

第6章 Simulink的应用

6.1 Simulink工作平台的启动

6.2 Simulink仿真原理

6.3 Simulink模块库

6.4 仿真模型的建立和模块参数及属性的设置

6.5 其他应用模块集及Simulink扩展库



6.1 Simulink工作平台

启动Simulink，通常有两种方法：

- (1) 在MATLAB命令窗口中直接输入Simulink命令；
- (2) 在MATLAB工具栏上单击Simulink按钮，如图6.1所示。



这样就可打开了Simulink的Simulink Library Brower(库模块浏览器)，如图6.2所示。

在菜单栏中执行File/New/Model命令，就建立了一个名为untitled的模型窗口，如图6.3所示。在建立了空的模块窗口后，用户可以在此窗口中创建自己需要的Simulink模型。

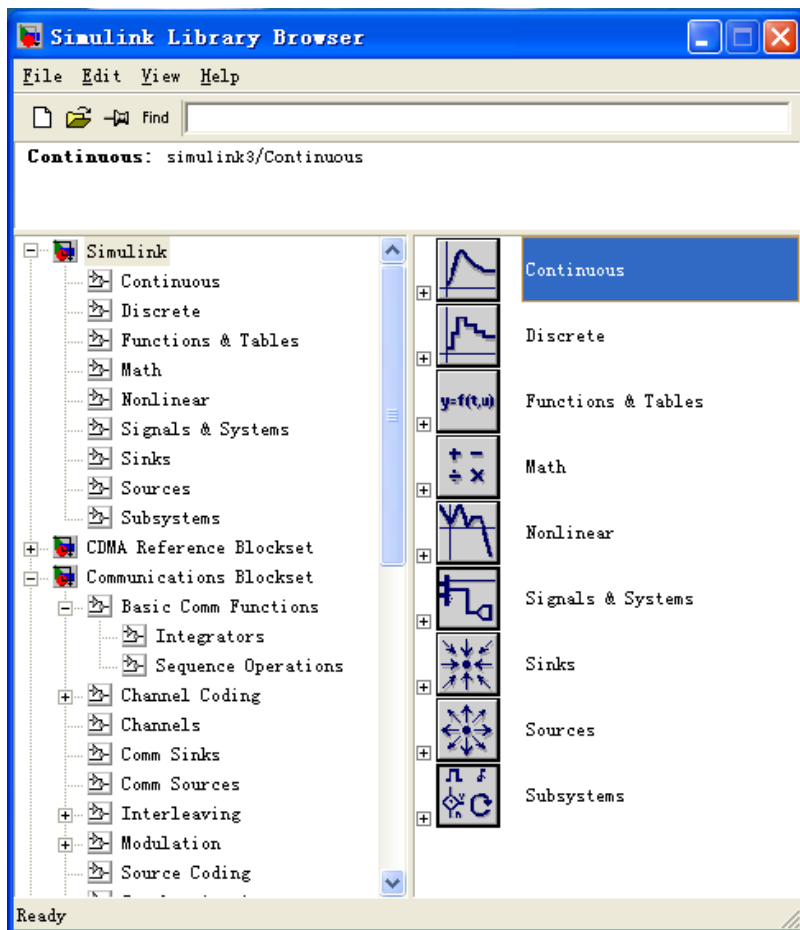


图6.2库模块浏览器

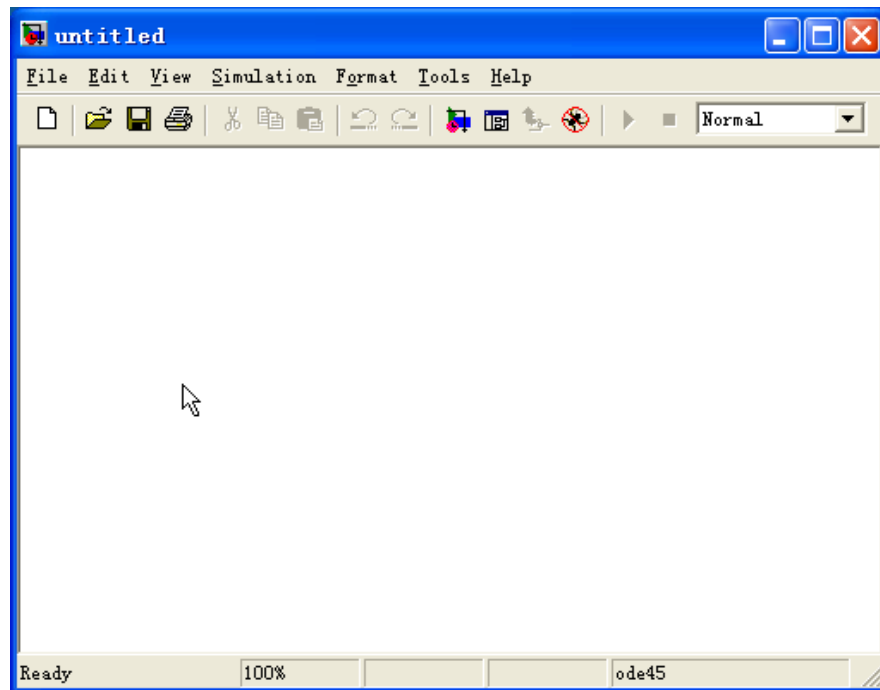
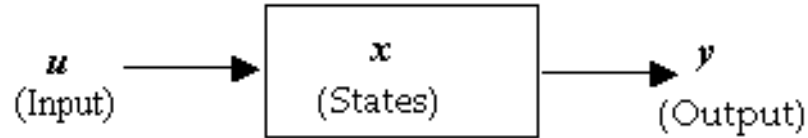


