

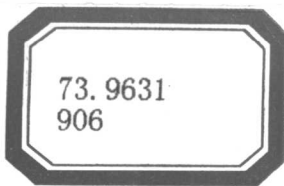
嵌入式系统 Windows CE 开发技巧与实例

傅曦 齐宇 编



化学工业出版社

工业装备与信息工程出版中心



郑州大学 01011247981Y

嵌入式系统 Windows CE

开发技巧与实例

傅曦 齐宇 编

图书在版编目(CIP)数据



化学工业出版社

工业装备与信息工程出版中心

·北京·

73.9631
906

6222

(京)新登字 039 号

图书在版编目(CIP)数据

嵌入式系统 Windows CE 开发技巧与实例/傅曦, 齐宇编.
北京: 化学工业出版社, 2003. 10
ISBN 7-5025-4854-8

I. 嵌… II. ①傅…②齐… III. 窗口软件, Windows CE
IV. TP316.7

中国版本图书馆 CIP 数据核字 (2003) 第 090889 号

嵌入式系统 Windows CE 开发技巧与实例

傅曦 齐宇 编

责任编辑: 李玉晖

责任校对: 陈 静

封面设计: 潘 峰

*

化 学 工 业 出 版 社 出版发行
工业装备与信息工程出版中心

(北京市朝阳区惠新里 3 号 邮政编码 100029)

发行电话: (010) 64982530

<http://www.cip.com.cn>

*

新华书店北京发行所经销

北京管庄永胜印刷厂印刷

三河市东柳装订厂装订

开本 787 毫米×1092 毫米 1/16 印张 22 字数 542 千字

2004 年 1 月第 1 版 2004 年 1 月北京第 1 次印刷

ISBN 7-5025-4854-8/TP·340

定 价: 45.00 元

版权所有 违者必究

该书如有缺页、倒页、脱页者, 本社发行部负责退换

本书实例代码下载及支持网站

书中提供了大量的程序实例，读者可以复制并使用。实例代码下载网址为：

<http://www.cnemb.com/wincetips.rar>

对于本书的相关问题和解答请到以下网址递交：

<http://www.cnemb.com/index.php?name=PNphpBB2&file=viewforum&f=42>

第 1 章 EVC 概述	1
1.1 EVC 的新特性	1
1.2 事件驱动和消息响应机制	1
1.3 熟悉 EVC 开发环境	2
1.3.1 建立应用程序	2
1.3.2 类及文件说明	5
1.3.3 编译和运行 Hello Windows CE 程序	7
1.3.4 辅助开发工具	7
1.3.5 Windows CE 程序和 Windows 程序的不同点	9
1.3.6 系统化移植的关键要素	11
第 2 章 系统	16
2.1 软输入面板 (SIP)	16
2.1.1 背景	16
2.1.2 显示/隐藏 SIP	16
2.1.3 移动 SIP	17
2.1.4 废除 SIP 的单词自动填充	17
2.1.5 恢复单词自动填充	17
2.1.6 SIP 标示要求	18
2.2 密码保护	19
2.2.1 背景	19
2.2.2 在 Pocket PC 上的特性	20
2.2.3 Pocket PC2002 的新特性	20
2.2.4 设置密码的延迟时间	20
2.2.5 使用 STPasswordManager 库	20
2.3 硬件键	21
2.3.1 背景	22
2.3.2 硬件键的虚拟代码	22
2.3.3 使用应用程序运行键	22
2.3.4 使用 STUtil 类	23
2.4 后台程序	24
2.4.1 背景	24
2.4.2 开始	24
2.4.3 创建没有窗口的程序	25
2.4.4 结论	26
2.5 创建控制面板程序	26

2.5.1	背景	26
2.5.2	创建控制面板程序的步骤	27
2.5.3	使用示例	28
2.6	存储卡	29
2.6.1	无驱动字符	29
2.6.2	存储卡定位	29
2.6.3	CStorageCard 类库	29
第 3 章	菜单、对话框和窗口	33
3.1	菜单	33
3.1.1	WM_COMMAND 消息和菜单响应函数	33
3.1.2	CMenu 类	34
3.1.3	上下文菜单	36
3.1.4	菜单操作技巧	37
3.2	对话框	38
3.2.1	使用资源编辑器编辑对话框	38
3.2.2	对话框的数据交换和数据检查	39
3.2.3	模态和非模态对话框	39
3.2.4	通用对话框	42
3.3	窗口	43
3.3.1	产生 CWnd 对象	43
3.3.2	消息映射	44
3.3.3	关闭窗口	44
3.3.4	CWnd 和句柄	45
3.3.5	SDMV 应用中的窗口切换函数	45
第 4 章	控件	49
4.1	Windows 标准控件和通用控件	49
4.1.1	静态控件	49
4.1.2	按钮控件	51
4.1.3	编辑控件	54
4.1.4	单选按钮	58
4.1.5	复选按钮	61
4.1.6	组合框控件	63
4.1.7	列表框控件	71
4.1.8	滚动条控件	74
4.1.9	树状控件	75
4.1.10	时间控件	78
4.1.11	月历控件	79
4.2	MFC 控件类介绍及使用范例	80
4.2.1	Ccebottom 类	80
4.2.2	Grid 类	85

