

# 快速入门

---

## Logic Developer - PLC PLC 编程软件

Version 4.00

Dec. 2003

GFK-1918D\_cn



All rights reserved. No part of this publication may be reproduced in any form or by any electronic or mechanical means, including photocopying and recording, without permission in writing from GE Fanuc Automation Americas, Inc..

### **Disclaimer of Warranties and Liability**

The information contained in this manual is believed to be accurate and reliable. However, GE Fanuc Automation Americas, Inc. assumes no responsibilities for any errors, omissions or inaccuracies whatsoever. Without limiting the foregoing, GE Fanuc Automation Americas, Inc. disclaims any and all warranties, expressed or implied, including the warranty of merchantability and fitness for a particular purpose, with respect to the information contained in this manual and the equipment or software described herein. The entire risk as to the quality and performance of such information, equipment and software, is upon the buyer or user. GE Fanuc Automation Americas, Inc. shall not be liable for any damages, including special or consequential damages, arising out of the use of such information, equipment and software, even if GE Fanuc Automation Americas, Inc. has been advised in advance of the possibility of such damages. The use of the information contained in the manual and the software described herein is subject to GE Fanuc Automation Americas, Inc. standard license agreement, which must be executed by the buyer or user before the use of such information, equipment or software.

### **Notice**

GE Fanuc Automation Americas, Inc. reserves the right to make improvements to the products described in this publication at any time and without notice.

© 2003 GE Fanuc Automation Americas, Inc. All rights reserved. CIMPPLICITY is a registered trademark of GE Fanuc Automation. Any other trademarks referenced herein are used solely for purposes of identifying compatibility with the products of GE Fanuc Automation Americas, Inc.

We want to hear from you. If you have any comments, questions, or suggestions about our documentation, send them to the following email address:

[doc@gefanuc.com](mailto:doc@gefanuc.com)

# 目录

---

<b>1 介绍</b>	<b>1</b>
系统需求 .....	3
安装 .....	4
产品注册 .....	5
技术支持 .....	7
<b>2 CIMPLICITY Machine Edition简介</b>	<b>9</b>
快速入门 .....	10
工程 .....	12
工具栏 .....	14
变量 .....	16
选项 .....	19
Machine Edition 帮助 .....	20
在线帮助 .....	20
信息浏览帮助 .....	20
<b>3 GE Fanuc PLC 目标对象</b>	<b>23</b>
增加, 配置和转换对象 .....	24
增加对象 .....	24
配置对象 .....	25
转换对象 .....	26
配置通讯 .....	28
配置 PACSystems 通讯 .....	28
在非 PACSystems PLC 中的通讯配置 .....	30
与 PLC 的交互操作 .....	33
检验对象 .....	33
在线/离线 .....	33
上载/下载 .....	35
运行/停止 .....	36
故障表 .....	37
参考变量察看表 .....	38
报告 .....	40

<b>4 硬件配置(HWC)</b>	<b>43</b>
PACSystems™ RX7i	43
PACSystems™ RX7i 硬件组态	44
冗余 (仅对 Series 90™-70 和 PACSystems™ RX7i )	50
冗余系统组态	52
Series 90™-30: DSM314 运动模块	55
Series 90™-70 远程 I/O 扫描器	57
VersaMax 远程 I/O	59
<b>5 逻辑编辑Logic Editors</b>	<b>63</b>
梯形图LD编辑器	66
LD编辑器离线工作方式	67
文本方式编辑逻辑	70
LD 编辑器的在线工作方式	72
字对字改变	73
下载改变过的程序到PLC对象	74
梯形图功能块	75
语句表编辑器	79
IL 编辑器离线工作方式	80
IL 编辑器在线工作方式	82
写修改到目标PLC	83
IL 语句表指令和功能块	84
C Blocks	86
对C语言块的操作	86
C语言程序	88
用户程序	88
用户程序工作方式	89
<b>6 运动编程 Motion Programming</b>	<b>91</b>
PLC运动编辑器	92
用运动编辑器编程	93
运动命令	94
本地逻辑	95
用本地逻辑编辑器工作	96
本地逻辑变量	97
本地逻辑命令和操作	99

CAM 编辑器 .....	100
用 CAM 编辑器工作 .....	101

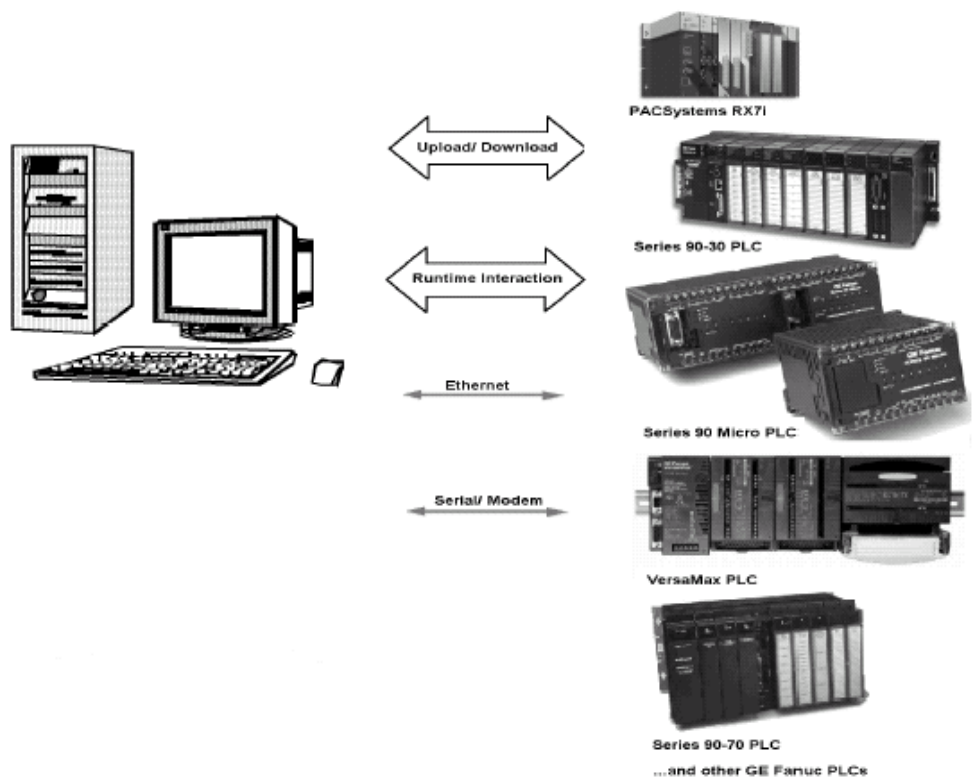


# 1

## 介绍 Logic Developer - PLC

欢迎使用Logic Developer - PLC，它是GE Fanuc PLC的编程工具，属于Machine Edition 系列自动化软件中的一个部分。这个软件包提供了多种工具用于创建功能强大的PLC控制程序。Logic Developer - PLC可以用于组态你的PLC控制器或远程I/O站，创建和修改PLC逻辑，上载和下载PLC工程，监视和调试正在执行的控制程序，工程文件也可以从Logicmaster, VersaPro, 和Cimplicity Control 的文件夹中导入。

### CIMPLICITY Logic Developer - PLC 环境



Logic Developer - PLC 可以在个人电脑上进行控制程序的开发，并且通过以太网或串行口/调制解调器下载到PLC中。

在 Machine Edition 的软件环境中，Logic Developer - PLC 提供了大量的先进的、实用的公共编程工具（见14页）。这些公共编程工具可以被Logic Developer - PC (PC Control), Logic Developer -State, CIMPLICITY View 和 CIMPLICITY 运动控制所共同使用，提供了一个集成的编程环境。Machine Edition 统一的编程环境和共用的开发工具为用户提供了数据共享和网络操作的功能。

Logic Developer - PLC的性能特点：

- Hardware Configuration (HWC)：硬件配置工具，用于组态GE Fanuc PLC 的机架式系统或远程 I/O系统。关于HWC更详细的情况，参见43页。
- LD Editor：一个智能的，基于图形的梯形图开发工具，用户可以自己定义开发界面和运行环境。关于LD Editor更详细的情况，参见63页。
- IL Editor：一个便于使用，自由格式的文本编辑器用于创建语句表控制逻辑，用户可以自己定义格式规则和颜色代码，从而使程序更加易读，关于IL Editor更详细的情况，参见79页。
- C 功能块 and C 程序：独立执行的代码，用C 语言进行编写，在Logic Developer - PLC之外进行开发，然后进行预编译并被工程调用，如果想知道C 功能块更详细的情况，参见86页，关于C 功能块 and C 程序详细的情况，参见88页。
- PLC Motion Editor：一个自由格式的运动控制文本编辑器，允许用户使用自己喜欢的风格进行程序开发，PLC Motion editor专门用于IC693DSM314 运动控制模块的程序开发，关于PLC Motion Editor更详细的情况，参见91页。
- Local Logic Editor：一个基于文本的编辑器，用于开发在 IC693DSM314 运动控制模块中执行的本地逻辑控制程序。本地逻辑控制程序和 PLC CPU 中的运动控制程序是同步执行的，但是独立于PLC CPU。关于Local Logic Editor更详细的情况，参见95页。
- CAM Editor：一个辅助工具用于Logic Developer - PLC Motion editor，特别用于开发IC693DSM314 运动控制。它提供了一个图形化的手段去创建，编辑，管理电子CAM文档，关于CAM Editor更详细的情况，参见100页。

---

## 系统需求:

为了使用 Logic Developer - PLC 和它的工具，需要下列支持:

软件需要:

- Windows® NT version 4.0 with service pack 6.0 或更新  
或  
Windows 2000 Professional  
或  
Windows XP Professional  
或  
Windows ME  
或  
Windows 98 SE。
- Internet Explorer 5.5 with Service Pack 2或更新。（在安装 Machine Edition 之前必须先安装IE5.5 SP2）

硬件需要:

- 500 MHz 基于奔腾的计算机。（建议主频1 GHz以上）
- 128 MB RAM（建议 256 MB）。
- 支持 TCP/IP 网络协议计算机。
- 150-750 MB 硬盘空间。
- 200 MB 硬盘空间用于安装演示工程(可选)。
- 另外需要一定的硬盘空间用于创建工程文件和临时文件。

## 安 装:

如果了解Machine Edition 产品最新的产品信息，发布信息，支持的硬件列表，可以参考ME文件光盘中的重要产品信息 (IPI) 文件。用户有多种方法可以查阅这个文档资料。

- 当安装Machine Edition时，在初始安装界面上选择选择**Important Product Information**。
- 通过Machine Edition 帮助菜单，选择**Important Product Information**。
- 当运行Machine Edition时，在浏览器窗口中点击InfoView标签栏，然后展开Getting Started目录，双击 **Important Product Information** 页。
- 当运行Machine Edition时，在InfoViewer的工具栏中点击Home按钮，然后在InfoViewer的窗口中点击在Get Started目录下面的联接What's New。

如果你的电脑中已经安装了一个老版本的Machine Edition，那么在安装新的版本前之必须先把老版本卸载。你可以通过选择WINDOWS中Machine Edition使用菜单中的“Uninstall CIMPLICITY Machine Edition” 功能来完成。卸载后，所有的工程文件，参数设定，软件注册信息都会保留。

## 安装 Logic Developer - PLC

1. 将 CIMPLICITY Machine Edition 光盘插入CD-ROM驱动器。

通常安装程序会自动启动，如果安装程序没有自动启动，你也可以通过直接运行在CIMPLICITY Machine Edition 光盘根目录下的*Setup.exe* 来启动。

2. 在安装界面中点击 **Install** 开始安装进程。
3. 跟随屏幕上的指令进行。

## 产品注册

在安装完Logic Developer - PLC后，你必须通过Product Authorization菜单进行软件的注册，如果不进行产品注册，那么你仅拥有4天的使用权限。注册的过程仅仅需要几分钟，注册完成后你就可以获得这个产品完全的技术支持。你可以通过电话，传真，电子邮件来进行注册。

### 注册 Machine Edition

1. 提供产品序列号

产品序列号可以在我们提供给用户的产品授权表中找到。

2. 通过 Start > Programs > Product Authorization, 运行Product Authorization程序。

进入产品注册对话框。

3. 点击 **Add**。

4. 选择你想注册的方式: Internet, 电话/传真/电子邮件, 或软盘传送, 然后点击 **Next**。

如果你选择了 Internet 选项, 进入第5步。

如果你选择了电话/传真/电子邮件选项, 进入第5步。

如果你选择了软盘传送选项, 则需保证有一张有注册功能的软盘。

5. 在对话框的信息栏中填写用户信息

带 (\*) 的信息栏必须要填写。

6. 如果通过下列方式注册:

- **Internet**, 点击 **Submit Authorization**, 你会通过电子邮件收到我们给你的注册码。
- **电话**, 点击 **Phone/Fax** 然后拨打屏幕上的电话号码, 根据提示你会获得产品的注册码。
- **传真**, 点击**Phone/Fax**. 在弹出的对话框中, 点击 **Print FAX**, 把产品注册信息打印出来, 然后根据打印信息上的传真号码把信息发送给我们, 我们会通过传真把注册码返回给你。
- **电子邮件**, 点击 **Send E-mail**, 在弹出的对话框中, 点击 **Authorize** 发送电子邮件给我们, 我们会通过电子邮件返回给你产品注册码。

当你把我们返回给你的产品注册码输入**Machine Edition** 中并被接受后, 产品的注册过程就完成了, 如果你购买了**Machine Edition**软件系列中的多种软件, 那么你需要多次运行Product Authorization 程序以完成所有软件的注册。

## 传送注册码到另一台电脑

你只能在拥有注册码的电脑上运行Machine Edition，如果你希望在另外的电脑上开发工程程序，那么你需要通过下列步骤把注册码从一台电脑传送到另外一台电脑。

1. 在目标电脑中安装 Logic Developer - PLC ，然后通过**Start menu > Programs > CIMPLICITY Machine Edition > Product Authorization**。运行 Product Authorization程序。
2. 在弹出的对话框中点击 **Software**。  
在对话框的右边有一串字符为目标电脑代码，仔细的把这一串字符记下来，必须要准确，在接下来的传送注册码的步骤中我们需要用到。
3. 点击 **Add**。  
出现软件注册向导窗口。
4. 点击 **Authorize by disk**。  
在这一步，你需要在有注册码的原电脑上进行操作，把注册码移动到软盘。
5. 在原电脑中运行Product Authorization 程序，点击 **Software**。
6. 点击 **Move**，然后点击 **OK**。输入目标电脑代码，即你在第3步记下来的那一串字符，然后点击 **Next**，经确认无误后点击 **OK**。
7. 查入一张经格式化的空白软盘并点击 **Next**。  
这时注册码将被移入软盘，同时系统会弹出一个对话框告诉你操作成功。
8. 点击 **OK**。
9. 返回目标电脑，这时屏幕显示需查入一张经注册的软盘，然后插入从原电脑取出的软盘。
10. 点击 **Next**。
11. 点击 **Finish**。  
系统会弹出一个对话框告诉你移动注册码已经成功。
12. 点击 **OK**。  
注册码就被成功的移动到目标电脑中。

---

## 技术支持

在成功的进行了产品注册后的90天内，你可以享受到我们免费的技术支持，如果想获得更长时间的技术支持，你可以向当地 GE Fanuc 的分销商购买此种服务。

如果在使用过程中遇到问题，你可以通过产品手册中的信息，在线帮助，有GE Fanuc产品使用经验人员的帮助来解决，如果还是无法解决，那么你也可以通过电话，传真或电子邮件来和GE-FANUC技术支持部联系。

如果通过电话联系，请尽量使用离你计算机最近的那部电话以便我们能和你及时的交流信息。同时如果你能提供以下的产品信息，我们就可以更有效的帮助你解决问题：

- 你所安装软件的序列号，名称，版本号，你可以通过 **Help>About** 弹出的对话框中获得。
- 你的系统中硬件的类型和型号。
- 操作系统的类型和版本号。
- 在出现故障前你所进行的操作步骤。

## 美国

技术支持热线：1-800-GEFANUC (1-800-433-2682) 传真：(780) 420-2049

网址：<http://www.gefanuc.com>

电子邮件：[support@gefanuc.com](mailto:support@gefanuc.com)

产品手册支持和帮助：[doc@gefanuc.com](mailto:doc@gefanuc.com)

公司地址：GE Fanuc

2700 Oxford Tower, 10235 - 101 St.

Edmonton, AB, Canada, T5J 3G1

## 中国

### 上海办事处

电话: 021-3222 4555 传真: 021-6279 3066  
地址: 上海南京西路1468号, 中欣大厦5层 邮编: 200040

### GE Fanuc Automation 北京办事处

电话: 010-6561 1166 传真: 010-6561 1530  
地址: 北京朝阳区光华路7号, 汉威大厦西区6层 邮编: 100004

### GE Fanuc Automation 广州办事处

电话: 020-8363 4828 传真: 020-8363 3128  
地址: 广州建设六马路33号, 宜安广场1512室 邮编: 510060

### GE Fanuc Automation 武汉办事处

电话: 027-8575 8336 传真: 027-8575 8332  
地址: 武汉建设大道566号, 国贸商业中心1018室 邮编: 430022

### GE Fanuc Automation 成都办事处

电话: 028-8619 8199 传真: 028-8619 8369  
地址: 成都西御路街77号, 国信大厦9层东座 邮编: 610015

### GE Fanuc Automation 西安办事处

电话: 021-32224555 传真: 021-62793066  
地址: 西安市南大街30号, 中大国际商务会馆606室 邮编: 700002

### GE Fanuc Automation 沈阳办事处

电话: 024-2281 2468 传真: 024-2281 2121  
地址: 沈阳和平北大街69号, 总统大厦C座907室 邮编: 110003

### GE Fanuc Automation 南京办事处

电话: 025-8471 7330 传真: 025-8472 8654  
地址: 南京汉中路2号, 金陵饭店世贸中心1661室 邮编: 210005



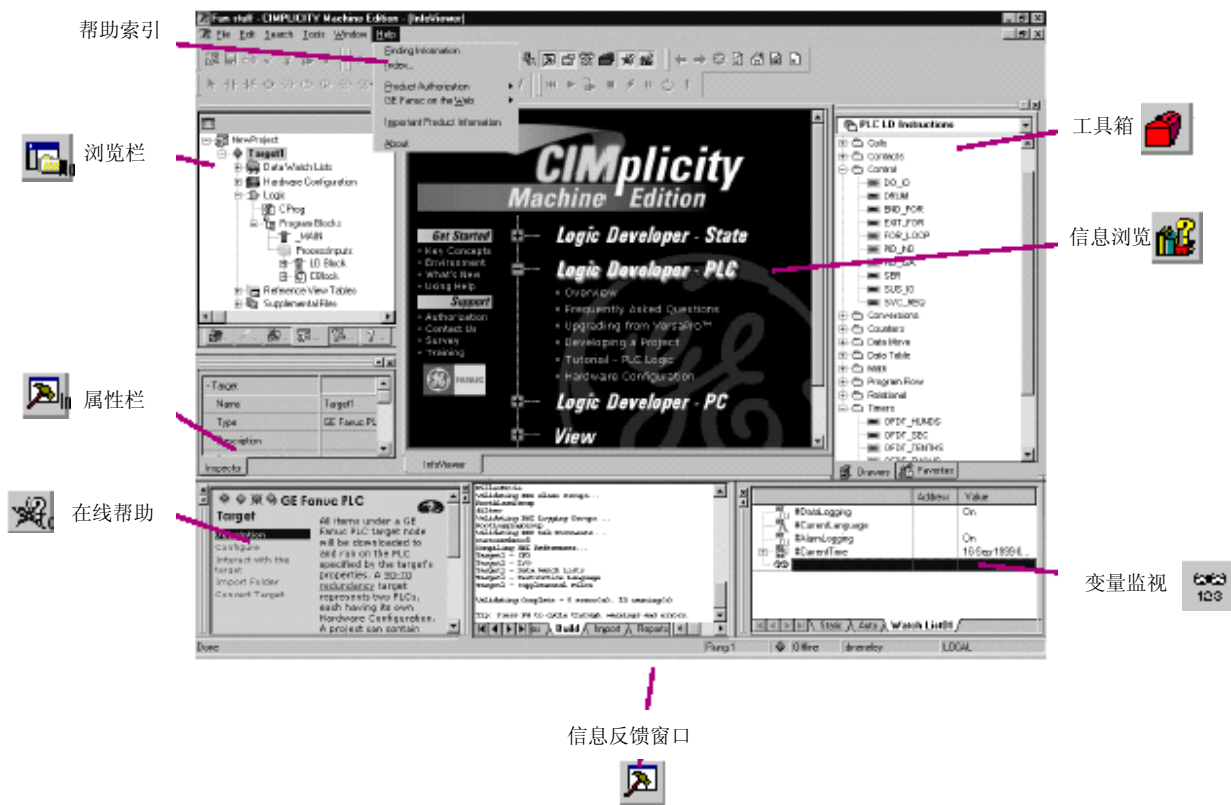
# 2

## CIMPLICITY Machine Edition 简介

Machine Edition 提供了一个统一的完整系统用于解决自动化控制方案。Machine Edition 特点在于提供了一个集成的开发环境，这样可以使用户花更多的时间在应用程序的开发上而不是学习如何使用软件上。所有Machine Edition 系列产品都被有效的集成在一个统一的开发环境。

- 所有软件的开发都使用同一个工具栏。
- 不同软件中的单元具有互相拖放的功能。
- 软件可以针对多种硬件平台进行应用程序开发，用户可以选择程序所运行的硬件平台。
- 开发过程中随着鼠标的点击，可以自动显示相关的帮助信息。


以下是一个Machine Edition工程的开发界面：



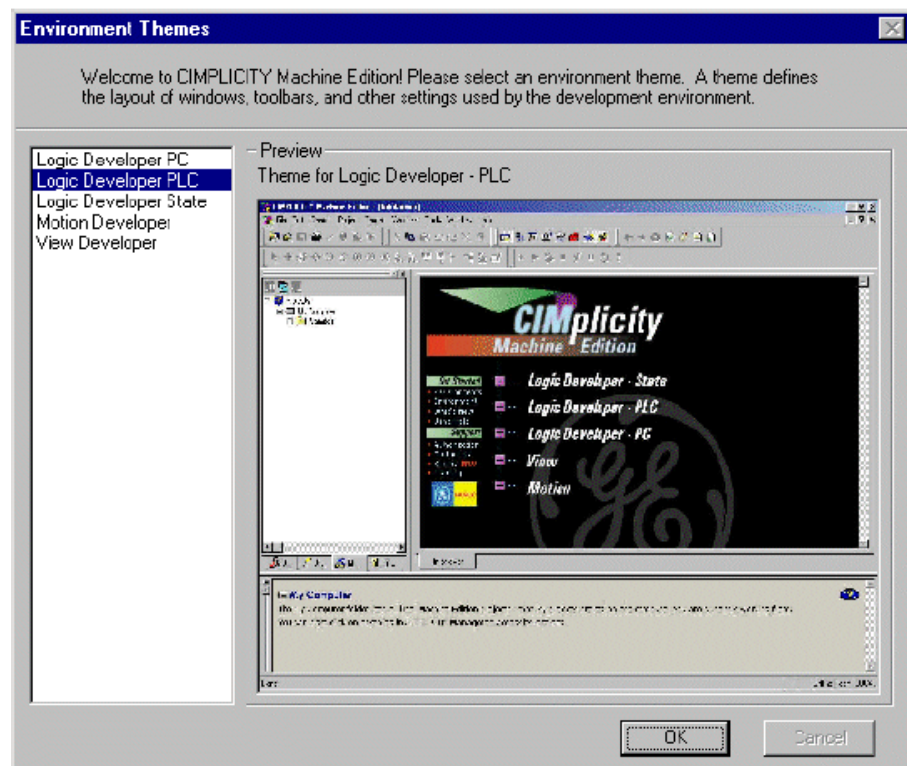
## 快速入门 (QUICK START)

下面我们将学习如何使用Machine Edition 创建PLC工程。

### 启动 Machine Edition

1. 点击  Start > Programs > CIMPLICITY Machine Edition 然后选择 CIMPLICITY Machine Edition。

在 Machine Edition 初始化后，进入开发环境选择窗口。

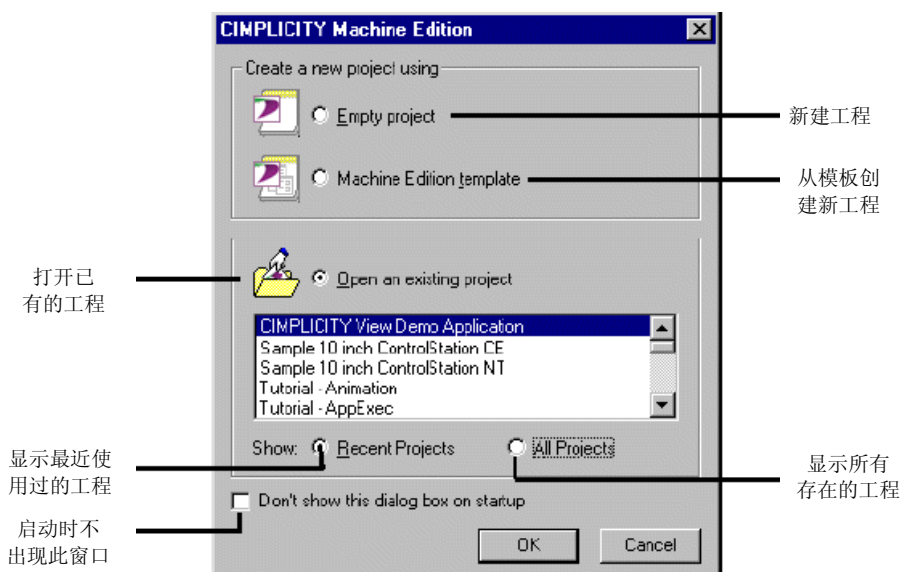


**注：**当你第一次启动 Machine Edition软件时，开发环境选择窗口会自动出现，如果你以后想改变显示界面，你可以通过选择 **Windows > Apply Theme** 菜单进行。

2. 选择 the Logic Developer - PLC 一栏.
3. 点击 **OK**.

