



# **Cadence软件应用： 如何在Allegro中执行Skill**

**How to Perform Skills in Allegro Tools**

**Doc Scope : Allegro Skill**  
**Doc Number : SFTEC11005**  
**Author : Daniel Zhong**  
**Create Date : 2011-11-09**  
**Rev : 1.00**

---

## 目录

---

1	Skill简介.....	3
2	在Allegro命令行中加载和调用Skill .....	3
2.1	命令行加载 Skill.....	3
2.2	命令行调用 Skill.....	4
3	在Allegro启动时加载Skill .....	5
3.1	allegro.ilinit 文件.....	5
3.2	编辑 allegro.ilinit.....	7
4	在Allegro菜单栏上调用Skill .....	8
4.1	Allegro 系列工具的菜单文件 (.men) .....	8
4.2	菜单文件搜索路径 .....	10
4.3	定制菜单.....	11

## 1 Skill简介

Skill是Allegro软件二次开发的接口，用此语言编译的文件可以在Allegro中载入和调用。

我们可以通过编写不同的Skill程序文件，在Allegro工具中实现各种原有命令不能实现或不方便实现的功能。Skill语言的语法结构和C++语言类似，工程师如果有一定C语言编程基础，使用Skill语言时就会很方便了。

对于Skill程序的用户，不需要对Skill语言有过多的了解，只需要知道如何在Allegro中加载和调用Skill即可，本文就是介绍了如何加载和调用Skill，可以采用两种方式：

- [在Allegro命令行中加载和调用](#)
- [在Allegro启动自动加载，在Allegro菜单上调用](#)

## 2 在Allegro命令行中加载和调用Skill

本章介绍如何在Allegro命令行中加载和调用Skill，分为以下两个小节：

- [命令行加载Skill](#)
- [命令行调用Skill](#)

### 2.1 命令行加载Skill

以[本站资源下载](#)中的一个skill程序clinecut.il为例，将此文件移动或复制allegro设计文件（.brd）所在目录。在Allegro工具中打开此brd文件，然后在命令行中输入

skill load("clinecut.il")



```
Command >  
Command >  
Command >  
Command >  
Command >  
Command >  
Command > skill load("clinecut.il")
```

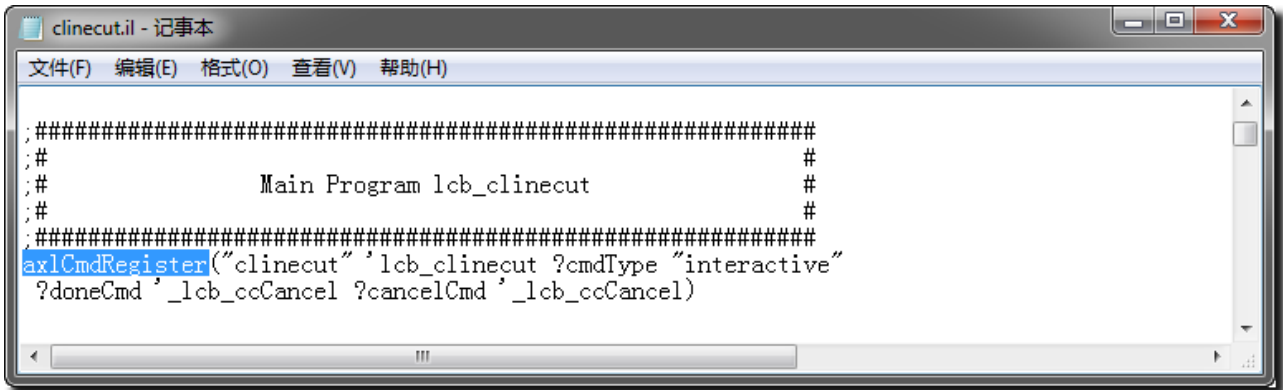
回车，然后命令行会显示skill程序加载的过程，具体文字信息会因skill不同而不同。



```
function _restoreSelSet redefined  
function findAllOFType redefined  
function ashFindAllClines redefined  
function ashFindAllClinesegs redefined  
t  
Command >
```