4.2.3 Process 类	93
第 5 章 屏幕绘图	97
5.1 基本文本操作	97
5.1.1 文本输出函数	97
5.1.2 文本属性	99
5.1.3 字符属性	101
5.1.4 字体	102
5.1.5 文本显示特殊技巧	104
5.2 基本图形图像操作	107
5.2.1 绘图函数	107
5.2.2 画笔	109
5.2.3 画刷	112
5.2.4 位图画刷	113
5.2.5 方便实用的画笔和画刷类	115
5.2.6 绘图模式	116
5.2.7 位图	117
5.2.8 位操作	118
5.2.9 图标	120
5.3 高级屏幕绘图	120
5.3.1 DIB 类	120
5.3.2 Pocket PC 2002 中未公开的 API	124
5.3.3 imgdecmp.dll 和 VOImage 类	124
5.3.4 IJG JPEG 库	126
5.3.5 Windows CE 高速图形库	128
5.3.6 二维实时图形	131
5.3.7 分析图表类库	132
第 6 章 多线程和多进程	140
6.1 Windows CE 的多任务机制	140
6.1.1 多线程和多进程机制	140
6.1.2 线程调度和优先级	140
6.2 多线程	141
6.2.1 创建线程	142
6.2.2 线程间同步	143
6.2.3 终止线程	147
6.2.4 多线程开发技巧	147
6.3 多进程	149
6.3.1 创建和终止进程	149
6.3.2 进程间共享数据	152
第 7 章 文件和数据库	155
7.1 文件操作	155

7.1.1	创建和打开文件	155
7.1.2	文件读写和关闭文件	156
7.1.3	基本文件操作示范	157
7.1.4	MFC 文件操作类	159
7.2	ADOCE 和 Pocket Access	161
7.2.1	ADOCE 对象	162
7.2.2	SQL 与 ADOCE	164
7.2.3	SQL 语句语法	169
7.2.4	ADOCE 与 OLE DB	171
7.2.5	通过 VOADO 类操作 Pocket Access	171
7.2.6	将 Access 转换成 Pocket Access	174
7.3	SQL Server 2000 For Windows CE 版	178
7.3.1	安装 SQL Server 2000 For Windows CE	178
7.3.2	SQL Server 2000 For CE 连接配置	179
7.3.3	SQL CE Query Analyzer	182
7.3.4	远程数据访问 (RDA)	182
7.3.5	合并复制 (Merge Replication)	188
7.3.6	STADOCE 类库应用实例	195
第 8 章	同步和远程连接	199
8.1	串行通信	199
8.1.1	RS-232 串行接口标准	199
8.1.2	打开和关闭串行端口	200
8.1.3	读写串行端口	201
8.1.4	配置串行端口	202
8.1.5	出错处理	204
8.1.6	异步多线程通讯	205
8.1.7	串口通信示例	205
8.2	红外线通讯	207
8.2.1	IrDA 协议概述	207
8.2.2	原始 IRDA	209
8.2.3	IrCOMM	210
8.2.4	IrSock	211
8.2.5	红外通信范例	216
8.3	RAPI	220
8.3.1	预定 RAPI 函数	220
8.3.2	自定义 RAPI 函数	221
8.3.3	RAPI 函数使用范例	222
8.4	ActiveSync	224
8.4.1	ActiveSync 概念	224
8.4.2	ActiveSync 提供者程序	224

