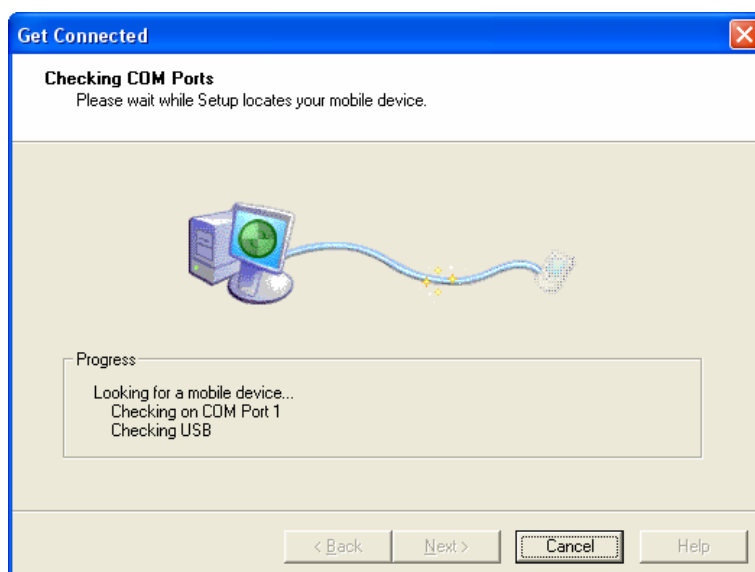
The PXA270 device should already be connected to the PC as described in chapter 2.1 and running.

You must start the tool *repllog* that handles the communication with ActiveSync. To do this, click on *Start* in the task bar of your PXA270 and select *Run...*

Open the virtual keyboard by clicking on the icon on the right corner of the taskbar. Set the focus back on the *Run* dialog and type the program name *repllog* into the dialog box using the virtual keyboard.

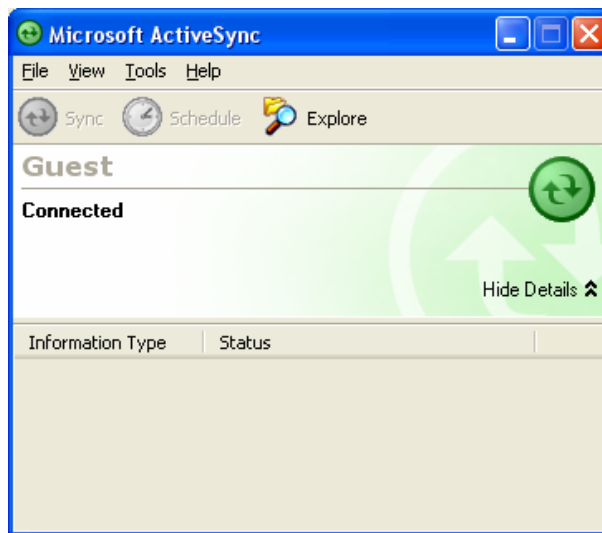


The program *repllog* will enable your device to connect with Active Sync which is running on your PC. Do NOT press *OK* yet. Instead click *Next* on the *Get connected* dialog running on your PC. This will cause Active Sync to start looking for a connection to a mobile device.



While Active Sync is looking for a connection click *OK* in the *Run..* dialog on your PXA270 device to enable the connection to your device. Establishing the connection might take a few seconds. Once

the connection has been established Active Sync will show a screen with status information.