当你打开一个工程后进入的窗口界面和在开发环境选择窗口中所预览到的界面是完全一样的。

CIMPLICITY Machine Edition对话框



4. 选择合适的选项来打开一个工程，打开**已有的工程**是系统缺省选择。

**注意：**

- 如果你选择了**新建工程**或者**从模板创建新工程**，你还需要通过新建工程对话框继续创建新工程进程（见12页）。
  - 如果你选择了**打开已有的工程**，你就可以从下面的功能选项中选择：**显示最近使用过的工程**或**显示所有存在的工程**，**最近使用过的工程**为系统缺省选择。
5. 如果你选择了**打开已有的工程**，那么你就可以在下部的列表框中选择你想要打开的工程。已有的工程中还包括了演示工程和教程，这样可以更快的帮助你熟悉Machine Edition。
  6. 如果有必要，可以选择“**启动时不出现此窗口**”选项。
  7. 点击 **OK**。

你的工程就按照你在前面选定的Machine Edition开发环境中打开了。

## 工程 (PROJECTS)

通过 Machine Edition 你可以在一个工程中创建和编辑不同类型的产品对象如：CIMPLICITY View, Logic Developer-PC, CIMPLICITY Motion, Logic Developer - State, 和 Logic Developer - PLC。在同一个工程中，这些对象可以共享 Machine Edition 的工具栏，提供了各个对象之间的更高层次的综合集成。

你也可以把以前用Logicmaster, CimplicityControl, 或 VersaPro创建的工程转换成 Logic Developer - PLC的工程，也可以通过创建多个工程来满足用户的各种要求。

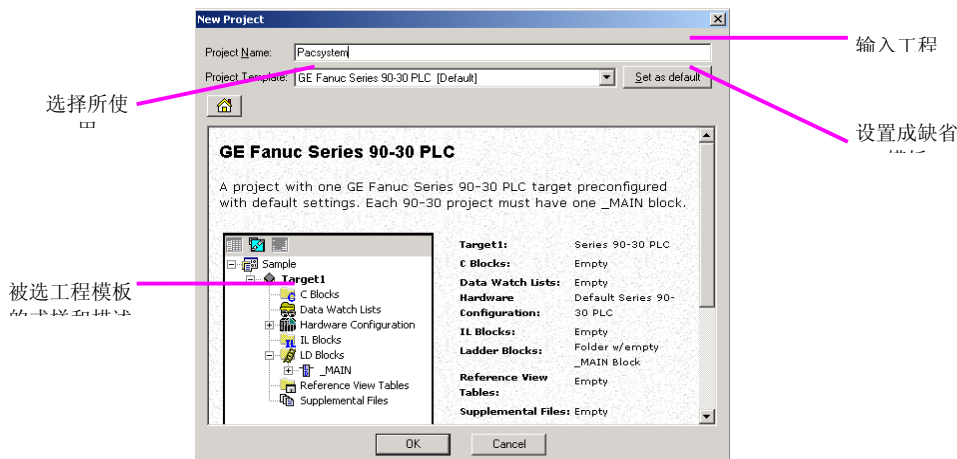
### 通过模板来创建工程

在创建一个工程前，你需要提供如下信息：

- 工程中主要包含那种类型的对象。
- 工程中将要使用的 PLC 类型。

1. 通过 File 菜单，选择 New Project, 或点击File工具栏中  按键。

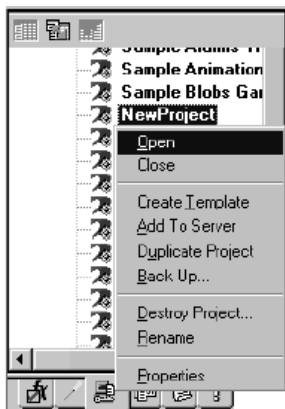
出现新建工程对话框。



2. 选择所需要的模板。
3. 输入工程名。
4. 点击 OK。

这样你的工程就在Machine Edition的环境中被打开了。

## 编辑一个已有的工程



浏览器：管理器标签栏

1. 打开 工程浏览窗口，然后选择 Manager 标签。

窗口中将显示工程列表。

2. 选择你想打开的工程单击右键选择 **Open**。

这样工程就被装入Machine Edition中，并随时可以被编辑。

---

**想了解更多信息？** 可以在帮助索引中寻找“Projects”。

---

## 从其它程序转换到 Machine Edition

1. 打开工程浏览窗口选择 **Project** 标签栏。
2. 选择你想使用的目标。
3. 右键单击这个目标，选择 **Import**，然后选择被转换工程的类型。
4. 在选择文件对话框中双击你想转换的文件。

---

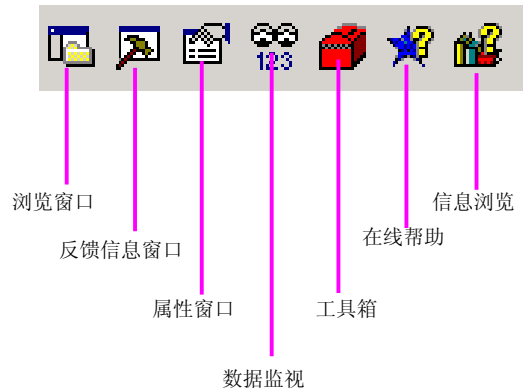
**想了解更多信息？** 可以在帮助索引中寻找“import a Cimplicity Control folder”，“import Logicmaster folder”，或“import a VersaPro folder”。

---

## 工具栏

Machine Edition 中的工具栏支持多种工程开发功能。每一种工具都可以从菜单或工具栏中的按钮打开或关闭。

以下是对每种工具的描述。



### 浏览窗口

浏览窗口是包含了一些标签栏的工具窗口，每一个标签栏都会把应用程序中相对应的信息以树状的形式在窗口中显示出来。哪些标签栏可以被用户使用则由安装 Machine Edition 选择安装软件的类型和你正在开发的应用程序的类型所决定，浏览窗口可以使用户清楚的了解整个应用程序的结构。

### 反馈信息窗口

反馈信息窗口是一个多重窗口，用来显示 Machine Edition 反馈的各种信息。信息反馈窗口通过不同的标签栏来分别显示不同类型的应用程序所反馈的信息。

### 属性窗口

属性窗口可以显示当前你所选择的物件的属性。你可以对各个属性进行编辑，如果你同时选中了多个物体，那么属性窗口则显示它们共同的属性。通过属性窗口来查看和修改物件的属性是一种非常简单和便捷的方法。



## 数据监视

数据监视工具是一个动态调试工具，它允许你在程序运行的时候监视和修改变量的数值，当在现场在线调试程序时，它是一个非常有用的工具，你可以监视单个的变量也可以监视由用户定义的变量表。变量监视列表可以被导入，导出或存储在工程中。



## 工具箱

工具箱是一个强大的物件仓库，你可以从中把你所需要的功能物件拖放到你的应用程序中去，同时你还可以自己定义功能物件，从而被Machine Edition 编辑并重复使用。在Machine Edition 中的工具箱还提供了创建物件向导功能。



## 在线帮助

在线帮助功能可以在开发应用程序的时候随时给你提示。当在线帮助功能被打开后，它会随时跟踪你的操作步骤并自动提供有用的帮助信息。在Machine Edition的环境中在线帮助是以当前被激活文本作为帮助主题。



## 信息浏览

信息浏览工具是一个内建的Web 浏览器，它主要用来显示：

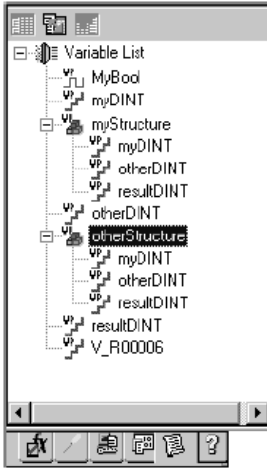
- Machine Edition 帮助
- Machine Edition 报告
- 工程或目标附带的文档资料

如果你非常熟悉Internet Explorer 或 Netscape Navigator，那么你很快就可以熟悉信息浏览工具的使用方法。和在线帮助一样，信息浏览也是以当前被激活文本作为帮助主题。在Machine Edition的环境中按 F1 就可以得到跟当前文本相关的任何帮助信息。同时一个信息列表将显示在信息浏览窗口中。



**想了解更多信息?** 可以在帮助索引中寻找“Tools: an Overview”。

## 变量（VARIABLES）



浏览器：变量标签栏

变量（有时被称为标签）是被命名的数据存储区。在工程中的所有变量都在变量浏览栏中被列出。每一个变量都会占用一个PLC的内存空间，所有的PLC系列（PACSystems™ RX7i除外）必须为每一个变量指定内存地址（如%R00001）。在 PACSystems™ RX7i中，你不需要对变量指定内存地址，系统会把它看作一个符号变量。Machine Edition会自动的分配这个变量在 PACSystems™ 中特殊内存空间。而其他属性如数据类型等则在属性窗口中进行配置。

Machine Edition支持数组和混合结构的变量。变量表可以被输出到多种格式的文件或从这些文件中导入，例如你可以用Excel来编辑，然后导入到Machine Edition的变量表中。Logic Developer - PLC中使用到的变量，同时也可以被CIMPLICITY View所浏览。下图就是典型的变量属性编辑窗口。

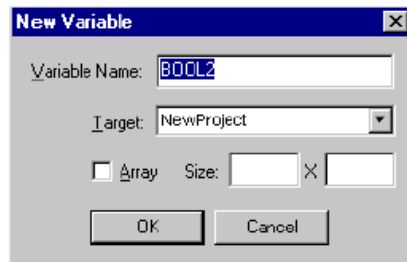


## 创建变量



新建变量菜单

1. 在 浏览窗口的 变量标签栏中，右键点击 Variable List，选中 **New Variable** 然后选择变量的数据类型后进入到新建变量对话框



2. 输入变量名后按回车 **ENTER**。

变量名可以由 1 到 32 个字符组成，以字母开头，包含大写字母和小写字母，数字 0 到 9，分隔符（“\_”）。

3. 选择变量基于的对象 **Target**。
4. 如果是创建一个数组变量，选中数组 **Array** 选项并输入数组的长度。
5. 点击 **OK**。

一个新的变量就在变量列表中被创建了。

## 为变量分配PLC内存地址

在把应用程序下载到PLC中之前，必须为变量分配PLC内存地址，除非它们是符号型的变量，由PACSystems™ RX7i自动分配。我们有两种方式来为非符号型的变量分配PLC内存地址。

### 第一种方式：

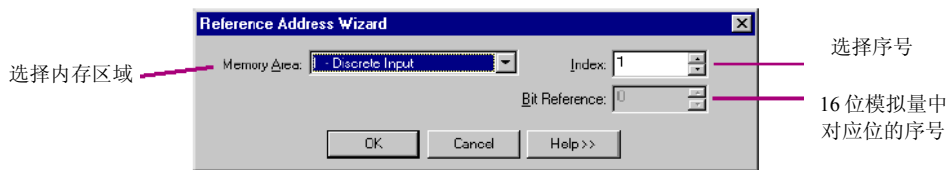
1. 在 工程浏览窗口的 变量标签栏中，右键点击变量并选择属性 **Properties**，进入变量的属性窗口。
2. 在 **Ref Address** 栏中，输入PLC内存地址，你可以：
  - 直接输入确定的地址，如%R00123 或 123R，那么%R00123就被分配给这个变量。
  - 仅输入地址的类型，如%R，PLC将为它自动分配一个连续的并且未被其他变量所使用的地址。例如，如果在其他变量中分配的PLC地址最高使用到%R00122，那么输入 %R 就将对应地址 %R00123。

## 第二种方式:

1. 在工程浏览窗口的变量表栏中，右键点击变量并选择属性 **Properties**，进入变量的属性窗口。
2. 在 **Ref Address** 栏中点击 **...** 按钮。



进入地址分配向导



3. 在内存变量选择列表中，选择你需要的PLC内存变量类型。
4. 在序号栏中，输入地址序号。
5. 选择对应位的序号。

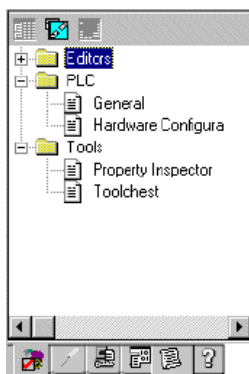
当你在 PACSystems™ PLC 需定义一个数字量，它是对应于16位模拟量中某一位时，**Bit Reference** 可选项才有效。

6. 点击 **OK**.

你的变量的PLC内存地址就分配完了。

**?** 想了解更多信息？可以在帮助索引中寻找“Variables”，“Importing variables from a file”，“Symbolic Variable”，和“Bit addressing in 16-bit memory”。

## 选项 (OPTIONS)



浏览器：选项标签栏

通过在浏览窗口中的选项Options标签栏可以修改 Machine Edition的环境设定。各种功能选项是以文件夹的形式显示的，通过鼠标点击可以打开或收起文件夹。

例如：

**选择** Editors > Ladder > View >Coil Justification: 可以定义编程环境中的列数，缺省为10列。

**选择** PLC> General > Duplicate Addresses: 可以设定是否对变量地址重复进行阻止，警告或忽略。.

**选择** PLC > Hardware Configuration > New Reference Assignment: 可以设定在做硬件配置时模块所使用的缺省地址。

### 进行功能和环境的设定

1. 在浏览窗口中的选项标签栏中，展开功能选项文件夹，然后右键点击功能类型选择属性以进入属性配置窗口。
2. 在属性配置窗口中，对想修改的功能进行编辑。

---

**想了解更多信息?** 可以在帮助索引中寻找“Options”，“Confirmations”，“PLC Options”，和“General Options”。

---


## MACHINE EDITION 帮助

CIMPPLICITY Machine Edition 内建了一个在线帮助系统，在你使用Machine Edition 的时候能随时显示相关的帮助信息。通过信息浏览或在线帮助都可以获得帮助。

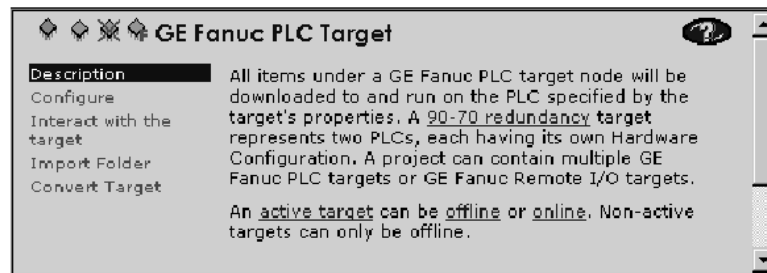
### 在线帮助

Machine Edition 的在线帮助系统在你工作时可以提供有用的提示和信息。当在线帮助功能被打开后，Machine Edition 就会自动跟踪你工作步骤并即时提示相关帮助信息。

### 使用在线帮助

1. 确认  在线帮助窗口已经打开。如果没有，按Shift+F11。
2. 在屏幕上任意点击。

与点击内容相关的帮助信息就显示在在线帮助窗口中。



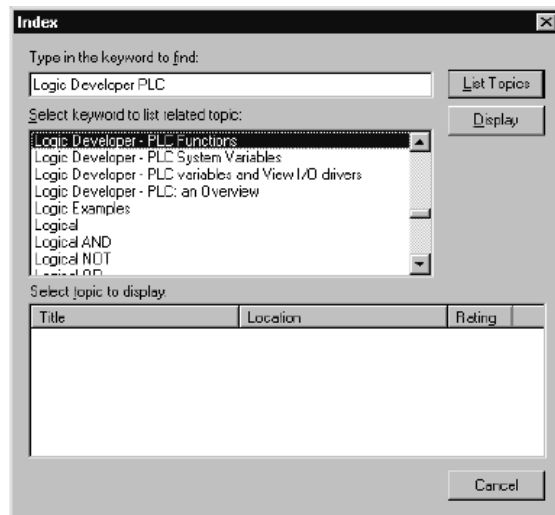
### 信息浏览器帮助

信息浏览器，是 Machine Edition 另一个帮助系统，也能提供详细的帮助信息。信息浏览器有自己的工具栏、内容列表、和搜索引擎，用于帮助你使用整个信息系统。和在线帮助一样，信息浏览器也是根据在信息浏览窗口中所选中的文本来显示帮助信息，你也可以在屏幕上选中一个条目然后按 F1 来显示相关信息。

## 使用帮助索引

1. 从 **Help** 菜单中，选择 **Index**。

进入帮助索引对话框。



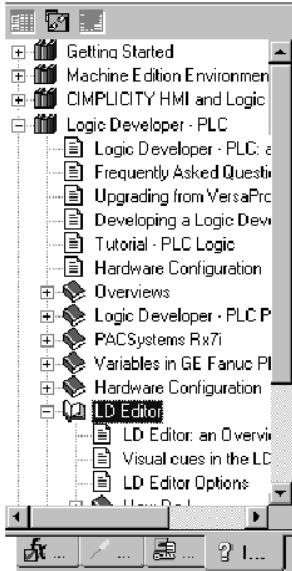
2. 在文本框中输入关键字或在下面的列表框中选择一个关键字。
3. 点击 **List Topics**。

相关主题列表就会显示在最下面的列表框中，所有主题是按照关键字在其中出现的频率的多少来排序的。

4. 在最下面的列表框中，选择你需要的主题。
5. 点击**Display**。

这个主题所有的内容都被显示在信息浏览窗口中。

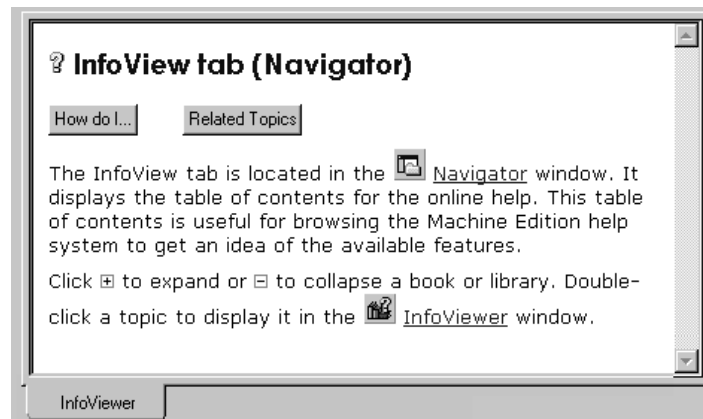
### 从帮助列表的内容中寻找相关主题



浏览器：信息浏览标签栏

1. 在浏览窗口中打开 信息浏览标签栏。  
帮助信息列表就会在信息浏览窗口中显示出来。
2. 展开 相关的帮助库或 电子书以查询所需的 帮助主题。
3. 双击 所需的主题。

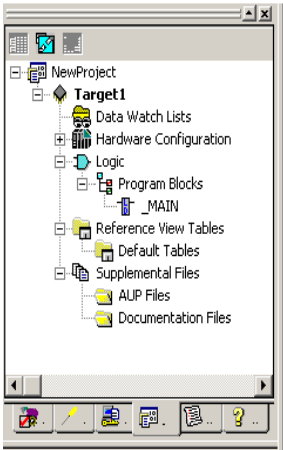
这个主题中所有的内容就显示在信息浏览窗口中。



# 3

## GE Fanuc PLC 对象 (Targets)

◆ 表示一个用 Logic Developer- PLC 编程的目标对象，每个对象包含所有可以下载到 PLC 的组件，Logic Developer- PLC 支持如下的 GE Fanuc PLC 对象系列：



浏览器：新 PACSystem™ 工程

- PACSystem™ RX7i
- Series 90™-70 PLC
- Series 90™-30 PLC
- VersaMax® PLC
- VersaMax® Nano/Micro PLC
- Series 90™ Micro PLC

和如下的 GE Fanuc 远程 I/O 接口的对象：

- GE Fanuc Series 90™-70 Genius 远程 I/O 扫描器
- GE Fanuc VersaMax® 以太网模块
- GE Fanuc VersaMax® Genius 模块
- GE Fanuc VersaMax® Profibus 模块

## 增加，配置和转换对象

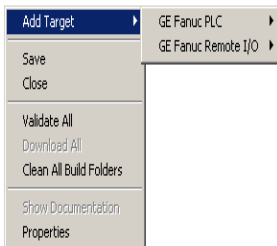
### 增加对象

通常，目标对象存在于用模版产生的工程中（参见第 12 页），一个工程项目可以包含多个目标对象。实际的对象的数量与你所使用的计算机的内存、硬盘大小及操作系统有关。




一个对象一般对应一个 PLC 或远程 I/O，只有当使用冗余的 CPU 时，一个对象才对应主/备两个 PLC，详情请参见第 50 页。

已有的对象可以从一种 GE Fanuc PLC 系列转换为另外一种，详情请参见第 26 页。

### 在工程项目中增加目标对象



右击工程弹出菜单





1. 在  浏览窗口的  工程标签中，右键单击  工程项目，并选择 **add Target**（增加对象）。
2. 指向“GE Fanuc PLC”或“GE Fanuc I/O”，选择 PLC 类型或远程 I/O 类型。。

## 配置对象

对象的属性包含确定 PLC 的类型，计算机和 PLC 的通讯连接方式，以及各种其他的设置。所有的属性都可以在属性栏中可以进行设置。如下表格是一个 PLC 对象的属性说明：

<b>Name</b>	对象名称
<b>Type</b>	PLC 的系列类型（只读）
<b>Description</b>	对象的描述（最大 255 个字符）
<b>Documentation Address</b>	工程文件存储的 URL 地址
<b>Family</b>	GE Fanuc PLC 的系列类型
<b>PLC Target Name</b>	PLC 对象名称
<b>Update Time (ms)</b>	在线时屏幕刷新时间
<b>Sweep Time (ms)</b>	在线时 PLC 扫描时间
<b>PLC Status</b>	PLC 在线/离线、运行/停止状态
<b>Dual HWC</b>	（S90-70 专有）硬件配置的冗余设置。设为 False 删除冗余配置
<b>Selected HWC</b>	（冗余设置为 True 时有效）选择不同的硬件配置
<b>Physical Port</b>	通讯端口（以太网/串口/Modem）
<b>IP Address</b>	（以太网协议有效）设置 IP 地址
 <b>Additional Configuration</b>	详细的通讯端口参数




## 配置 PLC 对象

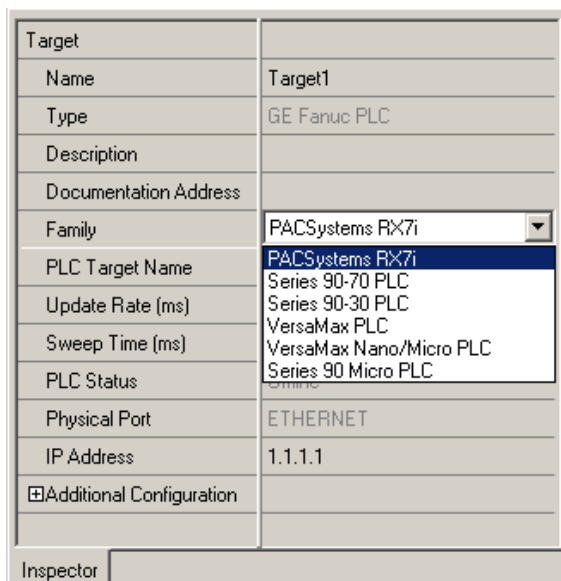
1. 在  浏览窗口的  工程标签中，右键单击  对象，并选择 **Properties**（属性）。
- PLC 的属性出现在属性窗口中。
2. 在属性窗口中， 可按需要来配置每个可选项。

## 转换对象

使用编程软件，目标对象可以在 GE Fanuc PLC 的不同系列之间转换。例如你可以将 Series 90™-70 PLC 程序，转换为 PACSystem™ RX7i 的程序。但是，一旦执行转换，该项操作不能撤销，建议备份工程之后，再执行转换。在实际的操作过程中，还要先进行合法性校验。

### 转换对象

1. 在  浏览窗口的  工程标签中，右键单击  对象 **Target**，并选择 **Properties**（属性）。
- 属性窗口中显示了所选对象的属性。
2. 在属性窗口中，点击 **Family**。
  3. 从下拉选项中选择你所希望转换的系列类型。



转换后的对象包含如下的改变：

- 原来配置中的模块都消失了，改变成新选系列的默认的硬件配置。

**注意：** PACSystem™ RX7i 支持大量的 Series 90™-70 模块和远程机架，当转换 Series 90™-70 PLC 到 PACSystem™ RX7i 时，其支持的模块和机架将保留，参数也改变为最相近的数值；其不支持的模块和机架将转换为最相近的模块和机架。所有的硬件配置改变都都登录在目标对象转换信息反馈窗中。


- 删除或增加对象组件、功能块、程序。

- 
- 刷新系统变量，包括故障定位变量。
  - 当只转换成 PACSystem™ RX7i 对象时，在信息显示窗中显示转换报表。

**警告：**改变 PLC 的系列类型，可能会引起工程中的某些程序逻辑结构变为不合法。

**注意：**转换后，必须先配置 PLC 硬件，再进行其他操作。关于硬件配置的详细信息，参阅 46 页。

---




 **想了解更多信息？** 可以在帮助索引中寻找“**Target Conversion**”。

---


## 配置通讯

为了建立 PLC 和编程软件 Logic Developer-PLC 之间的通讯，必须正确配置通讯连接的参数。在对象的属性窗中，可以按需要调整通讯参数。

### 配置以太网或串口通讯

1. 在  浏览窗口的  工程标签中，右键单击  对象 **Target**，并选择 **Properties** (属性)。

打开的属性窗口中显示了所选对象的属性。

2. 在  属性窗口中，设置 **Physical Port** (物理端口)，选择串行口、以太网或 Modem (如果安装了 Modem) 为通讯端口。
3. 如果是以太网连接端口，请设置对象的 **IP Address** (IP 地址)。  
**注意：**不能通过这种方式连接 PACSystem™ RX7i。
4. 双击 **Additional Configuration** (附加配置)，配置详细的参数。