8.4.3	AdoFiltr 程序	228
第 9 章	游戏开发	231
9.1	GapiDraw 总览	231
9.1.1	来自 DirectDraw 开发者观点的 GapiDraw	231
9.1.2	来自 GapiTools 开发者观点的 GapiDraw	234
9.2	创建项目文件	238
9.2.1	简介	238
9.2.2	Microsoft Visual C++ 6.0	240
9.2.3	Microsoft Embedded Visual C++ 3.0	242
9.3	创建 Alpha 映射	245
9.4	类	247
9.4.1	CGapiApplication	247
9.4.2	CGapiSurface	252
9.4.3	CGapiDisplay	268
9.4.4	CGapiMaskSurface	271
9.4.5	CGapiBitmapFont	273
9.4.6	CGapiInput	275
9.4.7	CGapiTimer	276
9.4.8	CGapiCursor	277
9.5	GapiDraw Structs	279
9.6	GapiTools 类概述	283
9.6.1	CGapiBuffer 类	283
9.6.2	CGapiTool 类	284
9.6.3	CGapiTimer 类	293
9.7	GapiDraw 使用范例	293
第 10 章	调试技巧	300
10.1	EVC 的设置编译选项	300
10.2	错误	301
10.3	调试	303
10.3.1	启动 Debugger	303
10.3.2	Debugger 窗口	305
10.3.3	Debugger 快捷键	305
10.4	几个重要的宏	306
10.5	日志文件	307
第 11 章	帮助和安装程序	310
11.1	帮助	310
11.1.1	创建一个帮助系统	310
11.1.2	向程序中添加帮助	313
11.1.3	PocketHelp 帮助制作工具	314
11.2	安装程序	317

11.2.1	CAB 向导	317
11.2.2	程序管理器	325
11.2.3	Windows CE 路径标识符	328
11.2.4	PocketSetup	328
11.2.5	EZSetup	336
参考文献		339

第 1 章 EVC 概述

1.1 EVC 的新特性

Embedded VisualC++(EVC) 3.0 是 Microsoft 公司推出的 Windows CE 程序的可视化开发工具, 是 Embedded Visual Tools 的重要组成部分。利用 EVC3.0 用户可以开发出规模较大, 功能较复杂的 Windows CE 应用程序和流式驱动程序。EVC3.0 具有的新特性概括起来有如下几点:

- 高效智能的开发环境
- 内置的某些特性提高了编辑代码速度
- 更快的编译输出
- 编译后优化代码
- 更容易组合和扩充控件
- ClassView 能自动动态更新
- 允许编译后的应用程序延时加载 DLL

和传统的桌面开发工具 Visual C++ 6.0 相比, EVC3.0 有一些为嵌入式开发而设计的特点, 表 1-1 列举了两者的不同点。

表 1-1 Visual C++ 6.0 和 EVC 的不同点

Visual C++ 6.0	EVC 3.0
Win32 API	Win32 API 的子集
C 运行时库	标准 C 运行时库的子集
C/C++ 编译器	● 为(x86)模拟器提供的 X86 C/C++编译器 ● 为 ARM 设备提供的 CLARM C/C++编译器 ● 为 MIPS 和 MIPSFP 设备提供的 CLMIPS C/C++编译器 ● 为 PowerPC 设备提供的 CLPPC C/C++编译器 ● 为 SuperH 设备提供的 SHCL C/C++编译器
通用开发工具	使用和 Windows 环境类似的开发工具, 还附加了一些为 Windows CE 设备提供的远程开发调试工具, 并具有能生成 Windows CE 格式资源的资源编辑器, 能生成 4 色位图等 Windows CE 格式的资源
支持微软基础类(Microsoft Foundation Classes MFC)	Windows CE MFC 是标准 MFC 类库的子集
支持活动模板库(Active Template Library ATL)	支持大多数 ATL 类
支持标准模板库(Standard Template Library STL)	不支持 STL

1.2 事件驱动和消息响应机制

Windows 程序设计是一种事件驱动的程序设计模式。在程序提供给用户的界面中有许多可操作的可视对象。用户从所有可能的操作中任意选择, 被选择的操作会产生某些特定的事件, 这些事件发生后的结果是向程序中的某些对象发出消息, 然后这些对象调用相应的消息处理函数来完成特定的操作。Windows 应用程序最大的特点就是程序没有固定的流程, 而只是针对某个事件处理有特定的子流程, Windows 应用程序可以说是由许多这样的

子流程构成的。

Windows 应用程序在本质上是面向对象的。程序提供给用户界面的可视对象在程序的内部一般也看做一个对象,用户对可视对象的操作通过事件驱动模式触发相应的消息处理函数。程序的运行过程就是用户的外部操作不断产生事件,这些事件又不断被处理的过程。

Windows 这种事件驱动模型的实质源于 Windows 的消息响应机制。在 Windows 程序中,事件产生消息,消息对应着事件。所谓事件响应,其实就是对各种消息的响应。用户要操作 Windows 程序,必须借助鼠标或键盘(在 Windows CE 中为触笔、硬件键等)等

