



北大宏博

# Pro/ENGINEER

Pro/ENGINEER 野火版系列丛书

野火版

## 二次开发 实用教程

张继春 编著



北京大学出版社  
<http://cbs.pku.edu.cn>



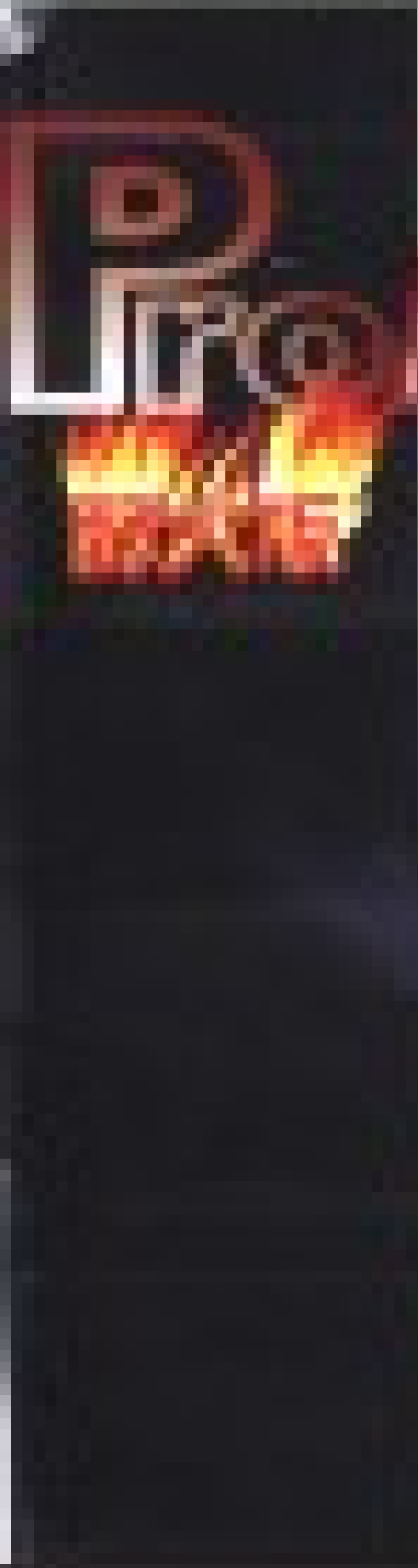
THE UNIVERSITY OF CHINA PRESS  
**ENGINEER**

# 二次开发 实用教程

——



清华大学出版社  
Tsinghua University Press



Pro/ENGINEER 野火版系列丛书

# Pro/ENGINEER 二次开发 实用教程

张继春 编著

北京大学出版社

· 北 京 ·

## 内 容 简 介

本书系统地介绍了 Pro/ENGINEER 软件的二次开发过程,全书共分 7 章,第 1 章介绍基于 Pro/TOOLKIT 二次开发的基础知识;第 2~4 章分别介绍二次开发程序中 Pro/ENGINEER 信息、菜单、对话框的实现;第 5 章介绍参数化设计程序实例;第 6 章介绍 VC 界面在 Pro/TOOLKIT 程序中的使用;第 7 章介绍利用 VB 进行 Pro/ENGINEER 二次开发的基本思路。

本书实例丰富,深入浅出,适合 Pro/ENGINEER 二次开发的初学者。本书可作为广大工程技术人员学习 Pro/ENGINEER 二次开发使用,也可作为教材供大专院校的高年级工科本专科学生、研究生学习使用,同时也可以供 CAD/CAM 高级用户和 CAD/CAM 软件开发人员作为参考书使用。