## 2.2 命令行调用Skill

上一节中已经用skill load命令加载了skill文件，但如何调用此skill呢？我们需要知道调用的命令。调用的命令不等于的skill文件名，它是在编写skill文件时定义的。要查找相关的命令，我们需要用文本编辑器（例如Windows自带的Notepad）打开skill文件clinecut.il，然后查找字符“axlCmdRegister”。



```

#####
;#
;#          Main Program lcb_clinecut          #
;#          #####
axlCmdRegister("clinecut", lcb_clinecut ?cmdType "interactive"
?doneCmd '_lcb_ccCancel ?cancelCmd '_lcb_ccCancel)
#####

```

可以看到“axlCmdRegister”紧接着“(“clinecut””，这就表示此skill的调用命令就是clinecut，刚好和文件同名。这里重申一下，调用命令和skill文件名是两个不相关的东西，可以同名可以不同名。

在命令行中输入

clinecut

回车

命令行会显示skill运行信息，具体文字视不同skill而定。

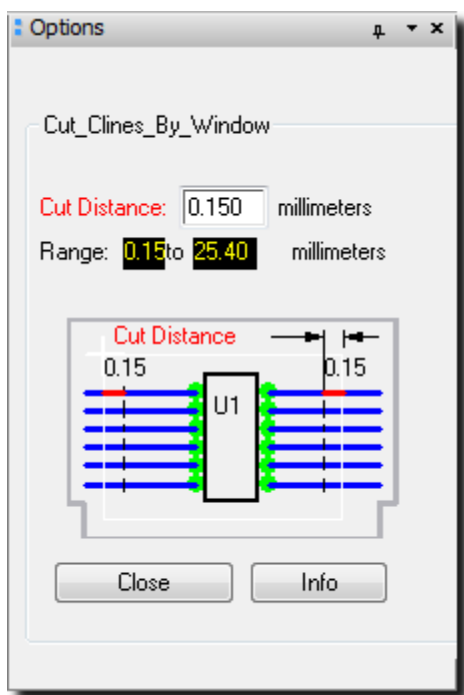


```

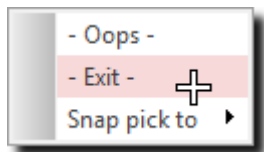
x function ashFindAllClineSegs redefined
t
Command > clinecut
- Enter First Window point -
Loading axlcore.cxt
Command >

```

这时，Allegro就进入到clinecut这一skill命令中，界面可能会因skill而改变，例如本例中，option栏就变成：



如何退出Skill命令，由Skill程序所定义。可以通过按某个按键（大多是“Esc”），可以通过鼠标右键菜单，也可以是调用其他命令时自动退出，等等。本例中，可以通过右键菜单或直接调用其他命令退出。



### 3 在Allegro启动时加载Skill

本章介绍如何在Allegro启动时自动加载Skill，分为以下两个小节：

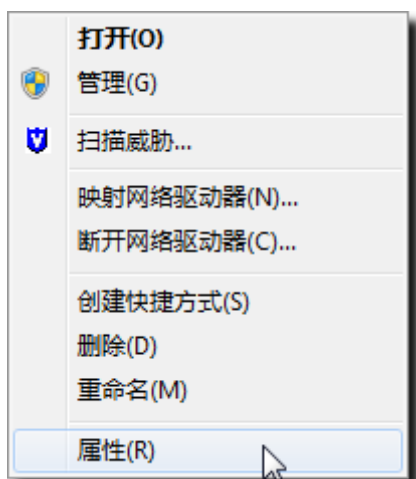
- [allegro.ilinit文件](#)
- [编辑allegro.ilinit](#)

#### 3.1 allegro.ilinit文件

可以从文件名看出allegro.ilinit文件的作用，allegro的il文件的initial文件，即allegro skill程序的初始化文件。此文件出现在Allegro的HOME目录下的pcbenv文件夹中，用于在allegro启动时自动加载skill。我们来看看如何找到这个文件。