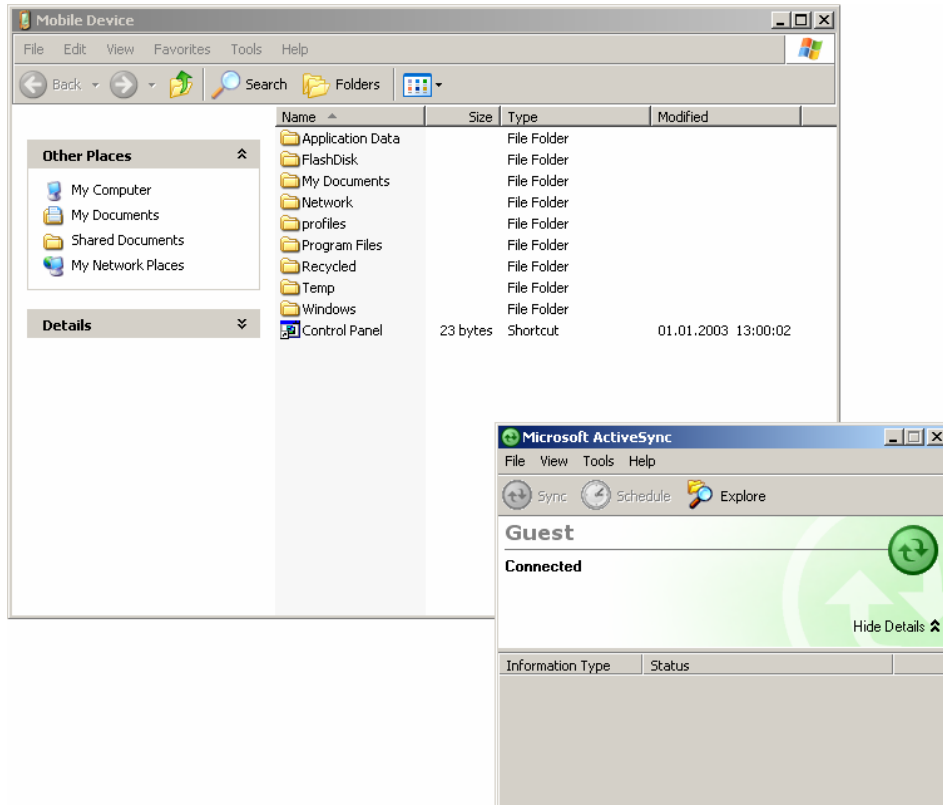


Your PC is now connected to your PXA270 device.

## 2.6 Downloading Example Code with Active Sync

To download the example code provided on the WinCE-Xscale-Disc, you need to have Active Sync running on your PC to be able to establish a connection to your PXA270 device as described in the previous chapter.

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 PXA270 device.



Open a file manager and navigate to the WinCE-Xscale-Disc in your CD-Rom drive. Go to the *WinCE50/Demo* folder. Select *HelloWorld.exe* and copy it to the *Mobile Device* window in the folder *FlashDisk* using drag and drop.



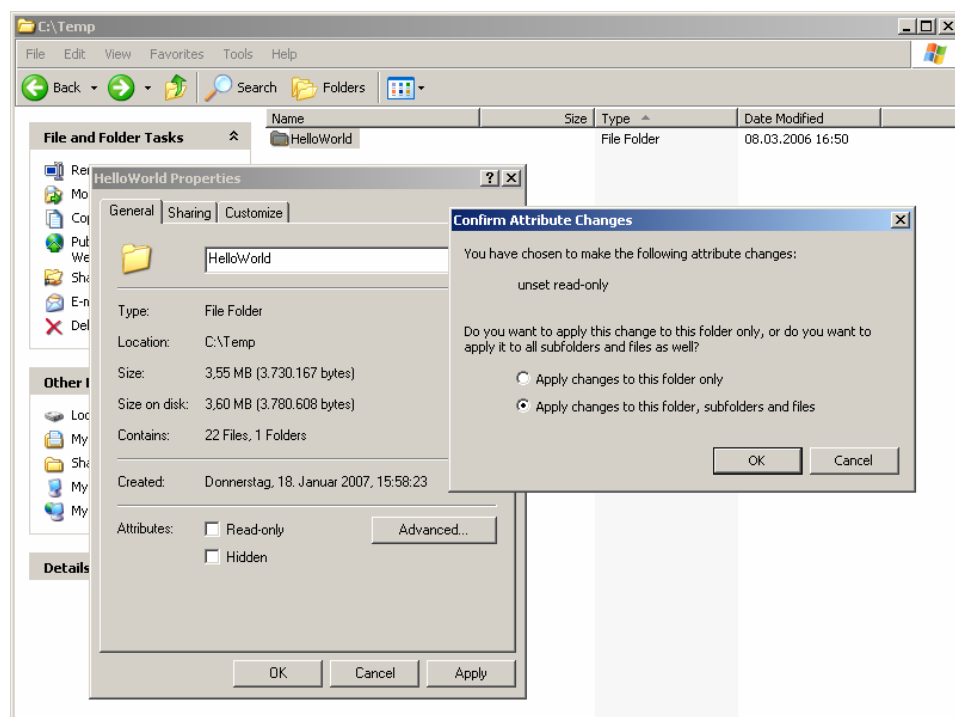
If you do not want the *HelloWorld* program to stay on your device after you turn it off you can copy it to any other location you wish. If you want the file to stay on your device after it has been turned off you need to copy it to your device's flash. You can do this by copying the file to the "FlashDisk" folder.

The program has now been copied to your PXA270 device. You will be able to access it by double-clicking the *My Device* icon on your PXA270 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 PXA270 device that will print a *Hello World* message.

## 2.7 Downloading Example Code with eMbedded Visual C++

PHYTEC provides you with example code to learn about downloading code onto your PXA270 device. To use this code, you need to copy it from the WinCE-Xscale-Disc to your local hard drive.

Open a file manager and navigate to the WinCE-Xscale-Disc in your CD-Rom drive. Go to the *WinCE50/Demo* folder. Select the *HelloWorld* folder and copy it to a location on your local hard drive. To be able to edit the example code you need to make sure that the files are not write-protected. To do this right-click on the folder you just copied and select *Properties*. In the *Properties* window deselect the *Read-only* attribute and click on *OK*. You will be asked if you wish to apply this change only to this folder or to all subfolders and files as well. Chose *Apply changes to this folder, sub-folders and files* and click on *OK*. The example code will now be editable.