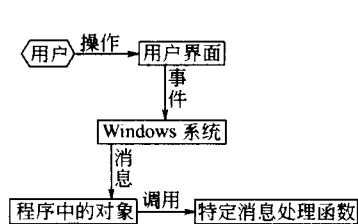


图 1-1 Windows 程序工作原理

人机交互工具。Windows 系统以特定的频率捕捉各种消息,当捕捉到发往本应用程序的消息时,Windows 应用程序将消息再传递给相关的消息处理函数做相应的处理。这种等待消息,响应消息的操作方式就是 Windows 特有的处理机制,称为消息处理机制。在 Windows 系统中,使用者与系统、系统与应用程序、应用程序之间主要就是采用了这种消息响应的处理机制。图 1-1 是 Windows 程序工作原理的示意图。

1.3 熟悉 EVC 开发环境

1.3.1 建立应用程序

(1) EVC 开发环境 在建立应用程序之前,读者应该先对自己所用的开发环境有一定的了解。开发环境如图 1-2 所示。

(2) 生成项目 在写 C++ 程序时,首先是要建立一个应用程序项目(Project)。EVC 项目中存放了建立应用程序所需要的全部信息,包括应用程序所有源文件名称及相互关系、所有库文件清单、编辑器、链接器、编译器等。

建立第一个应用程序。首先从菜单中选择 File 中的 New,则出现程序类型选择对话框,读者可以选中 WIN CE Pocket PC 2002 MFC AppWizard[exe] 标签,在 Project name 文本框中填入你的项目名称 chapter1_1,在 Location 中填上为该例程指定的目录,如图 1-3 所示。

如果想知道屏幕上操作的方法,可以单击窗口右上角的“?”按钮,然后点击想询问的操作位置。例如,如果想知道 Create new workspace 的含义,可以再单击此位置,则会出现说明相应的信息。

(3) 向导过程 进入程序向导后,AppWizard 会自动完成读者在向导中所指定的内容。在 AppWizard 的各步中读者可以做许多自定义。图 1-4 是 AppWizard 工作的第一步。在这一步中,读者需要为应用程序确定一个基本框架结构。有 3 种基本的框架结构,分别说明如下。

框 架 结 构	说 明
Single Document with DOC List	生成基于 SDI (单文档) 和 CCeDocList 项目
Single Document	生成基于 SDI (单文档) 项目
Dialog Base	生成基于对话框项目

然后,读者需要为应用程序选择一个语言资源

在此,读者单击 Single Document with DOC List 选项,建立一个单文档的应用程序。然后单击 Next,进入 AppWizard 的第二步。