首先需确认HOME环境变量的路径。

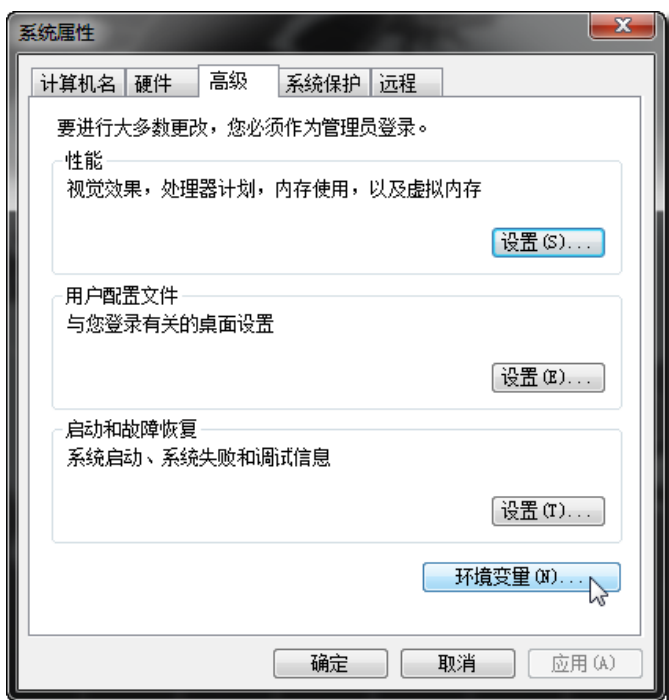
Win7下右击“计算机”，选择下拉菜单中的“属性”。



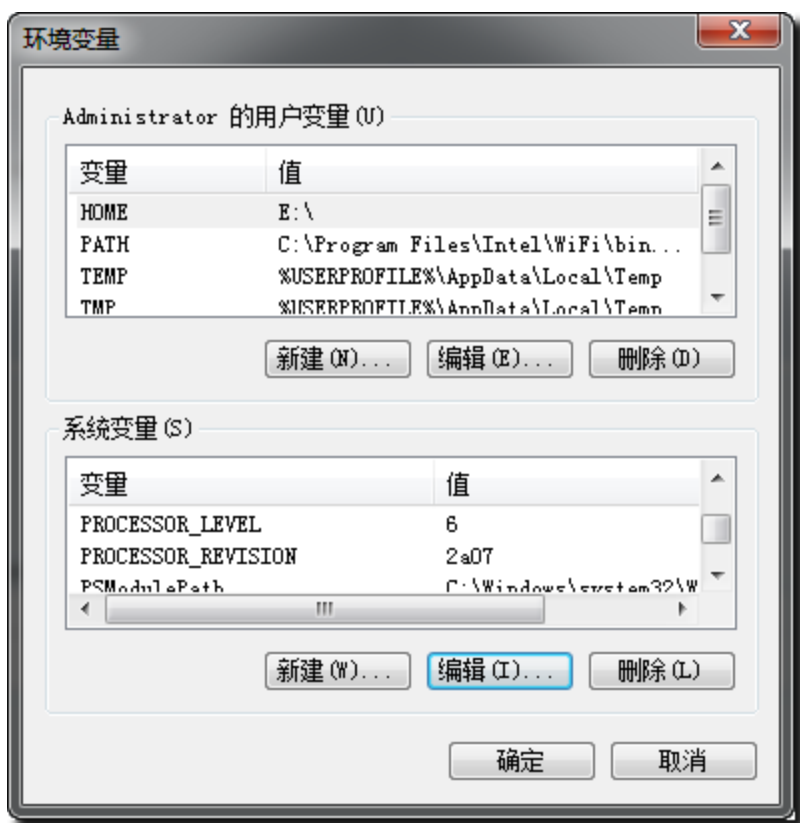
在打开的“控制面板\所有控制面板项\系统”页面左侧选择“高级系统设置”。



在弹出的“系统属性”窗口中选择“高级”栏眉，然后点选“环境变量”按钮。



在弹出的“环境变量”窗口中可以找到用户变量HOME的值是“E:\”，表示HOME目录是E盘根目录。

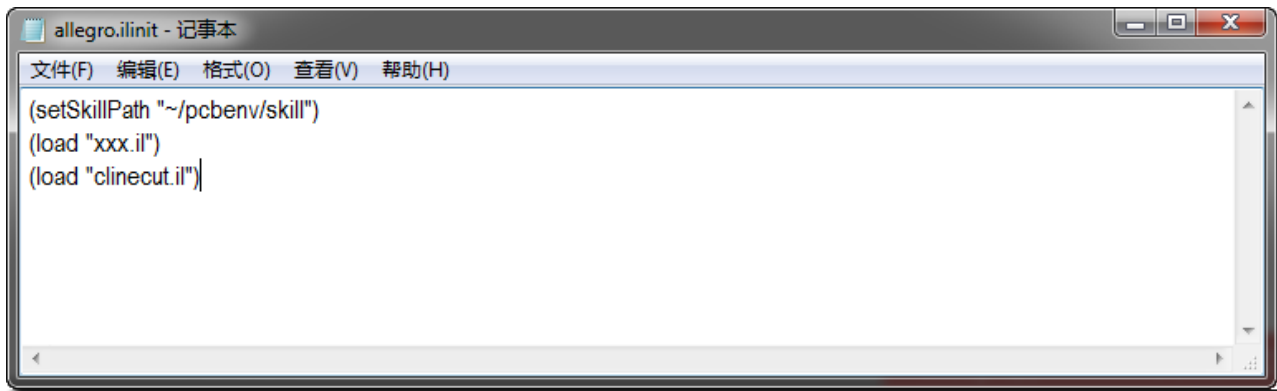


注：HOME环境变量在Allegro软件安装时已经要求设置，但我们也可以在这里编辑更改。

### 3.2 编辑allegro.ilinit

在HOME目录下，我们可以找到pcbenv文件夹，打开pcbenv文件夹，新建一个文本文件并改名为allegro.ilinit，并用文本编辑器（如Notepad）打开，并输入以下字段：

```
(setSkillPath "~/pcbenv/skill")