图6.3 新建的空白模块窗口

6.2 Simulink的仿真原理

6.2.1 Simulink仿真模块

通常，Simulink仿真系统包括输入(Input)、状态(states)和输出(Output)三个部分。



- 输入模块**：即信号源模块，包括常数字信号源和用户自定义信号；
- 状态模块**：即被模拟的系统模块，是系统建模的核心和主要部分；
- 输出模块**：即信号显示模块，它能够以图形方式、文件格式进行显示。

注意：在设计一个模型时，必须先确定这三个部分的意义，以及它们之间的联系；Simulink的仿真模型并非一定要完全包括这三个部分，它可以缺少其中一个或者两个；Simulink的状态模块可以是连续的、离散的，或者它们二者的结合。

6.2.2 Simulink仿真过程

1. 初始化阶段

- ① 对模型的参数进行估计，得到它们实际计算的值。
- ② 展开模型各个层次；
- ③ 按照更新的次序对模型进行排序；
- ④ 确定那些显式化的信号属性，并检查每个模块是否能够接受连接它们输入端的信号；
- ⑤ 确定所有非显式的信号采样时间模块的采样时间；
- ⑥ 分配和初始化存储空间，以便存储每个模块的状态和当前值的输出。

2. 模型执行阶段

模型仿真是通过数值积分来进行完成的，计算数值积分可以采用以下两步来进行：

① 按照秩序计算每个模块的积分；

② 根据当前输入和状态来决定状态的微分，得到微分矢量，然后把它返回给解法器，以计算下一个采样点的状态矢量。在每一个时间步中，Simulink依次解决下列问题：

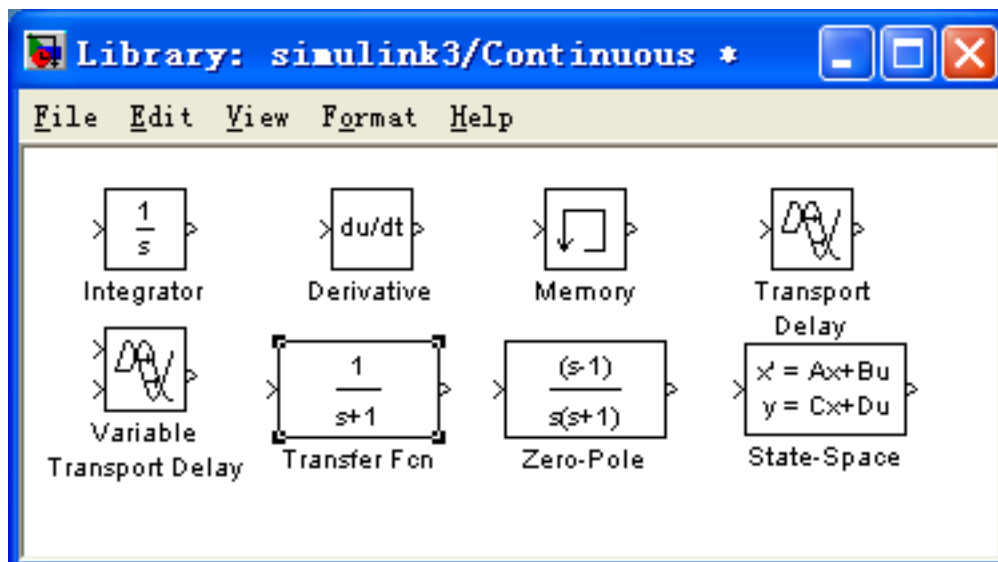
- 按照秩序更新模块的输出；
- 按照秩序更新模块的状态；
- 检查模块连续状态的不连续点；
- 计算下一个仿真时间步的时间。

6.3 Simulink模块库

在库模块浏览器中单击Simulink前面的“+”号，就能够看到Simulink的模块库，如图6.2所示。

6.3.1 连续模块库(Continuous)

在连续模块(Continuous)库中包括了常见的连续模块，这些模块如图所示。



1. 积分模块(Integrator):

功能: 对输入变量进行积分。说明: 模块的输入可以是标量, 也可以是矢量; 输入信号的维数必须与输入信号保持一致。

2. 微分模块(Derivative)

功能: 通过计算差分 $\Delta u / \Delta t$ 近似计算输入变量的微分。

3. 线性状态空间模块(State-Space)

功能: 用于实现以下数学方程描述的系统:

$$\begin{cases} x' = Ax + Bu \\ y = Cx + Du \end{cases}$$

4. 传递函数模块(Transfer Fcn)

功能: 用执行一个线性传递函数。

5. 零极点传递函数模块(Zero-Pole)

功能: 用于建立一个预先指定的零点、极点, 并用延迟算子s表示的连续。

6. 存储器模块(Memory)

功能: 保持输出前一步的输入值。

6. 传输延迟模块(Transport Delay)

