



# Matlab编程及其应用

---

## 第七讲



# Simulink概述

---

- 模型化图形输入，对动态系统建模与仿真。
- **Simu**link :Simulation **仿真**，
- **Simu****link** :**连接**，把一系列**模块**连接起来，  
构成复杂的系统模型。



## ■ Simulink仿真模型的一般结构





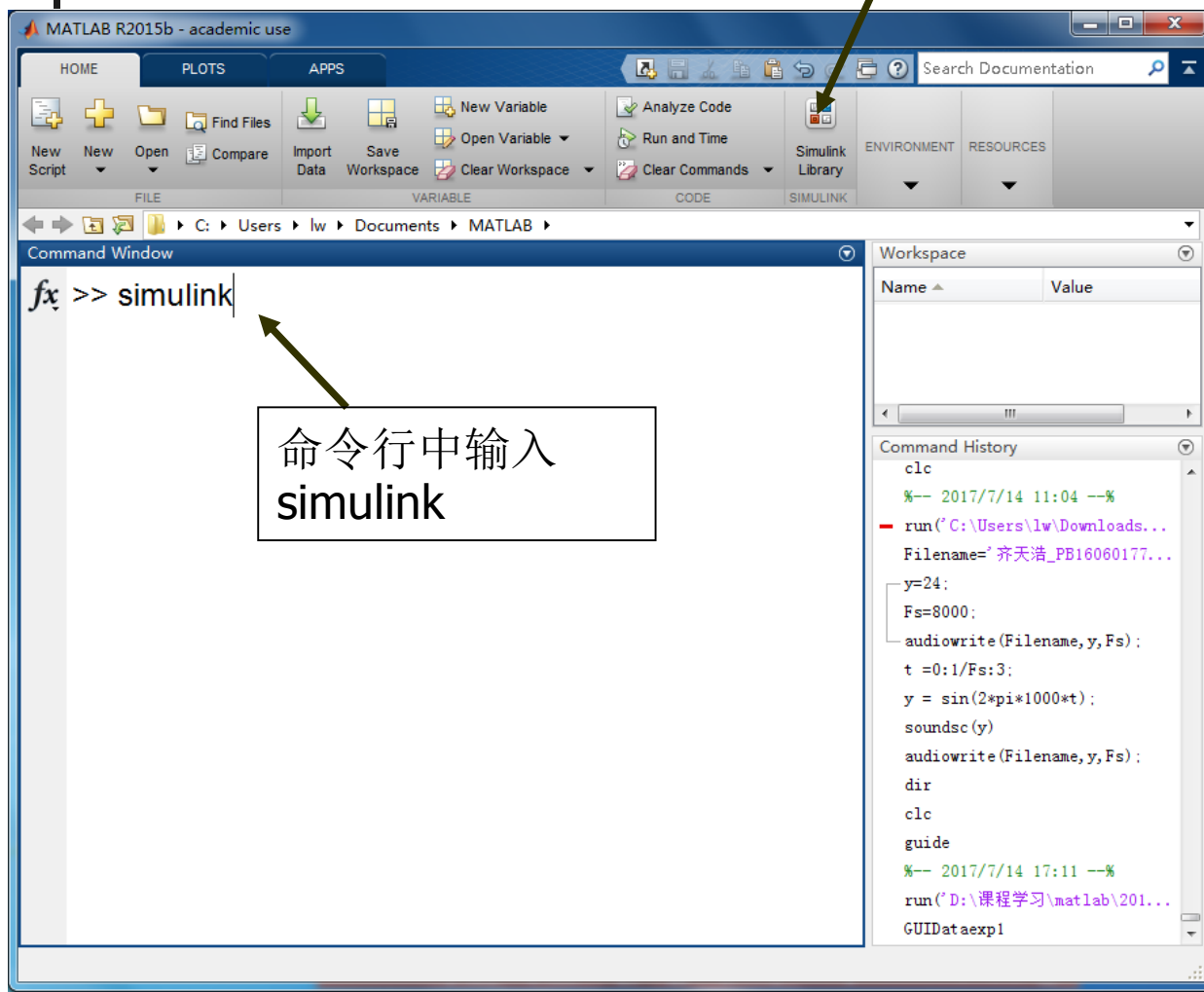
# Simulink基础

---

- 基本操作： 启动->建立新mdl文件->导入  
模块->连接->仿真设置->仿真