### 图书在版编目(CIP)数据

Pro/ENGINEER 二次开发实用教程/张继春编著.—北京:北京大学出版社,2003.6

ISBN 7-301-06373-3

I. P... II. 张... III. 机械设计:计算机辅助设计—应用软件, Pro/ENGINEER IV. TH122

中国版本图书馆 CIP 数据核字(2003)第 045896 号

书 名: Pro/ENGINEER 二次开发实用教程

著作责任者:张继春 编著

责任编辑:王方明

标准书号:ISBN 7-301-06373-3/TP·0716

出 版 者:北京大学出版社

地 址:北京市海淀区中关村北京大学校内 100871

网 址:<http://cbs.pku.edu.cn> <http://www.macrowin.net>

电 话:发行部 62750672 62765127 编辑室 62765126 邮购部 62752015

电子信箱:macrowin@macrowin.net

排 版 者:北京东方人华北大彩印中心 电话:62754190

印 刷 者:河北省滦县滦兴书刊印刷厂

发 行 者:北京大学出版社

经 销 者:新华书店

787 毫米×1092 毫米 16 开本 24.5 印张 588 千字

2003 年 7 月第 1 版 2003 年 7 月第 1 次印刷

定 价:48.00 元(含光盘)

# 作者自序

经历了近半年的艰辛，这本书终于要与读者见面了。回头看一下这几个月所走的路，我不得不谈一下这本书的产生过程。

接触 Pro/ENGINEER 已经有四年的时间了，这四年也可以说是 Pro/ENGINEER 由当初的初入中国到现在的极其广泛应用的发展过程。在这过程中，Pro/ENGINEER 的版本也从作者最初接触的 20 版升级到现在的野火版。Pro/ENGINEER 软件提出的单一数据库、参数化、基于特征、全相关的概念改变了机械 CAD/CAE/CAM 的传统观念，这种全新的概念已成为当今世界机械 CAD/CAE/CAM 领域的新标准。Pro/ENGINEER 的通用性非常强，其在中国的应用范围遍布汽车、机械、电子、模具等诸多行业，可以说 Pro/ENGINEER 是国外 CAD/CAE/CAM 软件系统在中国应用最广泛的软件之一。

然而就是由于 Pro/ENGINEER 软件的这种广而博的通用性，使它在具体应用时不能直接处理特定的产品，再加上国外设计观念、标准和国内的差异，在使用该软件进行具体产品的设计时会感到有些不是特别方便。基于这一点，为了使 Pro/ENGINEER 软件能够在特定的企业单位的特定产品的设计中最大限度地发挥潜力、创造效益，需要对 Pro/ENGINEER 进行二次开发。换一句话说，二次开发就是把 Pro/ENGINEER 进一步充实、完善、改进成为用户实用的（针对用户的）CAD/CAE/CAM 应用系统。

国内许多高校和研究所对 Pro/ENGINEER 的二次开发进行了大量的研究工作，这其中以高校所作的研究最多。许多青年工程师、教师、研究生以及 Pro/ENGINEER 的爱好者们都在对 Pro/ENGINEER 的二次开发工作进行苦苦的摸索，这其中也包括作者本人和作者的师兄师弟们。之所以称之为“摸索”，是因为国内没有任何一个人对 Pro/ENGINEER 的二次开发进行系统地、清晰地介绍过，而像作者这样的爱好者只能自己去一点点摸索、一点点咀嚼消化。而进行 Pro/ENGINEER 二次开发工作的主要工具是软件本身的帮助文档和国内外在此方面工作的一些学术论文。可想而知，这其中的重复性研究工作是多么巨大。包括到作者完成本书为止，还有多少科研工作者们在被 Pro/ENGINEER 二次开发的入门而搞得一头雾水？

然而 Pro/ENGINEER 二次开发并不复杂，其开发思路非常清晰，而且程序结构非常简单。只是在具体的产品设计时，随着产品的复杂程度和设计要求的不同二次开发的工作量会差别很大。可是只要用户能够清晰地建立二次开发程序框架，其他的工作量只是对具体细节的增删操作。

因此，为了对 Pro/ENGINEER 的二次开发过程进行清晰的介绍，帮助读者理清二次开发思路，作者萌生了编写本书的念头。正是由于作者在过去的几年中对 Pro/ENGINEER 的二次开发所进行的苦苦摸索，才使作者认为这样做十分有必要。