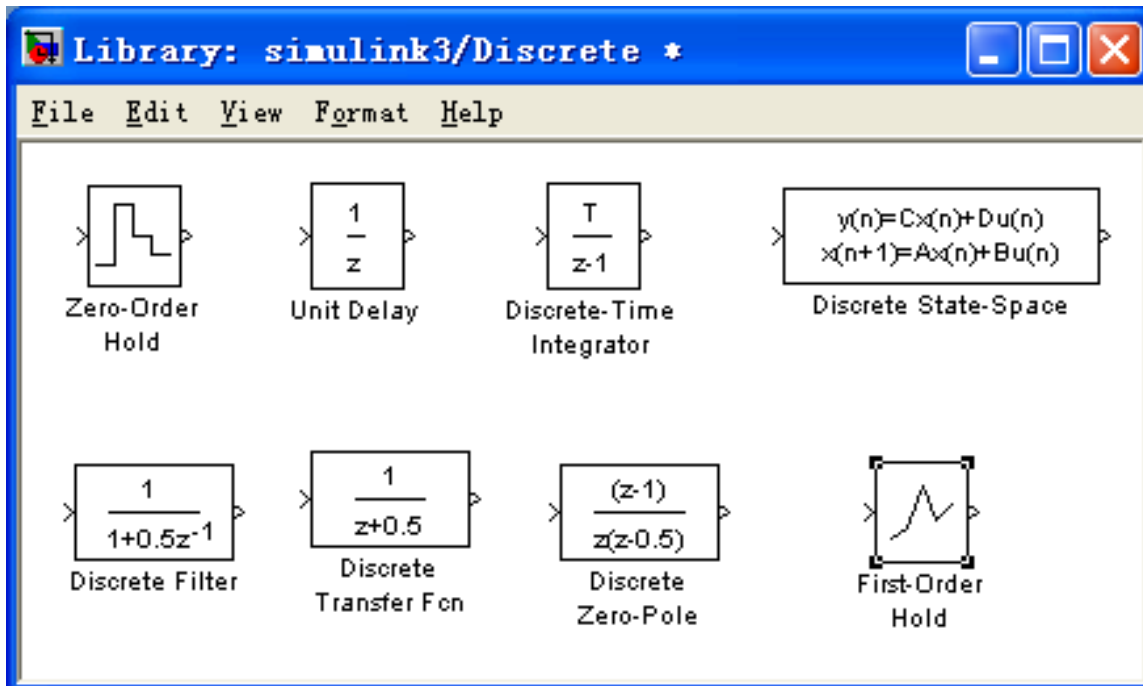
功能: 用于将输入端的信号延迟指定的时间后再传输给输出信号。

8. 可变传输延迟模块(Variable Transport Delay)

功能: 用于将输入端的信号进行可变时间的延迟。

6.3.2 离散模块库 (Discrete)

离散模块库 (Discrete) 主要用于建立离散采样的系统模型，包括的主要模块，如图所示。



1. 零阶保持器模块(Zero-Order-Hold)

功能: 在一个步长内将输出的值保持在同一个值上。

2. 单位延迟模块(Unit Delay)

功能: 将输入信号作单位延迟, 并且保持一个采样周期相当于时间算子 z^{-1} 。

3. 离散时间积分模块(Discrete Time Integrator)

功能: 在构造完全离散的系统时, 代替连续积分的功能。使用的积分方法有: 向前欧拉法、向后欧拉法、梯形法。

4. 离散状态空间模块(Discrete State Space)

功能: 用于实现如下数学方程描述的系统:

$$\begin{cases} x[(n+1)T] = Ax(nT) + Bu(nT) \\ y(nT) = Cx(nT) + Du(nT) \end{cases}$$

5. 离散滤波器模块(Discrete Filter)

功能: 用于实现无限脉冲响应(IIR)和有限脉冲响应(FIR)的数字滤波器。

6. 离散传递函数模块(Discrete Transfer Fcn)

功能: 用于执行一个离散传递函数。

6. 离散零极点传递函数模块(Discrete Zero-Pole)