**注意：**除 PACSystem™ RX7i 外，其他系列 PLC 可以先使用串口连接 (SNP 协议)，设置其 IP 地址，方可建立以太网通讯。

---

 **想了解更多信息？** 可以在帮助索引中寻找 “Configuring a direct connection”。

---

### 配置 PACSystem™ RX7i 通讯

1. 连接 PACSystem™ RX7i 到以太网。
2. 设置临时的 IP 地址，详情参考 “为 PACSystem™ RX7i 设置临时的 IP 地址”。

用临时 IP 地址能够建立最初与 PACSystem™ RX7i 的通讯





3. 做下面所述步骤的其中一个：

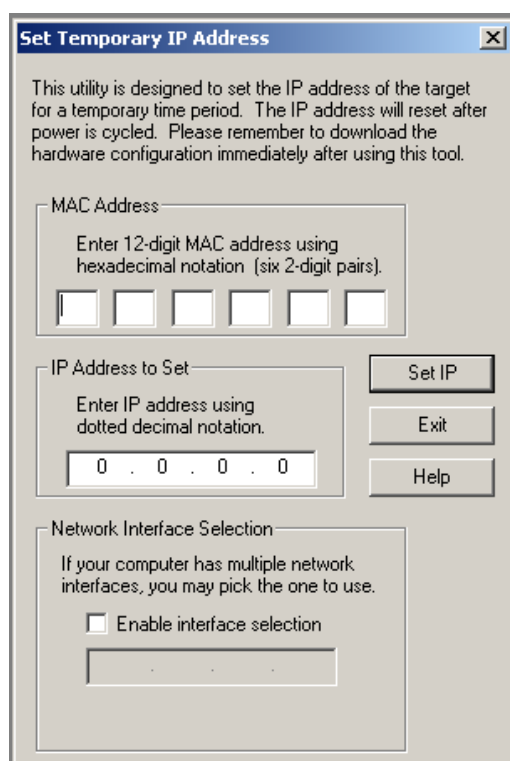
如果硬件配置已经存在于 PACSystem™ RX7i 中，其包含有永久的 IP 地址，则为了获取 PACSystem™ RX7i 的 IP 地址，可上载其硬件配置。

- 或者 -

设置永久的 IP 地址，（详情参考 “为 PACSystem™ RX7i 设置永久的 IP 地址”）。

## 为 PACSystem™ RX7i 设置临时的 IP 地址：

1. 在  浏览窗口的  实用工具标签中， 双击  来设置其临时的 IP 地址。  
出现设置临时 IP 地址的对话框。






2. 在“MAC Adress”栏中，输入该 PACSystem™ RX7i 的 MAC 地址。
3. 在“IP Adress”栏中，设置该 PACSystem™ RX7i 的临时 IP 地址。
4. 如果需要，可选择“Enable Interface Selection”有效，来指定 PACSystem™ RX7i 上安装的网络接口卡。
5. 点击“Set IP”按钮。

这个 PLC 的临时 IP 地址就设置好了。

**警告：**临时 IP 地址不能保持，如果重新上电，该临时 IP 地址将会丢失。您可以通过其它方法来设定永久的 IP 地址。


### 为 PACSystem™ RX7i 设置永久的 IP 地址

1. 在  浏览窗口的  工程标签中，右键单击  目标对象 **Target**，并选择 **Properties** (属性)。

属性窗口中显示了所选对象的属性。





2. 在对象的 **IP Address** (IP 地址) 属性中，设置其 IP 地址。
3. 下载硬件配置到 PACSystem™ RX7i (详情参考第 35 页)。

注意：下载前确信已配置了其它的硬件设置。

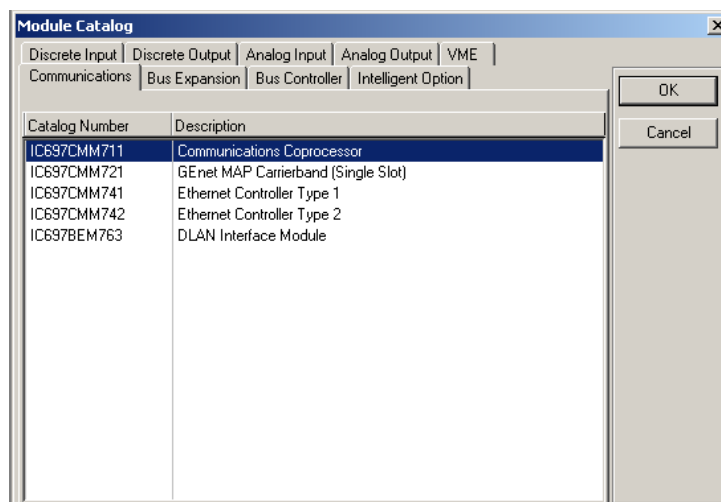
 **想了解更多信息？** 可以在帮助索引中寻找“Set Temporary IP Address utility”。

### 在非 PACSystem™ PLC 中的通讯配置

#### 设置 Series 90™-70 PLC 的 IP 地址

1. 在  浏览窗口的  工程标签中，展开  硬件配置节点。
2. 展开  主机架，并双击你所想要增加通讯模块的槽位。

出现模块产品目录选择窗



3. 选择 **Communications** (通讯) 标签，选中并增加通讯模块。  
选择 IC697CMM741 或 IC697CMM742 以太网模块。

4. 点击 **OK** 按钮。

出现参数编辑器，并显示所有以太网通讯模块的组态设定。

5. 在 **IP Address** 栏，双击 **IP Address Parameter** (IP 地址参数)。

IP 地址设置对话框出现



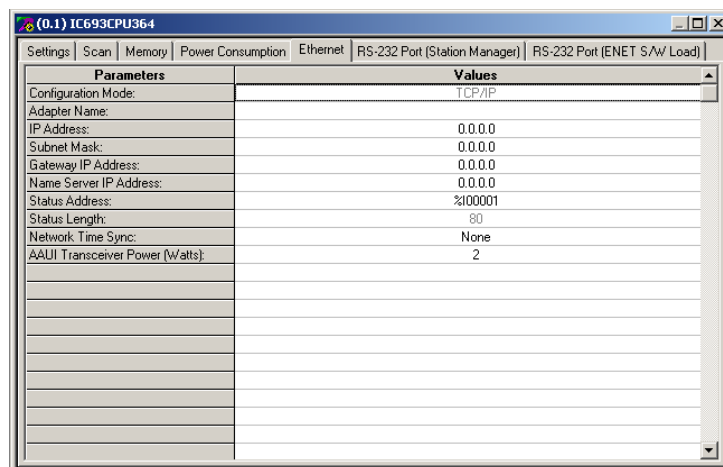
6. 输入 IP 地址，点击 “OK”。

## 设置 Series 90™-30 PLC 的 CPU364 和 CPU374 的 IP 地址

注意：对于 CPU364 和更高端的 CPU，以太网端口内置于 CPU 模块之中，但对于其他 CPU 模块，需要选择以太网模块进行通讯。




1. 需要一个 IP 地址，可能需要从网络管理员处获得。
2. 在 浏览窗口的 工程标签中，展开 *Hardware Configuration* (硬件组态) 文件夹，选择包含 CPU364 或 CPU374 的机架和插槽。
3. 右击 CPU 模块，选择 **Configure** (组态)  
参数编辑器显现。
4. 在参数编辑器中，点击 **Ethernet** 标签。

出现含有 Ethernet 标签内容的参数编辑器



5. 在 **IP Address** 栏，双击 **IP Address Parameter** (IP 地址参数)。  
IP 地址设置对话框出现
6. 输入 IP 地址，点击 “OK”。

### 设置 Series 90™-30 PLC 的 IP 地址 (不包括 CPU364 和 CPU374)

1. 在  浏览窗口的  工程标签中，展开  硬件配置节点。
2. 右击以太网模块 (IC693CMM321)，选择 “Configure”  
出现参数编辑器，并显示所有以太网通讯模块的组态设定。
3. 选择 “Setting” 栏，双击 “IP Address” 栏。  
IP 地址设置对话框出现
4. 输入 IP 地址，点击 “OK”。

### 下载 IP 地址到非 PACSystem™ PLC 中。

1. 在  硬件配置中设置好 CPU 模块或/和以太网通讯模块。
2. 右键单击  目标对象 **Target**，并选择 **Properties** (属性)。  
属性窗口中显示了所选对象的属性。
3. 在端口 **Physical Port** 的属性中，选择串口通讯。
4. 右键单击对象 **Target**，选择在线 **Go Online**。
5. 右键单击对象 **Target**，选择下载 **Download** 到 PLC。  
下载到 PLC 的对话框出现
6. 选择硬件配置 **Hardware Configuration**，单击 OK。
7. 右键单击对象 **Target**，选择离线 **Go Offline**。
8. 右键单击对象 **Target**，选择属性
9. 在端口 **Physical Port** 的属性中，选择 Ethernet 通讯。  
下一次再联机，Machine Edition 将用以太网进行连接。

## 编程器与 PLC 之间的交互操作



编程器只有在与 PLC 建立通讯后，才可以进行上装、下载程序，监视数据和 PLC 状态信息等操作。根据 PLC 的实际情况，可以使用串行口、以太网或 Modem 来建立编程软件 ME 和 PLC 之间的连接。

所有与对象 **Target** 的交互操作都在目标对象的右击菜单中出现。

### 检验对象

检验工程中可能存在的语法错误，并将错误的信息显示在反馈窗口中。含有错误的工程，不能下载到 PLC。基于这个理由，每次下载初始，ME 将会自动地检验程序。

#### 检验目标对象

- 在  浏览窗口的  工程标签中，右击对象 **Target**，并选择检验 **Validate**。

Logic Developer – PLC 对 **目标对象 Target** 的所有部件进行合法性检查。任何被查出的错误都被登录在  反馈窗口内。

**提示：** 双击在反馈窗口中的错误信息条目，可以自动的回到有错误的地方供检查。在线帮助的提示告诉你如何处理。打开在线帮助可以按 SHIT+F11 组合键。

### 在线/离线

如果处于离线状态，编程计算机和 PLC 之间就没有通讯，它们的交互操作在数量和类型上都是有限的，而实际的通讯连接时间也不需要象编辑程序那样长。如果已经配置了通讯的设置，并且 **离线 PLC 操作** 被允许，能进行如下的操作：

- 下载到 PLC
- 从 PLC 上载
- 启动 PLC
- 停止 PLC
- 清空 PLC 内存

当你使用上述的命令时，先连接通讯，然后执行操作，最后回到原来离线状态。

**注意：**可以在浏览窗口的选项标签 **Options tab** 中设置允许/不允许离线操作（详情参考第 19 页）。

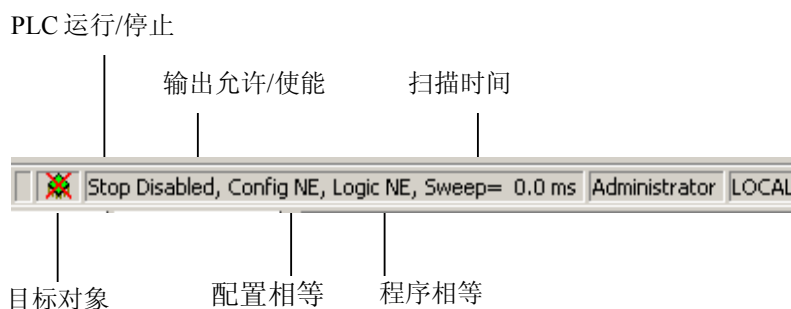
当 PLC **在线 Online** 时，存在编程器与 PLC 的通讯连接，并有实时的数据通讯，可以监视和控制 PLC 的运行和内存，包括上述的 **Offline** 时的操作命令。另外，当 **在线 Online** 时，且 PLC 的工程与编程器的工程相等时，LD 编辑器以图形化的方式显示指令执行。还可以在线编辑梯形图程序，并被提示下载“Word for Word”改变（关于“Word for Word”改变的详情，参考第 73 页）。

**?** 想了解更多信息？可以在帮助索引中寻找“Offline Versus Online”。

## 与 PLC 的在线

- 在浏览窗口的工程标签中，右击对象 **Target**，并选择**在线 Online**。

Logic Developer – PLC 连接一个工程到 PLC 时，**在线 Online** 状态显示在工程标签的对象 **Target** 图标上和状态栏中。



**注意：**当 PLC **在线 Online** 时，在浏览窗口的工程标签中的对象 **Target** 图标会显示成： 表示相等， 表示不等， 表示错误停止

## 与 PLC 的离线

- 在浏览窗口的工程标签中，右击对象 **Target**，并选择**离线 Offline**。

Logic Developer – PLC 与 PLC 离线。

**?** 想了解更多信息？可以在帮助索引中寻找“going online with a PLC”和“going offline from a PLC”。



---

## 上载/下载

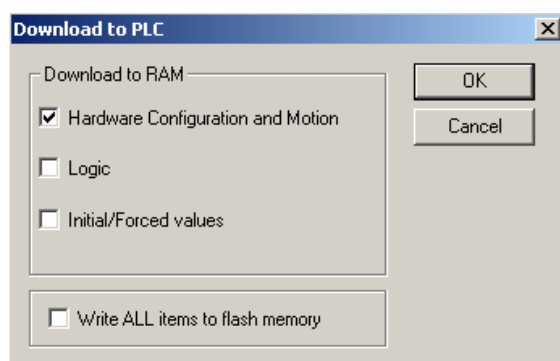
下载过程需要对对象的所有运行文件进行检验，编译工程需要同原先的配置比较（详情参考第 28 页）。

上载过程是将激活的 PLC 对象的工程获取激活的转换。

### 下载文件到 PLC

1. 在  浏览窗口的  工程标签中，右击想要下载的对象 **Target**，并选择 **下载到 PLC Download to PLC**。

出现下载到 PLC 的对话框




**注意：**如果 PLC 正在运行，则只能下载与 PLC 逻辑不相等的程序，不允许下载硬件配置，故不会出现该选项。


2. 选择你所希望下载到 PLC 的内容，单击 **OK** 按钮。

**提示：**为了把工程中的所有对象都下载，右键单击工程节点，并选择 **全部下载 Download All**。如果你希望只下载激活的目标对象，则选择 **下载激活对象 Download Active Target**。

**注意：**一个 PLC 对象在同一时刻，只能下载一个工程。如果一个对象设备已经存有一个工程，下载后原有的工程将被覆盖。

对每个对象的下载过程，Machine Edition 总是先保存工程，执行合法性检查，生成运行文件（the runtime files），尝试建立通讯的连接。在这个过程中发生的任何错误都将显示在 Build 标签的  反馈窗口中。如果通讯的连接成功，Machine Edition 传送所有的运行文件到 PLC。

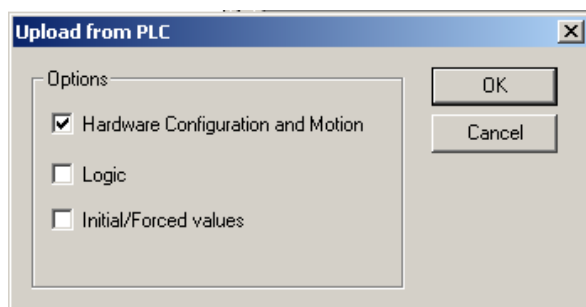
---

 **想了解更多信息？** 可以在帮助索引中寻找 “**downloading to PLC**”。

---

### 从 PLC 上载文件

1. 在 浏览窗口的 工程标签中，右击想要上载的 对象 **Target**，并选择 **上载到 PLC Upload from PLC**。



2. 选择你所希望从 PLC 上载的内容，单击 **OK** 按钮。

Logic Developer – PLC 编程软件连接到 PLC，并且上载所选定的内容。如果已经有一个打开了的 PLC 工程，则上载的工程合并入现有的工程，并确保上载的变量名全部保留。如果上载到一个空的对象，则所有的变量名将被使用默认的名称。如：%R00001 命名为 R00001。

**?** 想了解更多信息？可以在帮助索引中寻找“**downloading to PLC**”。





### 运行/停止 PLC

可以设置 PLC 的运行/停止模式，在停止模式下，还可以设置输出允许/禁止。

如果启动 PACSystem™ RX7i 或 Series 90™-70 PLC 运行，还可以设置输出允许/禁止。


---

## 启动 PLC

- 在  浏览窗口的  工程标签中，右击对象 **Target**，并指向 **在线命令 Online Commands** 或 **离线命令 Offline Commands**，然后选择 **启动 PLC Start PLC**。

目标 PLC 开始执行其程序。

## 停止 PLC


- 在  浏览窗口的  工程标签中，右击对象 **Target**，并指向 **在线命令 Online Commands** 或 **离线命令 Offline Commands**，然后选择 **停止 PLC Stop PLC**。

PLC 停止对话框出现，提示是否允许/禁止 PLC 的输出。

- 选择一个选项。
- 单击 OK。

目标 PLC 停止执行其程序。

---



 **想了解更多信息？** 可以在帮助索引中寻找 “starting the PLC” 和 “stopping the PLC”。

---

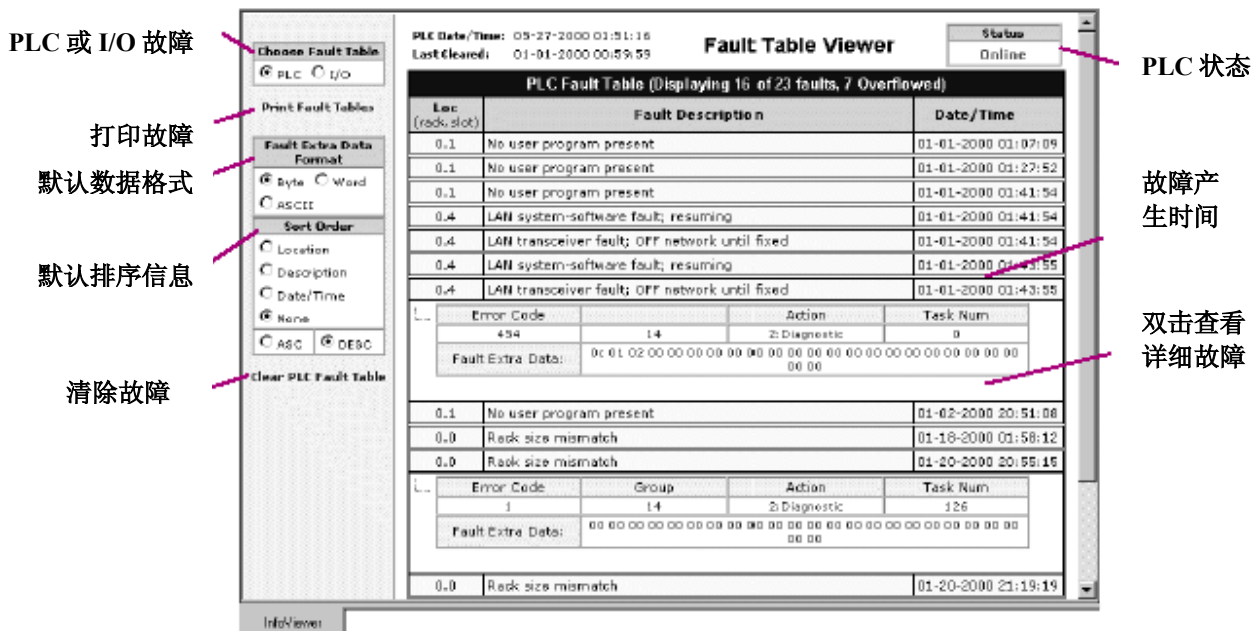
## 故障表

PLC 和 IO 故障表显示了由 PLC 的 CPU 或模块登录的故障信息。这些信息常用于确定 PLC 的硬件或软件的哪部分出了问题。浏览故障表，计算机与 PLC 必须处于在线状态。

## 查看故障表报告

- 在  浏览窗口的  工程标签中，右击想要查看的对象 **Target**，选择 **诊断 Diagnostics**。

故障显示在信息查看 **InfoViewer** 窗中。



**?** 想了解更多信息？可以在帮助索引中寻找“Working with the Fault Tables”。

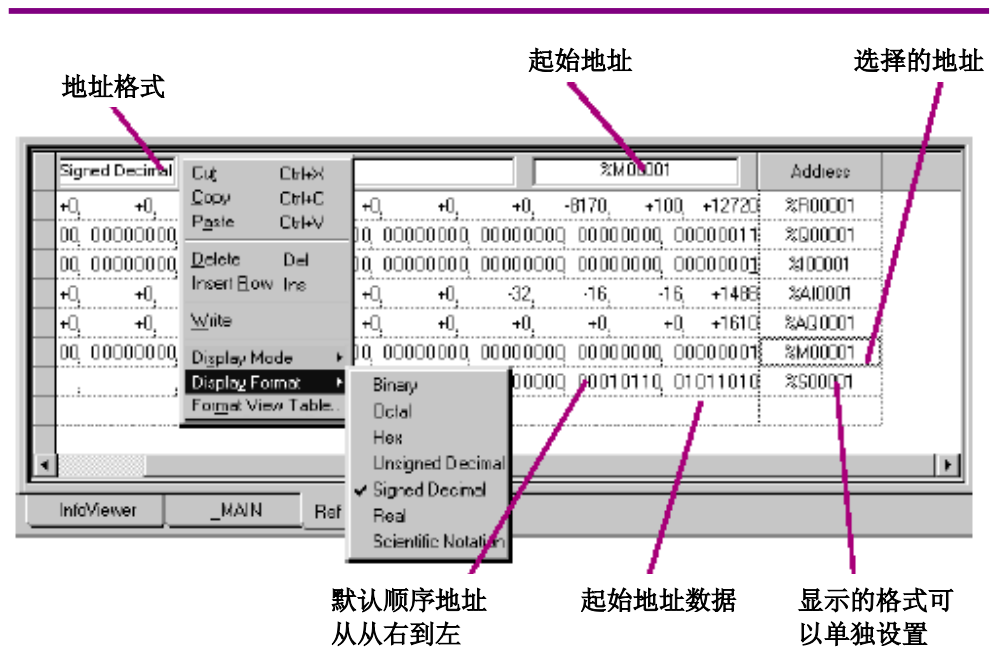
## 参考变量察看表

参考变量察看表（RVTs）是能够实时的监视和改变的变量地址表。在 浏览窗口的 工程标签中的参考变量察看表文件夹里，有默认的变量表，你也可以添加用户定义的变量表。一个对象可以有 0 个或多个用户定义的 RVTs 变量表。

RVTs 变量表中包含的变量数量并不影响性能，只是影响显示和刷新的视觉效果。

RVTs 变量表只显示激活且在线的目标 PLC 的变量，可以在浏览窗口的选项标签中配置该变量表的显示方式。

地址数值的默认显示方式是按指定的以起始地址，是从右到左的顺序排列。默认的或用户定义的 RVTs 变量表都是以离散地址每行 8 个单元，连续地址每行 10 个单元方式显示。显示数据的数量依赖于数据显示的格式。



### 建立用户定义的变量表

- 在浏览窗口的工程标签中，右击参考变量表 **Reference View Tables**，并选择“New”。

一个名为“default”的新表增加在 RVTs 变量表文件夹下。

### 操作变量表



工程变量表

- 在浏览窗口的工程标签中，展开变量表文件夹中，双击变量表。变量表显示在 **Machine Edition** 的主窗口中。
- 添加变量地址。  
**注意：**不能在默认的变量表内添加变量地址。
- 按需要选择变量显示格式。

想了解更多信息？可以在帮助索引中寻找“RVT”和“Working with Reference View Tables”。

## 报告



报告提供了工程的摘要和工程的各种信息，大部分报告显示在信息浏览窗里（详情参考第 20 页）。在 **报告 Reports** 标签的  反馈栏中包含了 **Machine Edition** 工程自本次打开后生成的所有报告。**Logic Developer - PLC** 软件可以生成和打印如下的报告：

地址使用报告	C 语言程序块报告*	在线强制数据报告
硬件配置报告	指令程序块报告*	
变量报告	本地逻辑块报告*	
CAM 报告	运动控制块报告*	

### 注意：

- 含有“\*”号的表示可以打印输出。
- 梯形图逻辑块的报告不能生成，只能从程序块中直接打印输出。


## 生成报告

- 在  浏览窗口的  工程标签中，右击所需生成报告的节点，并选择 **报告 Report**。

一个报告自动地生成在信息窗口中。

**注意：**生成**在线强制报告**，PLC 必须要求处于在线状态。右击目标对象，选择**在线命令 Online Commands**，再选择“**Online Force Report**”。

## 重新显示先前生成的报告


1. 在  反馈栏中，双击 **报告 Reports** 标签。  
所有先前生成的历史报告出现在反馈栏处。
2. 从清单中选择需要查看的报告。  
报告显示在信息查看窗口中。

---


报告中的许多条目有超级连接，点击有超级连接的条目，可跳转至这个条目在工程中的地方。例如：在一个报告中有一个具有超级连接的变量名，点击这个变量名，则可连接到变量表中的这个变量的位置。

大的报告可能分为好几页，为了查看报告中不同的页面，可以使用滚动条上下浏览。

## 打印信息浏览窗中显示的报告



1. 如下可做：
  - 生成你所需的报告来打印。
  - 或者-
  - 重新显示原先的报告。
2. 当报告显示后，右击  信息窗口，选择打印 **Print**。