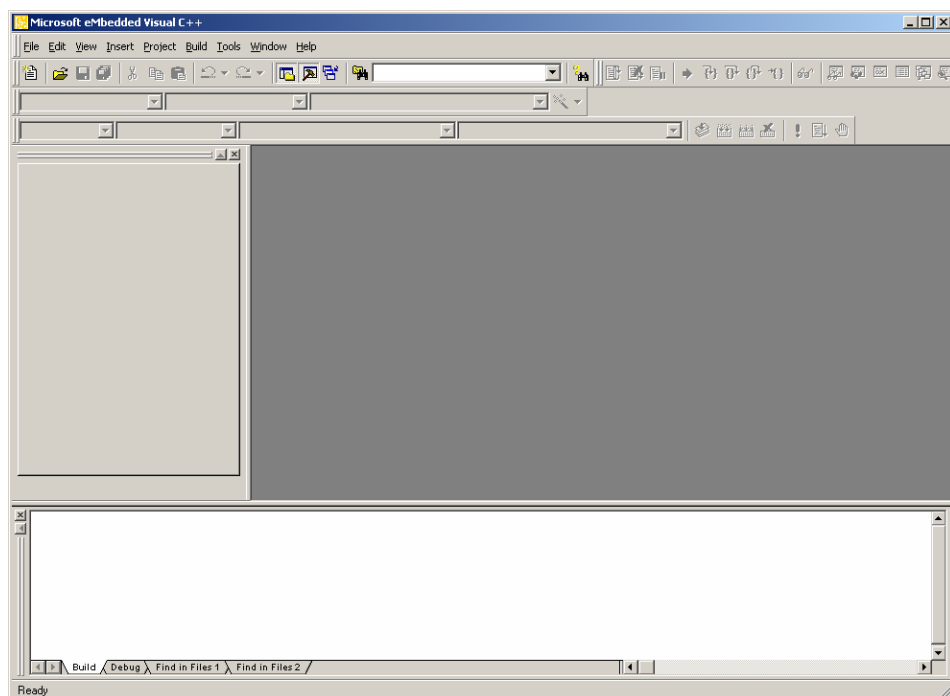
In sections 2.2 of this QuickStart manual you have installed the eMbedded Visual C++ software development tool chain.

**Note:**

It is recommended to use the Microsoft tool chain described in this QuickStart manual in order to complete this QuickStart Instruction successfully. Use of a different version can lead to version conflicts, resulting in functional problems<sup>1</sup>.

Start the tool chain by selecting eMbedded Visual C++ 4.0 from the programs group: *Start\Programs\Microsoft eMbedded Visual C++4.0* or by double-clicking the icon eMbedded Visual C++ 4.0 on your desktop.

After starting eMbedded Visual C++ 4.0 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 *Project / Close Project*.



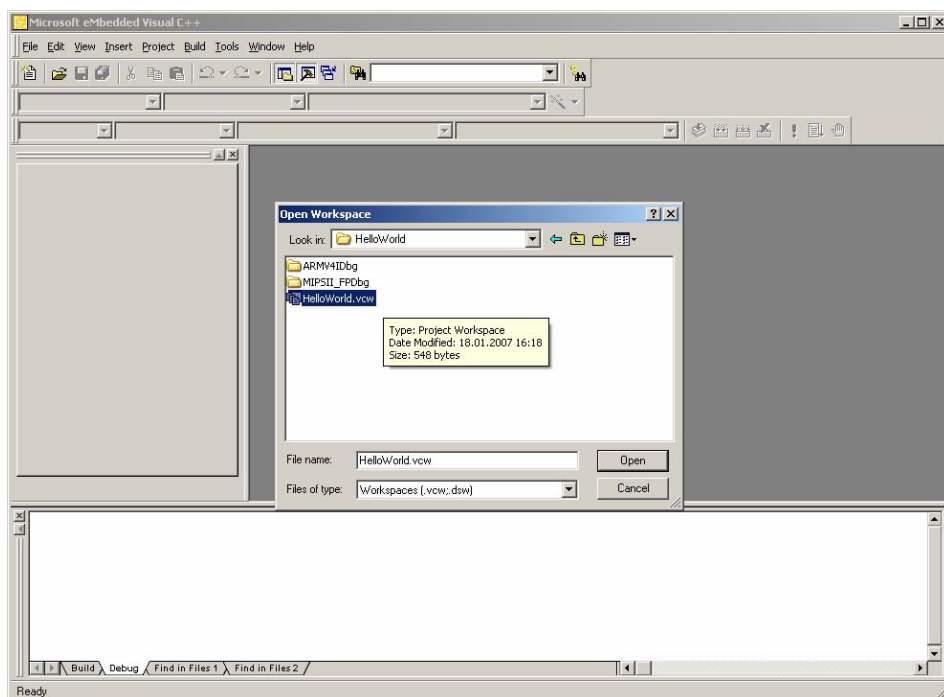
---

<sup>1</sup> : **Note:** If you have a full version of eMbedded Visual C++ already installed on your host-PC we recommend installing this version on a different desktop or laptop computer. If this is not possible we recommend using the same version of eMbedded Visual C++ that we use in this QuickStart manual. PHYTEC can not guarantee successful completion of these QuickStart instructions if a different version of eMbedded Visual C++ is used.



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 PXA270.