- 常用模块库简介
- 模块基本操作
- 连线基本操作
- 例

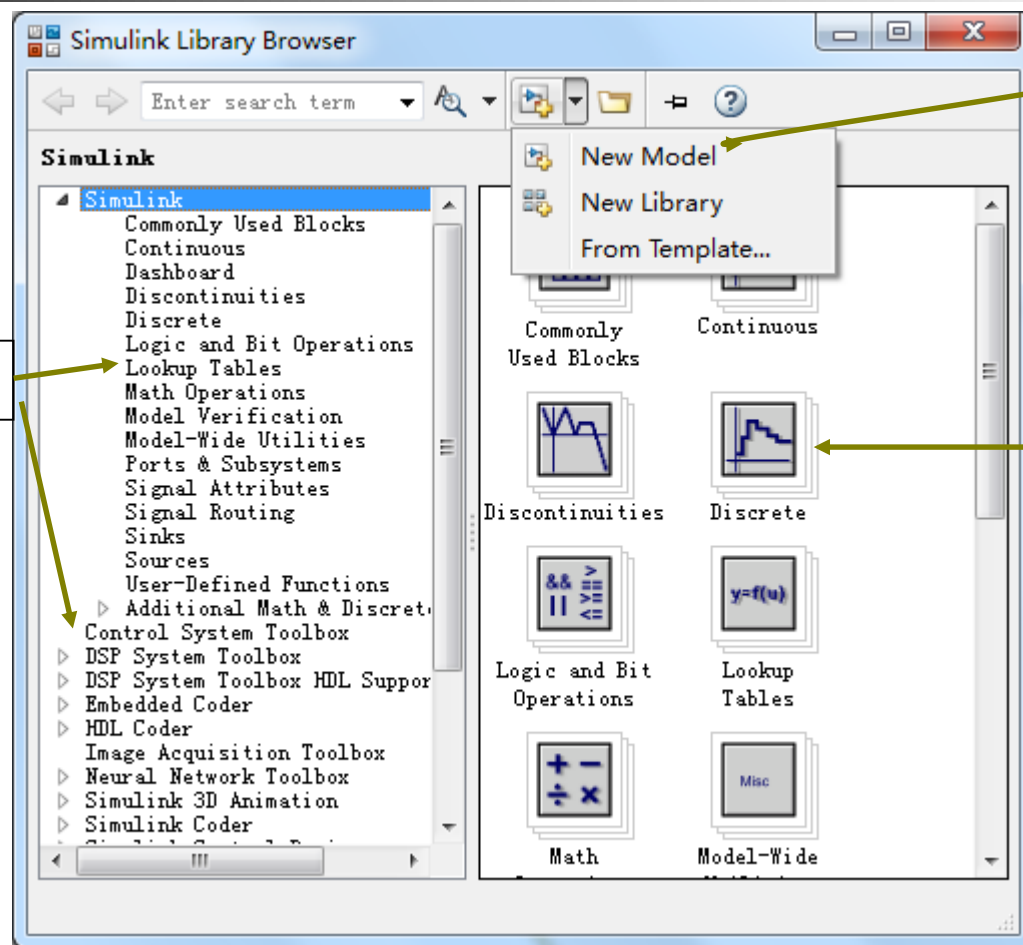
# 启动simulink

点击simulink图标



命令行中输入  
simulink

# Simulink模块库



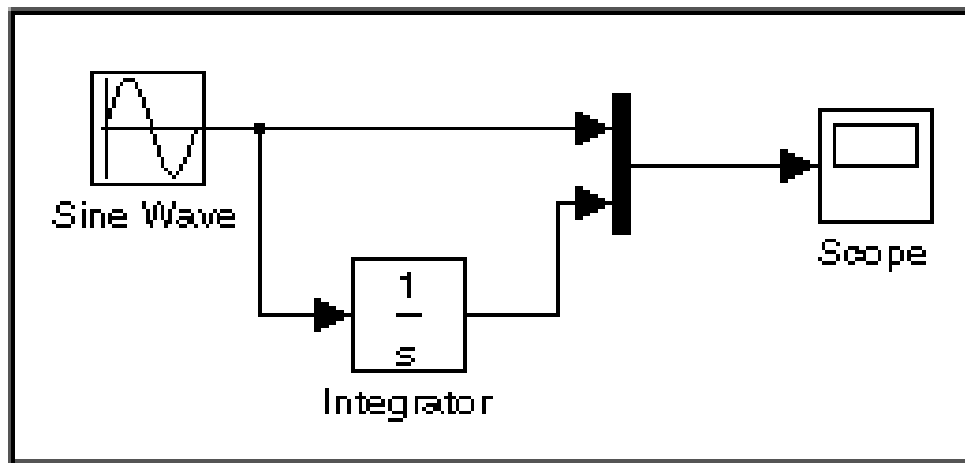
新建一个Simulink  
仿真文件

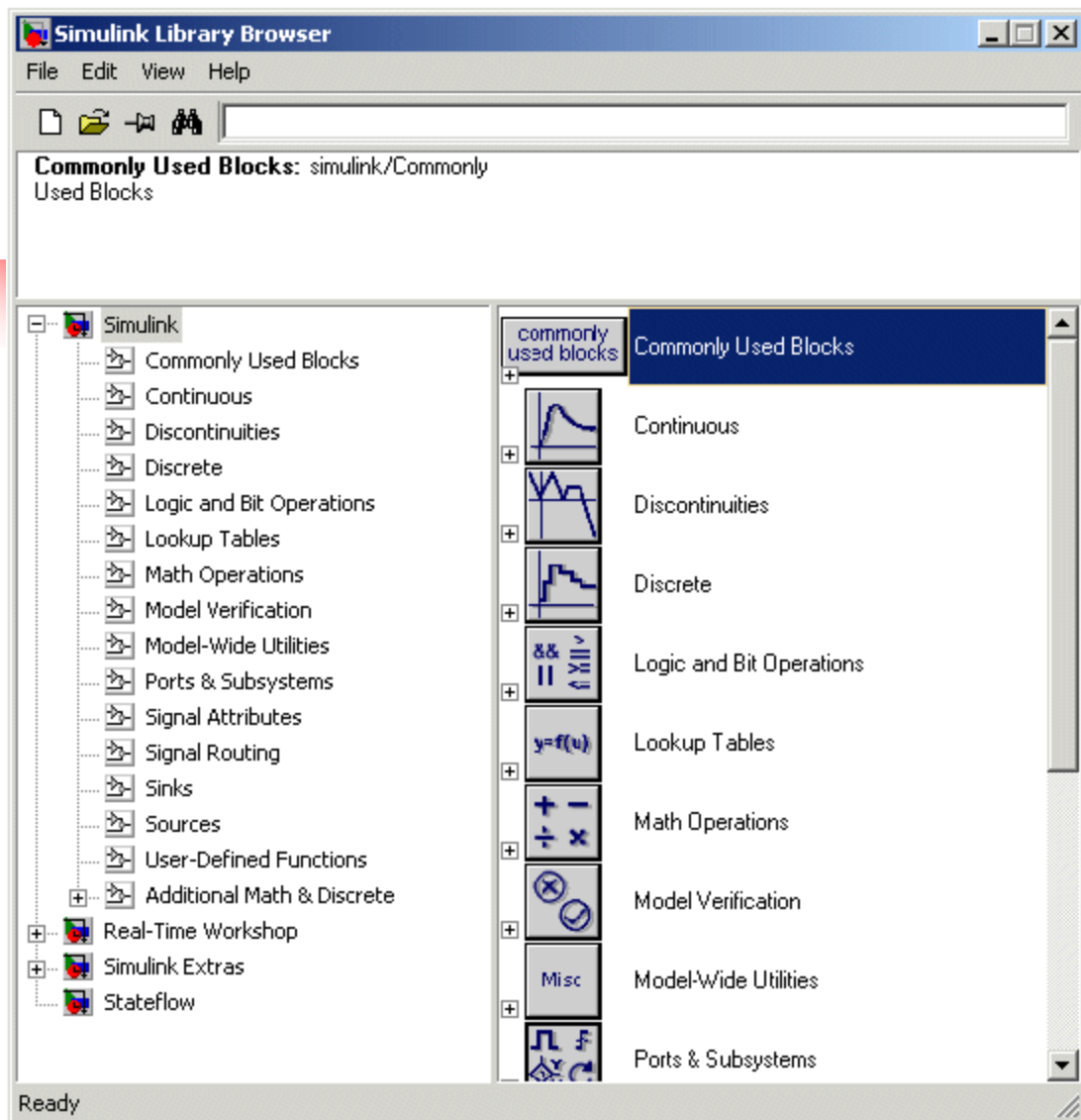
Simulink模块库

Simulink模块