于是在北大宏博的支持下，作者开始了本书的编写工作。在编写过程中作者本着一切以实用为出发点，切实做到简单易用，所以本书中给出了大量的二次开发程序实例。Pro/ENGINEER 二次开发的帮助文档（《Pro/TOOLKIT User's Guide》）中虽然也给出了大量的实例，但往往只是针对具体函数的程序语句段，而不是一个个完整的程序实例，所以读者参考起来有许多的不便之处。

因此，本书中作者所举的程序实例绝大多数都是完整的二次开发程序，这有助于读者清晰地掌握编程方法、思路。

Pro/TOOLKIT 是 Pro/ENGINEER 二次开发的主要工具。本着通用性、实用性的出发点，本书着重介绍了基于 Pro/TOOLKIT 的 Pro/ENGINEER 二次开发的一般性、基础性知识。这些一般性、基础性的知识是所有 Pro/ENGINEER 二次开发程序都必须进行的工作。这些工作包括 Pro/ENGINEER 二次开发程序框架的建立，Pro/ENGINEER 信息、菜单、对话框的编程实现等。所以这些内容是本书的重点，为了进一步巩固这些知识，作者列出了具体的开发实例加以说明。而对于 Pro/ENGINEER 二次开发的其他具体的模块，本书没有过多地介绍，这需要读者们在具体的开发过程中进一步深入研究。

为了进一步拓宽读者的思路，本书的最后一章给出了利用 VB 二次开发 Pro/ENGINEER 的一种方法。这种方法比较简单，作者通过简单的几个程序实例来说明这一种方法，目的是使读者对这一过程获得十分清晰的认识。

总之，作者认为本书是 Pro/ENGINEER 爱好者学习 Pro/ENGINEER 二次开发的一本非常实用的教材，因为这正是作者写此书的出发点。而且我也相信，只要读者能够认真阅读本书，并认真按照程序的实例进行练习，就一定会觉得 Pro/ENGINEER 二次开发是一件非常简单而且快乐的事情，而且一定能成为真正的 Pro/ENGINEER 二次开发工作者。

张继春

2003 年 5 月

# 前 言

Pro/ENGINEER 软件系统是美国参数化技术公司 PTC(Parametric Technology Corporation)的优秀产品, 提供了集成产品的三维模型设计、加工、分析及绘图等功能的完整的 CAD/CAE/CAM 解决方案。该软件以使用方便、参数化造型和系统的全相关性而著称。目前 Pro/ENGINEER 软件在我国的机械、电子、家电、塑料模具等行业取得了广泛的应用, 该软件在国内的应用数量大大超过了同类型的其它国外产品。

虽然 Pro/ENGINEER 软件功能非常强大, 通用性非常好, 然而在具体的使用过程中不可能满足各种设计要求, 特别是国外的 CAD/CAE/CAM 系统在设计标准、规范及标准件库等方面和国内存在较大的差异, 因此, 以 Pro/ENGINEER 软件为平台进行二次开发, 使之适合国内及企业设计的要求, 更大限度地发挥 Pro/ENGINEER 的作用, 已成为该软件应用过程中的一项重要工作。

Pro/TOOLKIT 是 Pro/ENGINEER 软件系统提供的用户化工具箱, 也称为用户接口(API), 利用 Pro/TOOLKIT, 可以用 C 语言进行二次开发工作。本书主要介绍如何利用 Pro/TOOLKIT 进行 Pro/ENGINEER 二次开发工作。为拓宽读者的思路, 本书的最后一章简单地介绍了如何利用 VB 进行 Pro/ENGINEER 的二次开发工作。

在本书创作的过程中, 本着简单实用的出发点, 列举出大量的二次开发程序实例, 这些实例绝大多数比较简单, 目的是使读者通过这些实例对基于 Pro/TOOLKIT 的二次开发过程获得非常清醒的认识。

本书共分 7 章, 第 1 章介绍基于 Pro/TOOLKIT 二次开发的基础知识; 第 2 章介绍 Pro/ENGINEER 交互信息的实现; 第 3 章介绍如何进行 Pro/ENGINEER 菜单编程; 第 4 章介绍了基于 Pro/TOOLKIT 的对话框程序的开发; 第 5 章介绍了参数化设计系统程序实例; 第六章介绍了如何在 Pro/TOOLKIT 程序中使用 VC 编制的界面; 第七章介绍了利用 VB 进行 Pro/ENGINEER 的二次开发的基本过程。