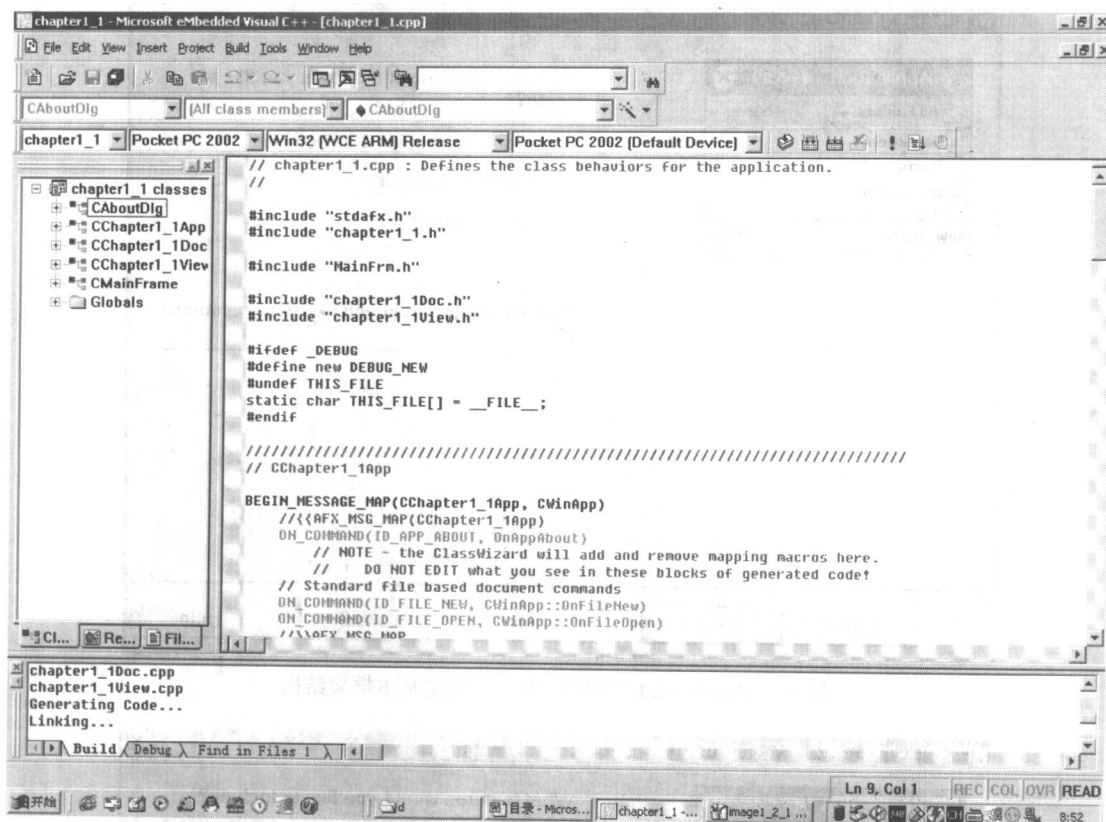


图 1-2 EVC 开发环境

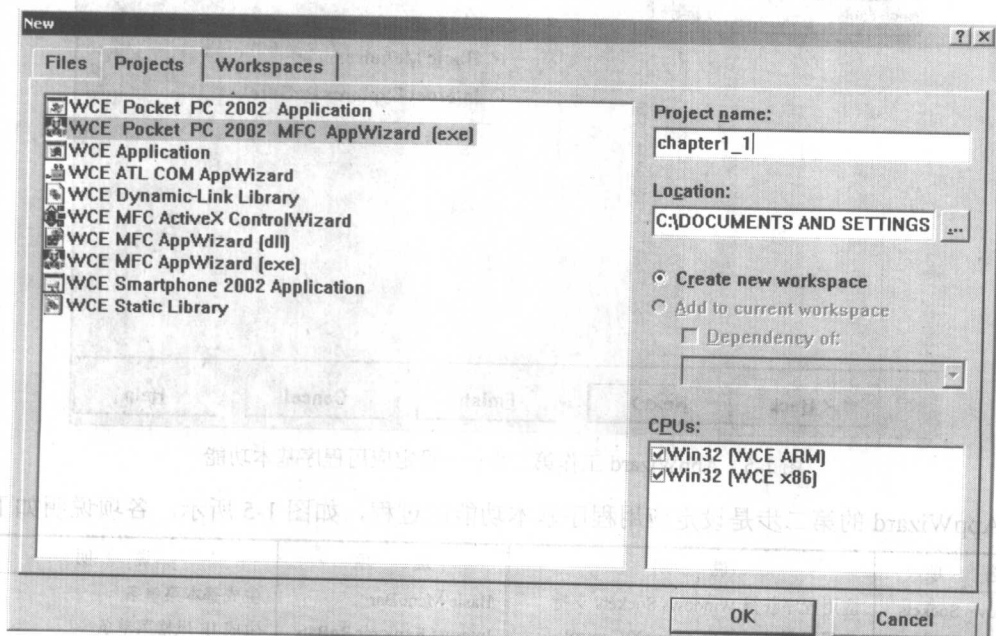


图 1-3 生成项目

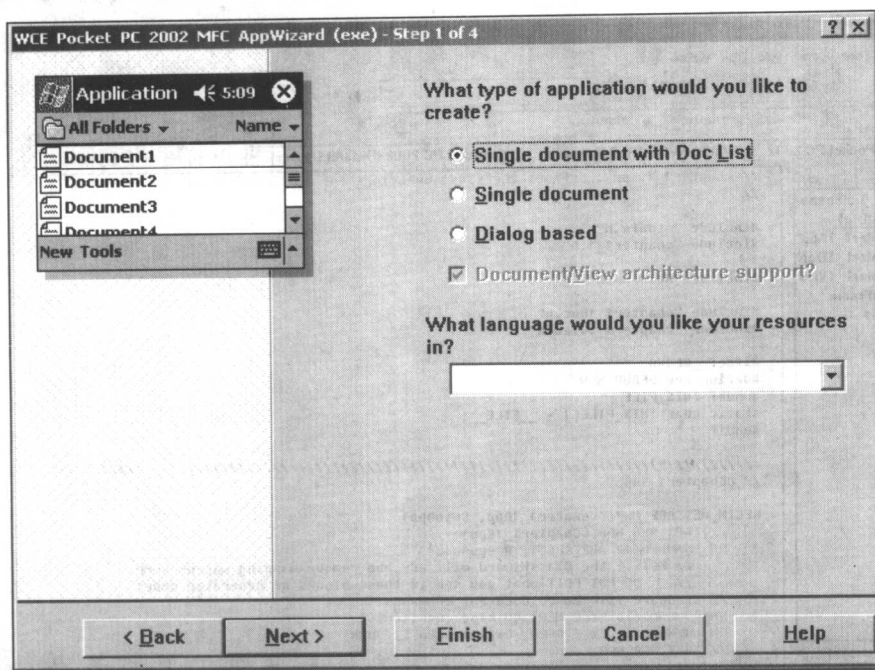


图 1-4 AppWizard 工作第一步——确定基本框架结构

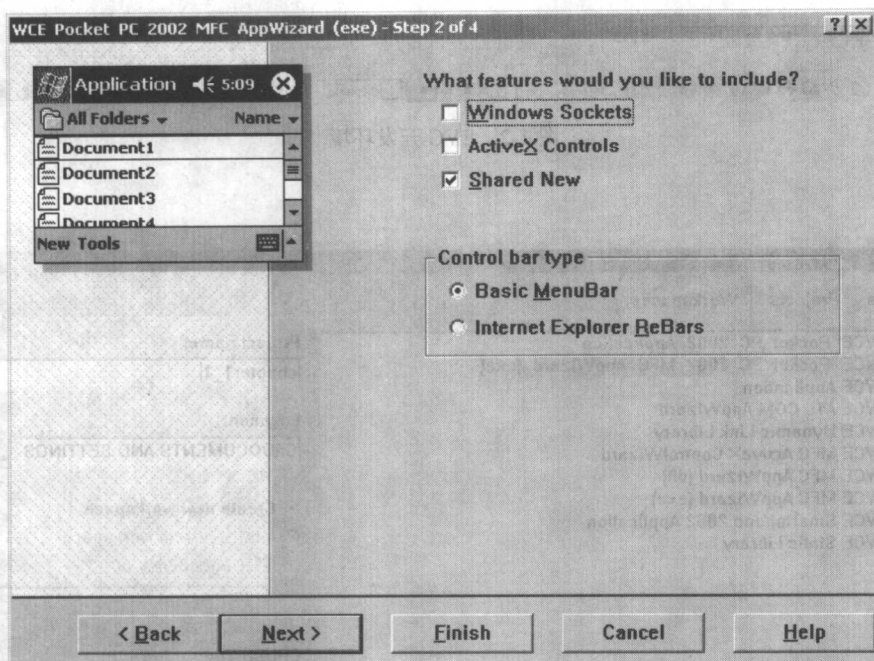


图 1-5 AppWizard 工作第二步——设定应用程序基本功能

AppWizard 的第二步是设定应用程序基本功能的过程，如图 1-5 所示。各项说明如下。

选 项	说 明	选 项	说 明
Windows Sockets	应用程序获得 Windows Sockets 支持	Basic MenuBar	生成基本菜单条
