

Inventor iLogic设计自动化技巧及案例分享

撰文 / 欧特克（中国）软件研发有限公司 蔡益江 刘静宜

Inventor iLogic模块提供了对设计自动化的支持。本文通过一些具体的案例，结合客户的实际需求，通过使用iLogic中提供的自动化函数、界面的支持以及良好的扩展性对设计实现自动化，从而使我们更方便的重用设计，提高效率。

上期在介绍Inventor i系列功能时，提到了iLogic的主要功能及其基本使用。本期，我们通过一些案例帮助用户了解iLogic在实际中的使用。

一、案例一：液压阀块及其装配

Inventor 自带的iLogic教程中，有一个液压阀块 (Manifold Block) 的例子，很好地诠释了iLogic的能力和具体应用的方法。

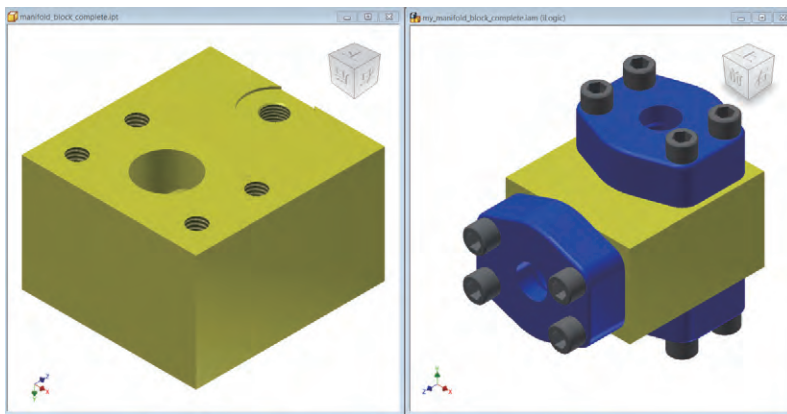


图 1 阀块及阀块装配

首先，分析这个阀块装配，其核心就是阀块（图 1 左）。在阀块的设计中，阀块的尺寸参数，联接类型、油口的孔径类型及尺寸都是设计中需要管理的关键参数。

1. 阀块类型（图 2、图 3）



图 2

说明：图 2 中，选择 Standard 类型时，三个面的油口孔径相同，对应变量参数名分别为：port_b_size, portc_c_size, port_a_size，通过 if...End if 条件判断和赋值语句（=），完成参数的数值定义。

说明：图 3 中，选择“Tee”型联接时，三个面的油口都有效；选择 Elbow（90° 弯头）联接时，下表面的油口（Port_B）开口及螺纹特征被抑制，通过

```

If block = "tee" Then
Feature.IsActive("Port_B") = True
Feature.IsActive("Port_B_Threads") = True
ElseIf block = "elbow" Then
Feature.IsActive("Port_B") = False
Feature.IsActive("Port_B_Threads") = False
End If

```

block	tee
tap_dim	n elbow
Port B Drill Depth	in tee

图 3

Feature.IsActive() 语句实现对特征状况的控制。

2. 油口位置

油口孔径有相应的规格, 在 Inventor 中支持多值列表型变量, 用户可以预先定义供用户选值。当数值较多时, Excel 表格是不错的选择, iLogic 提供了很强的 Excel 表格的操作能力, 实现在 Inventor 中按照设计规则完成查询及选值操作 (图 4)。

port_a_size	in	1.00 in
port_b_size	in	1.00 in
port_c_size	in	0.50
block	文本	1.25
component_type	文本	1.50
nbedding 1		2.00
		2.50
		3.00

图 4 多值列表型变量

在本例中, 采用了第三方嵌入对象的方式, 即 Excel 表格存放在 Inventor 文件内部。iLogic 也支持将 Excel 可以放置在外部, 区别就是在 iLogic 语句中, 给出完整的 Excel 文件的路径 (图 5)。

port_size	y_dist_between_screw	x_dist_between_screw	vert_offset	port_dia	block_depth
0.5	0.69	1.5	1.88	0.5	2
0.75	0.88	1.88	2.25	0.75	2
1	1.03	2.06	2.25	0.94	2
1.25	1.19	2.31	2.75	1.25	2.25
1.5	1.41	2.75	2.75	1.5	2.5
2	1.69	3.06	3	1.94	3
2.5	2	3.5	3.25	2.38	3.5
3	2.44	4.19	3.75	2.88	4