本书由北航汽车工程系徐斌教授主审, 并且在本书的编写过程中, 得到了哈工大汽车工程学院崔胜民院长、杨建国教授、林波教授、王彦岩等的大力支持与帮助, 在此表示最真诚的感谢。

本书在编写过程中, 参考了大量的文献, 在此对所参考文献的作者表示衷心的感谢。另外在本书的编写过程中, 得到了刘晓锋、王永光等的大力帮助, 在此一并表示感谢。

由于作者水平有限, 疏漏和错误之处在所难免, 恳请广大读者批评指正。

阅读本书对读者的要求如下:

要求学习者至少有 40 小时的 Pro/ENGINEER 学习时间, 对 Pro/ENGINEER 基础操作有较好的理解; 能够运用 Pro/ENGINEER 建立和修改简单的零件和组合件, 能够创建特征陈列, 会定义和使用用户定义特征(UDFS)。

要求使用者能够进行 C 语言编程, 要求熟练掌握下列内容: 指针, 指向指针的指针, 数组, for 循环, while 循环, switch 语句, 字符串和字符串函数, 结构体, 函数。

作者

2003 年 5 月

# 目 录

第 1 章 基于 Pro/TOOLKIT 二次开发基础知识.....	1
1.1 基于 Pro/TOOLKIT 二次开发基本过程.....	1
1.1.1 编写源文件.....	1
1.1.2 程序的编译和连接.....	2
1.1.3 应用程序的注册.....	5
1.1.4 运行应用程序.....	5
1.2 Pro/TOOLKIT 的安装与测试.....	6
1.2.1 Pro/TOOLKIT 的安装.....	6
1.2.2 测试 Pro/TOOLKIT.....	8
1.3 基于 Pro/TOOLKIT 的二次开发.....	12
1.3.1 Pro/TOOLKIT 的工作模式.....	12
1.3.2 Pro/TOOLKIT 应用程序的开发.....	13
1.4 Makefile 文件.....	19
1.5 注册文件.....	23
1.5.1 注册文件的语法格式.....	23
1.5.2 注册文件实例.....	24
1.5.3 注册文件自动注册搜索顺序.....	26
1.6 一个简单的程序.....	26
1.7 Pro/TOOLKIT 相关概念术语.....	30
1.7.1 Pro/TOOLKIT 对象句柄.....	30
1.7.2 Pro/TOOLKIT 对象.....	31
1.7.3 Pro/TOOLKIT 库函数.....	32
1.7.4 选择对象.....	33
1.7.5 访问函数.....	34
1.7.6 同步模式和异步模式.....	37
1.8 宽字符.....	38
1.8.1 宽字符的定义.....	38
1.8.2 设置硬件类型.....	38
1.8.3 检查宽字符 wchar_t 的定义.....	38
1.8.4 宽字符函数.....	39
1.9 可扩展数组.....	40
1.9.1 可扩展数组的含义.....	40
1.9.2 可扩展数组的相关函数.....	40
1.10 用户帮助向导的使用.....	44