ActiveX Controls	应用程序可以使用 ActiveX Controls	Internet Explorer ReBars	生成 IE 风格菜单条
Shared New	在应用程序的菜单栏生成 NEW 菜单项		

按下 Next 键, 进入 AppWizard 的第三步。

AppWizard 的第三步继续设定应用程序的基本功能, 如图 1-6 所示。

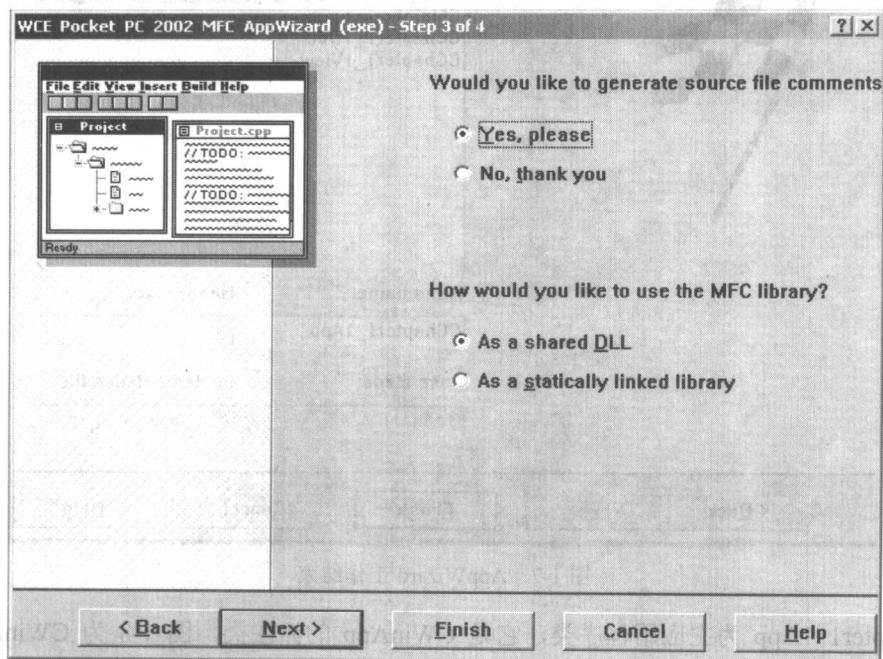


图 1-6 AppWizard 工作第三步——继续设定应用程序基本功能

在这里为用户提供了 2 个可选择设置。第一个设置询问用户是否让系统自动在程序框架文件中增加注释。一般建议用户使用程序注释, 因为这样便于修改程序代码。另一个设置是否使用 MFC 共享动态链接库 DLL, 默认设置为使用共享动态链接库。使用 MFC 动态链接库可以使可执行程序变得很小, 但是如果需要把程序复制到其他掌上设备时, 就需要同时复制一些必要的 MFC DLL 库。