---


 **想了解更多信息？** 可以在帮助索引中寻找“**Reports**”。

---

## 打印逻辑块

1. 在  浏览窗口的  工程标签中，展开 **逻辑 Logic** 节点。
2. 右击程序块节点，选择“**Print LD Blocks**”。
- 出现打印对话框。
3. 选择要打印的逻辑块。
  - 需要打印所有的梯形图程序块，选择“**All**”。
  - 只要打印某些梯形图程序块，先选中“**Selection**”可选项，然后选择每个需要打印的程序块前的复选框。
4. 设置打印选项，点击“**OK**”。

---

 **想了解更多信息？** 可以在帮助索引中寻找“**printing LD logic**”。

---



# 4

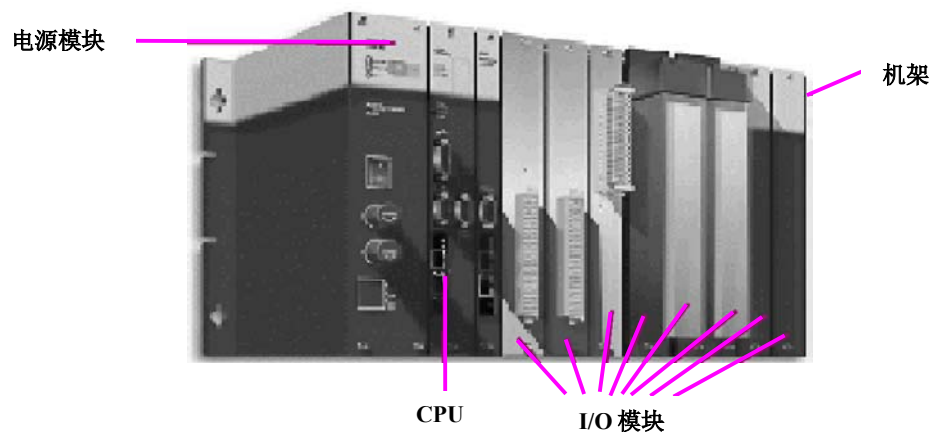
## 硬件组态 (HWC)

PLC 逻辑开发器 (**Logic Developer-PLC**) 支持 6 个系列的 GE Fanuc 可编程控制器 (PLC) 和各种远程 I/O 接口 (见 59 页), 包括它们各自所属的各种 CPU, 机架和模块。(见 95 页)。为了使用上述产品, 必须通过 **Logic Developer - PLC** 或其它的 GE Fanuc 工具对 PLC 硬件进行组态。**Logic Developer - PLC** 的硬件组态 (HWC) 组件为你的设备提供了完整的硬件配置方法。这一章详细描述了如何为您的应用进行硬件组态。

硬件组态的第一步是选择您想要组态的 CPU (见 24 页), 当您创建一个新的工程时, 您可以使用工程模板所包含的默认硬件组态, 或者创建一个空的工程, 然后自己进行组态。

### PAC Systems™ R X 7 I

下图所示的 PACSystems™ RX 7i 是 GE Fanuc 公司 PLC 产品中最新的一个系列。它的大部分特性基于 90™-70 系列 PLC, 而它的以太网性能是基于 90™-30 系列 PLC 的 CPU374。



PACSystems™ RX 7i 由主机架和最多 7 个扩展机架组成。PACSystems™ RX 7i 支持 2 种类型的主机架, IC698CHS017 和 IC698CHS117。同时它支持 5 种类型的 90™-70 系列扩展机架。所有的主机架和扩展机架都支持 3 种非智能型的 90™-70 系列风扇套件。

PACSystems™ RX 7i 的主机架拥有 18 个单槽宽度的 VME 插槽。0 号槽支持 2 种电源模块；1 号和 2 号槽安装双槽宽度的 CPU 模块。PACSystems™ RX 7i 支持 2 种 CPU，IC698CPE010（300MHz）和 IC698CPE020（700MHz）。

PACSystems™ RX 7i 的编程和组态只支持以太网通讯，并且 CPU 有一个内置的以太网适配器。在主机架上，可另增 3 个 IC698ETM001 以太网模块。以太网模块仅占用 1 个槽，所有的 IC698 模块只能在主机架上使用。

PACSystems™ RX 7i 支持多种 90™-70 系列模块。它们都是双槽宽度模块，因此，每个模块占用 2 个主机架插槽，但它仅占用 1 个 90™-70 系列扩展机架插槽。您可以将双槽宽度模块插入除了 0，1，2 号以外的任何两个相邻的插槽。因为 17 号槽是一个双槽宽度槽，任何插入 17 号槽的模块仅占用 17 号槽。17 号槽也可插入单槽宽度模块。

PACSystems™ RX 7i 支持下列的 90™-70 系列总线扩展模块

- IC697BEM711
- IC697BEM713

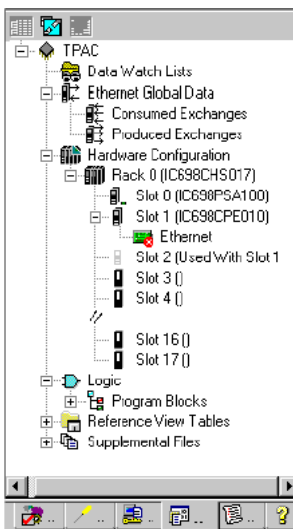
除对 90™-70 系列的支持外，PACSystems™ 也支持众多的 Genius 设备。

## PACSystems™ RX7i 硬件组态

当您创建了一个 PACSystems™ RX 7i 系统的对象，默认的硬件组态由主机架、0 号槽中的电源模块、1 号和 2 号槽中的一个 CPU 模块组成。您可以更改默认电源和 CPU 类型，并加入各种单宽度和双宽度的模块。最多您可以增加 7 个本地扩展机架，并在其上插入 90™-70 系列模块。

**注意：**在 PACSystems™ RX 7i 对象中，默认组态仅有 1 个主机架。而在 90™-70 和 90™-30 系列中，默认的硬件组态带有 7 个本地扩展，您不需要再增加。




以下是 PACSystems™ RX 7i 的详细组态步骤。对于其它 **Logic Developer - PLC** 支持的 PLC，步骤几乎是相同的。



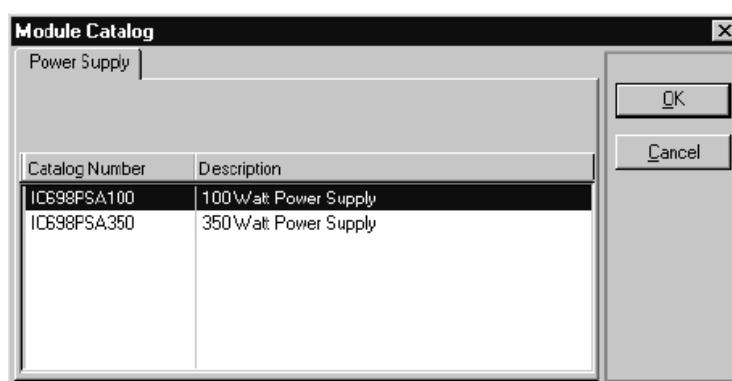
工程标签显示 PACSystem™  
RX7i 的硬件配置

## 选择电源

PACSystems™ RX 7i的默认电源模块是PSA100。改变电源模块的步骤：

1. 在  浏览窗口的  工程标签中，右键单击  **Slot 0** (0号槽) 并选择 **替换模块 Replace Module**。

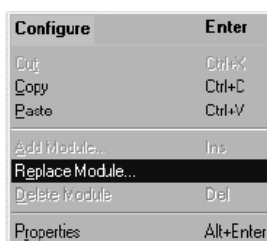
此时会显现一个列表，显示所有可用的电源模块。



**注意：**对于其它的GE Fanuc PLC系列，没有Slot 0（0号槽），代以对**PWR**节点右键单击。

2. 选择您安装在主机架上的电源模块并单击**OK**

## 选择一个CPU



HWC 右键单击菜单

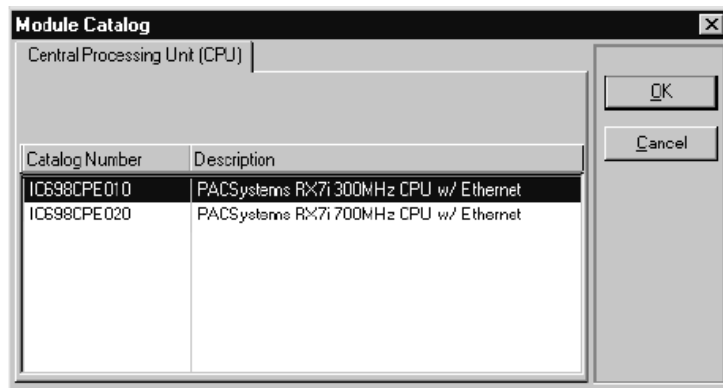
在所有的PACSystems™ RX 7i工程模板中，默认CPU都是CPE010，更改CPU的步骤：

1. 在  浏览窗口的  工程标签中，展开  硬件组态**Hardware Configuration**文件夹。

所有的机架都显现。

2. 展开  主机架
3. 右键单击  **Slot 1** 并选择 **Replace Module** (替换模块)。

出现模块目录对话框



4. 从表中选择一种CPU。
5. 单击**OK**

一个对话框出现，询问您是否希望保留现有CPU的设置。

6. 单击**Yes**或**No**

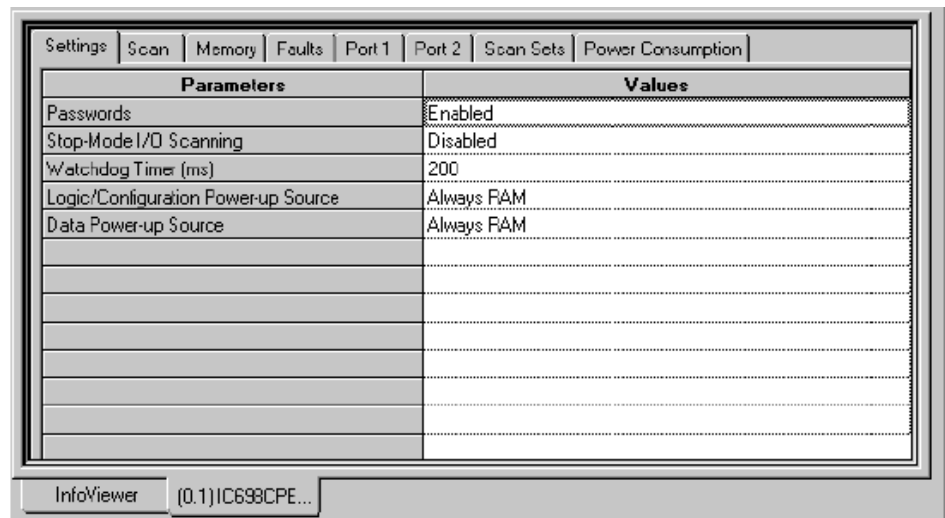
**注意：**除了PACSystems™外其它PLC，还将提示您确认替代操作。

该对象用新选的CPU来组态。

## CPU组态

1. 在 浏览窗口的 工程标签中，对一个包含CPU的槽单击右键并选择 **Configure**（组态）。

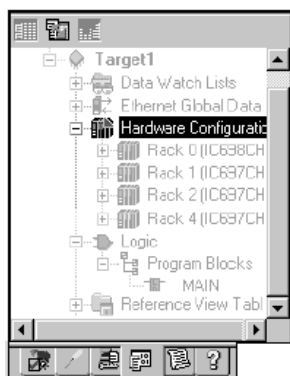
参数编辑器显现，并显示所有CPU的组态设定。



2. 按需要修改设定，关于参数信息，可选择参数，帮助会显示在在线帮助中。

技巧：按下SHIFT+F11可打开在线帮助。

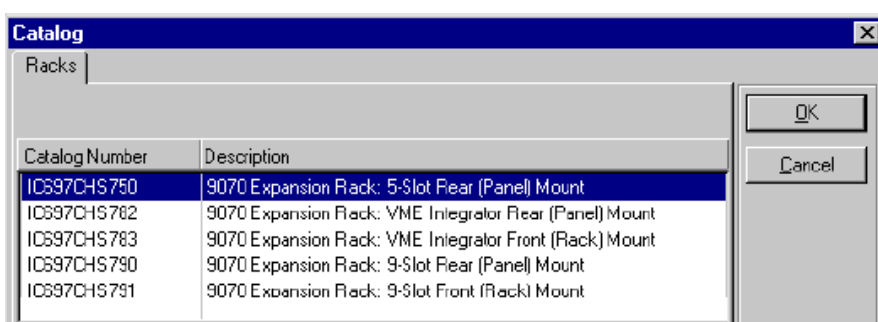
### 增加扩展机架（仅对PACSystems™ RX 7i）



注意：在90™-70和90™-30系列中，默认的硬件组态带有7个本地扩展，您不需要再增加。

1. 在 浏览窗口的 工程标签中，对 HWC 节点单击右键并选择 **Add Rack**（增加机架）。

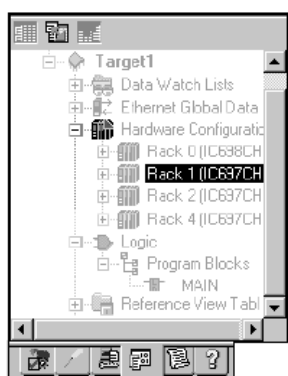
模块目录对话框显现，并列出所有可用的机架类型。



2. 选择一种机架并单击 **OK**。

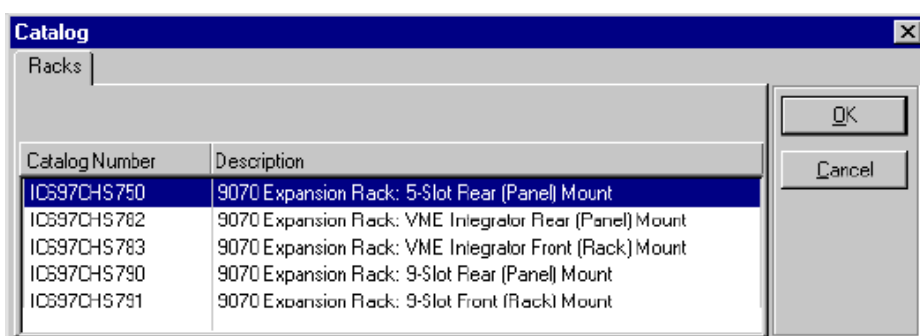
S90-70扩展机架被加到PACSystem RX7i系统中，除了已经使用的机架号外，1~7号都可以分配给这个机架。

### 替换一个机架



1. 在 浏览窗口的 工程标签中，对 HWC 节点单击右键并选择 **Replace Rack**（替换机架）。

模块目录对话框显现，并列出所有可用的机架类型。



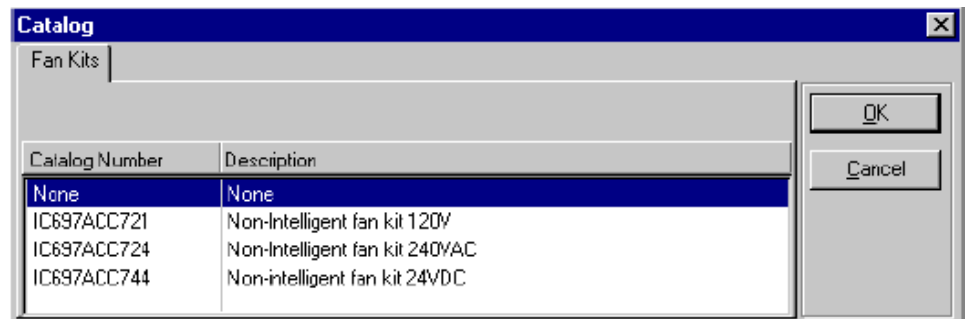
2. 选择一种机架并单击 **OK**。

### 为机架增加风扇套件（仅对PACSystems™ RX 7i）

1. 在浏览窗口的工程标签中，展开（硬件组态文件夹）。
2. 对一个机架单击右键并选择**Properties**（属性）。  
机架的属性会显示在属性窗口（Inspector）中。
3. 在属性窗口中的**Fan Catalog Number**特性的右边，单击。

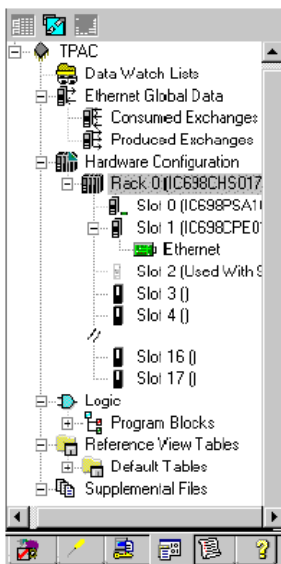


打开风扇产品目录菜单。







4. 选择想要增加的风扇套件，单击**OK**。

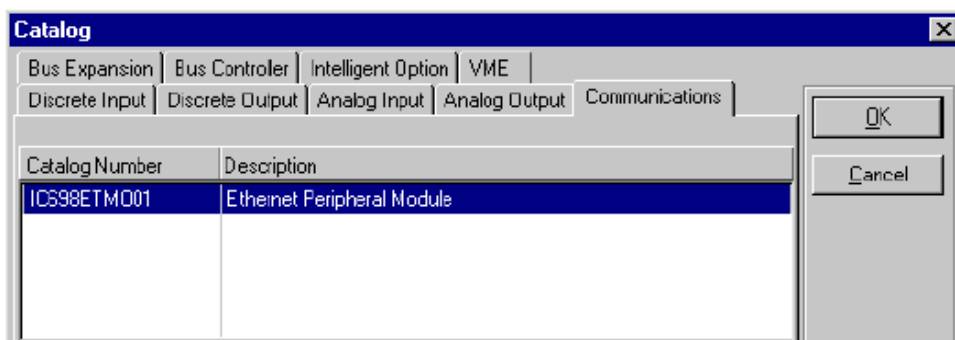
### 组态以太网子板（仅对PACSystems™ RX 7i）



1. 在浏览窗口的工程标签中，依次展开 **Hardware Configuration**（硬件组态）文件夹、**Rack 0**（0号机架）、**Slot 1**（1号槽）。
2. 双击 **Ethernet adapter**（以太网适配器）。  
参数编辑器窗口打开。
3. 按照需要，组态以太网适配器的参数。


### 增加以太网模块（仅对PACSystems™ RX 7i）

1. 在  浏览窗口的  工程标签中，依次展开  Hardware Configuration（硬件组态）文件夹、 Rack 0（0号机架，主机架）。
2. 右键单击需要增加以太网模块的插槽，并选择ADD Module（增加模块）  
模块目录打开



3. 在Communication（通讯）标签的模块列表中，选择以太网模块并单击OK。
4. 双击以太网模块（IC698ETM001）。  
参数编辑器打开。
5. 按照需要，组态以太网模块的参数。

---

 **想了解更多？** 在帮助索引中，搜索“Configuring a PACSystems™ RX 7i Rack System”。

---

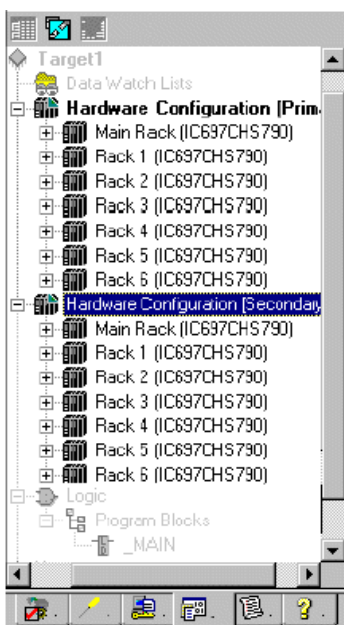
## 冗余（仅对90™-70系列和PACSystems™ RX 7i）

在冗余系统中，对两个单元进行设定和组态，让它们共同承担了原来一个单元承担的责任。如果其中一个单元出现故障或者离线，另一个单元可以无中断地接替整个系统的工作。

有三种类型的冗余系统：

- 基本CPU冗余（仅对90™-70系列）
- Genius冗余
- 基于Genius的CPU冗余（仅对90™-70系列）

### 基本CPU冗余



（仅对90™-70系列的CGR772和CGR935 CPU）为实现CPU冗余，两个物理意义上的PLC，1个主机Primary和1个从机Secondary，被关联到单独一个对象。两个PLC共享一个逻辑程序（Logic），但拥有各自独立的硬件组态：主机和从机。选中的硬件组态可进行联机，下载，上载等操作。

90™-70系列的机架系统支持三种类型的基本CPU冗余。这些冗余系统可以和Genius冗余方案结合使用。

- 单总线固主：使用一条Genius总线，在每个PLC中有一个或多个总线控制器。当系统初始化同步时，总是使主机为激活单元。
- 单总线浮主：使用一条Genius总线，在每个PLC中有一个或多个总线控制器。当系统初始化同步时，不会为了使主机单元激活而发生切换。
- 双总线浮主：使用两条Genius总线，在每个PLC中有一个或多个总线控制器。当系统初始化同步时，不会发生切换。为组态双总线网络，总线切换模块（BSMs）是必须的。这个选项对PLC和I/O总线都提供了冗余。

### Genius 冗余

(90™-70系列和PACSystems™ RX 7i) 为确保在部分组件故障或失效的情况下，Genius系统依然能正常操作，一个冗余的Genius系统是由一些双份的组件构成。冗余Genius系统可由90™-70系列冗余CPU系统组成。

您可以组态5种类型的冗余Genius系统

- Genius双网冗余（一个CPU，两个GBC，双网）
- Genius双网冗余（两个CPU，两个GBC，双网）
- Genius双GBC冗余（一个CPU，两个GBC，单网）
- Genius双GBC冗余（两个CPU，两个GBC，单网）
- Genius双网和双GBC冗余（两个CPU，4个GBC，双网）

### 架构在Genius之上的CPU冗余

（仅对90™-70系列）一个架构在Genius之上的CPU冗余系统是由一些双份的组件构成，在部分组件故障或失效的情况下，可确保系统依然能正常操作。

由各种冗余Genius系统和基本CPU冗余组合，可组成5种类型的架构在Genius之上的CPU冗余系统。

- CPU冗余（GHS），采用Genius双GBC冗余（两个CPU两个GBC）--固主单总线。
- CPU冗余（GDB），采用Genius双GBC冗余（两个CPU两个GBC）--浮主单总线。
- CPU冗余（GDB），采用Genius双网冗余（两个CPU两个GBC）--浮主单总线。
- CPU冗余（GDB），采用Genius双网和双GBC冗余--浮主双总线。
- CPU冗余（GDB），采用混合Genius冗余方案。

---





🔍 **想了解更多？** 在帮助索引中，搜索“Redundant HWC Overview”。

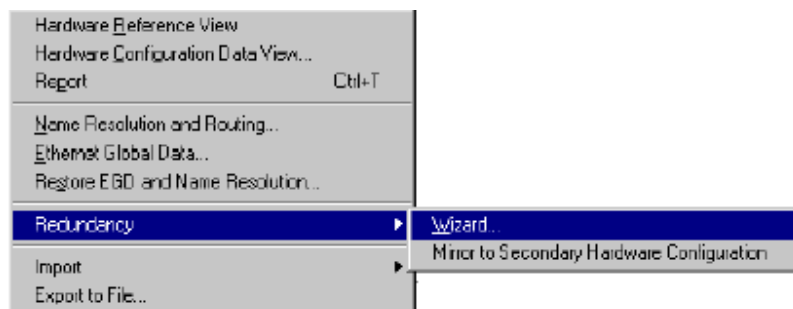
---

## 冗余系统组态

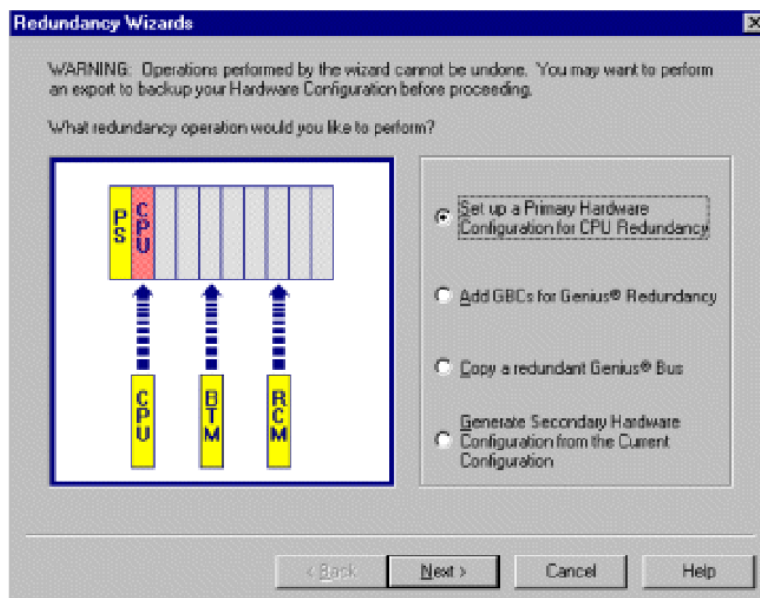
以下过程是组态冗余系统的通常步骤。对特定的冗余系统组态步骤，请查看在线帮助。

### 为CPU冗余设定主机硬件组态（仅对90™-70系列）

1. 在  浏览窗口的  工程标签中，展开您想设置为CPU冗余的  对象节点。
2. 对  硬件组态单击右键，指向冗余（Redundancy），选择向导（Wizard）