# Simulink基本操作

- **复制模块：** 确定模型中包含哪些模块，然后在库浏览器中找到所需要的模块，将需要的模块从模块库中复制到模型中。





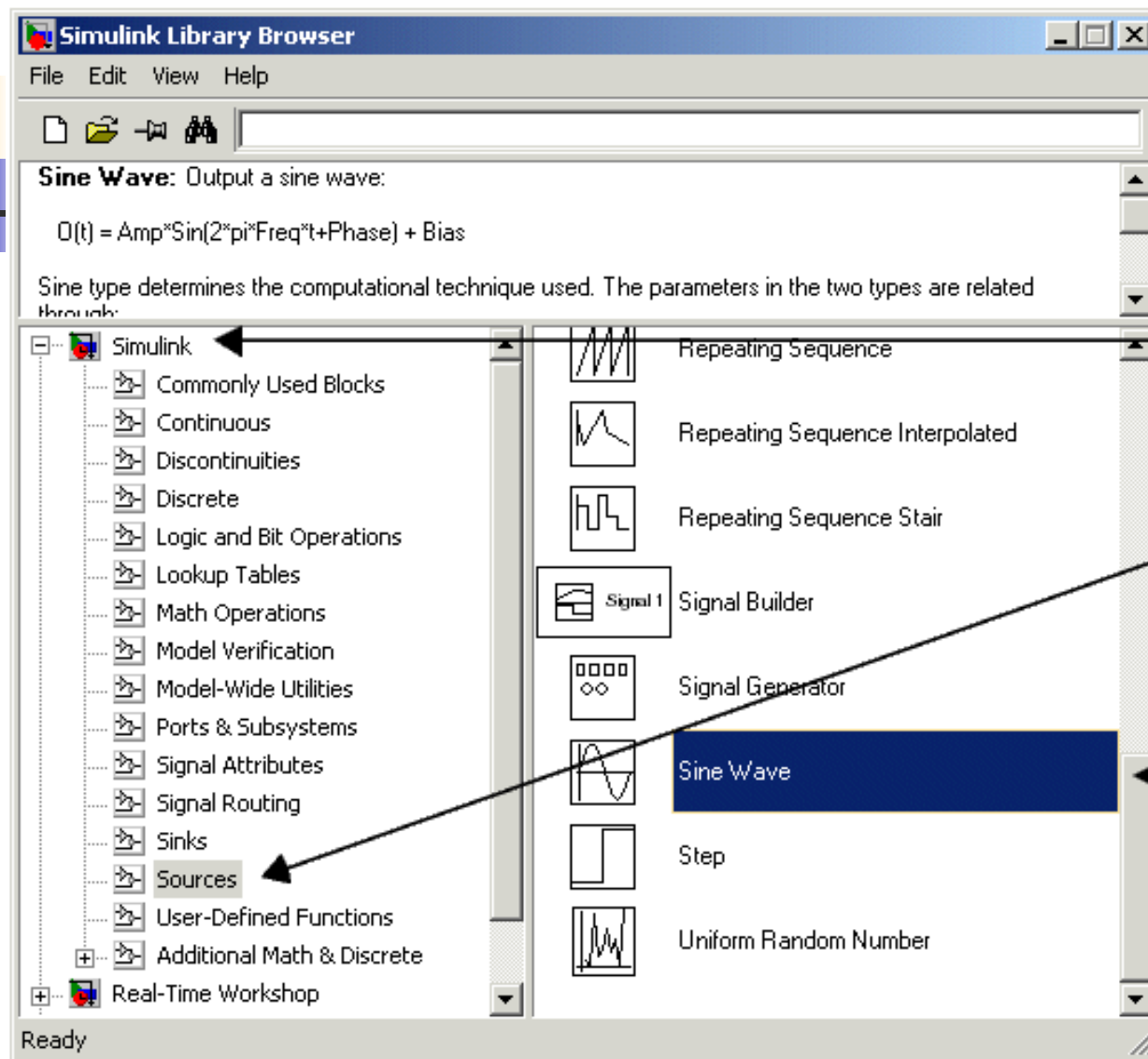
四个模块涉及的模块库分别为：

- **Sources**（信号源模块库）
- **Continuous**（连续模块库）
- **Sinks**（信号输出模块库）
- **Signal Routing**（信号路由模块库）

此例中，模型包括四个模块：

- 正弦波模块 (**sine Wave**)、积分模块(**Integrator**)、示波器模块(**Scope**)、组合模块 (**Mux**)