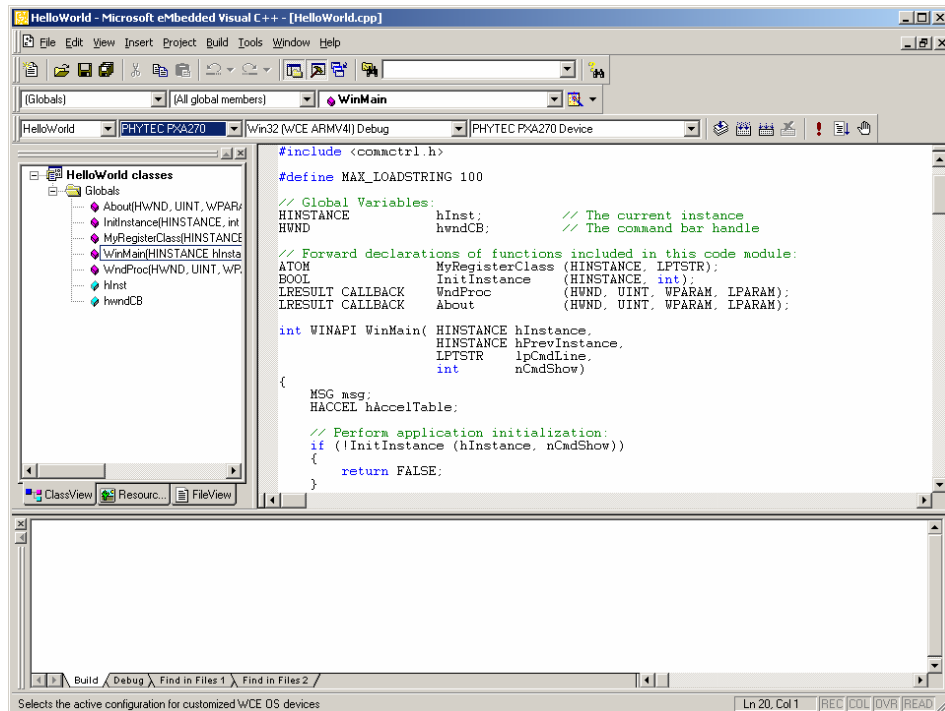
- To open the HelloWorld project select *File / Open Workspace...* from the eMbedded Visual C++ menu bar.
- Browse to the folder you copied the example code to and open the *HelloWorld* folder.
- Select the HelloWorld project *HelloWorld.vcw*.




- Click *Open*.

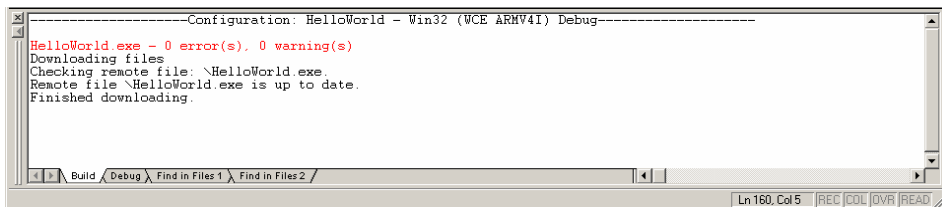


- In the **Select Active WCE Configuration** pull-down menu make sure that the *PHYTEC PXA270* configuration is selected. In the **Select Default Device** pull-down menu *PHYTEC PXA270 Device* should be selected.




## Build the Project

- Build the target by either selecting the *Build* icon  on the build toolbar or in the main menu bar select *Build / Build HelloWorld.exe*.
- 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 will be automatically downloaded to your PXA270 device. For this to work you need to have established an Active Sync connection between your PC and your PXA270 device as described in section 2.5.
- The individual steps of the download procedure can be viewed in the *Output Window - Build* tab.
- Wait until the download is complete. This is indicated by the *Finished downloading* message.



```
Configuration: HelloWorld - Win32 (VCE ARMV4I) Debug
HelloWorld.exe - 0 error(s), 0 warning(s)
Downloading files
Checking remote file: \HelloWorld.exe
Remote file \HelloWorld.exe is up to date.
Finished downloading.
```

- To execute the code click on the *Execute Program* icon  on the build toolbar of eMbedded Visual C++ 4.0 or in the main menu bar select *Build / Execute HelloWorld.exe*. eMbedded Visual C++ 4.0 will establish a connection to your device via Active Sync and trigger the execution of your program code. A window will open on your PXA270 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*