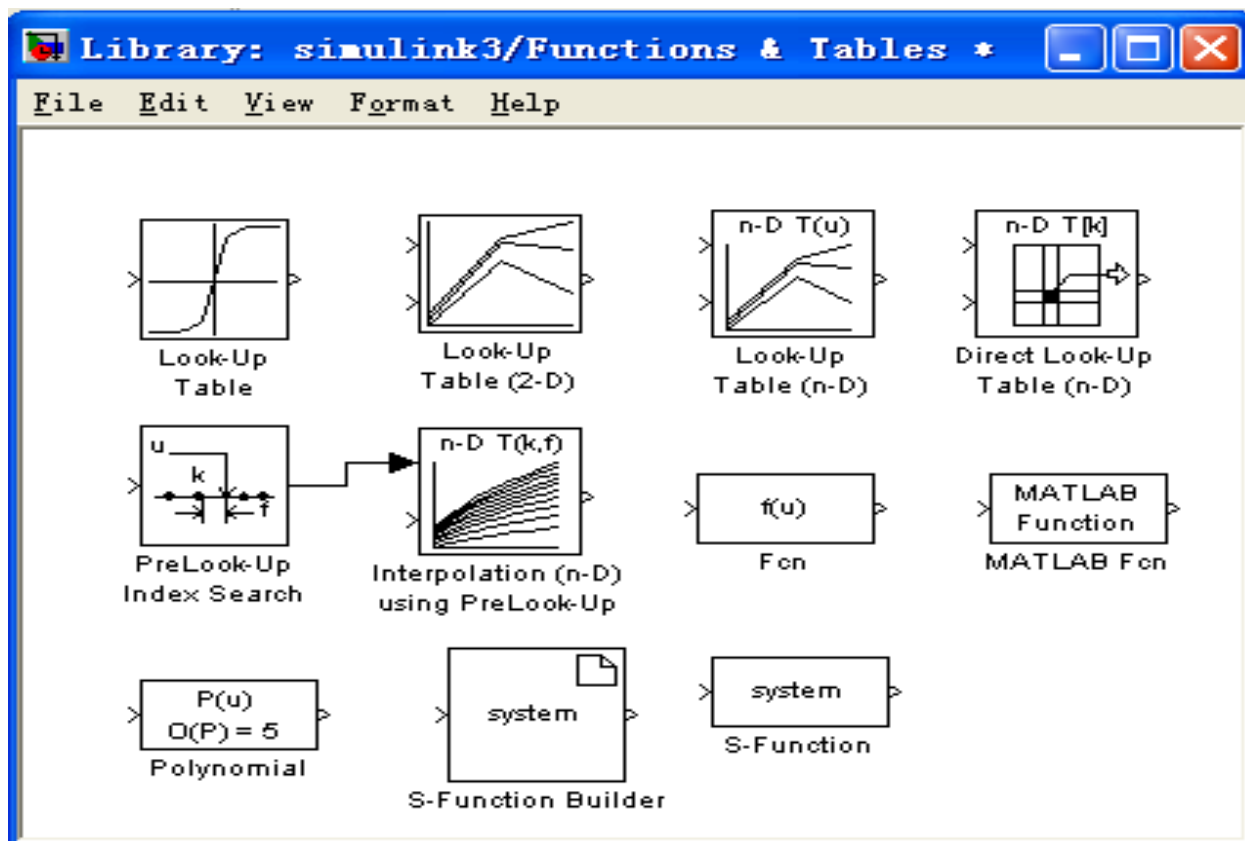
功能: 用于建立一个预先指定的零点、极点, 并用延迟算子 z^{-1} 表示的离散系统。

8. 一阶保持器模块(First Order Hold)

功能: 在一定时间间隔内保持一阶采样。

6.3.3 函数与表格模块库(Function & Table)

函数与表格模块库(Function & Table)主要实现各种一维、二维或者更高维函数的查表，另外用户还可以根据自己需要创建更复杂的函数。该模块库包括多个主要模块，如图6.6所示。



1. 一维查表模块(Look-Up Table)

一维查表模块(Look-Up Table)实现对单路输入信号的查表和线性插值。

2. 二维查表模块(Look-Up Table 2-D)

功能: 根据给定的二维平面网格上的高度值, 把输入的两个变量经过查表、插值, 计算出模块的输出值, 并返回这个值。

说明: 对二维输入信号进行分段线性变换。

3. 自定义函数模块(Fcn)

功能: 用于将输入信号进行指定的函数运算, 最后计算出模块的输出值。

说明: 输入的数学表达式应符合C语言编程规范; 与MATLAB中的表达式有所不同, 不能完成矩阵运算。

4. MATLAB函数模块(MATLAB Fcn)

功能: 对输入信号进行MATLAB函数及表达式的处理。

说明: 模块为单输入模块; 能够完成矩阵运算。

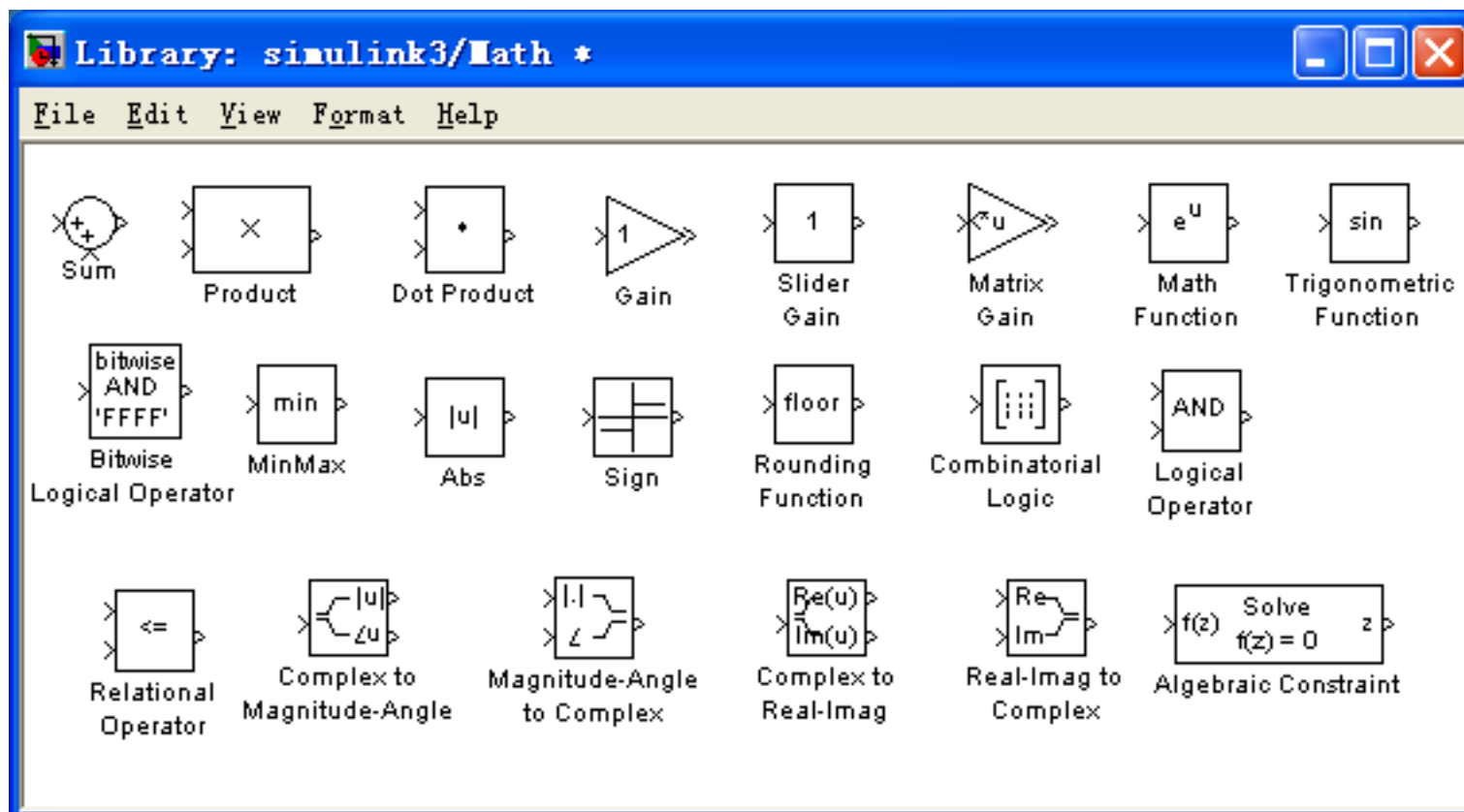
注意: 从运算速度角度, Math function 模块要比 Fcn 模块慢。当需要提高速度时, 可以考虑采用Fcn 或者S函数模块。