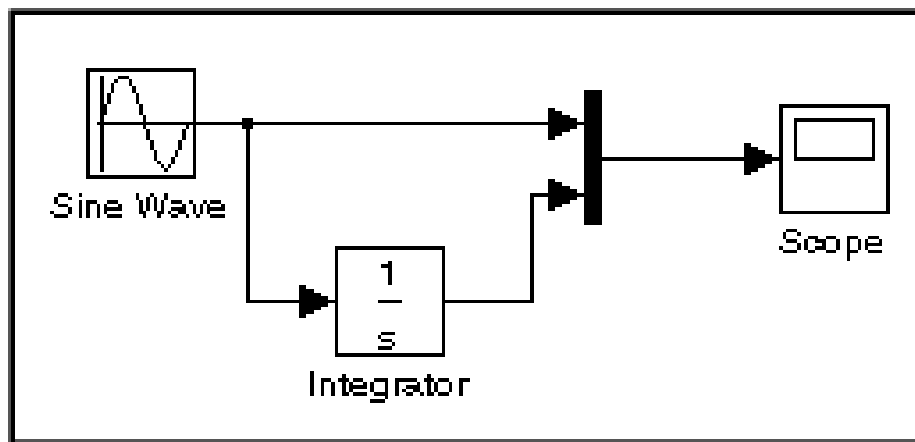
Simulink library

Sources library

Sine Wave block

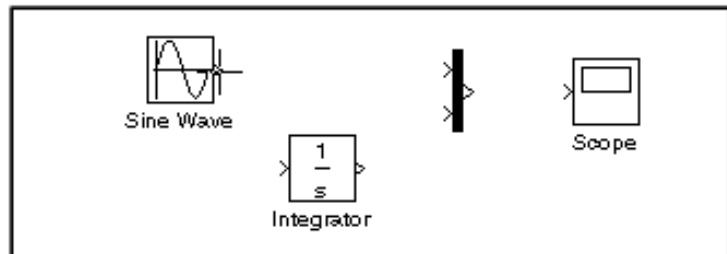
## 增加信号线

- 把一个模块的输出与另一个模块的输入连接起来
- 在一条已有的信号线上引出另一条信号线，这两条线将传送相同信号给各自对象。



## ■ 把一个模块的输出与另一个模块的输入连接起来

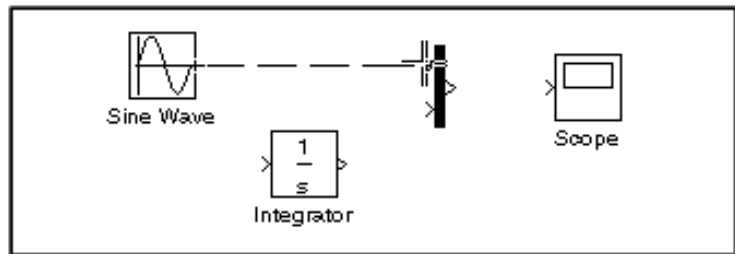
Now it's time to connect the blocks. Connect the Sine Wave block to the top input port of the Mux block. Position the pointer over the output port on the right side of the Sine Wave block. Notice that the cursor shape changes to crosshairs.



(1) 把鼠标指针移到第一个模块端口的任意位置，光标将变成十字形。

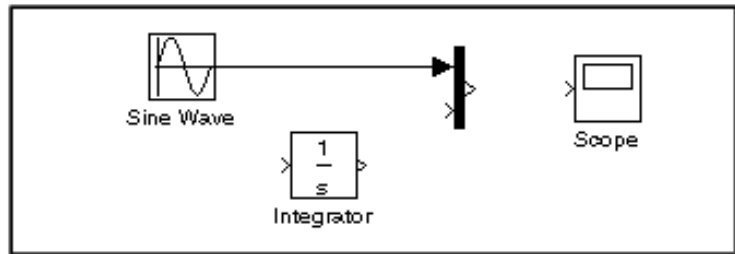
Hold down the mouse button and move the cursor to the top input port of the Mux block.

Notice that the line is dashed while the mouse button is down and that the cursor shape changes to double-lined crosshairs as it approaches the Mux block.



(2) 按下鼠标，拖动鼠标指针定位到第二个模块输入端口的位置。

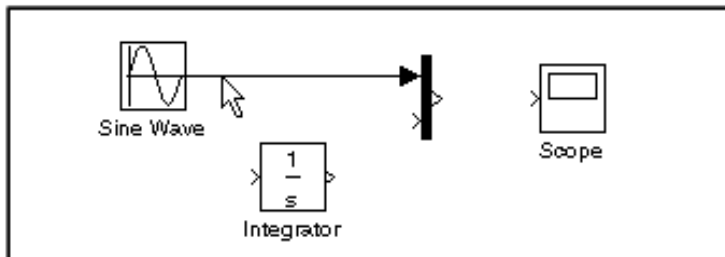
Now release the mouse button. The blocks are connected. You can also connect the line to the block by releasing the mouse button while the pointer is over the block. If you do, the line is connected to the input port closest to the cursor's position.



(3) 释放鼠标，**simulink**用一个带箭头的实线信号线代替端口的符号，用来表示信号的流向。

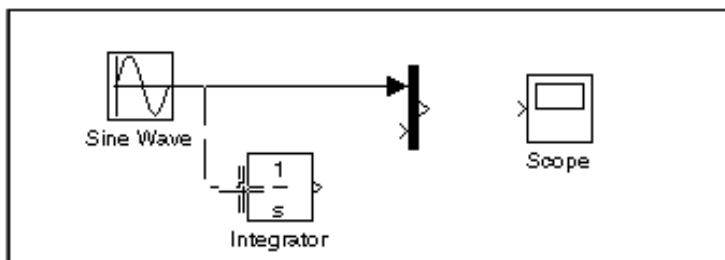
## ■ 在一条已有的信号线上引出另一条信号线

1. First, position the pointer on the line between the Sine Wave and the Mux block.



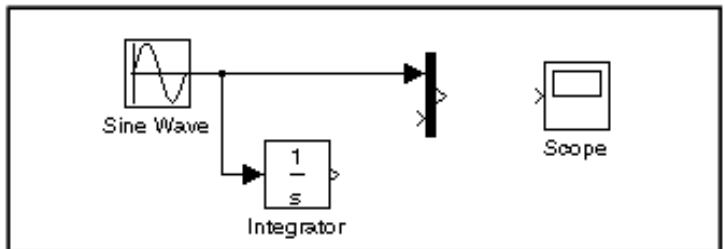
(1) 把鼠标指针移到这根信号线上的某个位置，这个位置就是引出新信号线的起始位置。

2. Press and hold down the **Ctrl** key (or click the right mouse button). Press the mouse button, then drag the pointer to the Integrator block's input port or over the Integrator block itself.