按下 Next 键, 进入 AppWizard 的最后一步操作。

在最后一步中, 读者可以知道 AppWizard 自动生成了哪些文件及其类名, 如图 1-7 所示。各文件的说明详见下一节。这一步中, 读者可以对类名进行修改 (但建议不要修改类名, 这样做会降低程序的可读性)。应该注意的是, 只有文本栏为高亮度方式显示时才可以修改, 否则为不允许。读者还可以根据程序开发要求, 修改已有类的基类。例如, 在选中程序框架中的视类时, 视类的 BASE CLASS 一栏是可选的, 我们可以把它修改为 CFormView、CListView、CTreeView 等不同的基类, 在以后的章节中会对此进行详细讨论。在此我们先不做任何修改, 点击 Finish 按钮, 完成向导设计过程。此时, AppWizard 已经在指定的路径下, 自动生成了应用程序的基本源代码, 并在窗口列出。我们可以进入下一步对程序进行编译、链接和运行了。

1.3.2 类及文件说明

(1) 类说明

在使用了 AppWizard 向导设置后, 读者已经知道总共生成了 4 个类, 其名称是: Cchapter1_1App, Cchapter1_1Doc, Cchapter1_1View 以及 CMainFrame。

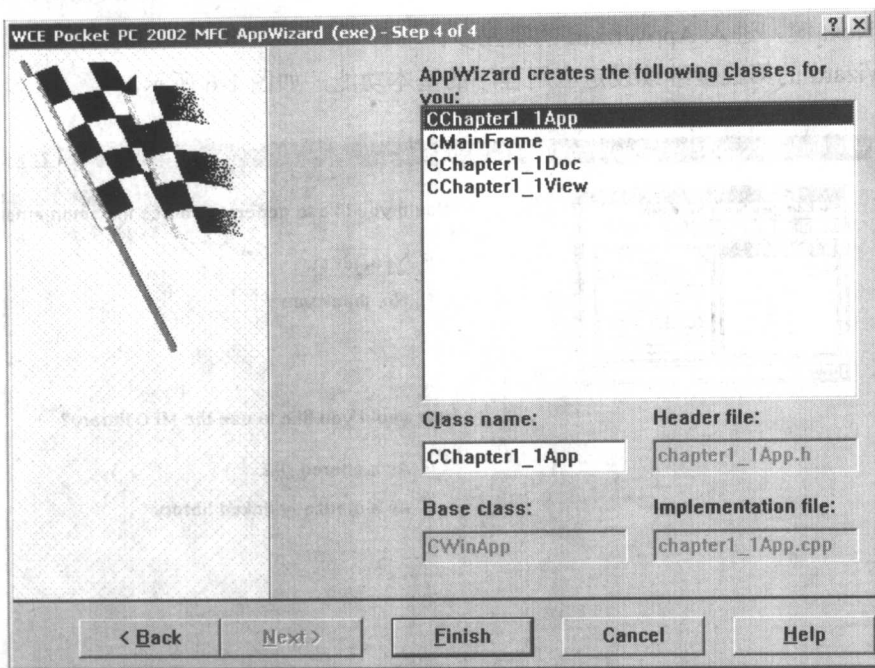


图 1-7 AppWizard 工作结束

Cchapter1_1App 为主应用程序类，它是 CWinApp 的派生类，图 1-8 为 CWinApp 类在 MFC 类层次结构中的位置。

Cchapter1_1Doc 是 CDocument 的派生类，用来储存应用程序数据，并负责处理文件存取。图 1-9 是 CDocument 类在 MFC 类层次结构中的位置。