(load "xxx.il")
(load "clinecut.il")
```



其中，第一行是定义Skill路径为HOME目录/pcbenv/skill文件夹，“~”表示HOME目录。这里也可以在引号中输入绝对路径，例如E:/Skill。二三行分别载入Skill路径中（本案例是E:/pcbenv/skill）的xxx.il和clinecut.il文件。

然后在新打开的Allegro程序中，就可以直接调用相关skill命令了，无需再在命令行中用skill load命令加载。

如果原先pcbenv目录下已经存在allegro.ilinit文件，可以新的skill文件（例如yyy.il）复制或移动到所定义的skill路径，然后编辑allegro.ilinit，新加一行(load "yyy.il")即可

## 4 在Allegro菜单栏上调用Skill

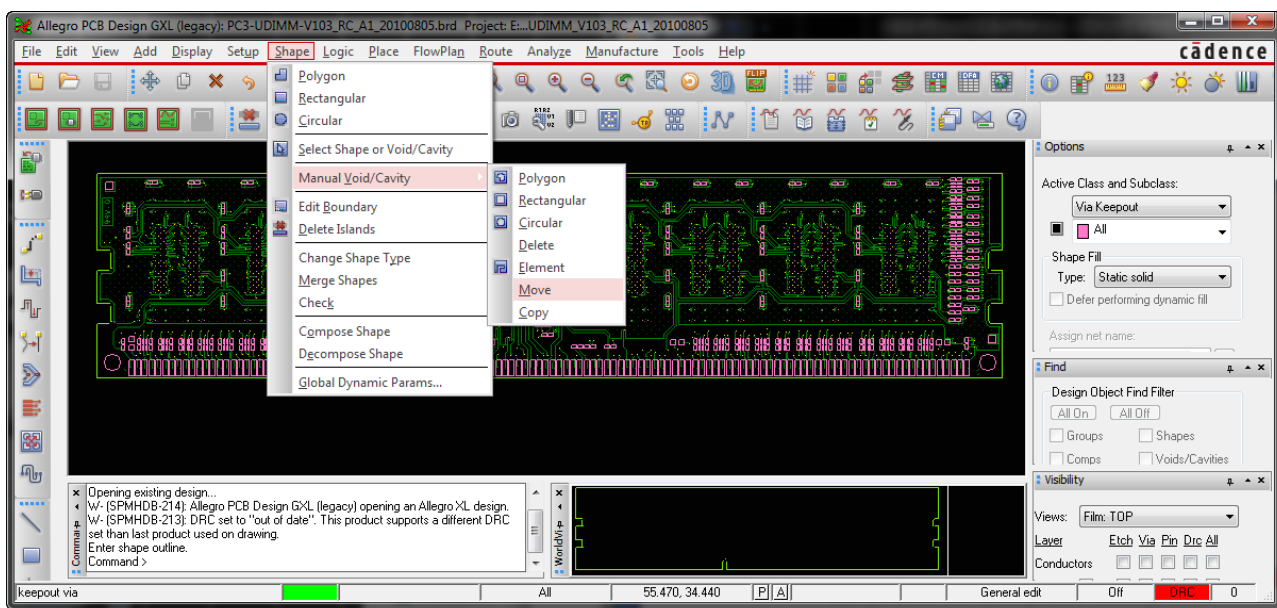
本章介绍如何编辑菜单文件(.men)，使得可以通过Allegro菜单栏上的命令调用Skill，分为以下三个小节：