### 3 Getting More Involved

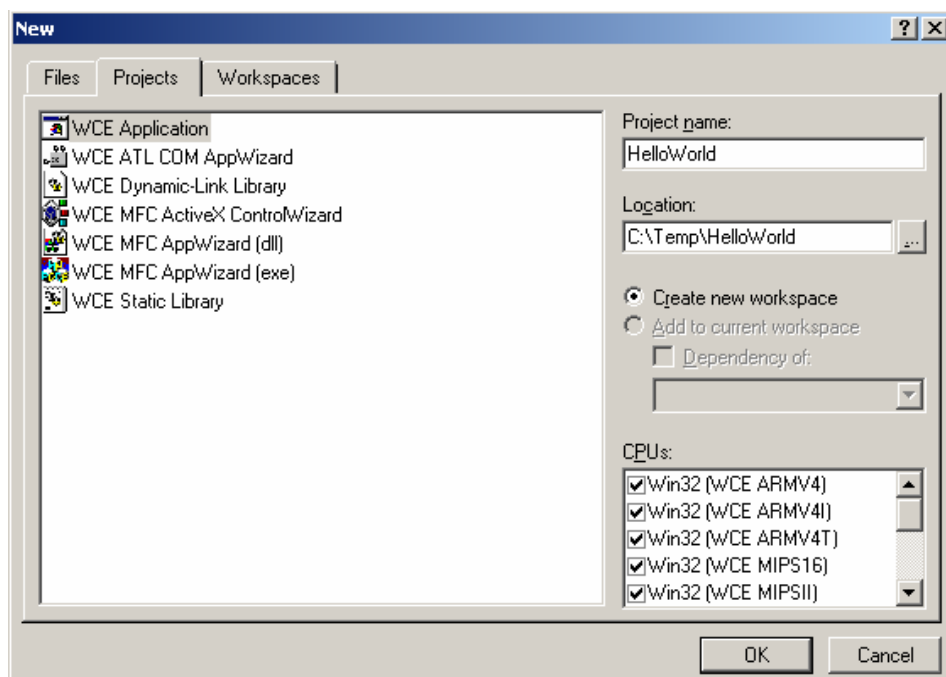


30 min

*In this chapter you will learn how to configure the eMbedded Visual C++ 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.*

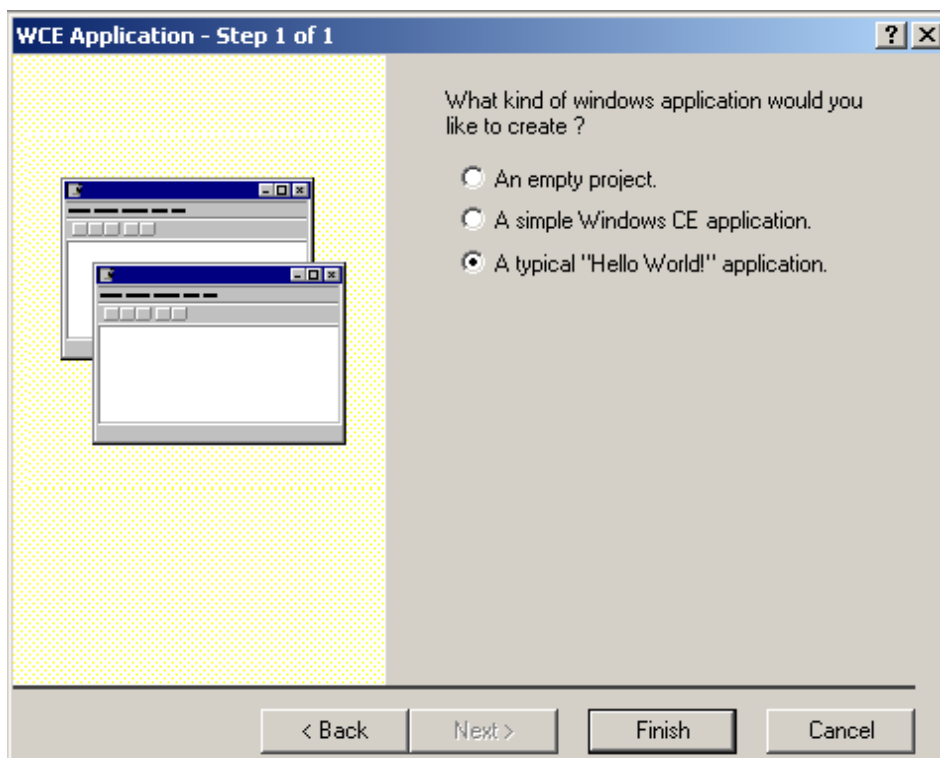
#### 3.1 Creating a New Project

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

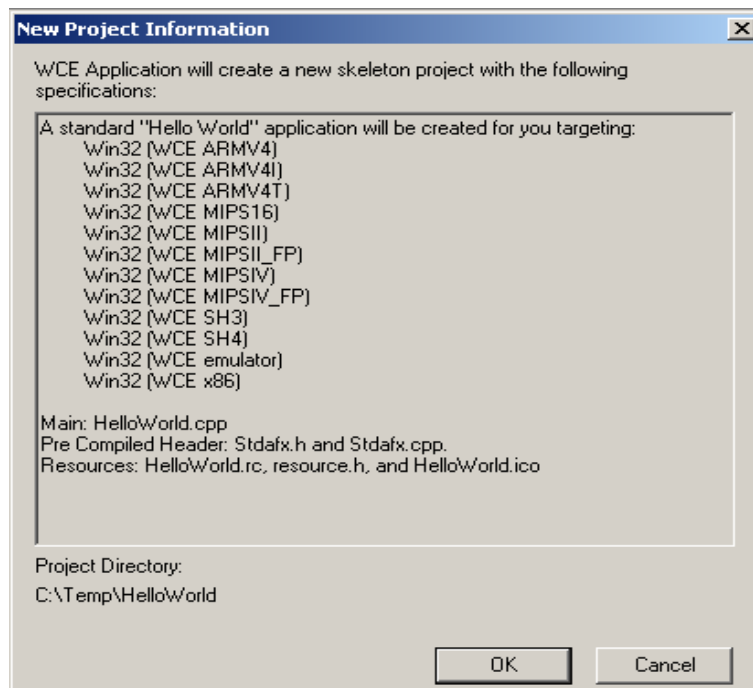


- Click on the *Projects* tab and select **WCE Application**.