5. S-函数模块(S-Function)

功能: 按照Simulink标准, 编写用户自己的Simulink函数。它能够将MATLAB语句、C语言等编写的函数放在Simulink模块中运行, 最后计算模块的输出值。

6.3.4 数学模块库(Math)

数学模块库(Math)包括多个数学运算模块，如图6.8所示。



1. 求和模块(Sum)

功能: 求和模块(Sum)用于对多路输入信号进行求和运算，并输出结果。

2. 乘法模块(Product)

功能: 乘法模块(Product)用于实现对多路输入的乘积、商、矩阵乘法或者模块的转置等。

3. 矢量的点乘模块(Dot Product)

功能: 矢量的点乘模块(Dot Product)用于实现输入信号的点积运算。

4. 增益模块(Gain)

功能: 增益模块(Gain)的作用是把输入信号乘以一个指定的增益因子，使输入产生增益。

5. 常用数学函数模块(Math Function)

功能: 用于执行多个通用数学函数，其中包含exp、log、log10、square、sqrt、pow、reciprocal、hypot、rem、mod等。

6. 三角函数模块(Trigonometric Function)

功能: 用于对输入信号进行三角函数运算, 共有10种三角函数供选择。

6. 特殊数学模块

特殊数学模块中包括求最大最小值模块(MinMax)、取绝对值模块(Abs)、符号函数模块(Sign)、取整数函数模块(Rounding Function)等。

8. 数字逻辑函数模块

数字逻辑函数模块包括复合逻辑模块(Combinational Logic)、逻辑运算符模块(Logical Operator)、位逻辑运算符模块(Bitwise Logical Operator)等。

9. 关系运算模块(Relational Operator)

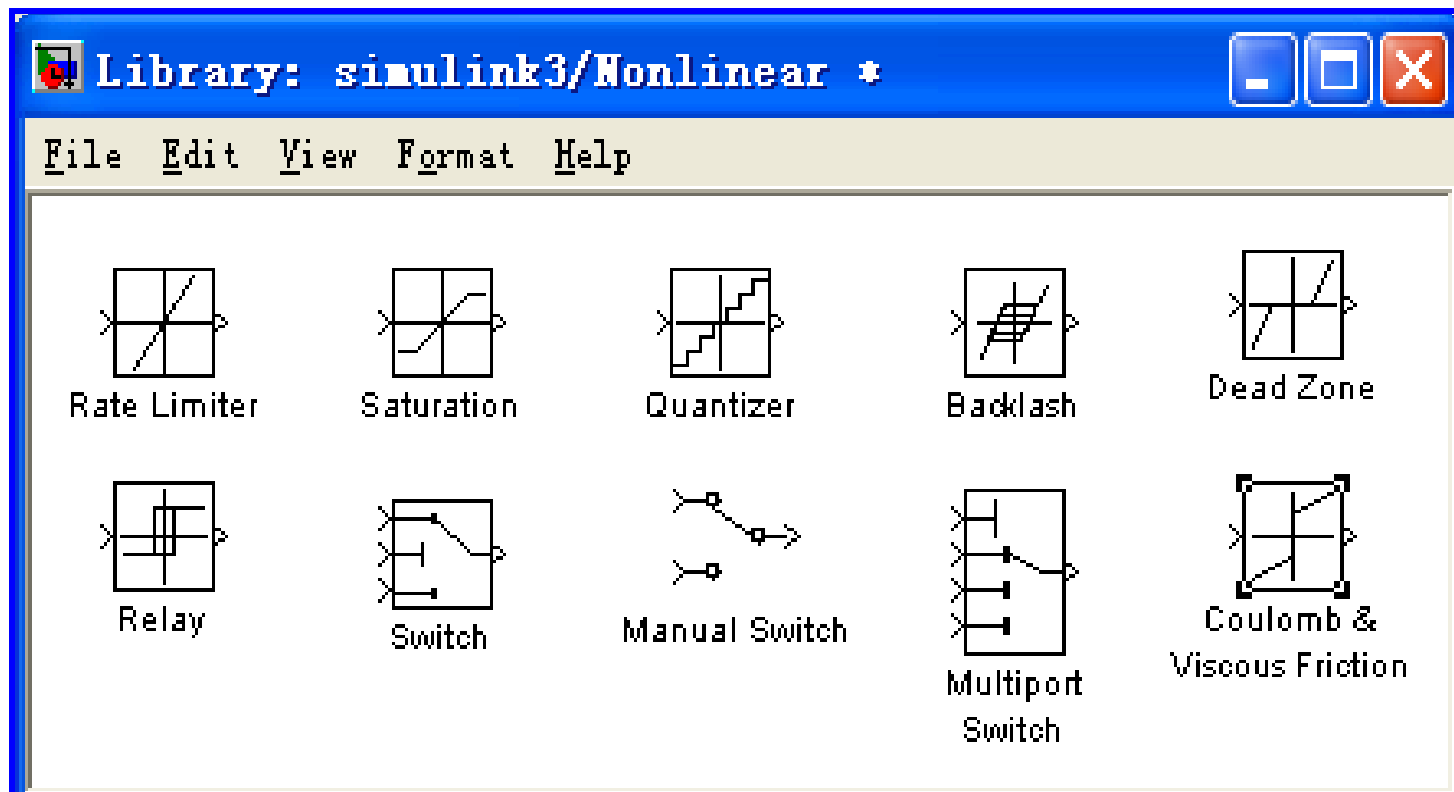
关系符号包括: == (等于)、≠ (不等于)、< (小于)、≤ (小于等于)、> (大于)、≥ (大于等于)等。