冗余向导对话框会打开，为CPU冗余设置主机硬件是默认选项。







2. 选择下一步 (Next) 并随着向导直到结束。向导会在组态中加入冗余的CPU，总线传送模块 (BTM) 和冗余通讯模块 (RCM)。

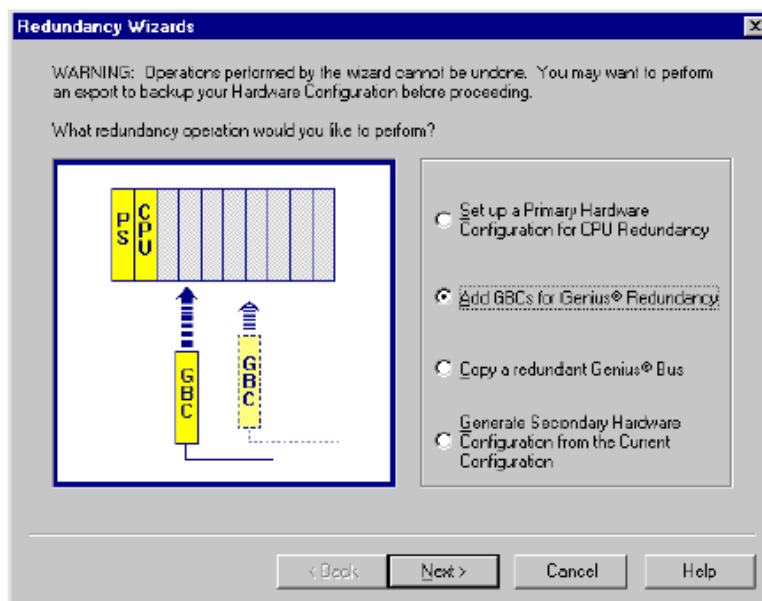
注意：对象属性中，Dual HWC，现在可以访问并被设为False。

Target	
Name	Target1
Type	GE Fanuc PLC
Description	
Documentation Address	
Family	Series 90-70 PLC
Program Name	Target1
Update Rate (ms)	250
Sweep Time (ms)	Offline
PLC Status	Offline
Dual HWC	False
Physical Port	COM1
Additional Configuration	

Inspector

在系统中加入**Genius总线控制器 (GBC)** (仅对90™-70系列和PACSystems™ RX 7i)




1. 在  浏览窗口的  工程标签中，展开您想设置为冗余的  对象节点。
2. 对  硬件组态单击右键，指向冗余 (Redundancy)，选择向导 (Wizard) 冗余向导对话框打开。

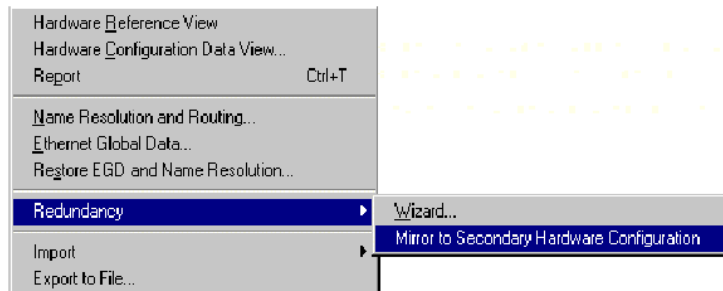


3. **enius冗余增加GBC (Add GBCs for Genius Redundancy)**、
4. 选择下一步 (**Next**) 并随着向导直到结束。向导允许您选择一个**Genius**冗余方案并定位**GBC**模块。在同一个系统中, 您可以多次运行这个向导来增加另外的一对冗余总线。

注意: 当您在90™-70系列机架上增加GBC模块时, 一条新的Genius总线会自动生成并与槽位号和GBC相关联。最多31个设备(块)可以通过这条Genius总线连接到GBC上。

### 从机的硬件组态 (仅对90™-70系列)

1. 在  浏览窗口的  工程标签中, 选择  硬件组态文件夹。
2. 右键单击硬件组态, 指向 **Redunrancy**, 选择**Mirror to Secondary Hardware Configuration**。



从机的机架系统是主机机架系统的拷贝。现在对象显示两个硬件组态 (HWCS), 一个标签是主机 (Primary) 另一个是从机 (Secondary)。主机的HWC是粗体的因为它被选中了。对象属性中, **Dual HWC**, 现在被设为True。

Target	
Name	Target1
Type	GE Fanuc PLC
Description	
Documentation Address	
Family	Series 90-70 PLC
Program Name	Target1
Update Rate (ms)	250
Sweep Time (ms)	Offline
PLC Status	Offline
Dual HWC	True
Selected HWC	Primary
Physical Point	COM1
<input type="checkbox"/> Additional Configuration	

Inspector

**注意：**在修改了主机硬件组态后，为了同步两个硬件组态，您可以多次进行镜像。每次对主机硬件组态镜像后，从机的硬件组态更新并反映出变化。

---

 **想了解更多？** 在帮助索引中，搜索“Redunrancy”。

---

### 90™-30系列：DSM314运动模块

90™-30系列支持一个其它GE Fanuc PLC系列不支持的模块：Motion Mate DSM314（见91页）。DSM314是一款高性能，易使用，多轴运动控制模块。它与90™-30系列的逻辑控制和通讯功能相兼容，DSM314支持下面两种伺服类型：

- 数字式—GE Fanuc数字伺服放大器和马达。
- 模拟式—GE Fanuc SL系列模拟伺服放大器和第三方马达。

DSM314可对4根轴独立组态成标准模式或随动模式。它可最多控制模拟模式4轴或数字模式2轴。当轴1和轴2在数字模式控制下时，轴3可控制模拟伺服。

为了在Logic Developer—PLC中完成运动控制程序，必须对DSM314进行正确的组态（见91页）。

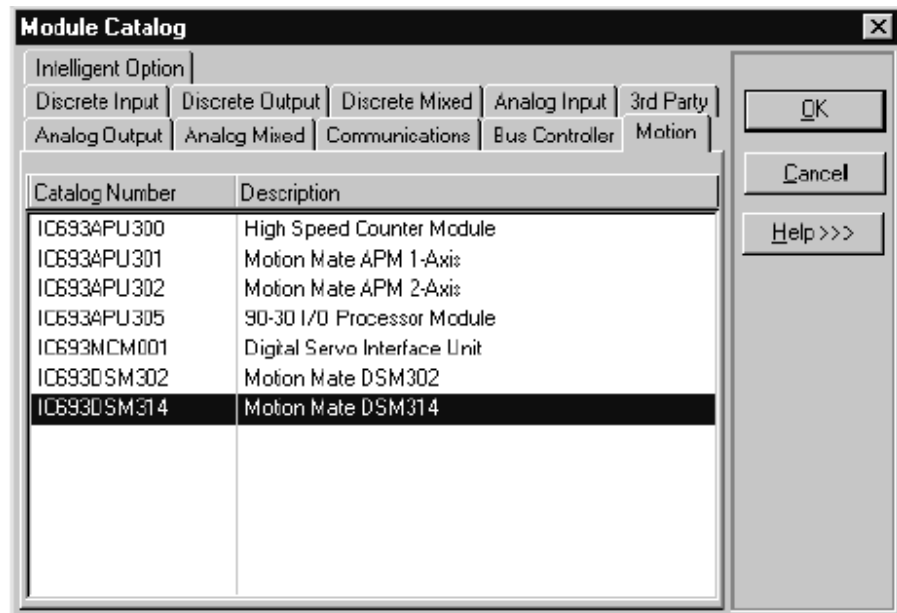
### 增加 DSM314 模块

1. 当  硬件组态中的一个机架节点展开后，在您想插入 DSM314 模块的空槽上，单击右键并选择增加模块（**Add Module**）。

模块目录对话框出现。

2. 在模块目录对话框中，选择运动（**Motion**）标签。  
可供选择的运动模块类型出现在模块目录对话框中。

3. 从表中选择 DSM314。



4. 单击 **OK**。

DSM314 已增加到工程的组态中。

## 对 DSM314 组态

1. 在 浏览窗口的 工程标签中，双击含有 DSM314 模块的槽位。

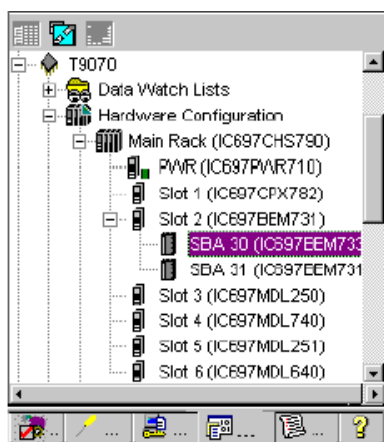
参数编辑器出现

Parameters	Values
Number of Axes:	4
%I Reference:	%I00001
%I Length:	80
%Q Reference:	%Q00001
%Q Length:	80
%AI Reference:	%AI0001
%AI Length:	84
%AQ Reference:	%AQ0001
%AQ Length:	12
Axis 1 Mode:	Analog Servo
Axis 2 Mode:	Analog Servo
Axis 3 Mode:	Auxiliary Axis
Axis 4 Mode:	Disabled
Local Logic Mode:	Disabled
Total Encoder Power (Watts):	0

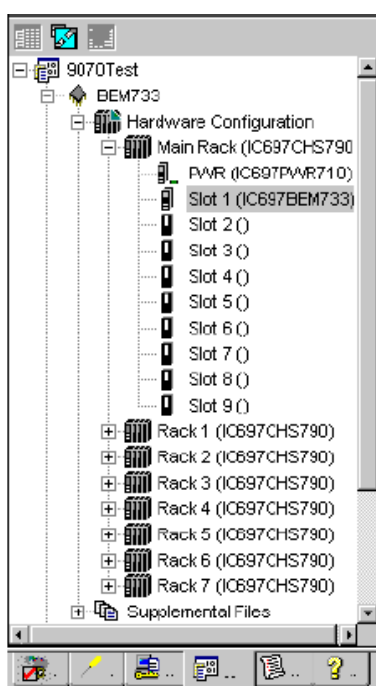
2. 使用参数编辑器对 DSM314 组态

**注意：** DSM314 的详细组态，请参阅 *GFK-1742, 90™-30 系列 Motion Mate DSM314 用户手册*。

## 90™-70 系列远程 I/O 扫描模块



远程 I/O 扫描模块 (IC697BEM733) 是 Genius 总线系统中的一种 Genius 设备。同样地，它作为一个 Genius 设备加入到 GBC 控制器中，在 90™-70 系列的对象中，被 Logic Developer—PLC 视为 GBC 节点下的一个 SBA 节点。



远程 I/O 扫描模块插在 90™-70 系列主机架的 1 号槽，I/O 模块可插在主机架的其余插槽和其它机架的所有插槽上，远程 I/O 扫描模块作为对象出现在浏览窗口的工程标签中，同时也出现在主机架 1 号槽中。

## 设置远程 I/O 扫描模块

1. 把远程 I/O 扫描模块作为一个 Genius 设备增加到 GBC 下。
2. 当出现提示，询问是否增加一个远程 I/O 扫描模块作为新对象时，单击 **Yes**。


Logic Developer—PLC 自动设置一些配置参数使远程 I/O 扫描模块连接到对应的 GBC。

3. 在作为对象的远程 I/O 扫描模块中，设置和作为 Genius 设备的远程 I/O 扫描模块相同的 SBA 值。

现在，无论何时只要你更新远程 I/O 扫描模块上的 I/O 布置，GBC 中的 I/O 布置也会自动更新

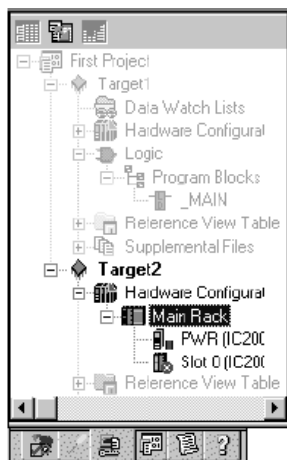
4. 组态远程 I/O 扫描模块的对象属性。
5. 在远程 I/O 扫描模块所在的机架上增加 I/O 模块并组态。
6. 组态远程 I/O 扫描模块的模块属性。
7. 组态远程 I/O 扫描模块的 Genius 总线设备属性。

---

 **想了解更多？** 在帮助索引中，搜索 “IC697BEM733, Remote I/O Scanner”。

---

## Versamax 远程 I/O



Versamax 远程 I/O 由网络接口单元 (NIU) (Ethernet, Genius 或 Profibus) 和一个或多个 I/O 模块组成。如果您的对象类型是远程 I/O, 您能用 Logic Developer—PLC 对它进行硬件组态。一旦组态后, 远程 I/O 就可以被 PLC 或 PC 控制器来控制。Versamax PLC 和远程 I/O 的区别在于远程 I/O 仅仅是拥有网络接口的输入/输出设备, 不象 PLC 那样拥有 CPU。

当进行 Versamax 远程 I/O 硬件组态时, Logic Developer—PLC 能对它增加机架, 组态模块和电源。

在您创建一个带模板的工程时, 远程 I/O 对象通常已经添加了 (见 12 页), 您也可以在以后再增加远程 I/O 对象。

### 从模板中创建一个含有 GE Fanuc 远程 I/O 对象的工程

1. 从 **File** 菜单, 选择 **New Project**。

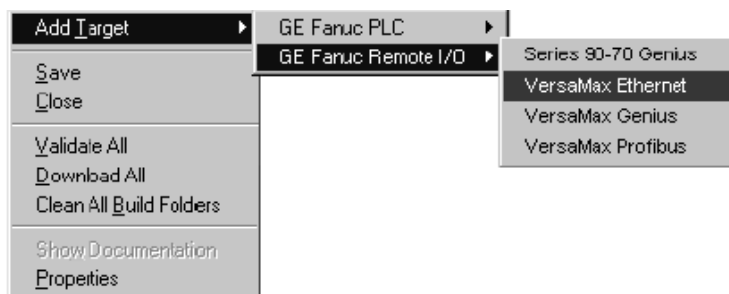
新工程对话框出现

2. 从工程模板列表中, 选择您想要添加的 **GE Fanuc 远程 I/O**。
3. 输入工程名称
4. 单击 **OK**。

一个新的远程 I/O 工程被建立了。




### 在已有的工程中添加 GE Fanuc 远程 I/O 对象

1. 在  浏览窗口的  工程标签中, 右键单击工程节点。
2. 指向 **Add Target**, 再指向 **GE Fanuc Remote I/O**, 选择您想添加的远程 I/O 种类。

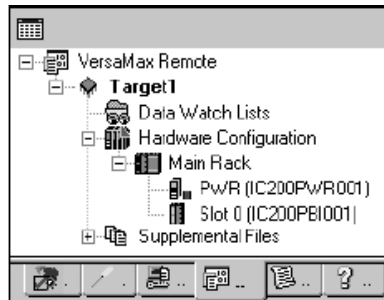


**注意:** **Series 90-70 Genius**选项是指90™-70系列远程I/O扫描模块 (见57页)。

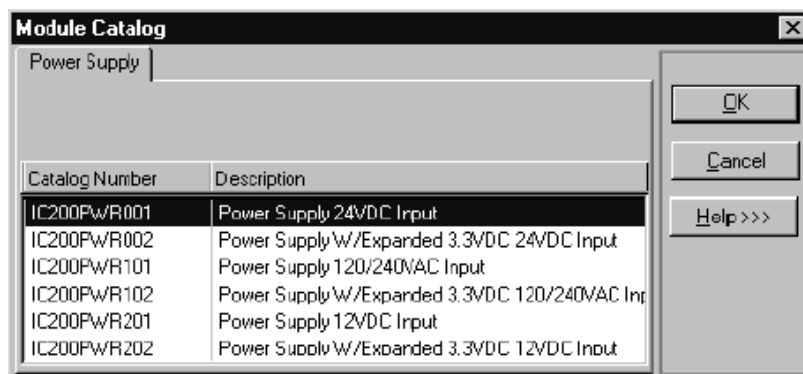
## 在远程 I/O 组态中替换电源

1. 在  浏览窗口的  工程标签中，展开远程 I/O 的  硬件组态节点。

浏览窗口的显示如下：



2. 右键单击  PWR 节点并选择 **替换模块 Replace Module**。



模块目录对话框出现。

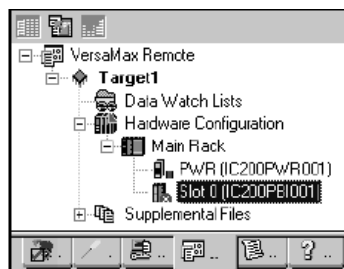
3. 从列表中，选择您想要组态的电源模块类型。


## 为您的 Versamax 远程 I/O 新增一个底座

1. 在  浏览窗口的  工程标签中，选择  远程 I/O 对象。

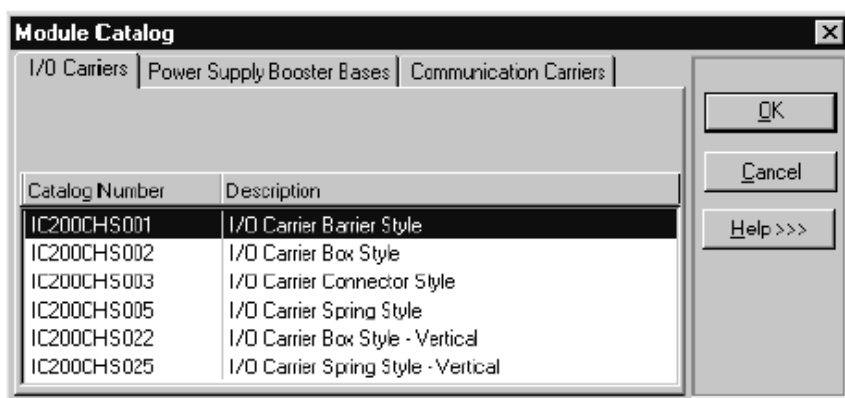
2. 展开硬件组态和主机架节点。

浏览窗口的显示如下：



3. 右键单击  **Slot 0** 节点并选择 **Add Carrier/Base**。

模块目录对话框出现。



4. 选择您想要添加的底座类型。

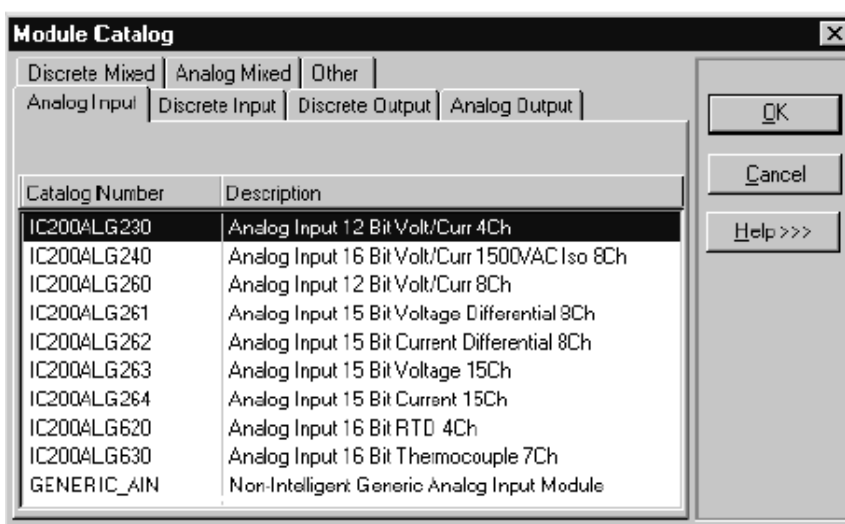
**注意：** 在每个 Versamax 机架中，最多可增加 8 个底座。

5. 单击 **OK**。


## 在底座上添加模块

1. 双击一个空的底座。

模块目录出现。



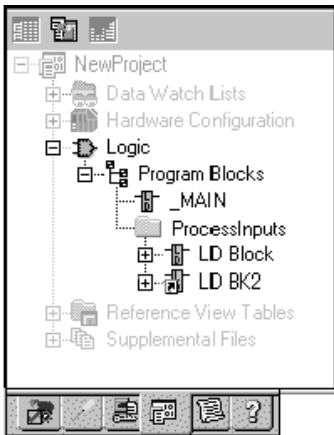
2. 从模块列表中，选择您想要添加的模块类型。

 想了解更多？在帮助索引中，搜索“Hardware Configuration”。



# 5

## 逻辑编辑



**Logic Developer – PLC** 软件（不包括运动编程）支持下列逻辑类型，并能对其编译和下载到 GE Fanuc PLC 对象，GE Fanuc PLC 对象的逻辑文件夹包括：

-  梯形图（LD）块
-  语句表（IL）块
-  C 语言块
-  C 语言程序


上述四项都可以下载到 PLC 运行，每个逻辑块和用户程序是 PLC 执行代码的一个部分，逻辑块可以放在程序块文件夹的用户自定义文件夹下，但 C 程序只能放在主逻辑文件夹下。

每个对象 **target** 必须有一个逻辑块或者叫做 “\_MAIN” 主用户程序，除 S90-70 firmware 版本 6 以上 PLC 外，其余 GE Fanuc PLC 中，程序总是首先执行主程序 “\_MAIN”。

除 “\_MAIN” 块外，其它程序块可定义为时间中断或 I/O 中断，不同的 PLC 支持不同的中断类型，详情请咨询 GE Fanuc 当地技术支持。






**注意：** S90-70 PLC firmware 版本 6 以上，LD 程序能自定义执行方式，也就是 S90-70 PLC 不需要首先执行 \_MAIN LD 主程序块。


---

 **想了解更多？** 在帮助索引中，查阅 “LD Program”。

---

## 创建用户自定义文件夹



1. 在  浏览窗口的  工程标签中，展开用户要组态对象 **target** 中的  逻辑文件夹 **Logic folder**。
2. 右击  程序块文件夹，指向“New”并选择文件夹 **Folder**。  
一个新的用户定义的文件夹  出现。
3. 可以输入文件夹名，文件夹名在程序文件夹中必须是唯一的。

 **想了解更多?** 在帮助索引中，查阅“PLC”，然后选择“Organizing your PLC logic”。



## 定义逻辑块执行方式

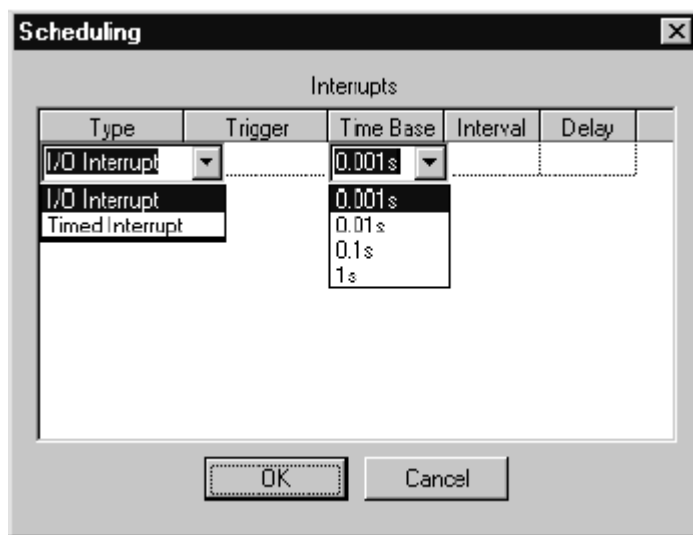
Block Properties	
Name	LDBK1
Description	
Block Type	BLK
Scheduling	...
Parameters	
Lock Settings	
Extra Local Words	128

Inspector


1. 在  浏览窗口的  工程标签中，右击已建立的 LD,C 或 IL 块，选择属性 **Properties**。

逻辑块的属性显示在  属性窗 **Inspector** 中。



2. 在  属性窗 **Inspector** 中，点击时序 **Scheduling** 特性栏，选择  按钮，弹出逻辑块执行方式定义对话框。



3. 组态时序 **Scheduling** 特性栏中各列参数。

 **想了解更多?** 在帮助索引中，查阅“Scheduling”。

## 设定块保护


1. 右击 LD,C 或 IL 块，选择属性 **Properties**，块属性在  属性窗 **Inspector** 中显示。
2. 在  属性窗 **Inspector** 中，展开保护设置 **Lock Setting** 特性。
3. 在块保护特性中，从列表中选择用户需要的保护类型。
4. 如选择密码保护，输入密码并牢记。

## 块文本查找和替换

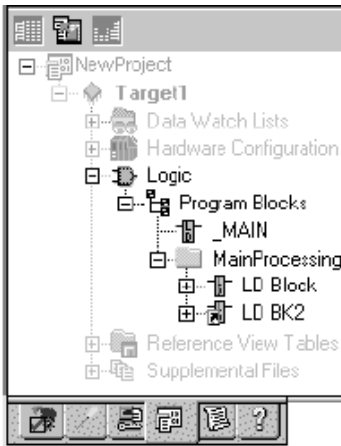
1. 右击打开 LD 或 IL 块。
2. 在主窗口 **搜索 Search** 菜单中，选择查找 **Find** 或替换 **Replace**。弹出对话框。
3. 在 **Find what** 或 **Text to find** 栏中，输入要查找的文本。
4. 若要替换文本时，在 **Replace with** 或 **New text** 栏中，输入要替换的文本。
5. 按要求选择或清除查找/替换选项。
6. 按要求点击 **Find**，**Find next**，**Replace**，**Replace All**，**Close** 或 **Cancel**。

**注意：** Logic Developer-PLC 提供在用户指定的对象 Target 中或其中部分，定义不同的选项实现精确查找和替换的功能。

---

 **想了解更多？** 在帮助索引中，查阅“Global”并选择“Conducting a Global Search”帮助主题。

---



## 梯形图 LD 编辑器

梯形图 LD(Ladder Diagram)编辑器用于创建梯形图语言的程序。它以梯形逻辑图显示 PLC 程序执行过程。

LD 的区段对应继电器的梯级 (rung)，一个指令和它的参数占有一个或几个区段。

在 LD 编辑器中可离线编辑用户程序和在线监控，编辑用户程序。

LD 编辑器的界面，可以根据每个人的爱好而设定

一个 LD 程序块编辑下载到 PLC 中运行，下表描述 GE FANUC PLC 支持的最大 LD 程序块数量。

CPU 类型	程序块数量	程序块最大数量
VersaMax	64 个子程序块 1 个 _MAIN 块	65
Serials 90-30	64 个子程序块 1 个 _MAIN 块	65
Serials 90-70	255 个子程序块 1 个 _MAIN 块	256
PACSystems™	511 个子程序块 1 个 _MAIN 块	512