- In the *Location* text field enter the folder you wish to have your project created in.
- In the text field *Project name*, enter the name of the project you wish to create.
- Leave the *CPUs* selection as it is and click on *OK*.
- Next you will be asked what kind of Windows application you wish to create. Chose *A typical "Hello World" application'* and click on *Finish*.



- An information screen will appear that shows you the files that will be created for your project as well as the targets it will be created for.

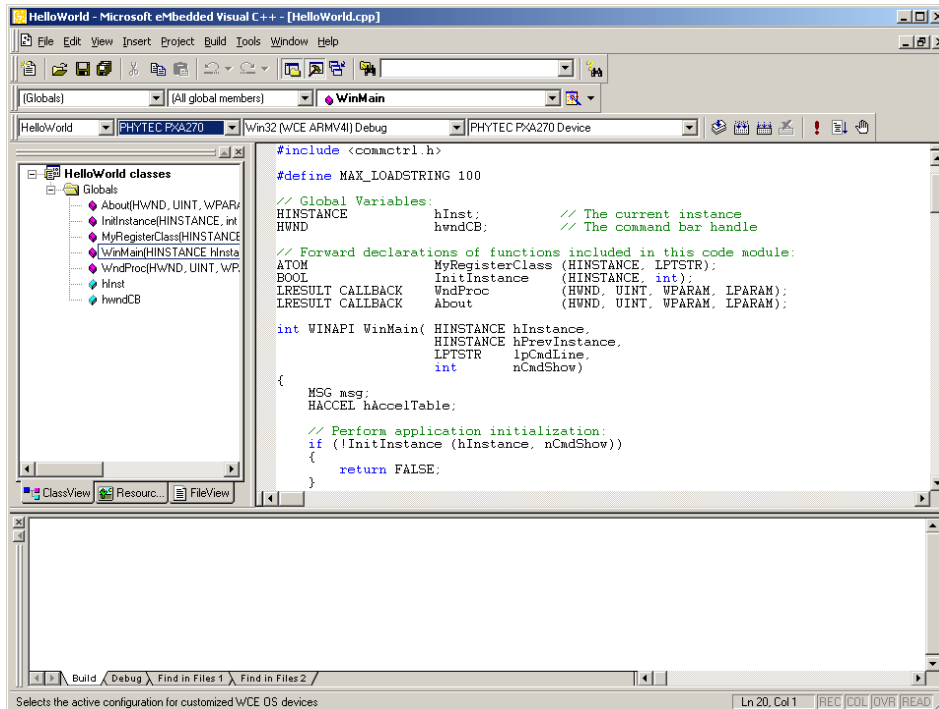


- Click on OK to finish the project creation and to open up your new project.

At this point you have created a new project. The next step is to modify the C source file before building your project. This includes compiling, linking, locating and creating the machine-readable file.

## 3.2 Modifying the Source Code

- Open up the *Globals* folder and double-click on the *WinProc* function in the *Class View* window to open the file in the source code editor.



- Locate the following code section to make your HelloWorld program print the text **Hello World!** 10 times instead of just one.

```

DrawText(hdc, szHello, _tcslen(szHello), &rt,
         DT_SINGLELINE | DT_VCENTER | DT_LEFT);


```

- Replace this code section with the following:

```

int i;
for(i = 1; i <= 10 ; i++)
{
    rt.top = rt.top + 30;
    DrawText(hdc, szHello, _tcslen(szHello), &rt,
             DT_SINGLELINE | DT_TOP | DT_CENTER);
}


```

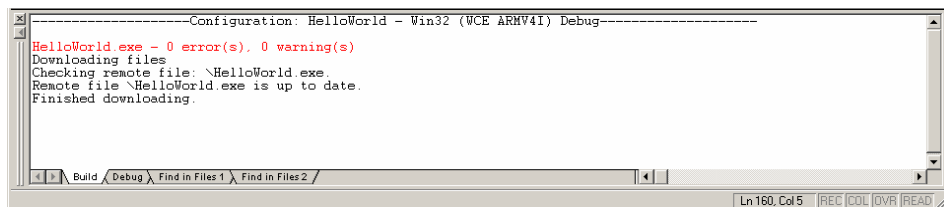
- Save the modified file by choosing *File / Save* or by clicking the floppy disk icon .