图 5 嵌入表格及表格内容

```

If block = "tee" Then
port = MaxOfMany(port_a_size,port_b_size,port_c_size)
ElseIf block = "elbow" Then
port = MaxOfMany(port_a_size,port_c_size)
End If

i = GoExcel.FindRow("3rd Party:Embedding 1", "Sheet1", "port_size", "=", port)
block_depth = GoExcel.CurrentRowValue("block_depth")
port_c_depth_from_front = GoExcel.CurrentRowValue("port_c_depth_from_front")
block_width = GoExcel.CurrentRowValue("block_width")
port_a_hor_offset = GoExcel.CurrentRowValue("hor_offset")
port_b_hor_offset = GoExcel.CurrentRowValue("hor_offset")
port_c_hor_offset = GoExcel.CurrentRowValue("hor_offset")

```

图 6

说明: 图 6 中, MaxOfMany(): iLogic 提供的一个数学函数, 即比较括号内的数值, 返回最大值, 这里是比较几个油口的孔径, 将最大油口孔径值赋给变量 port。

GoExcel.FindRow(): iLogic 提供的 Excel 查询函数, 这里是查询嵌入的表格, 从其表单 Sheet1, 对应的 Port_size 列, 找到数值等于 Port 的是第几行, 将返回数值赋给 i。

GoExcel.CurrentRowValue(): 配合 FindRow() 返回的行, 读取该行中对应变量名的数值, 本例中, 通过查询表格获得了各个面的油口的位置。

3. 属性定义

在实际使用中, 用户对设计零件的零件代号有特定的要求, 在该例中, 当设计为标准型时, “零件代号” 从嵌入的 Excel 表格中查询获得, 如果为用户自定义的, “零件代号” 设置为 “HomeMade”。

```

If component_type = "standard" Then
i = GoExcel.FindRow("3rd Party:Embedding 1", "Sheet1", "port_size", "=", port_a_size)
iProperties.Value("Project", "Part Number") = GoExcel.CurrentRowValue("model_code")
Else
iProperties.Value("Project", "Part Number") = "HomeMade"
End If

```

图 7

说明: 图 7 中, iProperties.Value (“Project”, “Part Number”), 对应 “Project” (iProperty 的项目页面) 中 “Part Number” (零件代号) 的数值, 从 iLogic 的 iProperty 对应的代码和 iProperty 对应的属性页面, 很容易找到其对应关系 (图 8)。



图 8 iLogic 属性方法及 iProperty 页面

4. 阀块装配中的 iLogic 应用

在本例中, 大家注意到阀块参数的变化是核心, 但是阀块装配中如何直接控制阀块参数呢? 之前, 我们需要借助链接外部参数的方式 (Excel 或 Inventor 文件), 而用 iLogic 方法就变得非常简单。如图 9 所示, 通过 Parameter() 函数, 我们可以直接将数值赋给装配下的零件的变量, 这样就实现了在顶层装配下对底层零部件的控制。

在阀块装配中还有将模型中的属性及参数值返回到 Excel 表格中的应用, 和前面读取 Excel 表格的例子类似,

```

Parameter("manifold_block:1", "block") = block
Parameter("manifold_block:1", "component_type") = component_type
Parameter("manifold_block:1", "port_a_size") = port_a_size
Parameter("manifold_block:1", "port_b_size") = port_b_size
Parameter("manifold_block:1", "port_c_size") = port_c_size