### 自定义 LD 编辑器

1. 在 浏览窗口的 可选项 **Option** 标签中，打开 **Editors** 文件夹，选择 **Ladder**。
2. 右击 (**Confirmations, Editing, Font and Colors, View**) 页面，选择属性 **Properties**，已组态的 LD 设置在属性窗 **Inspector** 中显示。
3. 在 属性窗 **Inspector** 中，按用户要求校正设置。在 **Find what** 或 **Text to find** 中，输入要查找的文本。

### 创建 LD block

1. 在 浏览窗口的 工程标签中，打开用户要组态对象 **target** 中的 逻辑文件夹 **logic folder**，右击程序块 ，点击 **New**，选择 **LD block**，创建一个新的 **LD block**。

**注意：**如果新建一个对象 **target** 或用模板建立对象 **target**，缺省添加的第一个块是“\_MAIN”主程序块，子程序缺省名为 LDBK1, LDKK2....。

2. 重新定义块名。

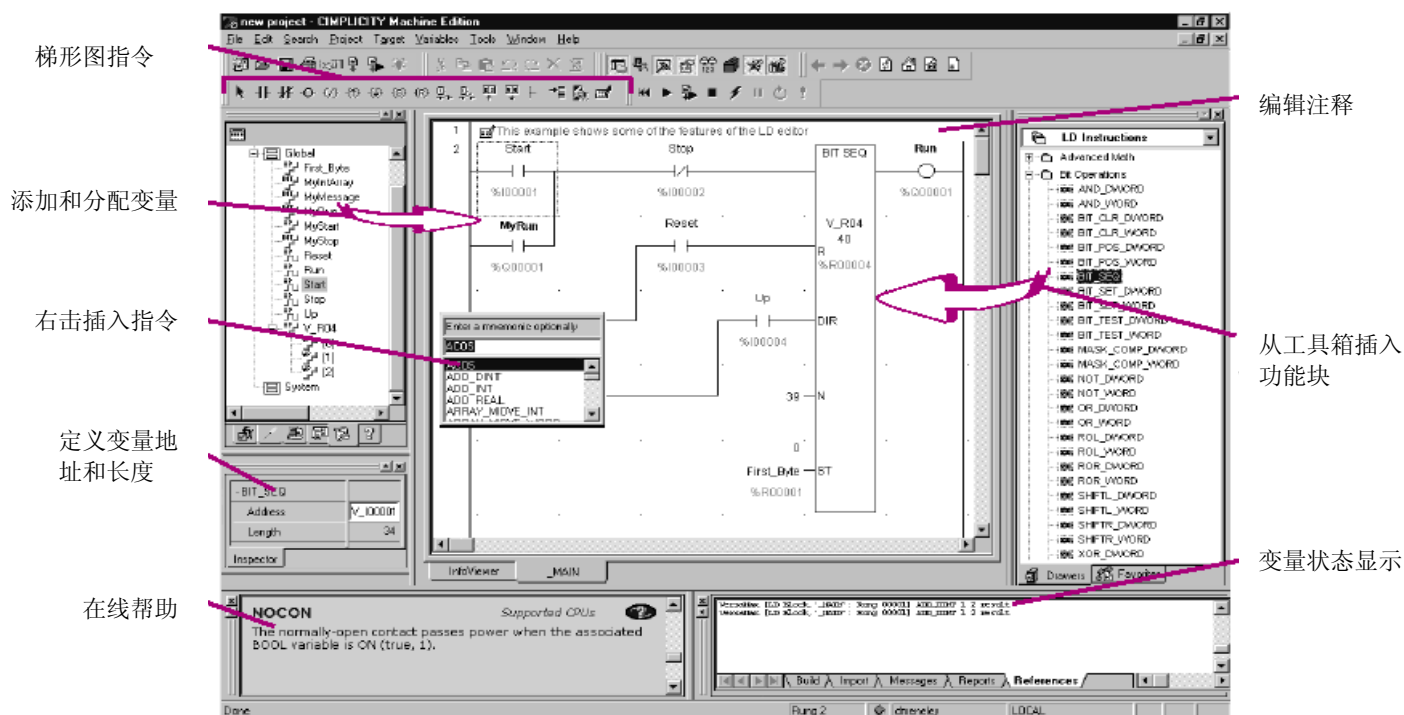
## 编辑 LD block

- 在 浏览窗口的 工程标签中，双击 LD block，在 LD 编辑器打开它。

**注意：**你可以在编辑器中打开多个程序块，选择编辑器下部的按钮可切换程序块显示。

## LD 编辑器离线工作方式

在离线工作方式，LD 编辑器与 PLC 之间未建立实时通讯，一般用户程序开发都在离线方式下进行。离线方式下常用操作见下图。



## 插入指令

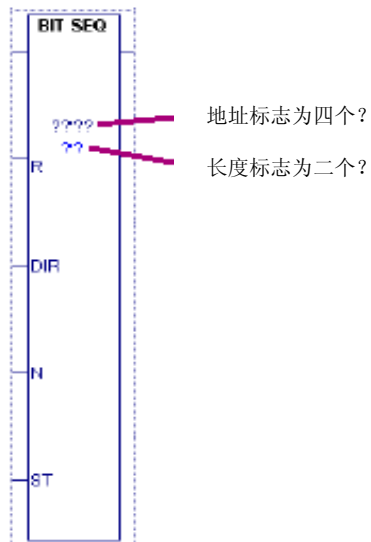
1. 在 LD 编辑器中，右击空区段，选择 **Place Instruction**。  
弹出智能指令表，其中列出了所有可用的指令助记符。
2. 选择一个指令，按回车键。  
**注意：**只有无输入参数的功能块可以放在第一列。

## 组态指令的长度和地址

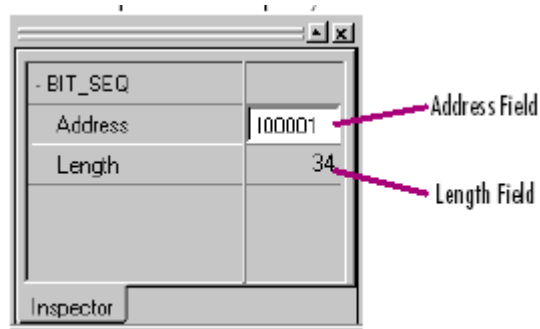
有二种可组态指令地址和长度

### 第一种：

1. 在 LD 编辑器中，插入一个需要组态地址和长度的指令。



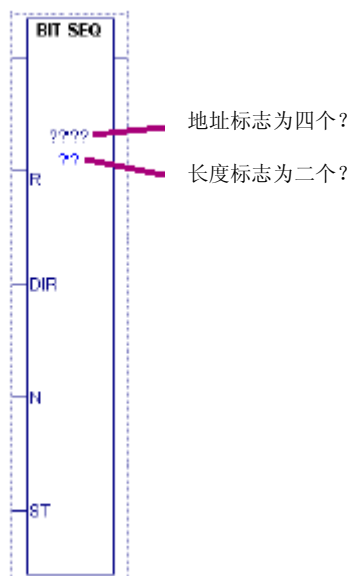
2. 右击指令选择属性 **Properties**，指令属性在属性窗 **Inspector** 中显示。



3. 在地址栏中，输入变量名或参考地址，作为指令占用内部地址的开始地址。  
**注意：**如果输入参考地址，系统自动转换为变量名。
4. 在长度栏中，输入指令占用内部地址的长度。

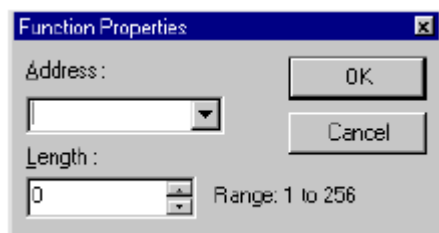
第二种:

1. 在 LD 编辑器中，插入一个需要组态地址和长度的指令。



2. 双击指令或选择指令并按回车。

弹出指令属性对话框。



3. 在地址栏中，输入变量名或参考地址，作为指令占用内部地址的开始位。

**注意：**如果输入参考地址，系统自动转换为变量名。

4. 在长度栏中，输入指令占用内部地址的长度

## 分配指令的输入/输出变量

1. 在 LD 编辑器中，双击指令的输入/输出参数，或选中输入/输出参数后按回车。

弹出一个智能变量列表，提示输入变量名或参考地址。

注意：如果在智能变量列表中，提示输入内存符号，则提供触点或连线。

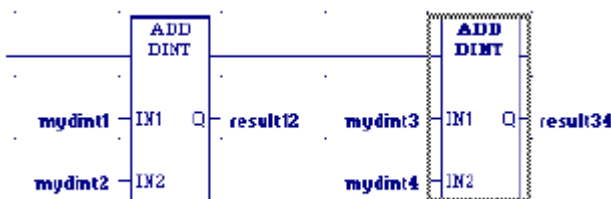
2. 键入或从列表中选择变量名或参考地址。

如果输入参考地址，系统自动转换为变量名。

## 文本方式编辑逻辑


在 LD 编辑器中，右击空区段，弹出智能指令列表，输入用户需要的指令和输入/输出参数，回车，输入的指令和输入/输出参数将在 LD 编辑器中出现。

例如，如果输入 **AD MyDint1 MyDint2 Result12; AD MyDint3 MyDint4 Result34**，以下逻辑将在 LD 编辑器中出现：



一条水平线插入在指令的 Enable 端，输入/输出参数自动填充在正确的位置，输入的 AD 代替“ADD\_DINT”指令，因为它是指令列表中“AD”开始的第一个合法指令。分号分开二个 ADD\_DINT 指令，逻辑关系为“和”。

---

 **想了解更多?** 在帮助索引中，查阅“**Keyboard Method**”。

---

可以在任何文本编辑器中输入梯形逻辑，然后拷贝和粘贴到 LD 编辑器中，或者从 LD 编辑器中拷贝和粘贴梯形逻辑到文本编辑器中。可以拷贝一个 LD 块或者一段梯形逻辑到文本编辑器，并在文本编辑器中编辑逻辑，再拷贝到 LD 编辑器中，实现 LD 编辑器和文本编辑器中逻辑和文本的完全对应。


当梯形逻辑拷贝为文本时，为了方便阅读，会使用详细的格式。但在 LD 编辑器中，以文本方式编辑梯形逻辑，当键盘输入时可以使用速记的形式。

---

## 以文本方式拷贝 LD 块

1. 验证  对象 **target**，保证 LD 块逻辑合法。
2. 在  浏览窗口的  工程标签中，展开对象 **target** 中的  逻辑文件夹 **logic folder**。
3. 展开  程序块文件夹,如果 LD 块在  用户定义的文件夹中，展开用户定义的文件夹。
4. 右击 LD 块，选择拷贝  
LD 块中逻辑拷贝到 Windows 粘贴板。
5. 把 Windows 粘贴板的内容粘贴到文本编辑器。  
以文本的方式粘贴了 LD 逻辑


## 拷贝一段 LD 逻辑到文本编辑器

1. 验证  对象 **target**，保证 LD 块逻辑合法。
2. 在 LD 编辑器中，选择要拷贝的逻辑区域。
3. 右击选中的逻辑，选择拷贝  
逻辑拷贝到 Windows 粘贴板。
4. 把 Windows 粘贴板的内容粘贴到文本编辑器。

## 拷贝一段文本到 LD 编辑器

1. 在文本编辑器中，选择拷贝逻辑文本到 Windows 粘贴板。
2. 在 LD 编辑器中，选择要覆盖的区段或者要粘贴的区段。
3. 右击选择粘贴。  
把 Windows 粘贴板的内容粘贴覆盖到 LD 编辑器中。

---

 **想了解更多?** 在帮助索引中，查阅 “**Copying LD Logic as Text: Overview**”。

---

## LD 编辑器的在线工作方式




在线模式，LD 编辑器与 PLC 之间建立了实时通讯连接，用户程序可在线编程调试。

当在线时，LD 编辑器反映 PLC 中逻辑执行实时数据变化。**GE Fanuc PLC** 支持 **Run Mode Store(RMS)**,即支持在线运行模式下载逻辑。

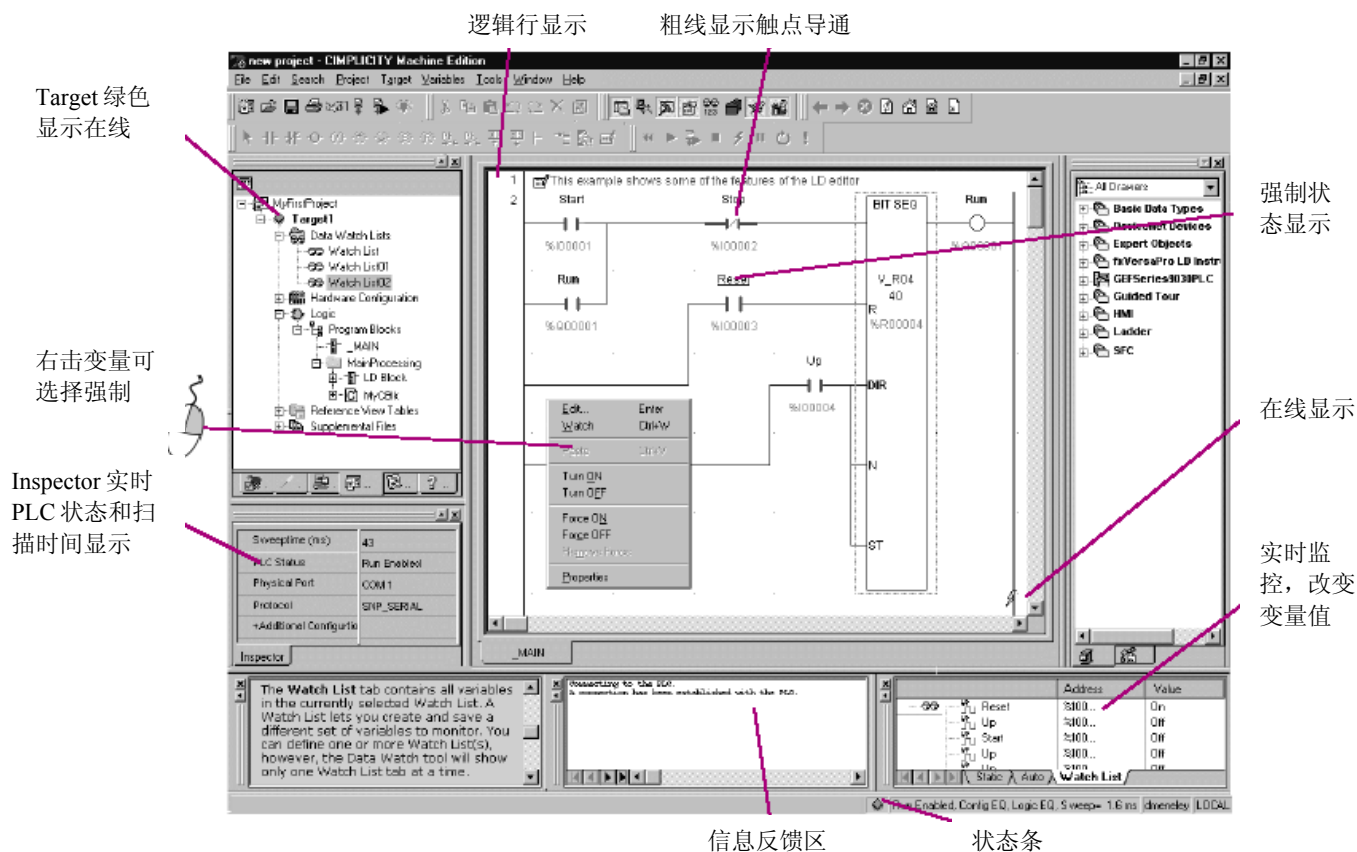
在一个在线运行 PLC 中，有三种在线修改、下载逻辑方式：

- **Online Editing（仅用于 PACSystems™）**：当修改逻辑后，LD 编辑器显示原逻辑和修改后的逻辑，但 PACSystems™在 CPU 中执行原逻辑。用户可以在线测试修改后的逻辑，当执行修改后逻辑时，PACSystems™保留原逻辑的备份。用户可以在线更新 PACSystems™CPU 中程序为修改后程序，或者返回到源程序，或者取消测试继续修改逻辑，或者放弃修改。
- **Word-for-Word Changes(字对字改变)**：在线时逻辑修改不改变原程序大小。例如改变指令的输入/输出参数通常是字对字改变，在线时字对字改变可直接下载到 PLC，不需要下载全部逻辑。
- **Other Logic Changes**：在线时逻辑修改已改变原程序大小。例如增加新逻辑功能或者增加逻辑行。当完成其它逻辑改变，可以在 PLC 运行状态下下载程序。**注意**：某些逻辑类型改变，必须在 PLC Stop(停止)下载程序。

### 设定 PLC 对象在线：

- 在  浏览窗口的  工程标签中，右击  对象 **Target**，选择 **在线 Go Online**。

灰色的对象 **Target** 图标变成绿色，表示对象 **Target** 已经在线。状态条更新显示成在线状态。LD 编辑器中的程序如果与 PLC 中程序相等，则会显示各种在线特性，如下图所示：



## Word-for-Word Changes (字对字改变) :

PACSystems™, 90-30 系列, 90-70 系列, 90 系列 Micro, Versamax PLC 支持在线字对字改变。(Versamax Nano/Micro 不支持)。

在线时只修改逻辑, 而不改变原程序大小就是字对字改变。例如改变逻辑触点和线圈类型, 或者改变功能块的输入/输出参数都是字对字改变。




### 执行 Word-for-Word Changes(字对字改变):

1. 当 PLC 对象 **Target** 在线方式时, 梯形逻辑修改不改变原程序大小。  
弹出确认窗提示是否下载。
2. 选择 **Yes**。  
修改过的逻辑直接写到 PLC。

## 下载改变过的程序到 PLC 对象：

如果改变程序大小，必须下载改变后的逻辑到 PLC，以保证 PLC 逻辑与 LD 编辑器相等。可以在 PLC 运行或者停止模式下下载程序。GE Fanuc PLC 所有系列都支持在运行模式下下载程序（但不是每个系列中每个 PLC 都支持）。

### 执行下载改变过的程序到 PLC 对象：

1. 在对象在线模式时，改变 LD 逻辑。
2. 在  浏览窗口的  工程标签中，右击  对象 Target, 指向 **Online Commands**, 选择 **Write Changes**。

新逻辑立即下载到 PLC，而不需停掉 PLC。如果 LD 逻辑的修改，不同于字对字改变，就会有提示将要下载到正在运行的 PLC，并检验整个对象，而不是仅仅下载修改过的程序块。

### turn on/off (置 1/置 0) 和 force (强制) 变量：

- 在 LD 编辑器中，右击梯形逻辑的 BOOL (布尔) 变量，选择 **Force ON** (强制 1)，**Force OFF** (强制 0)，**Turn ON** (置 1)，**Turn OFF** (置 0)。

**注意：**强制变量 ON 或者 OFF 优先级最高，此变量在程序实时运行中的状态不起作用。例如，如果一个变量 Force OFF (强制 0)，但是逻辑执行结果为 1，实际变量状态取 0。

---

 **想了解更多？** 在帮助索引中，查阅 **“Online Editing:an Overview”** 和 **“working online with LD editor: Overview”**

---

## 梯形图功能块:

下表为 **Logic Developer - PLC** 可用的所有功能块，附有\*号的 P (\*P) 表示此功能块只能在 PACSystems™ PLC 上执行。\*70 表示此功能块只能在 90—70 PLC 上执行。单\*表示此功能块只能在 PACSystems™ 和 90—70 PLC 上执行。

 **想了解更多?** 在帮助索引中，查阅“**LD Functions**”。

### Advanced Math

ACOS	COS	LN	SQRT_DINT	TAN
ASIN	EXP	LOG	SQRT_INT	
ATAN	EXPT	SIN	SQRT_REAL	

### Bit Operations

AND_DWORD*	BIT_SET_DWORD*	NOT_WORD	SHIFTL_DWORD*
AND_WORD	BIT_SET_WORD	OR_DWORD*	SHIFTL_WORD
BIT_CLR_DWORD*	BIT_TEST_DWORD*	OR_WORD	SHIFTR_DWORD*
BIT_CLR_WORD	BIT_TEST_WORD	ROL_DWORD*	SHIFTR_WORD
BIT_POS_DWORD*	MASK_COMP_DWORD	ROL_WORD	XOR_DWORD*
BIT_POS_WORD	MASK_COMP_WORD	ROR_DWORD*	XOR_WORD
BIT_SEQ	NOT_DWORD*	ROR_WORD	

### Coils

COIL	NEGCOIL	PTCOIL*P
CONTCOIL	NTCOIL*P	RESETCOIL
NCCOIL	POSCOIL	SETCOIL

### Contacts

CONTCON	LOALR*	NOCON	POSCON*
FAULT*	NCCON	NOFLT*	PTCON*P
HIALR*	NEGCON*	NTCON*P	

**Control**

DO_IO	EXIT_FOR*	PID_ISA	SVC_REQ
DRUM	FOR_LOOP*	SER	
END_FOR*	PID_IND	SUS_IO*	

**Conversions**

BCD4_TO_INT	DINT_TO_BCD8*	INT_TO_REAL	REAL_TO_WORD	UINT_TO_REAL*
BCD4_TO_REAL	DINT_TO_INT*	INT_TO_UINT*	TRUNC_DINT	WORD_TO_REAL
BCD4_TO_UINT*	DINT_TO_REAL	RAD_TO_DEG	TRUNC_INT	
BCD8_TO_DINT*	DINT_TO_UINT*	REAL_TO_DINT	UINT_TO_BCD4*	
BCD8_TO_REAL*	INT_TO_BCD4	REAL_TO_INT	UINT_TO_DINT*	
DEG_TO_RAD	INT_TO_DINT*	REAL_TO_UINT*	UINT_TO_INT*	

**Counters**

DNCTR	UPCTR
-------	-------

**Data Move**

BLK_CLR_WORD	BUS_TS_BYTE* <sup>P</sup>	DATA_INIT_UINT*	SWAP_WORD*
BLKMOV_DINT*	BUS_TS_DWORD* <sup>P</sup>	DATA_INIT_WORD*	VME_CFG_READ* <sup>70</sup>
BLKMOV_DWORD*	BUS_WRT_BYTE* <sup>P</sup>	MOVE_BOOL	VME_CFG_WRITE* <sup>70</sup>
BLKMOV_INT	BUS_WRT_DWORD* <sup>P</sup>	MOVE_DINT*	VME_RD_BYTE* <sup>70</sup>
BLKMOV_REAL	BUS_WRT_WORD* <sup>P</sup>	MOVE_DWORD*	VME_RD_WORD* <sup>70</sup>
BLKMOV_UINT*	COMM_REQ	MOVE_INT	VME_RMW_BYTE* <sup>70</sup>
BLKMOV_WORD	DATA_INIT_ASCII*	MOVE_REAL	VME_RMW_WORD* <sup>70</sup>
BUS_RD_BYTE* <sup>P</sup>	DATA_INIT_COMM*	MOVE_UINT*	VME_TS_BYTE* <sup>70</sup>
BUS_RD_DWORD* <sup>P</sup>	DATA_INIT_DINT*	MOVE_WORD	VME_TS_WORD* <sup>70</sup>
BUS_RD_WORD* <sup>P</sup>	DATA_INIT_DLAN* <sup>70</sup>	SHFR_BIT	VME_WRT_BYTE* <sup>70</sup>
BUS_RMW_BYTE* <sup>P</sup>	DATA_INIT_DWORD*	SHFR_DWORD*	VME_WRT_WORD* <sup>70</sup>
BUS_RMW_DWORD* <sup>P</sup>	DATA_INIT_INT*	SHFR_WORD	
BUS_RMW_WORD* <sup>P</sup>	DATA_INIT_REAL*	SWAP_DWORD*	

## Data Table

ARRAY_MOVE_BOOL	FIFO_WRT_WORD*	SEARCH_GE_UINT*	SEARCH_NE_DINT
ARRAY_MOVE_BYTE	LIFO_RD_DINT*	SEARCH_GE_WORD	SEARCH_NE_DWORD*
ARRAY_MOVE_DINT	LIFO_RD_DWORD*	SEARCH_GT_BYTE	SEARCH_NE_INT
ARRAY_MOVE_DWORD*	LIFO_RD_INT*	SEARCH_GT_DINT	SEARCH_NE_UINT*
ARRAY_MOVE_INT	LIFO_RD_UINT*	SEARCH_GT_DWORD*	SEARCH_NE_WORD
ARRAY_MOVE_UINT*	LIFO_RD_WORD*	SEARCH_GT_INT	SORT_INT*
ARRAY_MOVE_WORD	LIFO_WRT_DINT*	SEARCH_GT_UINT*	SORT_DINT*
ARRAY_RANGE_DINT*	LIFO_WRT_DWORD*	SEARCH_GT_WORD	SORT_WORD*
ARRAY_RANGE_DWORD*	LIFO_WRT_INT*	SEARCH_LE_BYTE	TBL_RD_DINT*
ARRAY_RANGE_INT*	LIFO_WRT_UINT*	SEARCH_LE_DINT	TBL_RD_DWORD*
ARRAY_RANGE_UINT*	LIFO_WRT_WORD*	SEARCH_LE_DWORD*	TBL_RD_INT*
ARRAY_RANGE_WORD*	SEARCH_EQ_BYTE	SEARCH_LE_INT	TBL_RD_UINT*
FIFO_RD_DINT*	SEARCH_EQ_DINT	SEARCH_LE_UINT*	TBL_RD_WORD*
FIFO_RD_DWORD*	SEARCH_EQ_DWORD*	SEARCH_LE_WORD	TBL_WRT_DINT*
FIFO_RD_INT*	SEARCH_EQ_INT	SEARCH_LT_BYTE	TBL_WRT_DWORD*
FIFO_RD_UINT*	SEARCH_EQ_UINT*	SEARCH_LT_DINT	TBL_WRT_INT*
FIFO_RD_WORD*	SEARCH_EQ_WORD	SEARCH_LT_DWORD*	TBL_WRT_UINT*
FIFO_WRT_DINT*	SEARCH_GE_BYTE	SEARCH_LT_INT	TBL_WRT_WORD
FIFO_WRT_DWORD*	SEARCH_GE_DINT	SEARCH_LT_UINT*	
FIFO_WRT_INT*	SEARCH_GE_DWORD*	SEARCH_LT_WORD	
FIFO_WRT_UINT*	SEARCH_GE_INT	SEARCH_NE_BYTE	