11555/05



1.10.1	用户帮助向导的内容 .....	44
1.10.2	用户帮助向导的安装与运行 .....	44
1.10.3	使用 Pro/TOOLKIT 用户帮助向导 .....	45
<b>第 2 章</b>	<b>信息 .....</b>	<b>49</b>
2.1	信息栏信息的分类 .....	49
2.2	信息栏信息文本 .....	50
2.2.1	信息文本文件的命名 .....	50
2.2.2	信息文本文件的内容 .....	50
2.3	信息栏信息的显示和滚动 .....	52
2.3.1	函数 printf() 的用法 .....	53
2.3.2	函数 ProMessageDisplay() 的用法 .....	53
2.3.3	函数 ProMessageClear() 的用法 .....	54
2.3.4	程序举例 .....	54
2.4	键盘输入的实现 .....	59
2.4.1	ProMessageDoubleRead() 函数和 ProMessageIntegerRead() 函数 .....	59
2.4.2	ProMessagePasswordRead() 和 ProMessageStringRead() 函数 .....	60
2.4.3	程序实例 .....	61
2.5	弹出式信息对话框 .....	65
2.5.1	弹出式信息对话框函数 .....	65
2.5.2	弹出式信息对话框程序实例 .....	66
<b>第 3 章</b>	<b>菜单 .....</b>	<b>73</b>
3.1	菜单概述 .....	73
3.1.1	菜单栏菜单 .....	74
3.1.2	菜单管理器菜单 .....	74
3.2	菜单栏菜单 .....	75
3.2.1	菜单栏菜单创建的一般过程 .....	75
3.2.2	在菜单栏菜单中添加命令按钮 .....	82
3.2.3	在菜单栏菜单中添加复选按钮和单选按钮 .....	91
3.2.4	添加菜单 .....	100
3.2.5	在工具条上添加命令图标 .....	106
3.3	菜单管理器菜单 .....	111
3.3.1	菜单管理器创建的一般过程 .....	111
3.3.2	编写菜单资源文件 .....	113
3.3.3	将菜单资源文件调入内存 .....	115
3.3.4	为菜单按钮设置激活函数 .....	116
3.3.5	定义菜单关闭函数 .....	119
3.3.6	创建、显示并激活菜单 .....	119

---

3.3.7 关于菜单管理器菜单的其他操作 .....	120
3.3.8 数据菜单 .....	126
3.3.9 设置菜单按钮的选中状态和可访问状态 .....	128
<b>第 4 章 用户对话框</b> .....	<b>131</b>
4.1 用户对话框创建的基本过程 .....	131
4.1.1 创建对话框资源文件 .....	131
4.1.2 编写 C 程序 .....	132
4.1.3 编译程序 .....	134
4.1.4 编写注册文件 .....	134
4.1.5 注册、运行应用程序 .....	134
4.2 对话框资源文件 .....	135
4.2.1 Pro/TOOLKIT 对话框控件种类 .....	135
4.2.2 对话框资源文件语法格式 .....	136
4.2.3 对话框控件公共属性 .....	151
4.2.4 预览对话框 .....	153
4.3 Pro/TOOLKIT 对话框编程 .....	154
4.3.1 对话框属性 .....	155
4.3.2 对话框函数 .....	155
4.3.3 程序实例 .....	158
4.4 命令按钮控件 .....	163
4.4.1 命令按钮控件属性 .....	164
4.4.2 对命令按钮控件的基本操作 .....	164
4.5 编辑框控件 .....	166
4.5.1 编辑框控件的属性 .....	167
4.5.2 对编辑框控件的操作 .....	167
4.6 复选按钮控件 .....	171
4.6.1 复选按钮属性 .....	171
4.6.2 对复选按钮的操作 .....	171
4.7 单选按钮控件 .....	173
4.7.1 单选按钮属性 .....	173
4.7.2 对单选按钮控件的操作 .....	174
4.8 列表框控件 .....	177
4.8.1 列表框属性 .....	177
4.8.2 对列表框控件的操作 .....	178
4.9 组合框控件 .....	183
4.9.1 组合框属性 .....	183
4.9.2 对组合框控件的操作 .....	183
4.10 标签控件与布局控件 .....	188