- [Allegro系列工具的菜单文件\(.men\)](#)
- [菜单文件搜索路径](#)
- [定制菜单](#)

### 4.1 Allegro系列工具的菜单文件(.men)

如果仅仅是通过allegro.ilinit自动加载skill文件，调用时还是需要在命令行中输入skill文件所定义的命令。这样会带来一些不便，特别是当定制的skill程序越来越多时，我们需要记忆的命令也会更多，一旦命令的任何一个字母输入错误，命令将无法执行。另一个方面，由于skill命令存在与skill文件中，如果不做整理，没有直观的方式可以查看，我们也就难以很快的找到某一个skill功能所对应的命令。有什么方法可以解决这个问题吗？答案是我们可以编辑allegro菜单，然后直接在菜单中调用skill命令。

allegro工具使用.men文件来控制菜单栏的所有菜单，我们可以通过编辑此文件添加和删除菜单上的命令。





默认的菜单文件存放目录是<CDSROOT>/share/pcb/text/cuimenu，其中<CDSROOT>表示安装目录，如果是安装在D:/Cadence/SPB\_16.5下，那么完整的目录就是D:/Cadence/SPB\_16.5/share/pcb/text/cuimenu。打开此目录，我们可以看到下面有多个men文件，其中allegro.men文件管理Allegro PCB Editor菜单（无论属于何种产品系列），apd.men文件管理Package Designer菜单，specctraquest.men文件管理PCB SI菜单，等等。我们可以通过用文本编辑器打开.men文件查看其管理的工具。例如，打开allegro.men，可以在靠近文件头处找到如下文字：

```
// Loadable Menu File
// ***** Commoen menu file for 4 tiers of Allegro, expert,
//          designer, pcb and librarian expert
// This does not support the symbol menus or the old-style allegro_librarian
// menu

// ENVIRONMENT VARIABLES USED
// _ALLEGRO_PCB_GXL
// _ALLEGRO_EXPERT
// _ALLEGRO_DESIGNER
// _ALLEGRO_STUDIO - plus librarian xl
// _ALLEGRO_STUDIO_ANL
// _ORCAD_DESIGNER - professsional
// _ORCAD_EDITOR - standard
// _ORCAD_DEMO
// __unix
// enable_ee_beta_functionality
//
// Product options
// _allegro_partition
// _allegro_rfpcb
// _orcad_si
// _allegro_option_highspeed
// _allegro_option_minaturization
//
// _ALLEGRO_GRE_ALL
// _ALLEGRO_GRE_FEAS
// _ALLEGRO_GRE_IFD
```

以上文字描述了使用此菜单文件的工具及系列，其中“//”表示本行为注释。

allegro.men文件在上述文字的下方就正式进入菜单描述了，其主要的关键字有BEGIN、POPUP、END、MENUITEM、#ifdef和#endif等。

其中BEGIN和END表示菜单的开始和结束，在某一个BEGIN开始后，直到遇上同一级别的END，之间的菜单为同一级别。POPUP表示进入下一级菜单，会紧接一个BEGIN，然后直到遇上同一级别的END。对于第一级别的菜单，会显示在Allegro菜单行，以下级别依次按下拉菜单的方式显示。