(2) 在按下**Ctrl**键的同时，按下鼠标，拖动鼠标到目标端口。

3. Release the mouse button. Simulink draws a line between the starting point and the Integrator block's input port.



(2) 释放鼠标按钮和**Ctrl**键，那么**Simulink**就在起始位置和目标端口之间创建了一条新信号线。

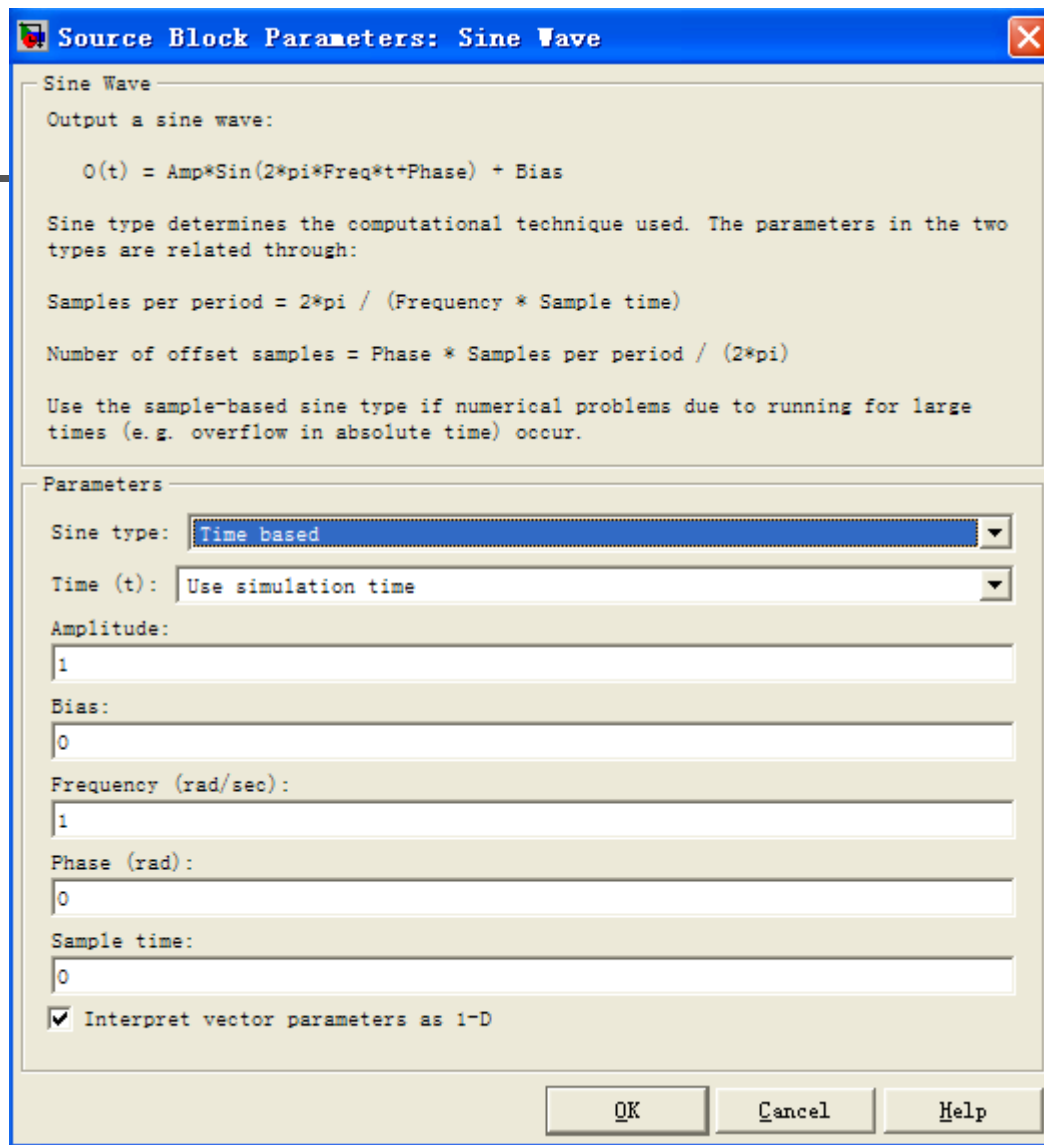


## 确定模型参数

---

- 在模型窗口选中一个模块，用鼠标**双击**该模块，**Simulink**就打开**模块基本属性**对话框。
- 在模型窗口选择【**Simulation: Configuration parameters**】菜单，可进行仿真参数设置。

## ■ Sine wave 模块属性设置对话框



**Source Block Parameters: Sine Wave**

Sine Wave

Output a sine wave:

$$O(t) = \text{Amp} * \sin(2 * \pi * \text{Freq} * t + \text{Phase}) + \text{Bias}$$

Sine type determines the computational technique used. The parameters in the two types are related through:

$$\text{Samples per period} = 2 * \pi / (\text{Frequency} * \text{Sample time})$$
$$\text{Number of offset samples} = \text{Phase} * \text{Samples per period} / (2 * \pi)$$

Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.