4.10.1	标签控件.....	188
4.10.2	布局控件.....	189
4.11	控件程序实例(一).....	189
4.11.1	程序实例一.....	189
4.11.2	程序实例二.....	195
4.11.3	程序实例三.....	201
4.12	进度条控件.....	209
4.12.1	进度条属性.....	210
4.12.2	对进度条控件的操作.....	210
4.13	滑块控件.....	211
4.13.1	滑块控件的属性.....	211
4.13.2	对滑块控件的操作.....	212
4.14	文本显示控件.....	214
4.14.1	文本显示控件属性.....	214
4.14.2	对文本显示控件的操作.....	214
4.15	微调旋钮控件.....	217
4.15.1	微调旋钮控件属性.....	217
4.15.2	对微调旋钮控件的操作.....	218
4.16	切换键控件和分隔符控件.....	220
4.16.1	切换键控件.....	220
4.16.2	分隔符控件.....	222
4.17	对话框菜单控件.....	223
4.17.1	级联按钮控件.....	223
4.17.2	菜单栏控件.....	223
4.17.3	菜单控件.....	225
4.18	表单控件.....	226
4.18.1	表单控件属性.....	226
4.18.2	对表单控件的操作.....	227
4.19	控件程序实例(二).....	239
4.19.1	程序实例一: 文本显示控件的使用.....	239
4.19.2	程序实例二: 表单控件的使用.....	242
<b>第 5 章</b>	<b>应用程序开发实例.....</b>	<b>253</b>
5.1	参数化 CAD/CAM 原理.....	253
5.1.1	参数化 CAD/CAM 简介.....	253
5.1.2	Pro/ENGINEER 系统中自动化建模的实现方法.....	254
5.2	齿轮参数化系统.....	254
5.2.1	创建齿轮参数化实体模型.....	254
5.2.2	编写资源文件.....	255

---

5.2.3 编写程序源文件.....	263
5.2.4 程序注册运行.....	271
5.3 发动机连杆参数化设计系统.....	273
5.3.1 创建发动机连杆参数化实体模型.....	273
5.3.2 编写资源文件.....	273
5.3.3 编写程序源文件.....	295
5.3.4 应用程序注册运行.....	322
<b>第 6 章 VC 界面开发实例.....</b>	<b>326</b>
6.1 调用 VC DLL 程序实例.....	326
6.1.1 基本 DLL 理论.....	326
6.1.2 程序框架.....	326
6.1.3 程序的具体创建过程.....	327
6.2 异步模式程序实例.....	340
<b>第 7 章 利用 VB 进行二次开发.....</b>	<b>349</b>
7.1 Automation GATEWAY 二次开发简介.....	349
7.1.1 什么是 Automation GATEWAY.....	349
7.1.2 Automation GATEWAY 的安装.....	350
7.2 程序实例.....	351
7.2.1 齿轮参数化建模程序实例.....	351
7.2.2 单位设置及质量属性计算程序实例.....	355
7.2.3 特征选择及尺寸显示修改实例.....	361
7.2.4 Excel VBA 程序实例.....	368

# 第 1 章 基于 Pro/TOOLKIT 二次开发基础知识

Pro/TOOLKIT 是 Pro/ENGINEER 系统的客户化开发工具包，即应用程序接口(API)。它提供了大量的 C 语言库函数，能够使外部应用程序(客户应用程序)安全有效地访问 Pro/ENGINEER 的数据库和应用程序。通过 C 语言编程及应用程序与 Pro/ENGINEER 系统的无缝集成，客户和第三方能够在 Pro/ENGINEER 系统中添加所需的功能。

## 1.1 基于 Pro/TOOLKIT 二次开发基本过程

本节里将通过 Hello World 程序使大家对 Pro/TOOLKIT 二次开发基本过程获得一定的感性认识。该程序的作用是在 Pro/ENGINEER 的当前窗口内显示“Hello World”欢迎信息。

### 1.1.1 编写源文件

源文件包括资源文件和程序源文件：资源文件包括菜单资源文件、窗口信息资源文件、对话框资源文件(分别用来完成创建和修改 Pro/ENGINEER 菜单、窗口信息和对话框等功能)等；程序源文件指我们所要编写的 C 语言程序，它是整个 Pro/TOOLKIT 程序开发的核心部分。

“Hello World”程序非常简单，其功能只是在 Pro/ENGINEER 窗口内显示欢迎信息，没有对菜单资源等进行的修改，所以此程序没有资源文件。该程序的程序源代码如下：

```
//头文件包含
#include "ProToolkit.h"
#include "ProGraphics.h"
/*=====*\
函数: user_initialize
功能: 用户初始化函数
\*=====*/
int user_initialize(int argc, char *argv[])
{
//函数声明
    int hello_function();

//调用函数 hello_function
    hello_function();

    return (0);
}
```

```
/*=====*\
函数: user_terminate
功能: 用户结束中断函数

\*=====*/
void user_terminate()
{
    return ;
}

/*=====*\
函数: hello_function()
功能: 显示出“Hello World”欢迎信息

\*=====*/
int hello_function()
{
    wchar_t wtext[64];
    double pt[3];

    pt[0] = 150.0;
    pt[1] = 375.0;

    //设置变量wtext内容为"Hello World"
    ProStringToWstring(wtext, "Hello World");

    //显示出"Hello World"信息
    ProGraphicsTextDisplay(pt, wtext);

    return (0);
}
```