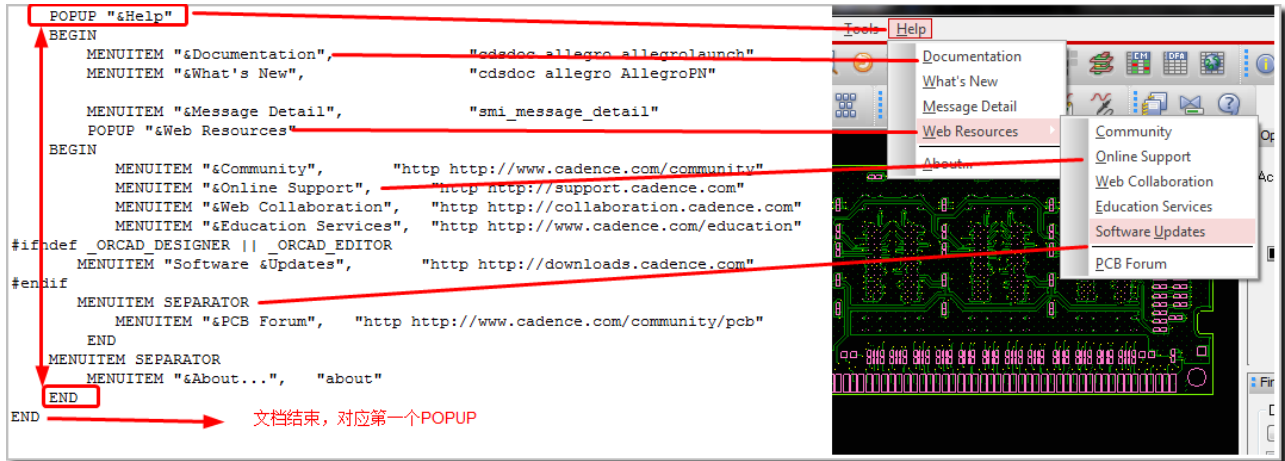
MENITEM定义菜单显示文字和对应的命令，如

```
MENITEM "&New...", "new"
```