10. 复数运算模块

复数运算模块包括计算复数的模与幅角(Complex to Magnitude-Angle)、由模和幅角计算复数(Magnitude-Angle to Complex)、提取复数实部与虚部模块(Complex to Real and Image)、由复数实部和虚部计算复数(Real and Image to Complex)。

6.3.5 非线性模块(Nonlinear)

非线性模块(Nonlinear)中包括一些常用的非线性模块，如图6.9所示。



1. 比率限幅模块(Rate Limiter)

功能: 用于限制输入信号的一阶导数, 使得信号的变化率不超过规定的限制值。

2. 饱和度模块(Saturation)

功能: 用于设置输入信号的上下饱和度, 即上下限的值, 来约束输出值。

3. 量化模块(Quantizer)

功能: 用于把输入信号由平滑状态变成台阶状态。

4. 死区输出模块(Dead Zone)

功能: 在规定的区内没有输出值。

5. 继电模块(Relay)

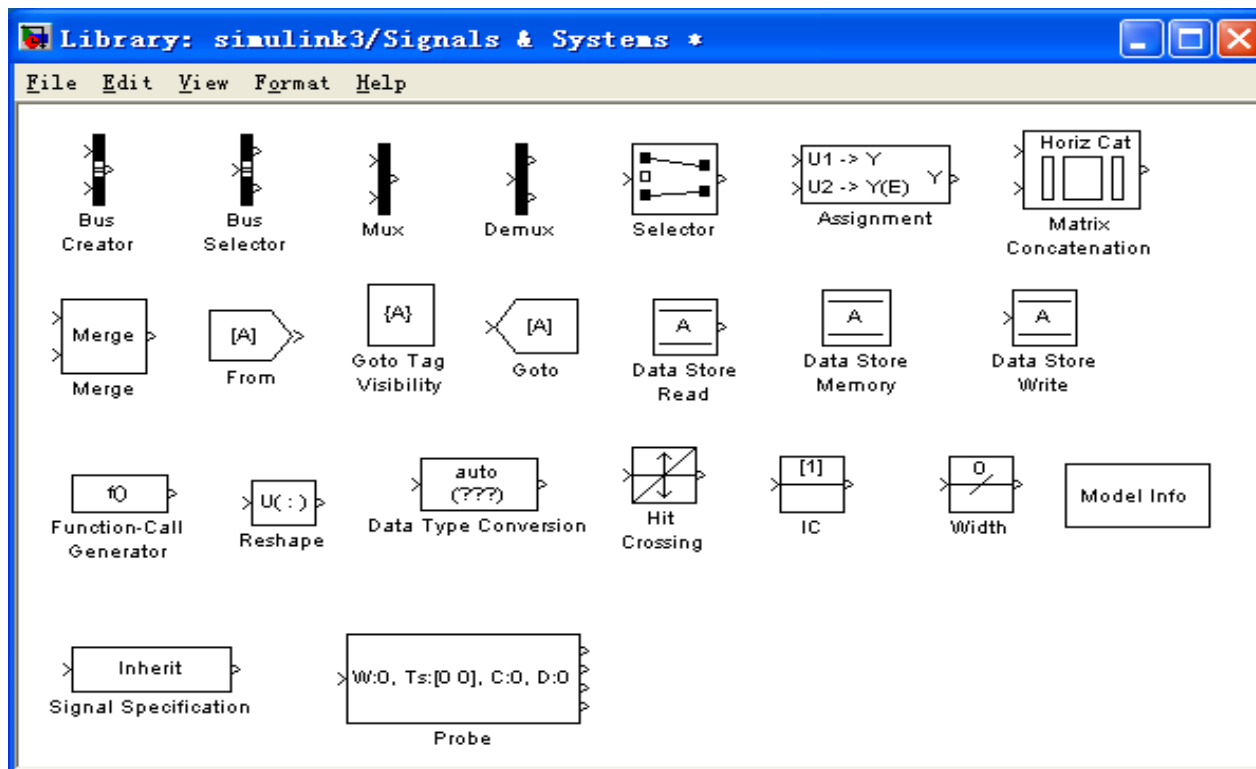
功能: 继电模块(Relay)用于实现在两个不同常数值之间进行切换。

6. 选择开关模块(Switch)