Parameters

Sine type: Time based

Time (t): Use simulation time

Amplitude: 1

Bias: 0

Frequency (rad/sec): 1

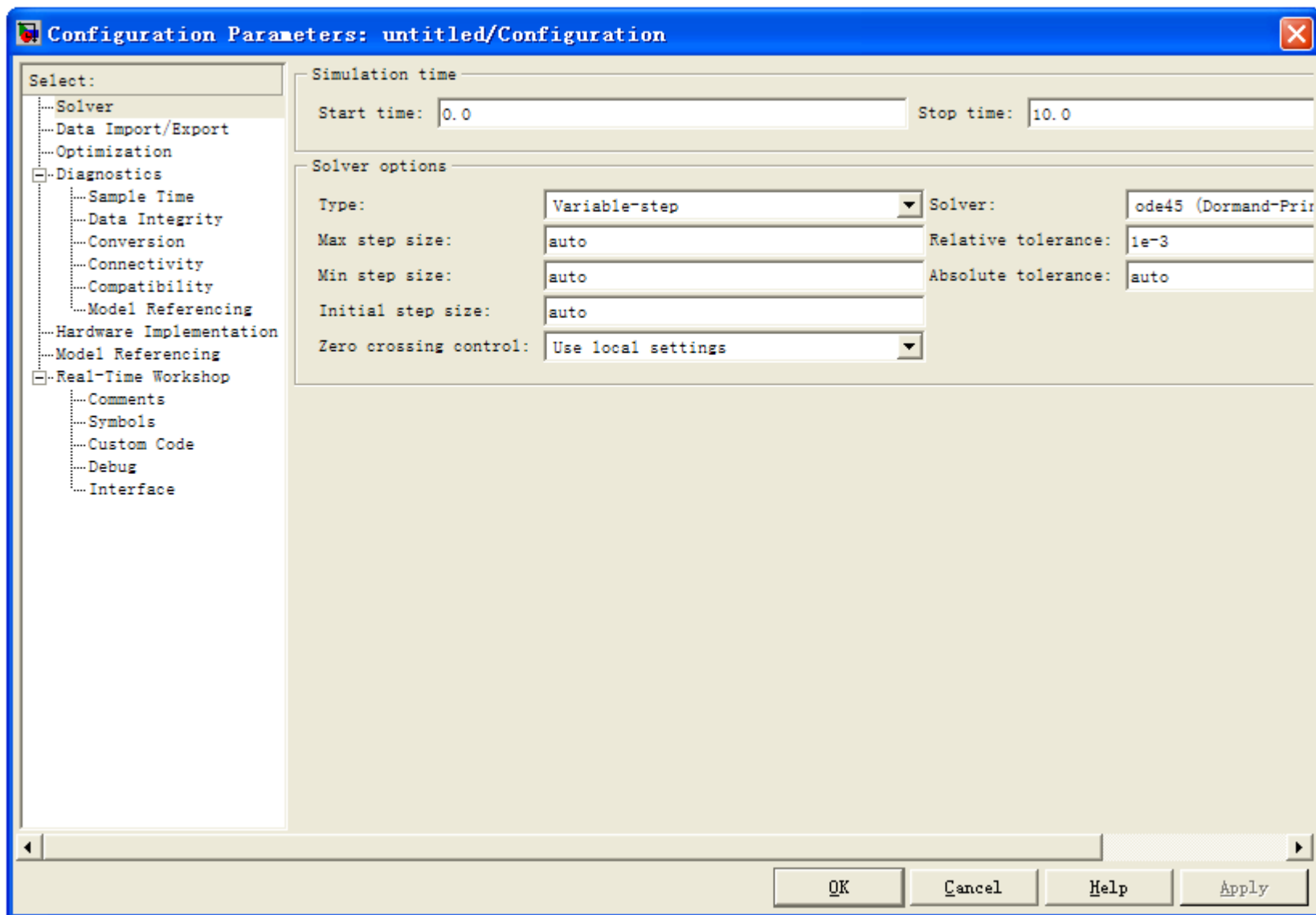
Phase (rad): 0

Sample time: 0

☒ Interpret vector parameters as 1-D

OK Cancel Help

## ■ 仿真参数设置





## 求解器(Solver)的设置

---

### (1) Simulation time （仿真时间设置）

- 修改仿真的开始和结束时间。

### (2) Solvers options （求解器选项设置）

- **Solver:** Simulink模型仿真一般需要采用微分方程或微分方程组的数值解法，用户可以根据仿真模型的特点，选择最合适的求解方法；
- **Type:** 选择可变步长或固定步长；
- 在可变步长中，有Max step size, Min step size, Initial step size
- **Zero-crossing control** （零点穿越控制）
- **Relative tolerance, Absolute tolerance** （容许误差控制）





# 仿真

---

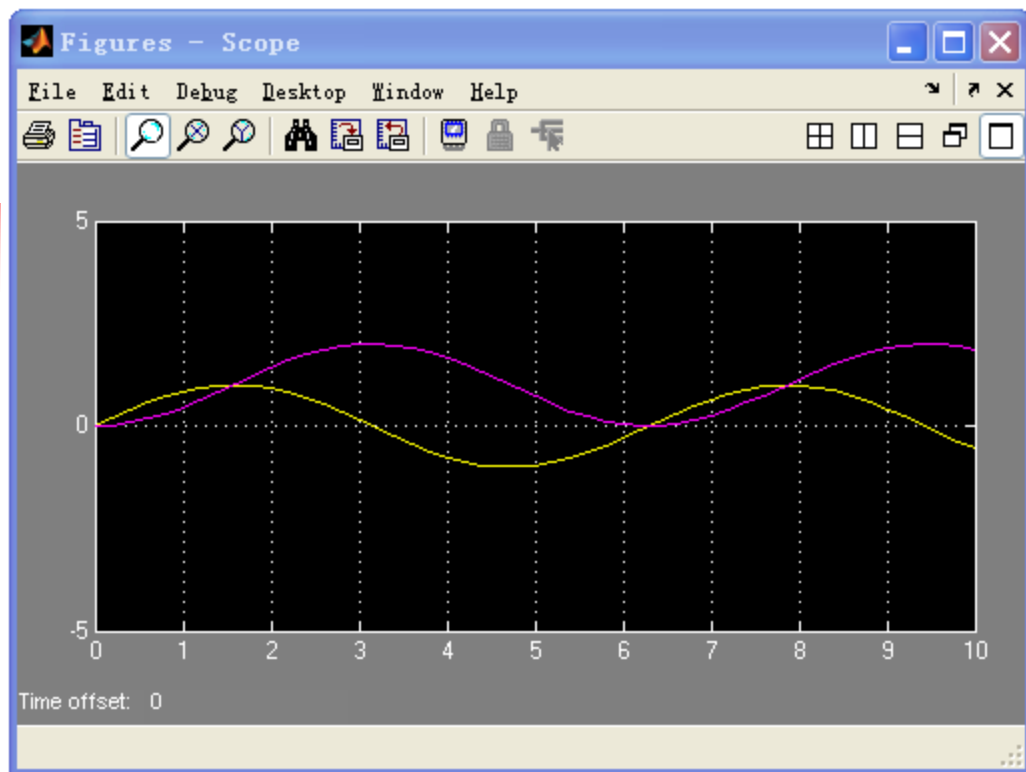
- 仿真过程的启动


在模型窗口选择 **【Simulation】** → **【Start】**

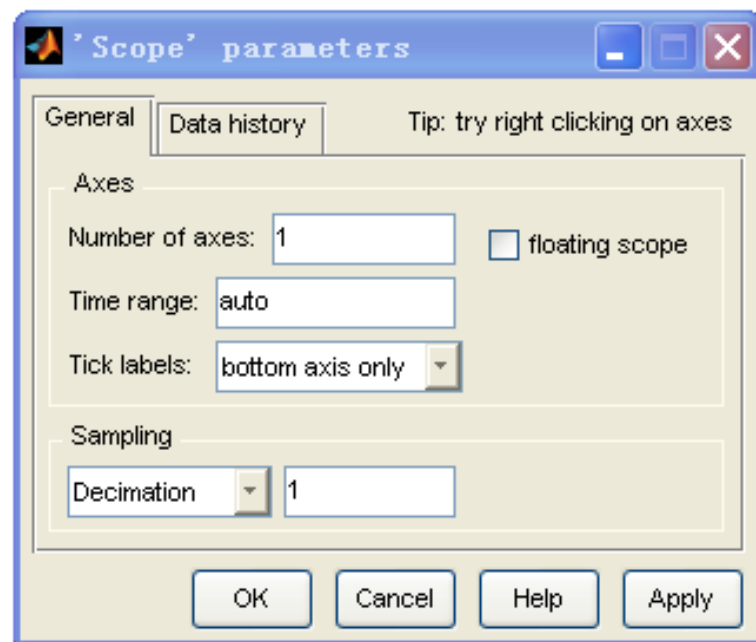
- 仿真过程的诊断

如果仿真过程出现错误，仿真一般会自动停止，并弹出一个诊断对话框显示错误的相关信息。

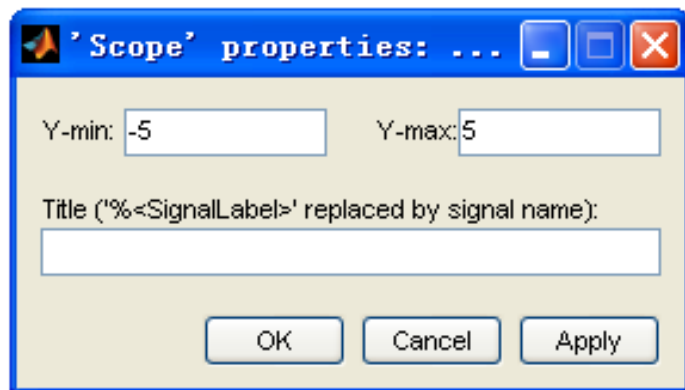
- **Message:** 错误类型，如模块错误或警告；
- **Source:** 发生错误的模块名称；
- **Fullpath:** 导致错误的对象的完整路径；
- **Summary:** 错误的简单说明；
- **Reported by :** 报告错误的组件。