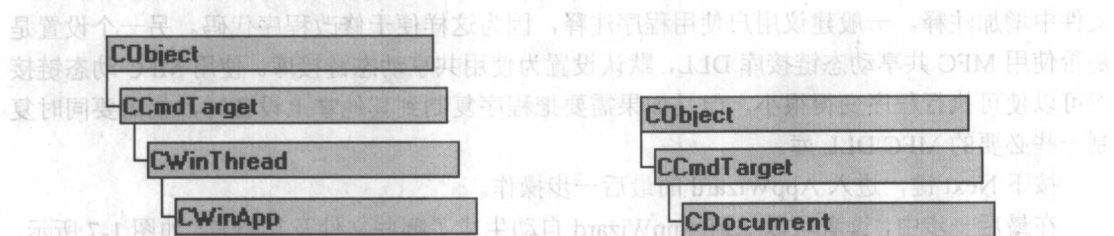


图 1-8 CWinApp 类的位置

图 1-9 CDocument 的位置

Cchapter1_1View 为 CView 的派生类，负责将数据显示到窗口的客户区，图 1-10 是 CView 在 MFC 类层次结构中的位置。

CMainFrame 主要处理窗口框架各项消息，对于单文档结构的项目，它派生于 CFrameWnd 类。图 1-11 是 CMainFrame 在 MFC 类层次结构中的位置。

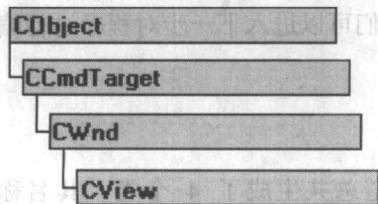


图 1-10 CView 的位置

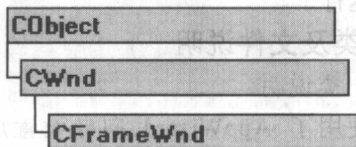


图 1-11 CMainFrame 的位置

(2) 文件说明

AppWizard 生成的类是以类文件的形式存在的。此外, AppWizard 还同时生成了资源文件, 说明文件等, 以下将详细说明之。

chapter1_1.h 应用程序头文件, 应用程序利用此文件映入项目程序所需要的其他头文件, 此文件中包含了 Cchapter1_1App 类的说明。

chapter1_1.cpp 应用程序类 Cchapter1_1App 的类实现文件。

chapter1_1.rc 应用程序资源文件, 可在 RES 子目录下找到所有的资源文件, 并可以在集成开发环境中修改资源。

Res\chapter1_1.ico 应用程序的图标文件, 包含在 chapter1_1.rc 中。

Res\chapter1_1.rc2 应用程序资源描述文件, 不能在集成开发环境中编辑。

chapter1_1.clw 记录 ClassWizard 编辑现有类或增加类时所需要的信息。

MainFrm.h、MainFrm.cpp 主框架窗口类 CmainFrame 的头文件和类实现文件。

Res\Toolbar.bmp 主框架窗口中工具栏所需的位图资源文件。

chapter1_1Doc.h、chapter1_1Doc.cpp 文件类 Cchapter1_1Doc 的头文件和类实现文件。

chapter1_1View.h、chapter1_1View.cpp 视图类 Cchapter1_1View 的头文件和类实现文件。

StdAfx.h、StdAfx.cpp 预编译头文件和实现文件。用来建立预编译文件头(chapter1_1.pch)和预编译文件类型(StdAfx.obj)

1.3.3 编译和运行 Hello Windows CE 程序

在跟着 AppWizard 完成程序预设计后, 我们还要为应用程序选择合适的目标设备和目标设备相应的 CPU。如图 1-12 所示, 在此, 我们在 WCE configuration 中选择 Pocket PC 2002 Emulation 为目标设备, 选 Win 32 (WCE x86) Release 为当前活动配置。

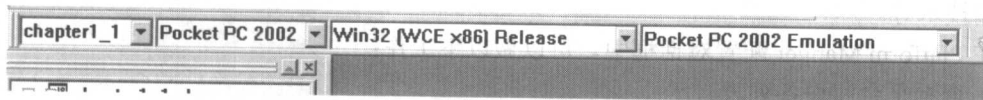


图 1-12 选择目标设备

接下来我们展开集成开发环境中 WORKSPACE 下的类目录树, 并移动到 CChapter1_1View 类的 OnDraw 函数上, 双击, 在代码编辑区打开 OnDraw 函数, 然后修改 OnDraw 函数为如下内容:

```
void CChapter1_1View::OnDraw(CDC* pDC)
{
    CChapter1_1Doc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    CString str;
    str=L"Hello Windows CE!!!";
    CRect rect(10,100,200,200);
    pDC->DrawText(str,rect,DT_CENTER|DT_NOCLIP);
}
```

然后来选 Build 菜单中的 Compile 菜单栏开始编译程序并生成可执行程序, 在集成开发环境下方的 OUTPUT 窗口可以看到编译过程信息。

1.3.4 辅助开发工具

EVC 提供远程操控工具, 可以通过桌面计算机对基于 Windows CE 目标设备实施多种控