## Math

ABS_DINT*	ADD_UINT*	MOD_DINT	MUL_REAL	SUB_REAL
ABS_INT*	DIV_DINT	MOD_INT	MUL_UINT*	SUB_UINT*
ABS_REAL*	DIV_INT	MOD_UINT*	SCALE_INT	
ADD_DINT	DIV_MIXED*	MUL_DINT	SCALE_WORD	
ADD_INT	DIV_REAL	MUL_INT	SUB_DINT	
ADD_REAL	DIV_UINT*	MUL_MIXED*	SUB_INT	

**Program Flow**

CALL	ENDMCR	JUMP	LABELN	V_WIRE
COMMENT	ENDMCRN	JUMPN	MCR	
END	H_WIRE	LABEL	MCRN	

**Relational**

CMP_DINT*	EQ_UINT*	GT_REAL	LT_INT	RANGE_DINT
CMP_INT*	GE_DINT	GT_UINT*	LT_REAL	RANGE_DWORD*
CMP_REAL*	GE_INT	LE_DINT	LT_UINT*	RANGE_INT
CMP_UINT*	GE_REAL	LE_INT	NE_DINT	RANGE_UINT*
EQ_DINT	GE_UINT*	LE_REAL	NE_INT	RANGE_WORD
EQ_INT	GT_DINT	LE_UINT*	NE_REAL	
EQ_REAL	GT_INT	LT_DINT	NE_UINT*	

**Timers**

OFDT_HUNDS	ONDTR_HUNDS	TMR_HUNDS
OFDT_SEC*	ONDTR_SEC*	TMR_SEC*
OFDT_TENTHS	ONDTR_TENTHS	TMR_TENTHS
OFDT_THOUS	ONDTR_THOUS	TMR_THOUS



## 语句表编辑器 (IL EDITOR)

**Instruction List (IL)** 语句表是 IEC 61131-3 标准定义的程序语言。语句表类似于对微处理器编程的汇编语言，指令通过语句表程序使用 PLC 内存的累加器来执行。PLC 定义二种累加器：一个用于数字的模拟量累加器和八个用于离散逻辑的布尔量累加器，布尔量累加器支持 8 层布尔量表达式嵌套。IL 编辑器是一种应用标准规则的自由形式的编程器，其外观和功效是可以由用户自定义

**注意：**只有 S90-30, VersaMax PLC, Versamax Nano/Micro 支持 IL 功能。

### 组态累加器：

1. 在 浏览窗口的 工程标签中，右击 程序块文件夹，选择 **属性 Properties**。  
*属性窗 Inspector* 被打开，显示累加器地址属性。
2. 在布尔量起始特性中，为布尔量累加器输入 8 个 PLC 内存的起始地址。结束地址由系统自动计算，地址类型必须是 %T, %M, %Q 中的一种。
3. 在模拟量起始特性中，为模拟量累加器输入 PLC 内存的起始地址。结束地址由系统自动计算，地址类型必须是 %R, %AI, %AQ 中的一种。

### 创建 IL 块：

1. 在 浏览窗口的 工程标签中，右击 程序块文件夹，指向 **New**，并选择 **IL Block**。  
一个缺省名为 “ILBK $n$ ” 的 IL 块添加到文件夹中，这里  $n$  是一个唯一的数字。
2. 按用户需要修改文件名。

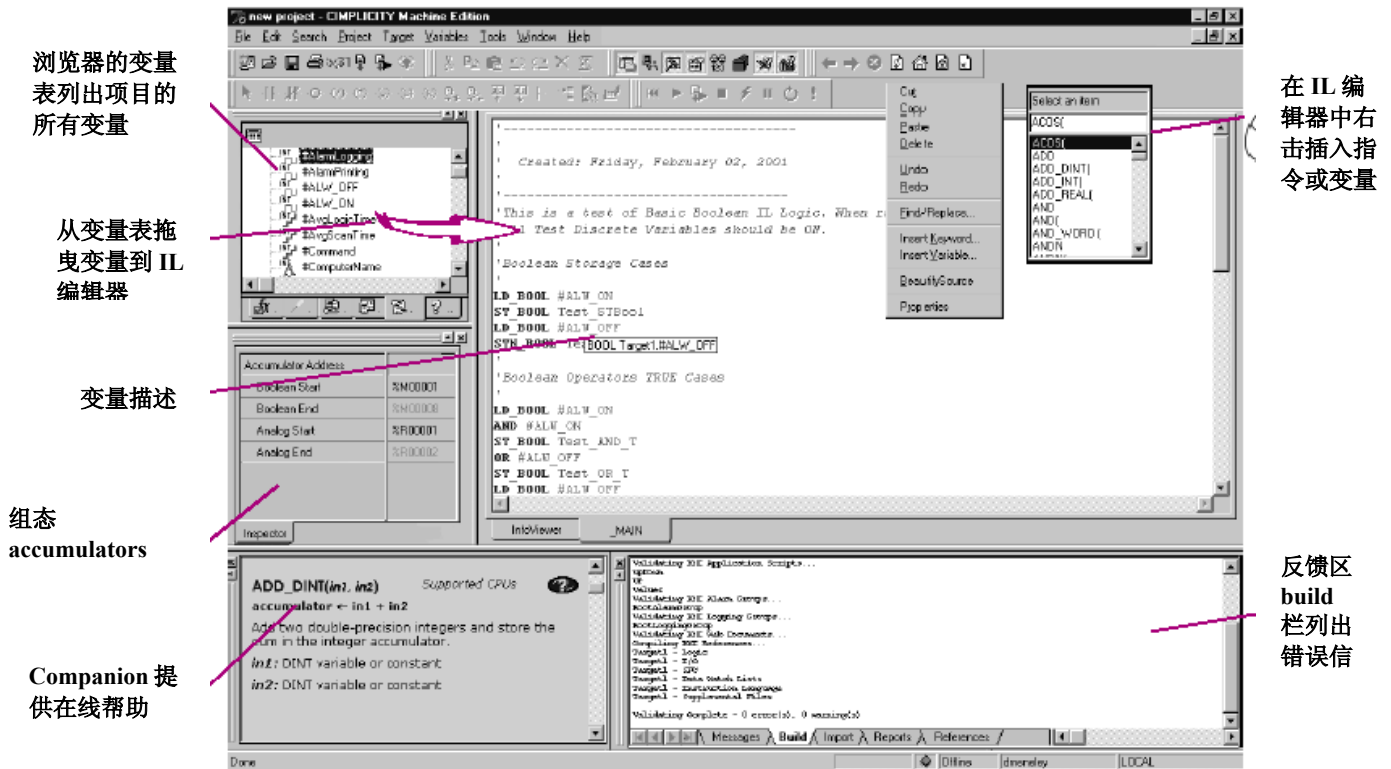
### 打开并编辑 IL 块：

- 在 浏览窗口的 工程标签中，右击 IL 块，选择 **打开 Open**。  
IL 编辑器中显示打开的 IL 块。

**注意：**你可以在 IL 编辑器中打开多个 IL 块供编辑，选择编辑器下部的按钮可切换 IL 块显示。

## IL 编辑器离线工作方式:

在离线工作方式，IL 编辑器与 PLC 之间未建立通讯，一般用户程序开发都在离线方式下进行。和 Machine Edition 的工具配合，离线编辑提供了最大的灵活性，离线方式下常用操作见下图。




## 插入一条指令:

1. 在 IL 编辑器中，右击选择**插入关键词 *Insert Keyword***。  
弹出智能指令列表，列出所有可用的指令助记符。
2. 在指令列表中选择一个指令，按回车键。  
指令被插入你的逻辑中。

## 分配指令参数:


1. 在 IL 编辑器中，右击选择**插入变量 *Insert Variable***。  
弹出智能变量列表，显示所有已被定义的变量。  
键入或从变量列表中选择一个变量名或物理地址，按回车键。  
变量名出现在你的逻辑中。  
**注意:** 如果输入一个物理地址或新变量，必须在变量表中创建此变量。

### 由物理地址创建一个变量:

1. 在  IL 编辑器中, 右击一个物理地址, 选择 **Create" name" as**, 然后选择数据类型。

一个变量被创建, 并给予一个缺省的名字。例如, 如果物理地址是 %R0032, 自动创建的变量缺省名为 R00032。


### 由名字创建一个变量:

1. 在  IL 编辑器中, 右击一个名字, 选择 **Create" name" as**, 然后选择数据类型。


一个带有对应名字的变量被创建。

2. 映射变量到 PLC 内存 (见第 17 页)。


### 移动和拷贝 IL 逻辑:

1. 在  IL 编辑器中, 选择一段逻辑。
2. 如要移动, 按住选择的逻辑, 用鼠标拖曳到目标位置。如要拷贝, 按住选择的逻辑和 Ctrl 键, 用鼠标拖曳到目标位置。当放开鼠标键, 要移动和拷贝的逻辑就放在新的位置。


### 插入一行注释:

1. 在  IL 编辑器中, 单击要插入行注释的地方。
2. 先键入省略号 ('), 再输入注释文本。  
回车结束输入。

### 插入块注释:

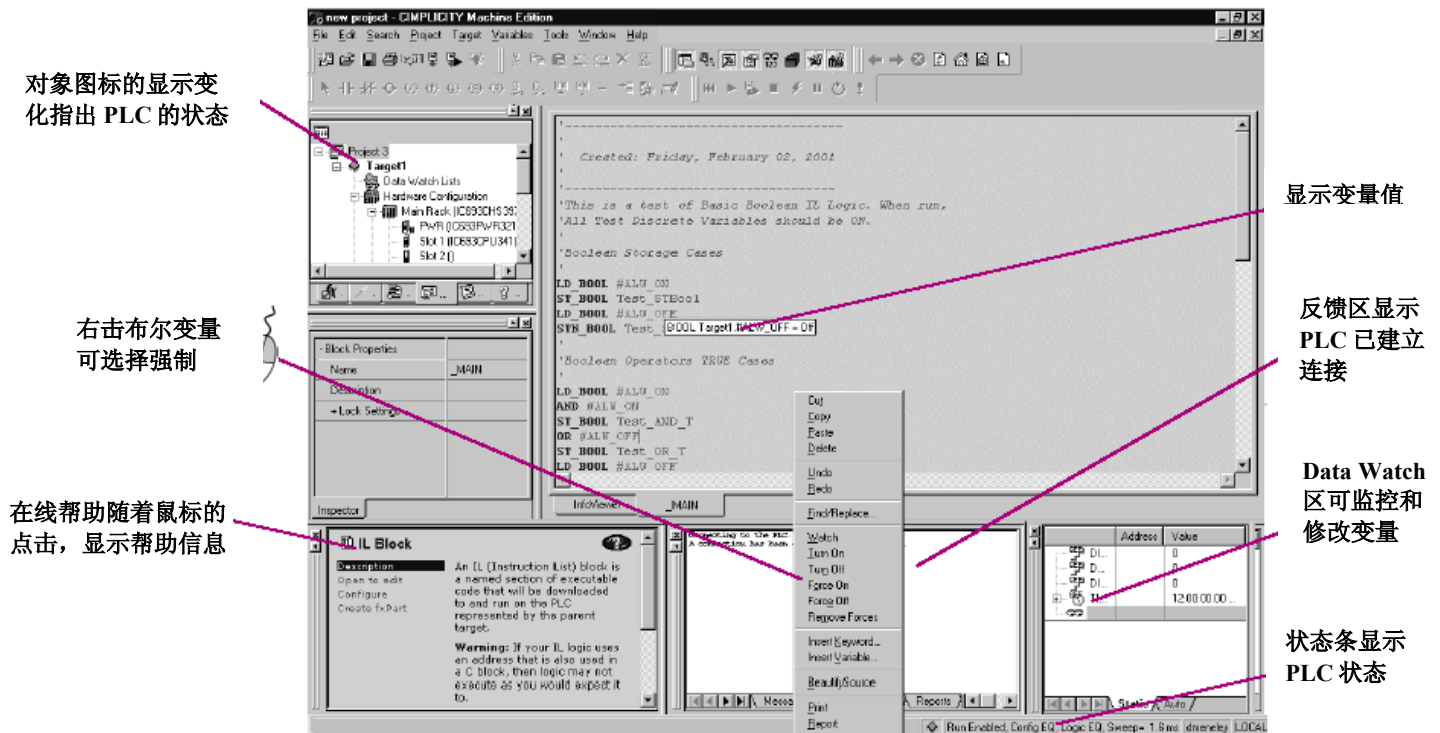
1. 在  IL 编辑器中, 单击要插入块注释的地方。
2. 先键入 (\*, 再输入注释文本。  
块注释可包含任何数字, 并可跨行输入。
3. 键入 \*), 完成块注释。

### 重新布置 IL 语句:

- 在  IL 编辑器中, 右击并选择 **Beautify Source**。  
IL 编辑器中文本按缺省 (缩排) 的方式重新布置。

## IL 编辑器在线工作方式：

在在线工作方式，IL 编辑器可以实时地调试程序，监视数据数值，改变或强制布尔变量。这些允许测试正在执行的逻辑，在执行的工程中触发一个事件或从正在执行的逻辑中去掉一个元素。



### 监控变量：

- 在 IL 编辑器中点击任何地方，移动鼠标指向一个变量。

出现粘附提示，显示变量当前值。但这个显示值不会自动更新，可以移动鼠标到别的地方再返回，重新显示新的变量值。

### 改变布尔变量的状态：

- 在 IL 编辑器中，右击要改变的布尔变量，并选择 **Turn On**（置 1）或者 **Turn Off**（置 0）。

当命令被接受时，PLC 对象的变量状态被改变。这个状态一直保持到 PLC 的逻辑的作用为止。

### 强制布尔变量状态:

- 在 IL 编辑器中, 右击要一个布尔变量, 选择 **Force On** (强制置 1) 或者 **Force Off** (强制置 0)。

不管逻辑执行结果如何, 被强制布尔变量值保持不变。

### 取消布尔变量强制状态:

- 在 IL 编辑器中, 右击一个强制的布尔变量, 选择 **Remove Forces**。




布尔变量值取消强制, 变量值由逻辑控制。

## 写变化到目标 PLC

用户可以在线修改 IL 逻辑, 但必须在 PLC 运行或者在线方式时把修改过的逻辑写到 PLC 中, 以保证逻辑相等。


**注意:** 如果在离线方式, 用户需用要常规的方式下载修改过的逻辑。

### 写变化到目标 PLC 步骤:

- 当目标 PLC 在线时, 修改 IL 逻辑。
- 在  浏览窗口的  工程标签中, 右击  对象 **target**, 指向 **Online Commands**, 并选择 **Write Change to PLC**。

你将被提示确认是否在运行模式存储, 确认后, 修改后的 IL 逻辑下载到 PLC 中, 而不停止 PLC 运行。

---

 **想了解更多?** 在帮助索引中, 查阅 “**Writing Changed Logic to the PLC**”

---

**IL 语句表指令和功能块:**

下表为 **Logic Developer - PLC** 可用的所有语句表指令和功能块，附有 (\*) 表示此功能块只能在 **VersaMax PLC** 中执行。在线帮助显示那种 **PLC** 支持其它的功能块。

**Basic Instructions**

ADD	CAL	GT	MOD	NT	R	SUB
AND	DIV	LE	MUL	OR	RET	XORN
AND_WORD	EQ	LT	NE	ORN	RETC	
ANDN	GE	MCRN	NOT	PT	S	

**Advanced Math**

ACOS	COS	LN	SQRT_INT	TAN
ASIN	EXP	LOG	SQRT_DINT	
ATAN	EXPT	SIN	SQRT_REAL	

**Bit Operations**

AND_WORD	BIT_SEQ	MASK_COMP_DWORD	OR_WORD	SHIFTL_WORD
BIT_CLR_WORD	BIT_SET_WORD	MASK_COMP_WORD	ROL_WORD	
BIT_POS_WORD	BIT_TEST_WORD	NOT_WORD	ROR_WORD	

**Control**

DO_IO	PID_ISA
DRUM	SER
PID_IND	SVC_REQ

**Conversions**

BCD4_TO_INT	DINT_TO_REAL	RAD_TO_DEG	REAL_TO_WORD	WORD_TO_REAL
BCD4_TO_REAL	INT_TO_BCD4	REAL_TO_DINT	TRUNC_DINT	
DEG_TO_RAD	INT_TO_REAL	REAL_TO_INT	TRUNC_INT	

**Counters**

DNCTR	UPCTR
-------	-------

**Data Move**

BLK_CLR_WORD	BLKMOV_DINT	MOVE_BOOL	MOVE_WORD
BLKMOV_INT	COMM_REQ	MOVE_REAL	SHFR_WORD
BLKMOV_REAL	MOVE_INT	SHFR_BIT	

**Data Table**

ARRAY_MOVE_BOOL	SEARCH_EQ_INT	SEARCH_GT_DINT	SEARCH_LT_BYTE	SEARCH_NE_WORD
ARRAY_MOVE_BYTE	SEARCH_EQ_WORD	SEARCH_GT_INT	SEARCH_LT_DINT	
ARRAY_MOVE_DINT	SEARCH_GE_BYTE	SEARCH_GT_WORD	SEARCH_LT_INT	
ARRAY_MOVE_INT	SEARCH_GE_DINT	SEARCH_LE_BYTE	SEARCH_LT_WORD	
ARRAY_MOVE_WORD	SEARCH_GE_INT	SEARCH_LE_DINT	SEARCH_NE_BYTE	
SEARCH_EQ_BYTE	SEARCH_GE_WORD	SEARCH_LE_INT	SEARCH_NE_DINT	
SEARCH_EQ_DINT	SEARCH_GT_BYTE	SEARCH_LE_WORD	SEARCH_NE_INT	

**Math**

ADD_DINT	DIV_DINT	MOD_DINT	MUL_DINT	SCALE_WORD*	SUB_REAL
ADD_INT	DIV_INT	MOD_INT	MUL_REAL	SUB_DINT	
ADD_REAL	DIV_REAL	MUL_INT	SCALE_INT*	SUB_INT	

**Program Flow**

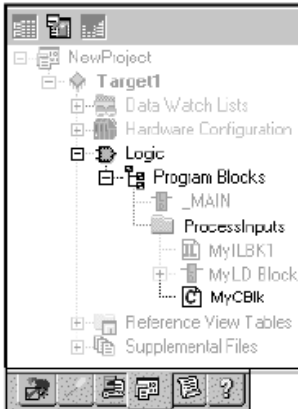
CALL	ENDMCR	JUMP	LABELN	V_WIRE
COMMENT	ENDMCRN	JUMPN	MCR	
END	H_WIRE	LABEL	MCRN	

**Relational**

EQ_DINT	GE_DINT	GT_DINT	LE_DINT	LT_DINT	NE_DINT	RANGE_DINT
EQ_INT	GE_INT	GT_INT	LE_INT	LT_INT	NE_INT	RANGE_INT
EQ_REAL	GE_REAL	GT_REAL	LE_REAL	LT_REAL	NE_REAL	RANGE_WORD

**Timers**

OFDT_HUNDS	ONDTR_HUNDS	TMR_HUNDS
OFDT_TENTHS	ONDTR_TENTHS	TMR_TENTHS
OFDT_THOUS	ONDTR_THOUS	TMR_THOUS



Navigator: Project tab

A C Block

## C Block

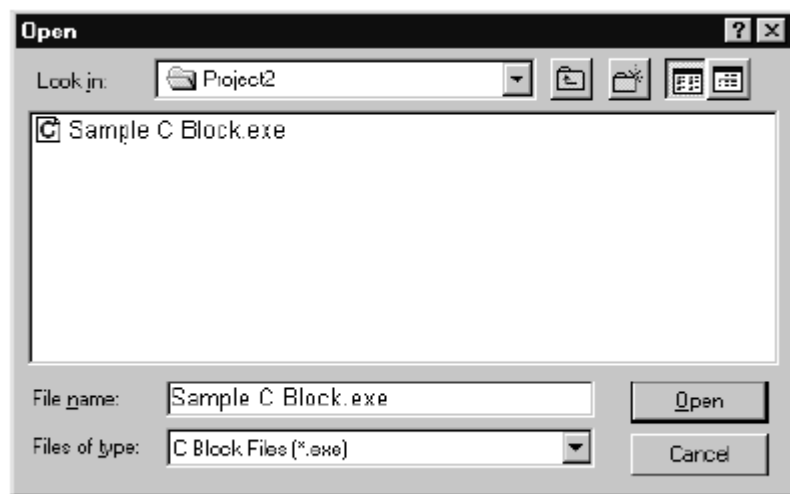
C 语言块是一个独立的用 C 编程语言编写的可执行代码，它被下载到 PLC 中执行。C 语言块由 GE Fanuc's **C Programming Toolkit** (C 语言程序工具包) 在外面创建，然后 **Import** 输入到工程中。编译后的 C 语言块在 PAC 中的扩展名是 **.gefelf**，在 90-30 和 90-70 中的扩展名是 **.exe**。C 语言块作为子程序能被其它程序块 (LD 或 IL) 调用，但是 C 块不能调用其它功能块。

有关开发 C 语言块的更多信息，请查阅 *C programmer's Toolkit for Series 90 PLCs (GFK-0646)* 和 *PACSYSTEMS RX7i C Toolkit User's Guide (GFK-2259)*。

## 对 C 语言块的操作

### 输入一个 C 语言块

1. 在 浏览窗口的 工程标签中，右击 程序块文件夹，并选择 **Add C block**。弹出对话框如下图。



2. 选择要输入的 C 语言块 (扩展名 **.exe** 或者 **.gefelf**)，并点击 **Open** 键。

被选择的 C 语言块添加到 程序文件夹。

**注意：** 也可以把 C 块添加到 用户自定义文件夹。

---






**想了解更多?** 在帮助索引中，选择“C”和“User-Defined Folder”。

---

---


## 设置 C 语言块的参数：

**注意：**只有 PAC 和 90-70 可以按以下步骤设置带参数 C 语言块的参数。

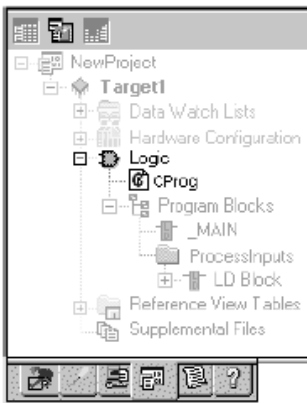
1. 在  浏览窗口的  工程标签中，右击  C 语言块，选择属性 **Properties**。  
在  属性窗中显示这个块的属性。
2. 在属性窗 Inspector 中，选择参数属性并点击 。  
弹出参数对话框。
3. 对每个需要的输入和输出参数，键入名字和描述。

关于每个参数的信息，需要参考 C 语言块的编写说明文档。输入的名字会在 **CALL** 指令中显示。当移动鼠标指向一个 **CALL** 指令时，粘附帮助将显示参数的名字和描述。

---

 **想了解更多？** 在帮助索引中，输入 “**Parameterized**” 并选择 “**C Block**” 主题。

---



Navigator: Project Tab

A C Program

## C 语言程序

**注意：**只有 S90-70 PLC firmware 版本 6.00 或以上的 CPU 才支持 C 语言程序。

C 语言程序是一个独立的用 C 编程语言编写的可执行代码，它被下载到相关的对象 PLC 中执行。C 语言程序由 GE Fanuc's **C Programming Toolkit (C 语言程序工具包)** 在外部创建，然后 **Import** 输入到工程中。

使用 C 程序请查阅 GE Fanuc's *C Programmer's Toolkit for Series 90 PLCs User's Manual* (GFK-0646)。

C 程序可访问除主程序 \_MAIN LD 块中 %P 和其他子程序 LD 块中 %L 地址外的 PLC 所有地址，并且可以通过 C programmer's Toolkit 调用许多 PLC 内嵌的功能块。

C 语言程序不能被其它作为子程序调用，只能通过时序安排来执行。同时 C 程序也不能调用其它块作为它的子程序。

C 语言程序通过设定参数，可以直接访问 PLC 内存。当 C 语言程序开始执行时，为所有参数读取数据并做了备份。如果 C 语言程序是以中断方式或多线程扫描的时间片方式来执行，当它每次返回重新开始时，C 语言程序用原先拷贝备份的数据继续执行。

在 S90-70 对象中，只有当主程序 \_MAIN block 是 LD 块时，才能添加 C 程序。这时的 LD 程序可以作为时序安排的用户程序。最多添加 15 个 C 程序。

如果 C90-70 中有状态逻辑，不管状态逻辑与 LD 逻辑是否共存，状态逻辑被编译成为叫 \_STATE 的 C 语言程序。如果状态逻辑没有与 LD 逻辑共存，\_STATE 仅仅是目标对象支持的 C 语言程序。

C 程序可以与状态逻辑可共存在 S90-70 PLC 中。对于这样一个对象，状态逻辑被编译成为 C 语言程序。如果没有 LD 逻辑，C 程序命名为 \_MAIN，可是如果状态逻辑与 LD 逻辑共存，状态逻辑被编译为叫 \_STATE 的 C 语言程序，并且一个 LD 块必须是 \_MAIN。