在示波器窗口单击工具按钮 ，弹出 **Scope** 模块的参数设置窗口：



在示波器窗口单击鼠标右键，弹出：





# 模块基本操作：

- **模块的选定** 选多个模块时，按下Shift键，依次选定所需的模块，或者按住鼠标左键，拉虚线框
- **模块的移动** 按住鼠标左键
- **改变模块的方向** 单击【**Format**】→【**Flip Block**】菜单项，可将模块左右镜像翻转，单击【**Format**】→【**Rotate Block**】菜单项，可将模块旋转90°
- **复制模块**



## 模块基本操作：

---

- **改变大小**：选中模块，对模块出现的4个黑色标记进行拖曳即可。
- **模块命名**：先用鼠标在需要更改的名称上单击一下，然后直接更改即可。名称在功能模块上的位置也可以变换**180度**，可以用**Format**菜单中的**Flip Name**来实现，也可以直接通过鼠标进行拖曳。**Hide Name**可以隐藏模块名称。



## 模块基本操作：

---

- 颜色设定： **Format**菜单中的**Foreground Color**可以改变模块的前景颜色，**Background Color**可以改变模块的背景颜色；而模型窗口的颜色可以通过**Screen Color**来改变。



# 信号线操作

- **改变粗细**：线所以有粗细是因为线引出的信号可以是标量信号或向量信号，当选中**Format**菜单下的**Wide Vector Lines**时，线的粗细会根据线所引出的信号是标量还是向量而改变，如果信号为标量则为细线，若为向量则为粗线。选中**Vector Line Widths**则可以显示出向量引出线的宽度，即向量信号由多少个单一信号合成
- **设定标签**：只要在线上双击鼠标，即可输入该线的说明标签。也可以通过选中线，然后打开**Edit**菜单下的**Signal Properties**进行设定，其中**signal name**属性的作用是标明信号的名称，设置这个名称反映在模型上的直接效果就是与该信号有关的端口相连的所有直线附近都会出现写有信号名称的标签。



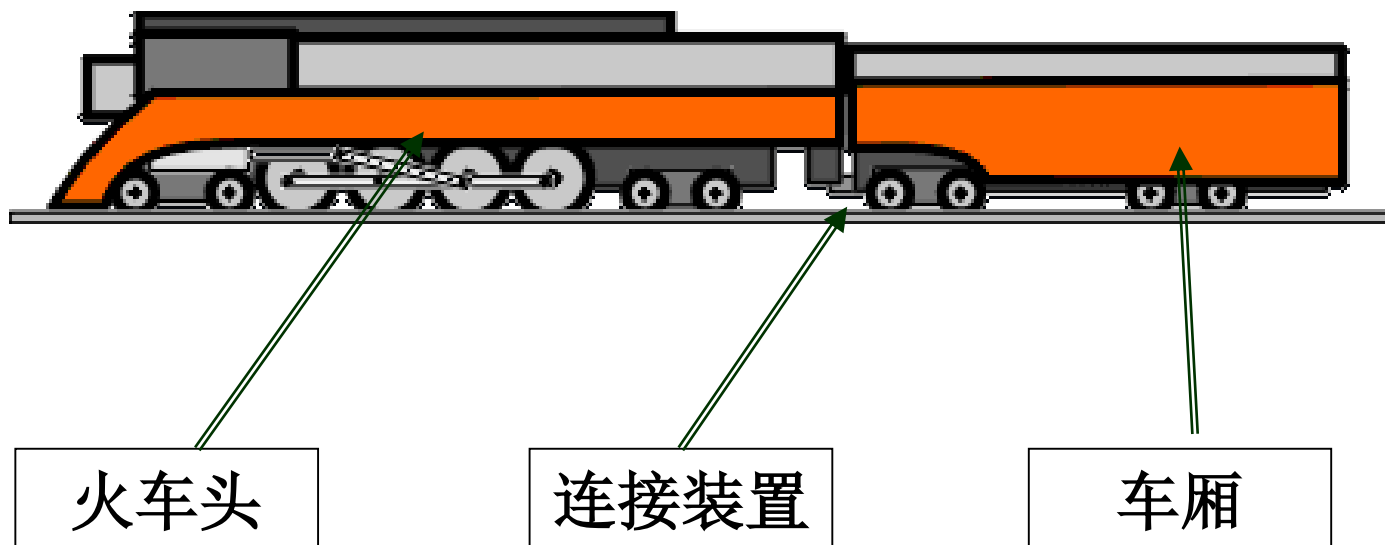
# 信号线操作

---

- **线的折弯：** 按住**Shift**键，再用鼠标在要折弯的线处单击一下，就会出现圆圈，表示折点，利用折点就可以改变线的形状。
- **线的分支：** 按住鼠标右键，在需要分支的地方拉出即可以。或者按住**Ctrl**键，并在要建立分支的地方用鼠标拉出即可。