功能: 根据设置的门限来确定系统的输出。

6.3.6 信号与系统模块库(signals &Systems)

信号与系统模块库(signals &Systems)包括的主要模块如图6.11所示。



1. Bus 信号选择模块(Bus Selector)

功能： 用于得到从Mux模块或其它模块引入的Bus信号。

2. 混路器模块(Mux)

功能： 把多路信号组成一个矢量信号或者Bus信号。

3.分路器模块(Demux)

功能： 把混路器组成的信号按照原来的构成方法分解成多路信号。

4. 信号合成模块(Merge)

功能： 把多路信号进行合成一个单一的信号。

5. 接收/传输信号模块(From/Goto)

功能： 接收 / 传输信号模块 (From/ Goto) 常常配合使用， From 模块用于从一个 Goto 模块中接收一个输入信号， Goto模块用于把输入信号传递给From模块。

6. 初始值设定模块(IC)

功能： 初始值设定模块(IC)用于设定与输出端口连接的模块的初始值。

6.3.6 信号输出模块(Sinks)

信号输出模块(Sinks)包括的主要模块如图6.12所示。

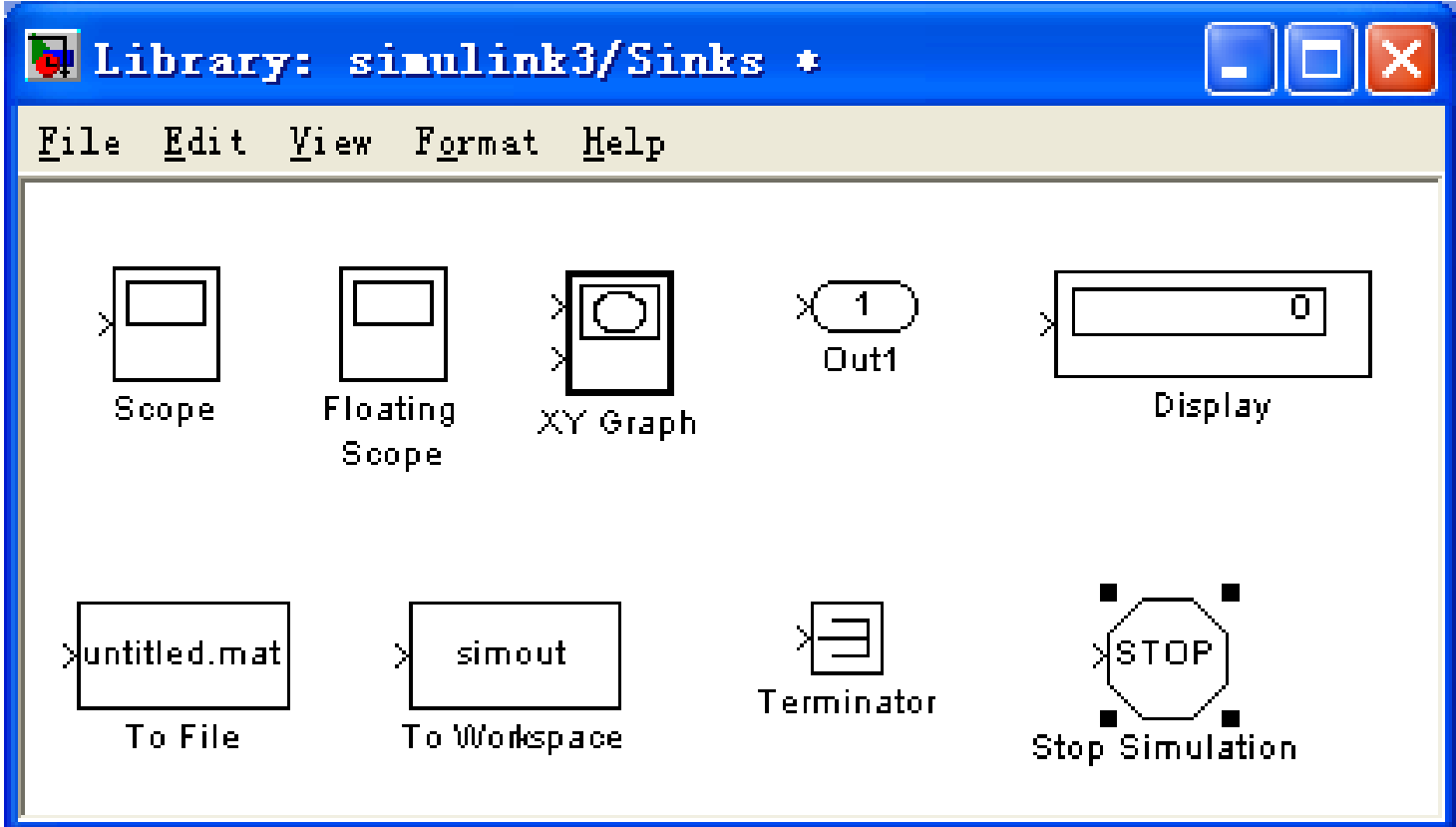


图6.12 输出显示模块库

1. 示波器模块(Scope)