这个程序是非常简单的，不过从中可以看出 Pro/TOOLKIT 程序的基本结构。整个程序由文件包含部分、用户初始化函数 `user_initialize()` 和用户结束中断函数 `user_terminate()` 组成。关于用户初始化函数 `user_initialize()` 和用户结束中断函数 `user_terminate()` 的作用以及具体用法，将会在后面的章节中进一步介绍。

### 1.1.2 程序的编译和连接

为了编译连接所编制的程序代码，一般需要制作 `Makefile` 工程文件，可以根据 Pro/TOOLKIT 自带的 `Makefile` 修改。该文件主要指定库文件、头文件、源文件的位置及要生成的可执行文件和动态连接库名称等。也就是说，此文件是用来说明如何进行应用程序编译和连接的，当然也可以在 VC++6.0 环境中指定上述各种项目。

“Hello World”程序的 `Makefile` 文件 `make_hello` 内容如下：

```
//文件开始
# File Name //定义make文件名称
MAKEFILENAME = make_hello
```

```
# Machine Type//定义机器类型
PRO_MACHINE_TYPE = i486_nt

# Executable names//定义生成的执行文件
EXE = hello.exe
EXE_DLL = hello.dll

# Pro/Toolkit Source & Machine Loadpoint (EXTERNAL USE -
  DEFAULT)//定义#Pro/Toolkit 安装根目录
PTC_DIR = D:\PTC\proeWildfire
PROTOOL_SRC = $(PTC_DIR)/protookit
PROTOOL_SYS = $(PROTOOL_SRC)/$(PRO_MACHINE_TYPE)
PRODEV_SRC = $(PTC_DIR)/prodevelop
PRODEV_SYS = $(PRODEV_SRC)/$(PRO_MACHINE_TYPE)

# Pro/Toolkit Source & Machine Loadpoint (PTC INTERNAL
  USE)
#PROTOOL_SRC = $(PTCSRC)/protookit
#PROTOOL_SYS = $(PTCSYS)
#PRODEV_SRC = $(PTCSRC)/prodevelop
#PRODEV_SYS = $(PTCSYS)

# Include File Paths//定义包含文件所在目录
INCS = -I. -I$(PROTOOL_SRC)/protk_appls/includes
  -I$(PROTOOL_SRC)/includes \
  -I$(PRODEV_SRC)/includes

# Compiler Flags //定义编译特征值
CC = cl
MACH = -DPRO_MACHINE=29 -DPRO_OS=4
CCFLAGS = -Dhypot=_hypot -DMSB_LEFT -Dfar=ptc_far
  -Dhuge=p_huge -Dnear=p_near -c -W2 -G4 -Za -D_X86_=1
  -D_WSTUDIO_DEFINED
CFLAGS = $(CCFLAGS) $(INCS) $(MACH)

# Libraries//定义库文件路径
PTCLIBS = $(PRODEV_SYS)/obj/prodevelop.lib
  $(PROTOOL_SYS)/obj/protookit.lib
PTCLIBS_DLL = $(PRODEV_SYS)/obj/prodev_dll.lib
  $(PROTOOL_SYS)/obj/protk_dll.lib
LIBS = libc.lib kernel32.lib user32.lib wsock32.lib
  advapi32.lib mpr.lib winspool.lib

# Object files//定义目标文件
OBJS = hello.obj

# Linker//编译器
LINK = link

$(EXE) : $(OBJS) $(PTCLIBS)
  @echo Start at 'date'
```

```

@echo PROTOOL_SRC = $(PROTOOL_SRC)
@echo PROTOOL_SYS = $(PROTOOL_SYS)
$(LINK) /subsystem:console -out:$(EXE) /debug:none
/debugtype:coff /machine:i386 @<<longline.list
$(OBJS) $(PTCLIBS) $(LIBS)
<<
@echo Finished at 'date'
del $(OBJS)

# Dll target//定义 dll 文件
dll: $(EXE_DLL)

$(EXE_DLL) : $(OBJS) $(PTCLIBS_DLL)
@echo Start at 'date'
@echo PROTOOL_SRC = $(PROTOOL_SRC)
@echo PROTOOL_SYS = $(PROTOOL_SYS)
$(LINK) /subsystem:console -out:$(EXE_DLL) /debug:none
/debugtype:coff /machine:i386 /dll @<<longline.list
$(OBJS) $(PTCLIBS_DLL) $(LIBS)
<<

@echo Finished at 'date'
del $(OBJS)

# object dependencies//定义目标文件依附关系
PROTK_APPLS = E:\download\protoolkit\hello_word

hello.obj: $(PROTK_APPLS)/hello.c
$(CC) $(CFLAGS) $(PROTK_APPLS)/hello.c

clean :
del $(OBJS)
del $(EXE)
nmake -f $(MAKEFILENAME)

clean_dll :
del $(OBJS)
del $(EXE_DLL)
nmake -f $(MAKEFILENAME) dll
//文件结束