```

图 9

在此不赘述。

二、案例二：用 iLogic 实现货架设计

我们的一位客户设计商店里的货架（图 10），根据用户的要求，规格尺寸（长、宽、高），层数、列数、材料和颜色都不尽相同。



图 10 货架产品示意图

(1) 在利用 iLogic 解决货架模型之前，我们首先建立正确的货架模型结构并确定关键的驱动参数。根据货架的设计，分为三层装配（图 11）。



图 11 货架的产品结构

(2) 基于确定的结构和关键参数创建货架的三维模型。进行零件模型创建时，首先定义如下关键参数，并且命名以便于后续查找和 iLogic 应用（图 12）。

(3) 进行子部件模型创建时，首先添加约束，确定部

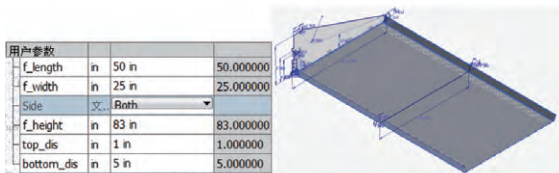


图 12 货架零件的关键参数

件的位置关系；然后创建 iLogic 规则，定义参数关系和参数传递（图 13）。

(4) 创建总装时，首先放置子部件、零件，并且约束和定位；接下来创建用户参数，来定义关键参数（长、宽、高、层数、列数材料和颜色）；使用 iLogic 语句，实现顶

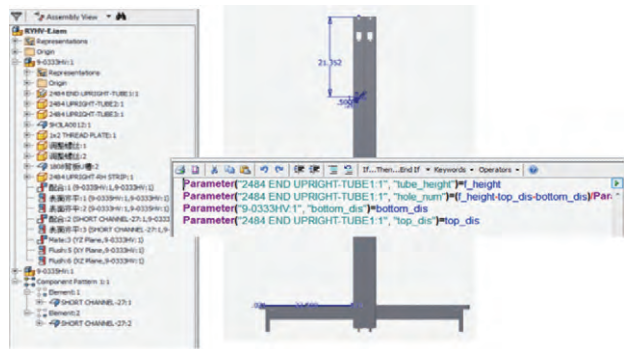


图 13 参数关系

层参数向下游零件的参数传递（图 14）。

用户参数			Parameter("RYHVE.1", "f_length")=Sheet_Length
Side	文	Single	Parameter("RYHVE.1", "f_width")=Sheet_Width
Sheet_Length	in	50 in	Parameter("RYHVE.1", "f_height")=HV_Height
Sheet_Width	in	25 in	Parameter("RYHVE.1", "Side")=Side
HV_Height	in	83 in	Parameter("2448 前底板.2", "f_width")=Sheet_Width
RowNum	ul	4 ul	Parameter("2448 前底板.2", "f_length")=Sheet_Length
Material	文	Steel	Parameter("J0946.1", "f_length")=Sheet_Length
Color	文	As Material	Parameter("J0946.1", "f_width")=Sheet_Width
ColumnNum	ul	3 ul	Parameter("KIKE PLATE-48 9-2719.2", "f_length")=Sheet_Length
X_Brace_...	in	10.031 in	Parameter("2248 轻型层板.1", "f_length")=Sheet_Length
Channel_C...	in	0.000 in	Parameter("2248 轻型层板.1", "f_width")=Sheet_Width
X_Brace_Dis	in	45.000 in	Parameter("X BRACE-WIRE.1", "f_length")=Sheet_Length
			Parameter("assembly sketch for x brace.1", "f_length")=Sheet_Length

图 14 装配下的参数传递

(5) 在本例中，定义了材料和颜色列表变量，在这里，我们使用了 iLogic 的 document 函数直接访问 Inventor API，获取到装配中的每一个零部件（For Each...In...），然后将材料和颜色赋值给相应的零部件（图 15）。

```

MultiValue List("Material") = iProperties.Materials

doc = ThisDoc.Document
Dim occur As ComponentOccurrence
Dim oPart As partdocument
For Each occur In doc.componentDefinition.Occurrences

    Component.Color(occur.Name)=Color

    iProperties.MaterialOfComponent(occur.Name) = Material

Next
    
```

图 15

(6) 创建 iLogic 用户界面，并添加产品配置关键参数和属性控件。iLogic 提供了界面创建工具，方便用户交互操作。图 16 即为根据用户的产品特点，创建的用户参数交互界面。

三、案例三：iLogic 在工程图的应用

经常听到用户希望把视图的比例和质量（去除单位）填写到工程图的标题栏内，以前用户必须通过程序调用 Inventor 的 API 来实现，而借助 iLogic，实现的过程就变得异常简单。