表示菜单显示New，对应的命令是new。

#ifdef和#endif用于判断产品的系列，并针对不同产品系列显示不同的菜单。




我们可以从下图中较直观的看出菜单文件的语法和结构。

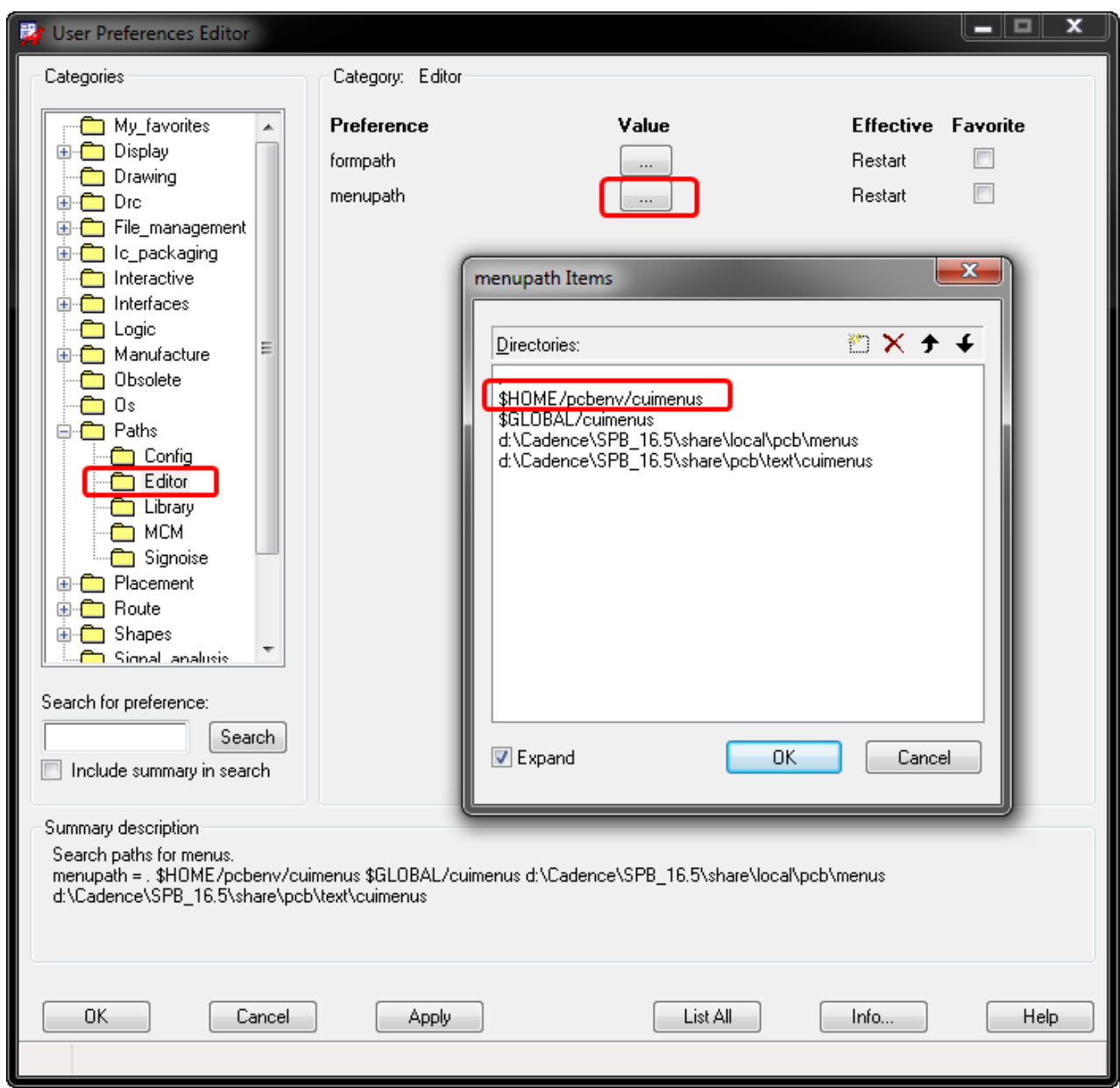


## 4.2 菜单文件搜索路径

虽然直接编辑默认的菜单文件是允许的，但这样做并不是一个很好的方式，Allegro允许我们自行设定多个菜单文件的搜索路径，并按照路径填写的顺序将第一个符合的菜单文件作为软件显示的菜单。

例如，我们自行定义了一个allegro.men的菜单并存于路径A，然后在设计菜单文件路径时添加上路径A并提至最前，Allegro PCB Editor在启动时就会使用路径A中我们自定义的allegro.men菜单文件了。

菜单文件搜索路径可以按照两种方式设定，其一是通过Allegro的User Preferences设定。在Allegro中选择Setup - User Preferences，在弹出的User Preferences Editor中选择Paths - Editor，然后点击出现的menupath右方的按钮 。再在弹出menupath Item窗口中点击右方的New按钮 ，输入\$HOME/pcbenv/cuimenu，并通过Move Up按钮  将此路径上移至最高。



在本案例中，HOME目录是“E:\”，也就是说我们可以将自定义的菜单文件allegro.men存放到“E:\pcbenv\cuimenu”路径下，Allegro就会在启动时优先加载此菜单文件。

### 4.3 定制菜单

将<CDSROOT>/share/pcb/text/cuimenu目录下原始allegro.men文件拷贝到E:\pcbenv\cuimenu目录下，用Uedit32打开，找到最后一行。这里我们不对原始的命令有任何修改，只是在原始菜单后添加新的菜单。

如果我们只想加入clinecut命令到菜单中，可以在最后一行END上加一行：

**MENUITEM "Cline Cut By Window", "clinecut"**

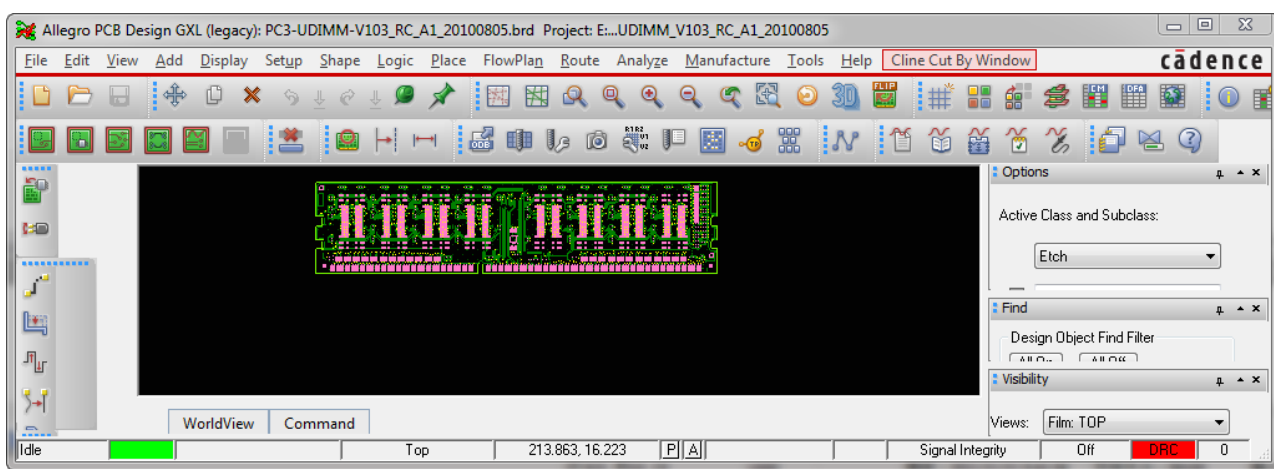
在文本中的效果如下图。

```

        MENUITEM "&PCB Forum",    "http http://www.cadence.com/community/pcb"
    END
    MENUITEM SEPARATOR
        MENUITEM "&About...",    "about"
    END
    MENUITEM "Cline Cut By Window",    "clinecut"
END