```


关于 **make** 文件的格式及改写，我们会在后面详细介绍，这里只是使大家对此文件获得一定的感性认识。

将此 **make\_hello** 文件改名为 **make\_hello.mak**，打开 VC 6.0 的 File 主菜单，单击 Open Workspace 按钮，在打开的 Open Workspace 对话框中将文件类型改为 Makefiles(.mak)，打开文件 **make\_hello.mak**。VC6.0 就会根据 **make\_hello.mak** 文件生成一个工程文件，编译连接该工程，生成 **hello.dll** 文件。



### 1.1.3 应用程序的注册

要使应用程序能够集成到 Pro/ENGINEER 系统中运行, 必须制作一个扩展名为 .dat 的注册文件, 用该文件进行应用程序的注册。可以采用两种注册方式, 一种为自动注册的方式, 将注册文件放在指定的目录下(如放在 Pro/ENGINEER 的启动目录下), 运行 Pro/ENGINEER 时将根据注册文件自动注册运行所指定的应用程序(此时注册文件必须名为 protk.dat); 另一种为手动注册, 即在运行 Pro/ENGINEER 后, 在 Tools 菜单下选择 Auxiliary Applications 菜单项, 然后在对话框中选取 Register, 指向所要注册的注册文件, 单击 start 按钮便可以运行应用程序了。

 **提示:** 如果采用手动注册方式, 注册文件只需扩展名为 .dat 即可, 命名为 protk.dat 不是必须的。比如, 可以将一个注册文件命名为 menu1\_reg.dat。


“Hello World”程序的注册文件为 protk.dat, 假设放在 Pro/ENGINEER 启动目录下, 此文件内容如下:

```
name hello
startup dll
exec_file d:\work\hello\hello.dll
text_dir text
allow_stop true
revision 24
end
```

关于注册文件的格式, 会在后面的内容中详细介绍。

### 1.1.4 运行应用程序

在含有上述 protk.dat 文件的目录下启动 Pro/ENGINEER, “Hello World”应用程序会自动执行, 执行结果为在 Pro/ENGINEER 图形窗口内显示出“Hello World”欢迎信息, 如图 1.1 所示。

 **提示:** Pro/ENGINEER 系统默认的显示字体是比较小的, 为使“Hello World”信息的显示效果如图 1.1 所示, 必须将 Pro/ENGINEER 的配置文件 config.pro 中 text\_height\_factor 项的值设为 50。

至此, 我们已经完成了第一个 Pro/TOOLKIT 应用程序 Hello World 的全部开发过程, 通过对这个过程的分析, 相信大家对于基于 Pro/TOOLKIT 的基本开发过程已经获得较好的感性认识。