首先我们在工程图的模板中，创建两个自定义属性，比如比例和质量。

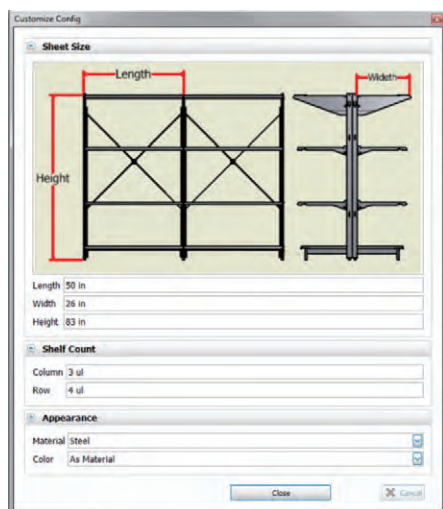


图 16 iLogic 用户交互界面

接下来我们把创建的自定义属性添加到标题栏的定义中(图 17)。



图 17 iLogic 在工程图中的应用

根据 iLogic 提供的工程图函数和属性函数,把指定的工程视图的比例和质量赋给相应的变量。

```

If ActiveSheet.View("视图1").Scale >= 1 Then
    iProperties.Value("Custom", "比例") = ActiveSheet.View("视图1").Scale & ":1"
Else
    iProperties.Value("Custom", "比例") = "1:1" & (1/ActiveSheet.View("视图1").Scale)
End If

doc = ThisDrawing.ModelDocument
iProperties.Value("Custom", "质量") = Round(doc.ComponentDefinition.MassProperties.Mass, 3)
InventorVb.DocumentUpdate()

```

图 18

注意:图 18 中,ActiveSheet.View().Scale 获取当前激活工程图下,指定视图名称的比例。

Round() 是 Inventor 支持的取整函数,这里,是对质量保留 3 位小数。

InventorVB.DocumentUpdate () 常用来当参数变更时,执行更新的操作,类似手工运行管理页面中的“更新”命令。

类似对工程图的操作方法还有很多,比如对放置视图位置的控制、视图高度的控制、图幅、标题栏样式和引出序号等。

四、案例四：使用 iLogic 导出文件

iLogic 支持把 Inventor 文件导出到常用的格式,比如 AutoCAD DWG、DWF、PDF 和 JPG 等。请看如图 19 所示打开的工程图文件,我们可以很方便地使用 iLogic 提供的自动化函数,把它存成其他格式。

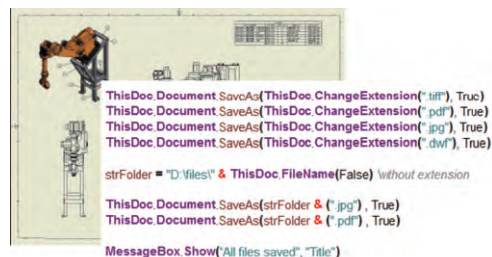


图 19 iLogic 输出文件

注意:Document.SaveAs () 是 iLogic 中提供的文件另存为的函数。

MessageBox.Show () 是 iLogic 提供的信息对话框方式,便于与用户交互。

五、语结

对于用户, iLogic 将 Inventor 在设计重用方面的能力提升到新的高度,归纳如下。

- (1) 设计规则和模型数据相结合的高效方式。
- (2) 更强的 Excel 和模型数据交互操作能力。
- (3) 对 Inventor 的支持面广,从文档类型上涵盖了零件、部件和工程图,从功能上覆盖了零部件参数、物理和外观属性和文档属性、特征控制和零部件控制(包括 iPart, iAssembly)、BOM 属性。
- (4) 支持规则运行条件定义(Trigger),便于自动化处理。
- (5) 它采用类似 VB 的解释型语言,用户易学易上手。
- (6) iLogic 既可以嵌入在文档内(内部规则)和也可以从外部调用(外部规则),方便灵活。
- (7) 支持用户自定义 UI 调用。
- (8) 对 Inventor API 的支持,提供了拓展空间。

限于篇幅,更多的 iLogic 应用案例无法一一在此展开,也欢迎大家和我们交流。如果读者对文中提到的模型和 iLogic 规则感兴趣,可以通过邮件联系我们(mfg.usergroup.china@autodesk.com)。