功能: 显示在仿真过程中产生的输出信号，用于在示波器中显示输入信号与仿真时间的关系曲线，仿真时间为x轴。

2. 二维信号显示模块(XY Graph)

功能: 在MATLAB的图形窗口中显示一个二维信号图，并将两路信号分别作为示波器坐标的x轴与y轴，同时把它们之间的关系图形显示出来。

3. 显示模块(Display)

功能: 按照一定的格式显示输入信号的值。可供选择的输出格式包括：short、long、short_e、long_e、bank等。

4. 输出到文件模块(To File)

功能: 按照矩阵的形式把输入信号保存到一个指定的MAT文件。第一行为仿真时间，余下的行则是输入数据，一个数据点是输入矢量的一个分量。

5. 输出到工作空间模块(To Workspace)

功能: 把信号保存到MATLAB的当前工作空间，是另一种输出方式。

6. 终止信号模块(Terminator)

功能: 中断一个未连接的信号输出端口。

6. 结束仿真模块(Stop simulation)

功能: 停止仿真过程。当输入为非零时，停止系统仿真。

6.3.8 信号源模块库(Sources)

信号源模块库(Sources)包括的主要模块如图6.13所示。

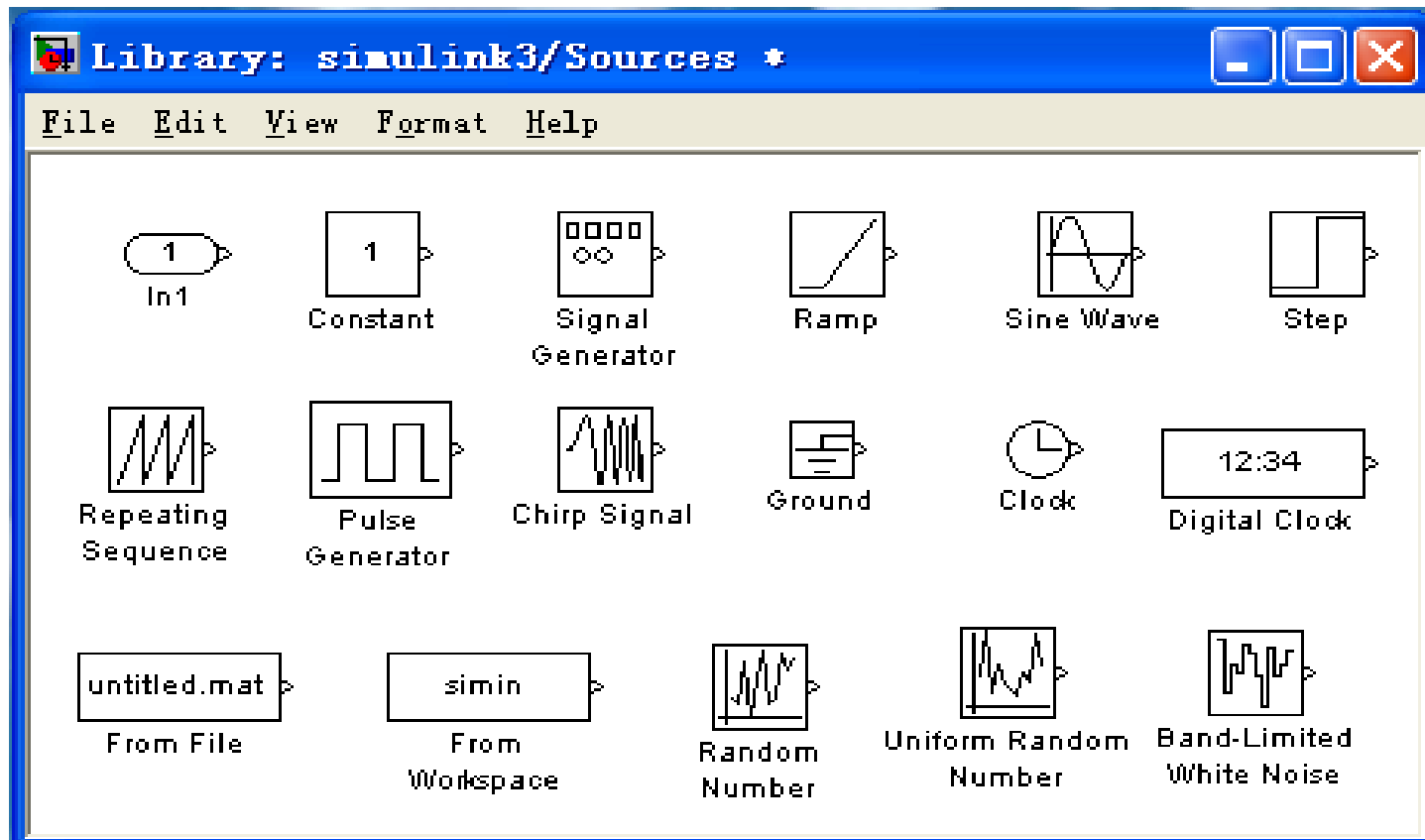


图6.13 信号源模块库

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/628011101064006121>