## 用户程序

用户程序包括：

- C 语言程序
- LD 程序，如果 \_MAIN block 是 LD 块。

## 用户程序工作方式

### 组态执行时序

1. 在  浏览窗口的  工程标签中，右击  C 语言程序或者 `_MAIN LD` 块，并选择属性。




在  属性窗中显示用户程序的属性。

2. 在属性窗中，展开 **时序 Scheduling** 属性。
3. 选择设定时序模式。
4. 选择设定时序模式中的其它属性。
5. 按 **OK**。

**注意：**如果做了任何设定输入，按 **OK** 后系统会自动检测错误。当出现错误时，用户必须改正错误才能关闭对话框。

### 设置 C 语言程序的参数：

**注意：**必须提供 C 语言程序需要的所有输入和输出参数清单。LD 程序没有用户参数。

1. 在  浏览窗口的  工程标签中，右击  C 程序，并选择属性。


在  属性窗中显示 C 语言程序属性。

2. 在属性窗中，选择参数 **Parameters** 属性，并单击 。

打开参数对话框。

3. 在输入和输出栏，输入最多 8 个输入参数和 8 个输出参数。每个参数有它自己的行，双击参数单元格输入以下数据。
  - **名字：**参数名。
  - **类型：**参数的数据类型。
  - **长度：**输入或输出地址长度。
  - **变量：**同参数相关的第一个数据项。
  - **描述：**（可选项）参数的描述。

---

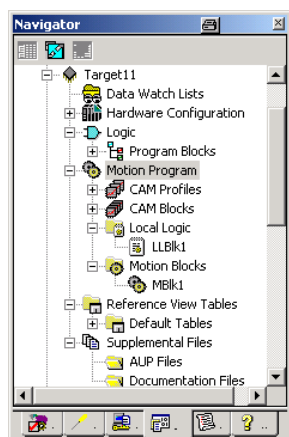
 **想了解更多？** 在帮助索引中，选择“C Program”。

---






# 6

## Motion Programming



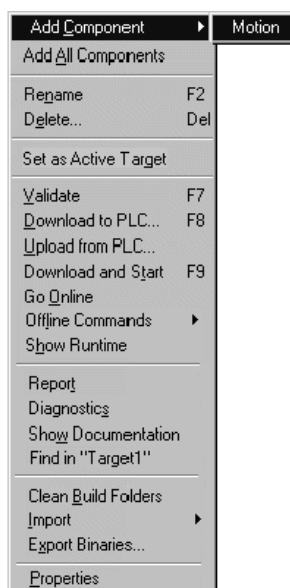
**Logic Developer - PLC** 支持运动控制模块 DSM314 的运动编程。运动控制模块 DSM314 是一个高性能、容易使用、多轴控制模块，它高度集成了 S90-30 PLC 逻辑控制和通讯功能。

DSM314（见 55 页）支持 10 个运动程序，40 个子程序和最大共 1000 条运动程序语句。**Logic Developer - PLC** 能支持下面运动编辑器的运动编程：

-  PLC 运动编辑器
-  本地逻辑编辑器
-  CAM 编辑器

本章概括了基本步骤以使你通过 **Logic Developer – PLC**，使用这三个运动编辑器来创建运动控制程序。

### 增加一个运动选件到一个任务



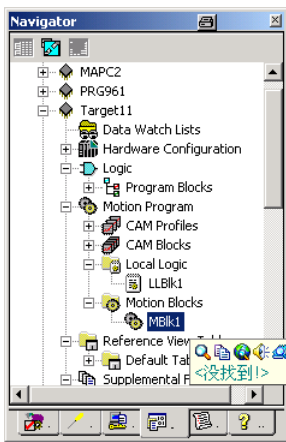
- 在浏览窗口的工程标签中，右击对象 **Target**，指向添加选件 **Add Component**，并选择运动 **Motion**。





一个运动程序节点被加到工程中，包括空的运动块、本地逻辑、CAM 轮廓和 CAM 块文件夹。

## PLC 运动编辑器

**Logic Developer - PLC** 包含一个 PLC 运动编辑器，它能使你为 DSM314 创建 PLC 运动程序块。这个文本格式的编辑器能配置它的外观和行为。注释和空白处不被认作 PLC 运动编程语句。运动编程和本地逻辑有不同的语法。

### 增加一个运动块






1. 在  浏览窗口的  工程标签中，展开  运动程序文件夹。
2. 右击  运动程序块文件，并选择 **新建 New**。

一个以缺省名命名的新的空 PLC 运动程序块加到你的工程中。

3. 按照需要重新命名该块。


### 打开一个运动程序块来编程

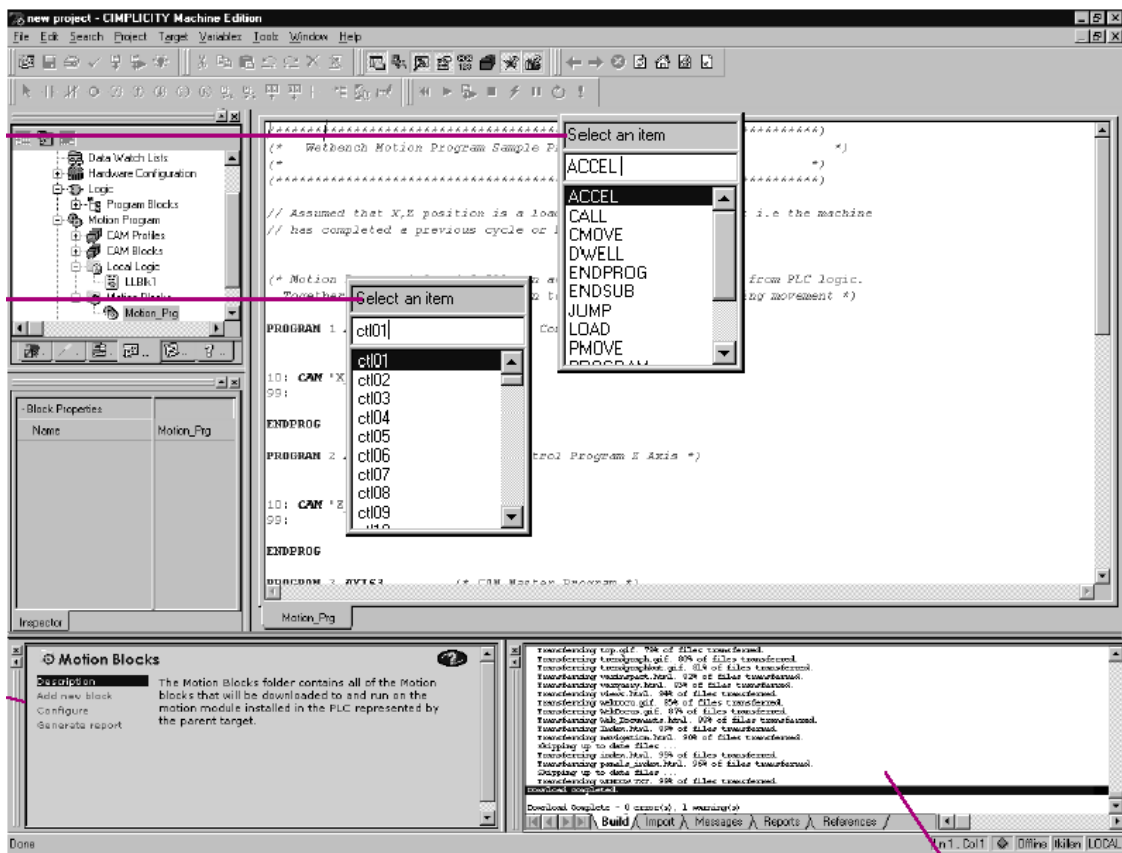
- 在  浏览窗口的  工程标签中，右击  运动程序块节点，并选择 **打开 Open**。

在 PLC 运动编辑器中，一个运动程序块被打开供编辑。

## 用运动编辑器编程

### 插入一个命令

1. 在  运动编辑器中，右击并选择插入关键字 **Insert Keyword**。 .  
一个智能列表出现，显示所有可用的运动命令。
2. 从智能列表中，选择适当的命令并按 **ENTER** 回车键。  
命令被放到运动编辑器中。

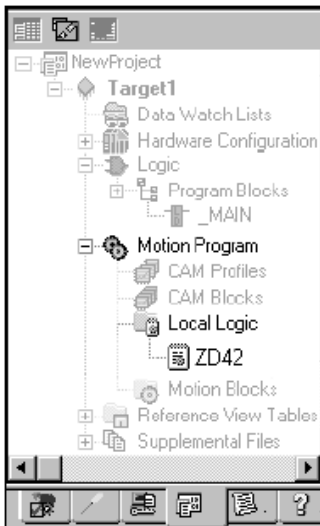


想了解更多? 在帮助索引中，选择“PLC Motion Editor”。

## 运动命令

<b>ACCEL</b>	加速度 <b>ACCEL</b> 指令用来设置顺序运动轴的加速度，并保持这个作用在已给定程序上，直至被改变。注意：如果一个运动指令在加速 <b>ACCEL</b> 以前执行，标签加速度被使用了
<b>BLOCK NUMBER</b>	块数可用作 <b>JUMP</b> 命令的目标，块数必须是唯一的，可以在 1 至 65535 之间选择
<b>CAM</b>	<b>CAM</b> 表达启动 CAM 运动和指定退出条件
<b>CAM-LOAD</b>	<b>CAM-LOAD</b> 为 CAM 从轴加载含有起始位置的参数寄存器。
<b>CAM-PHASE</b>	<b>CAM-PHASE</b> 设置 CAM 命令的相位
<b>CALL</b>	<b>CALL</b> 命令调用另一个程序块作为一个子程序。
<b>CMOVE</b>	<b>CMOVE</b> 命令靠使用指定的位置和加速度方式编辑一个连续的运动
<b>DWELL</b>	<b>DWELL</b> 引起运动在执行下一命令之前停止一指定的时间
<b>ENDPROG</b>	<b>ENDPROG</b> 结束 PLC 运动程序
<b>ENDSUB</b>	<b>ENDSUB</b> 结束 PLC 运动子程序
<b>JUMP</b>	在当前的程序和子程序中跳到某个块号或同步块。跳转可有条件或无条件基于 CTL 位的状态
<b>LOAD</b>	用 32 位双字整数初始化或改变一个参数数据寄存器
<b>PMOVE</b>	<b>PMOVE</b> 命令靠使用指定的位置和加速度方式编辑一个位置运动
<b>PROGRAM</b>	<b>PROGRAM</b> 是在运动程序中第一个表达式， <b>PROGRAM</b> 确定程序号（1-10）和轴的配置。 <b>PROGRAM</b> 不能嵌套
<b>SUBROUTINE</b>	<b>SUBROUTINE</b> 是运动子程序中第一个表达式， <b>SUBROUTINE</b> 确定子程序号（1-40）和轴的配置
<b>SYNCBLOCK</b>	<b>SYNCBLOCK</b> 是一个程序块号的特殊情形。 <b>SYNCBLOCK</b> 仅在多轴程序中应用
<b>VELOC</b>	设置过程的速度。使用顺序运动程序的移动命令并保持这个作用，直到被另一个 <b>VELOC</b> 表达改变为止

## 本地逻辑



一个本地逻辑程序与运动程序是同时运行的，但是本地逻辑独立于 PLC CPU 的扫描之外。这就允许 DSM314 能以最快的速度与界面连接器上的运动输入/输出信号交互，如果这些信号的控制逻辑在运行 PLC 主梯形图程序中处理。

本地逻辑语言利用自由格式，基于文本的回路和包括基本的数学和逻辑结构。灵活的编程环境，使当工作于连接 PLC 逻辑程序和运动程序之间时，本地逻辑编程句法允许你分配逻辑任务的变化到你的运动程序。由于它使用简单、易懂的语法，很容易地精通使用编辑器。

本地逻辑编程语言支持分配、条件的指令，算术、逻辑和关系的指令。本地逻辑提供用户存取运动控制数据、使用一套固定变量的参数、控制和状态位。

- 参数数据：来自于本地逻辑、主 PLC 和运动程序。参数数据类似于程序中变量。
- CTL 位：允许本地逻辑编程或主 PLC 发出信号到运动程序来启动一个事件。
- 运动程序块标号：仅在编写一个特定的运动部分期间，当前块号能用于本地逻辑程序或主 PLC 中，产生一个动作。

### 创建一个本地逻辑块

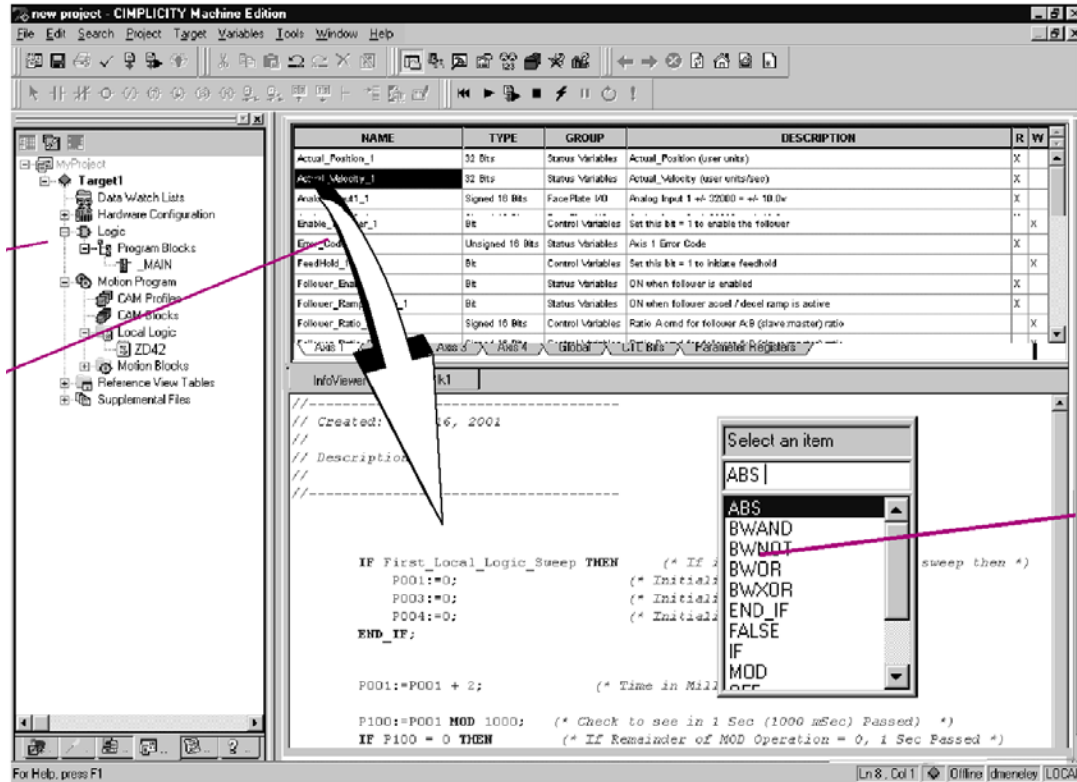
1. 在 浏览窗口的 工程标签中，右击 本地逻辑块和选择 **新建 New**。  
一个以缺省名命名的新的本地逻辑块就被创建了。
2. 按照需要重新命名该本地逻辑块。

### 打开一个本地逻辑块来编程

- 在 浏览窗口的 工程标签中，右击 本地逻辑块并选择 **打开 Open**。  
本地逻辑块在本地逻辑编辑器中打开准备来编辑。

## 用本地逻辑编辑器工作

本地逻辑编辑器有它自己独特的语法来构造本地逻辑程序。



## 插入一个本地逻辑命令

1. 在本地逻辑编辑器中，从菜单中右击并选择插入关键字。

一个智能列表出现，显示所有有效的本地逻辑命令。

2. 在智能列表中选择所需的命令并按回车键。

命令被插入。

**注意：**你可以从本地逻辑变量表中拖曳变量到本地逻辑编辑器里。






## 本地逻辑变量

本地逻辑被设计用来补充 PLC 的逻辑和数学运算能力。解决小的本地逻辑和数学运算式与被控制的运动需要一个紧密的同步。

**Logic Developer - PLC** 包括一个含有本地逻辑变量的表，叫本地逻辑变量表（LLVT），你可以拖曳本地逻辑变量到逻辑编程里。按照下面图表的例子，本地逻辑变量表有几个标签，按种类来管理变量。

NAME	TYPE	GROUP	DESCRIPTION	R	W
Actual_Position_1	32 Bits	Status Variables	Actual_Position (user units)	X	
Actual_Velocity_1	32 Bits	Status Variables	Actual_Velocity (user units/sec)	X	
Analog_Input1_1	Signed 10 Bits	FacePlate I/O	Analog Input 1 +/- 32000 = +/- 10.0v	X	
Analog_Input2_1	Signed 16 Bits	FacePlate I/O	Analog Input 2 +/- 32000 = +/- 10.0v	X	
Axis_OK_1	Bit	Status Variables	ON when axis is ready for commands	X	
Block_1	Unsigned 10 Bits	Status Variables	Motion program block number	X	
Commanded_Position_1	32 Bits	Status Variables	Commanded_Position (user units)	X	
Commanded_Torque_1	32 Bits	Status Variables	Reports digital servo torque in units of 0.01%	X	
Commanded_Velocity_1	32 Bits	Status Variables	Commanded_Velocity (user units/sec)	X	
Digital_Output1_1	Bit	FacePlate I/O	Set this bit = 1 to turn on 24v output OUT1_A		X
Digital_Output3_1	Bit	FacePlate I/O	Set this bit = 1 to turn on 5v output OUT3_A		X
Drive_Enabled_1	Bit	Status Variables	ON when enable output to servo is active	X	
Enable_Follower_1	Bit	Control Variables	Set this bit = 1 to enable the follower		X
Emur_Code_1	Unsigned 16 Bits	Status Variables	Axis 1 Error Code	X	
FeedHold_1	Bit	Control Variables	Set this bit = 1 to initiate feedhold		X
Follower_Enabled_1	Bit	Status Variables	ON when follower is enabled	X	
Follower_Ramp_Active_1	Bit	Status Variables	ON when follower accel / decel ramp is active	X	

### 观看本地逻辑变量表

- 在  浏览窗口的  信息查看标签中，展开  Logic Developer - PLC 图库，然后展开  本地逻辑编辑器书，并双击  本地逻辑变量表。

本地逻辑变量表出现在信息浏览器中。

每个标签的变量和数据是：

Axis 1 轴 1	指定轴 1 的变量
Axis 2 轴 2	指定轴 2 的变量
Axis 3 轴 3	指定轴 3 的变量
Axis 3 轴 4	指定轴 4 的变量
Global 全局	全局数据如模块状态码
CTL 位	DSM 一般的控制/状态位
参数寄存器	DSM 参数数据

表有六列：

名字	本地逻辑程序中用的变量名字
类型	变量的数据类型。如 32-bit 表示这是一个 32 位变量
组	变量所属的组。如面板 I/O 显示面板模块的变量
描述	变量的文字描述。当用户在描述上盘旋鼠标点时，工具顶部显示描述
只读 R	变量的文字描述。当用户在描述上盘旋鼠标点时,工具顶部显示描述
写入 W	变量变量在本地逻辑程序中只能被读入在本地逻辑程序中只能被写入

### 插入一本地逻辑变量

1. 在本地逻辑编辑器中，右击和选择 **插入变量 Insert Variable**。  
一个智能列表出现，提示你选择一个本地逻辑变量名字。
2. 在列表中选择一个变量并按 **回车键 ENTER**。  
变量被插入本地逻辑中。


---

## 本地逻辑命令和操作

本地逻辑能够使你在 DSM 模块中执行基本的逻辑和算数功能。命令只能使用大写字母。

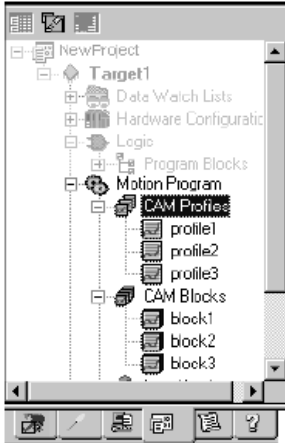
ABS	TRUE	- (减)	<= (小于或等于)
BWAND	FALSE	/ (除)	<> (不等)
BWOR	IF	* (乘)	
BWXOR	THEN	:= (赋值)	
BWNOT	END_IF	> (大于)	
ON	MOD	< (小于)	
OFF	+ (加)	>= (大于等于)	

---

 想了解更多? 在帮助索引中, 选择“Local Logic: an Overview”。

---

## CAM 编辑器



CAM 编辑器是 **Logic Developer - PLC** 运动编程的一个附件，它提供一个创建、编辑和管理电子 CAM 轮廓图的方法。每个 CAM 轮廓图是一个曲线图，它指示一从对主伺服位置刻度的响应。CAM 轮廓图通过名字在相关的运动程序中被引用，并且组合在 CAM 块中。每个块被设法通过 PLC 控制器下载到一指定的运动模块。在硬件配置中为相应的项目工程指定了硬件选项。

CAM 轮廓图被用户定义在图库中，然后组合到 CAM 块里并赋以别名。这就允许你在多 CAM 块中重复使用 CAM 轮廓图。

### 创建一个 CAM 块

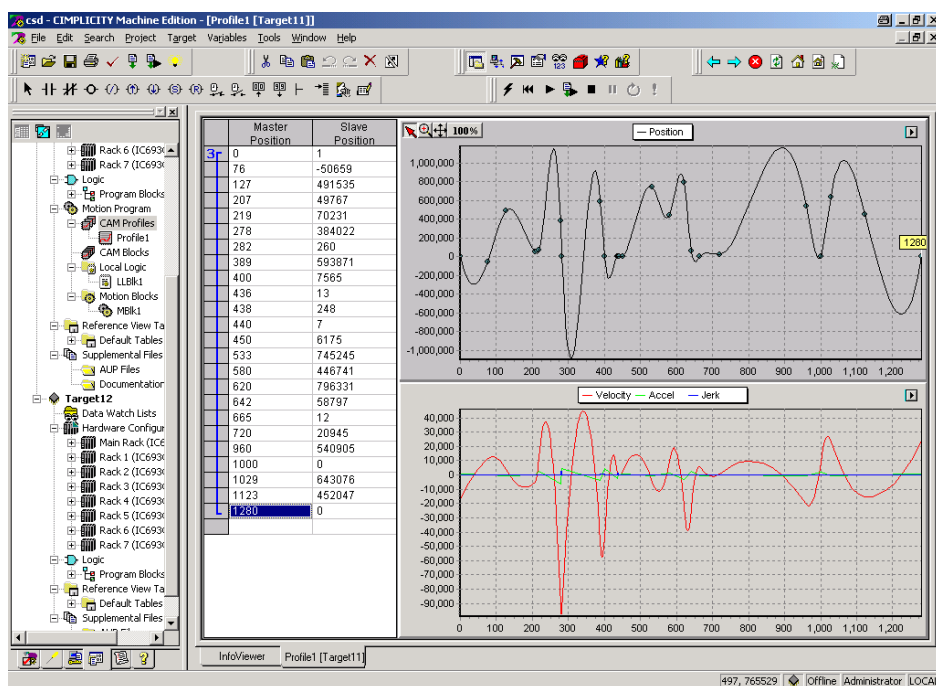
1. 在 浏览窗口的 工程标签中，展开 运动程序文件夹。
2. 右击 CAM 块文件夹，并选择 **新建 New**。  
一个以缺省名命名的新的 CAM 块就被创建了。
3. 按照需要重新命名该 CAM 块。

### 输入 CAM 块

1. 在 浏览窗口的 工程标签中，展开 运动程序文件夹。
2. 右击 CAM 块文件夹，并选择 **从文件输入 Import from folder**。  
打开文件的对话框出现。
3. 浏览你想输入的 CAM 块（后缀为 *.csv* 或 *.text* 的文件）。
4. 点击 **打开 Open** 键。  
被输入的块出现在你的项目工程中。

## 用 CAM 编辑器工作

你能调整你的 CAM 轮廓图的曲线来满足你项目的特殊需要。用 CAM 编辑器，在主/从位置曲线图中定义点来产生轮廓图。临近点的组被指定一区间，每个区间为曲线赋予一个适合的多项式次序（1，2，3），它指定曲线是如何在点之间内插值。



### 创建一个 CAM 轮廓图

- 在浏览窗口的工程标签中，右击 CAM 轮廓文件夹，并选择新建 **New**。

一个以缺省名命名的新的 CAM 轮廓图被加入你的项目中。





### 配置一 CAM 轮廓图

- 在浏览窗口的工程标签中，右击一 CAM 轮廓文件夹，并选择属性。

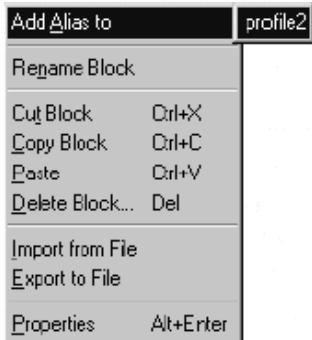
属性窗显示 CAM 文件的属性。




- 在属性窗中，调整 CAM 轮廓属性，指定它的类型和边界条件。

## 编辑—CAM 轮廓图

1. 在  浏览窗口的  工程标签中，打开  运动程序文件夹，并双击  CAM 轮廓文件。  
一个轮廓图形出现在轮廓编辑器中和一各数字出现在轮廓表格中。
2. 在轮廓编辑器或表格中插入和移动点。
3. 在轮廓表格中组合点到区间，并对每个区间分配适合的次序给曲线。


## 加一别名到 CAM 块



- 在  浏览窗口的  工程标签中，右击  CAM 块，指向 **添加别名到 Add Alias to**，然后选选择一个轮廓图。

**注意：**别名必须与 CAM 块中的 CAM 轮廓相符合。为了创建 CAM 块的别名，必须先创建 CAM 轮廓图。

---

 **想了解更多？** 在帮助索引中，选择 “CAM”。

---