### 3.3 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 YourProgramName.exe*.
- 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 will be automatically downloaded to your PXA270 device. For this to work you need to have established an Active Sync connection between your PC and your PXA270 device as described in section 2.4.
- The individual steps of the download procedure can be viewed in the *Output Window - Build* tab.
- Wait until the download is complete. This is shown by the *Finished downloading* message.



```

Configuration: HelloWorld - Win32 (VCE ARMV4T) Debug
HelloWorld.exe - 0 error(s), 0 warning(s)
Downloading files
Checking remote file: \\HelloWorld.exe
Remote file \\HelloWorld.exe is up to date.
Finished downloading.

```

- To execute the code click on the *Execute Program* icon  on the build toolbar of eMbedded Visual C++ 4.0 or in the main menu bar select *Build / Execute YourProgramName.exe*. eMbedded Visual C++ 4.0 will establish a connection to your device via Active Sync and trigger the execution of your program code. A window will open on your PXA270 device that will print a **Hello World** message at the location you specified in your code modification.



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

## 4 Debugging



**40 min**

*This Debugging section provides a basic introduction to the debug functions included in eMbedded Visual C++ 4.0. 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.*

### 4.1 Starting the Debugger

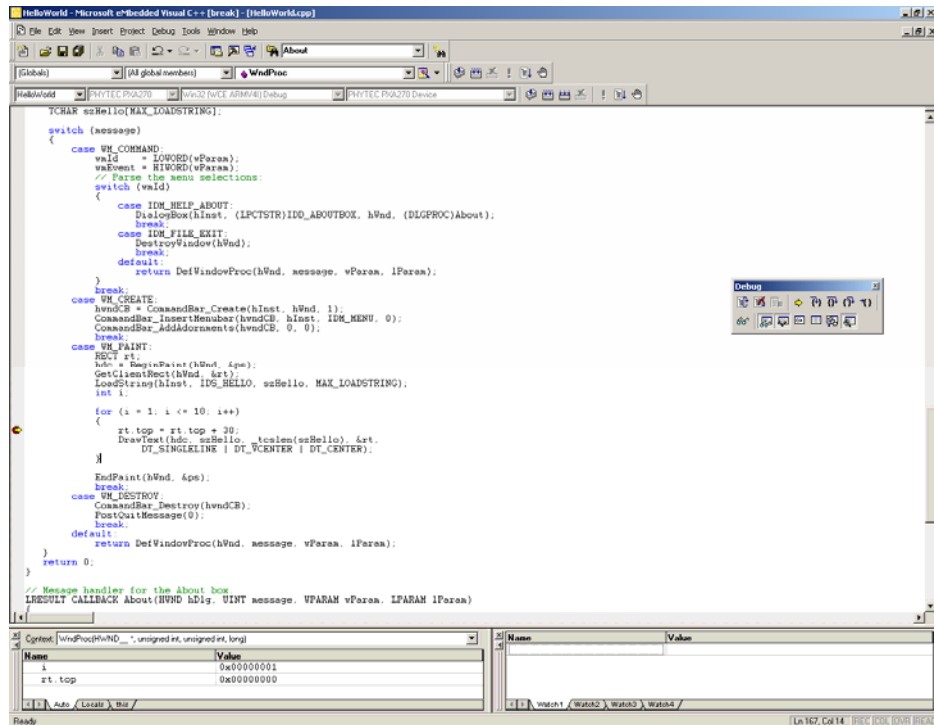
- Make sure that ‘Win32 (WCE ARMV4I) Debug’ is selected in the Select Active Configuration pull-down menu.
- Open up the *Globals* folder and double-click on the *WinProc* function in the *Class View* window to open the file in the source code editor.
- Locate the following code section

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

- Click on *Insert/Remove Breakpoint* icon  on the *WCE Configuration* toolbar 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 eMbedded Visual C++ 4.0, click on the *Go* icon  on the *WCE Configuration* toolbar 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 PXA270 device as described in section 2.4.

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.



```

TCHAR szHello[MAX_LOADSTRING];

switch (message)
{
    case WM_COMMAND:
        wParam = LOWORD(wParam);
        wParam = HIWORD(wParam);
        // Parse the menu selections:
        switch (wParam)
        {
            case IDM_HELP_ABOUT:
                DialogBox(hInst, (LPCSTR)IDD_ABOUTBOX, hWnd, (DLGPROC)About);
                break;
            case IDM_FILE_EXIT:
                DestroyWindow(hWnd);
                break;
            default:
                return DefWindowProc(hWnd, message, wParam, lParam);
        }
        break;
    case WM_CREATE:
        hWndCB = CommandBar_Create(hInst, hWnd, 1);
        CommandBar_InsertMenuBar(hWndCB, hInst, IDM_MENU, 0);
        CommandBar_AddAdornments(hWndCB, 0, 0);
        break;
    case WM_PAINT:
        RECT rt;
        hdc = BeginPaint(hWnd, &pe);
        GetClientRect(hWnd, &rt);
        LoadString(hInst, IDS_HELLO, szHello, MAX_LOADSTRING);
        int i;
        for (i = 1; i <= 10; i++)
        {
            rt.top = rt.top + 30;
            DrawText(hdc, szHello, _tcslen(szHello) &rt,
                DT_SINGLELINE | DT_CENTER | DT_CENTER);
        }
        EndPaint(hWnd, &pe);
        break;
    case WM_DESTROY:
        CommandBar_Destroy(hWndCB);
        PostQuitMessage(0);
        break;
    default:
        return DefWindowProc(hWnd, message, wParam, lParam);
}
return 0;
}
// Message handler for the About box
RESULT CALLBACK About(HWND hDlg, UINT message, WPARAM wParam, LPARAM lParam)