```

由于本例中新加的行属于第一级菜单，所以MENUITEM后第一对引号内的文字显示在菜单行中，Allegro菜单的效果如下图（真实图中无红框）。如果点击红框内的字符，会执行第二对引号中的clinecut命令，此命令已经在allegro.ilinit文件中被自动加载了。



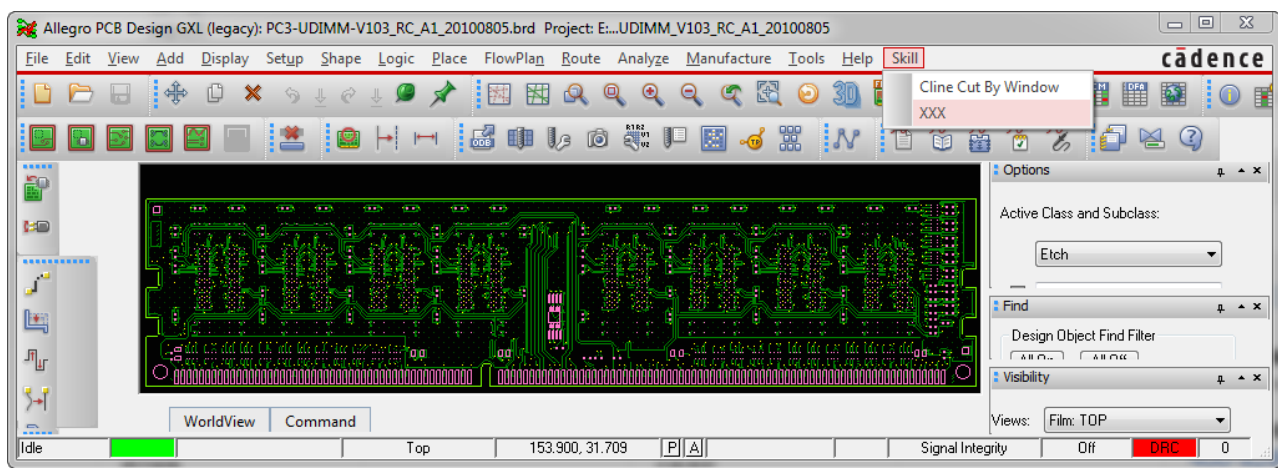
如果希望加入下拉菜单，可以按下图编辑allegro.men。

```

        MENUITEM "&PCB Forum",    "http http://www.cadence.com/community/pcb"
    END
    MENUITEM SEPARATOR
        MENUITEM "&About...",    "about"
    END
    POPUP "Skill"
    BEGIN
        MENUITEM "Cline Cut By Window",    "clinecut"
        MENUITEM "XXX", "xxx"
    END
END

```

效果如下图。



这里，如果选择了菜单中的“XXX”，由于其对应的“xxx”命令没有定义，所以Allegro会在命令行中报错。

按照上述方式，我们可以将所有常用的Skill命令，以及脚本执行命令（目的通常是一键执行设置类的脚本文件），甚至是其他外部程序命令加入到我们定制的菜单中，从而更方便我们的设计。

——[全文完]——

以上技术文档由上海索服科技提供

更多EDA技术文档请访问：<http://www.sofer.cn>