```

Name	Value
i	0x00000001
rt.top	0x00000000



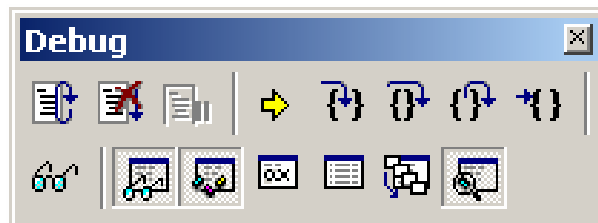
You may need to open, resize and /or move some windows to make your screen look similar to the screen capture. You can open inactive windows by choosing the desired window from the *View* pull-down menu.

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 *Insert/Remove Breakpoint* icon on the *WCE Configuration* toolbar or press F9 again to remove the breakpoint you have set.

## 4.2 eMbedded Visual C++ 4.0 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*.



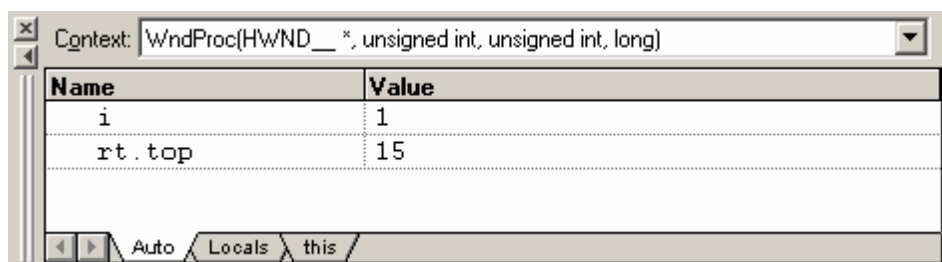
- The first button on the debugger toolbar is the *Restart* button. The *Restart* command sets the program counter to 0.
- The *Stop Debugging* button stops the debugger and will lead you back to your project screen.
- The *Break Execution* button stops the execution of the program.
- The *Show Next 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 last button on the debugger toolbar performs the *Run to Cursor* command. The *Run to Cursor line* command executes the program to the current cursor position within the code window. This allows use of the cursor line as a temporary breakpoint.

### 4.3 Using the eMbedded Visual C++ 4.0 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 10. 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.
- As a last example, the variable **szHello** will be evaluated. Go to source code line #130 where the variable **szHello** is declared. Mark **szHello** by double-clicking it and drag it to the *Watch 1* tab in the *Watch Window*.

Name	Value
⊕ szHello	0x1a02f9c4 "Hello World!"
◀ ▶ Watch1 Watch2 Watch3 Watch4	

- The constant is shown with its address and a small sign in front which indicates that **szHello** is an array with a group of array elements. Click the sign to expand the view and to see all array elements of **szHello**.

Name	Value
⊖ szHello	0x1a02f9c4 "Hello World!"
— [0x0]	0x0048
— [0x1]	0x0065
— [0x2]	0x006c
— [0x3]	0x006c
◀ ▶ Watch1 Watch2 Watch3 Watch4	

#### 4.4 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 PXA270 device.
- You can use the *Stop Debugging*  button to stop program execution at any time.

## 4.5 Changing Target Settings for the "Executable Version"

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<sup>®</sup>-PXA270.

- Make sure the program execution is stopped.
- Exit the current debug session by selecting *Stop Debugging*.
- In the *Select Active Configuration* pull down menu select 'Win32 (WCE ARMV4I) 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 PXA270 device.



*You have successfully finished this QuickStart.*

## 5 Summary

This QuickStart Instruction gave a general "Rapid Development Kit" description, as well as software installation advice and an example program enabling quick out-of-the box start-up of the phyCORE®-PXA270 in conjunction with WinCE.

In the Getting started section you learned how to configure your host to provide a basis for working with your target platform. You installed the Rapid Development Kit software and you learned to copy and run a program on the target.

In the Getting More Involved section you got a step-by-step instruction on how to modify the example, create and build new projects and copy output files to the phyCORE -PXA270 using eMbedded Visual C++.

The Debugging part of this QuickStart gave you information on setting up and using the debugger with eVC++. You learned how to set breakpoints, watching and changing variables content.





---

**Document: WinCE-Kit with phyCORE®-PXA270 QuickStart  
Instructions**  
**Document number: L-675e\_2, November 2006**

---

**How would you improve this manual?**

---

---

---

---

**Did you find any mistakes in this manual?** \_\_\_\_\_ page

---

---

---

**Submitted by:**

Customer number: \_\_\_\_\_

Name: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_

\_\_\_\_\_

**Return to:**

PHYTEC Technologie Holding AG  
Robert-Koch-Str. 39  
55129 Mainz, Germany  
Fax: +49 (6131) 9221-26