# 火车系统仿真

## ■ 物理系统：







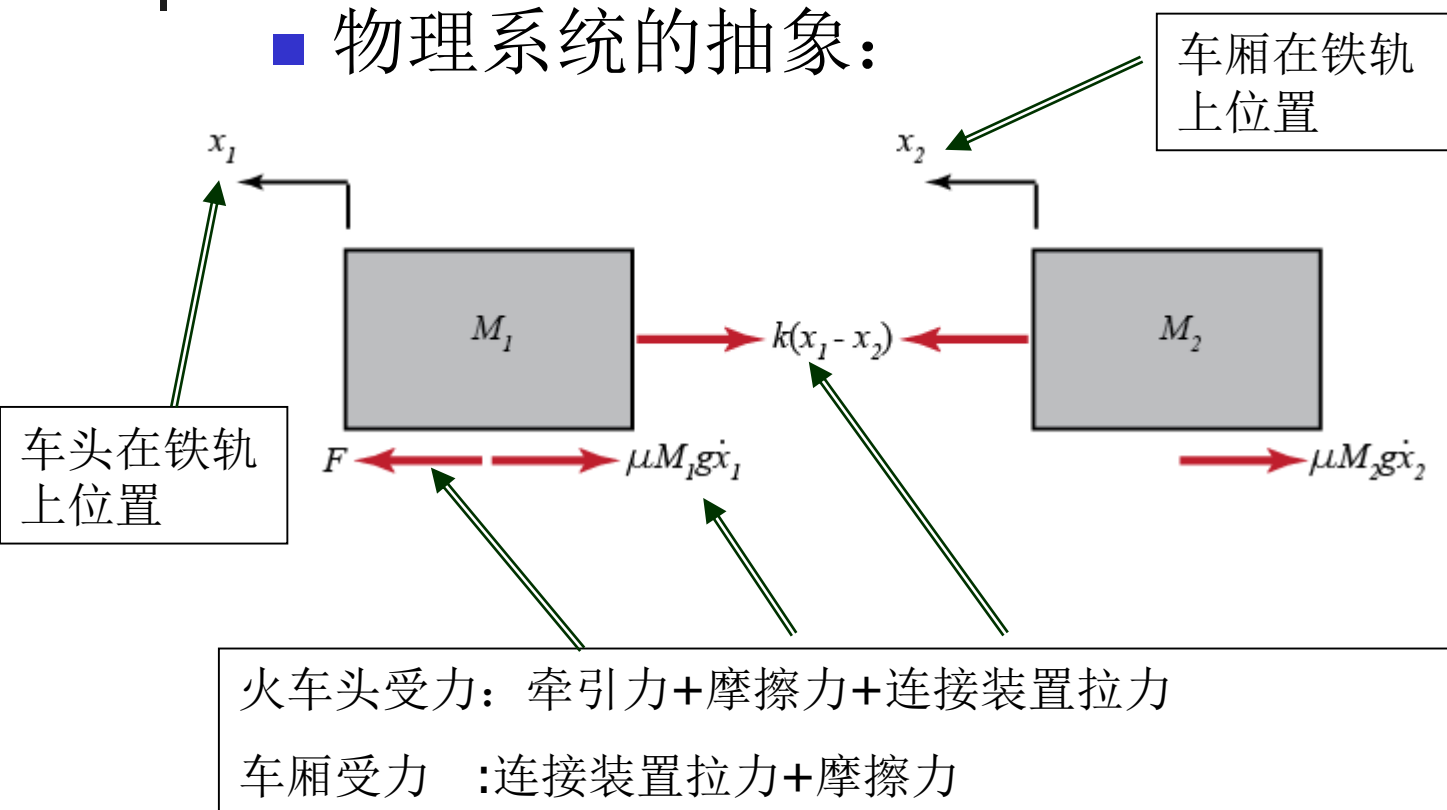
# 火车系统仿真

---

- 仿真目标：车速与火车头牵引力 $F$ 的关系
- 设火车头质量为 $M1$ ，车厢质量 $M2$ ；
- 连接装置设为弹簧，弹性系数为 $k$
- 希望 $F$ 改变时，火车速度能尽快改变并与 $F$ 成正比。

# 火车系统仿真

## ■ 物理系统的抽象：





# 火车系统仿真

---

- 物理系统的数学模型：

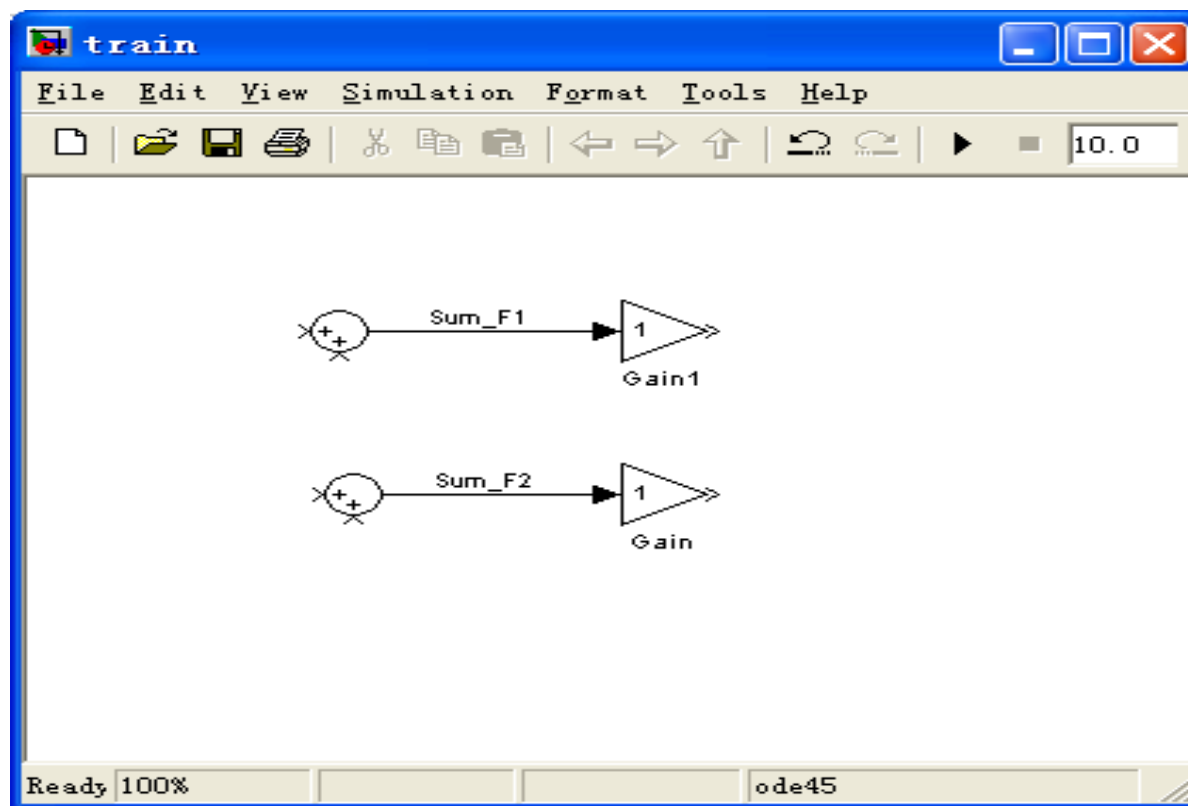
牛顿第二定律： **$F = ma$**

$$\Sigma F_1 = F - k(x_1 - x_2) - \mu M_1 g \dot{x}_1 = M_1 \ddot{x}_1$$

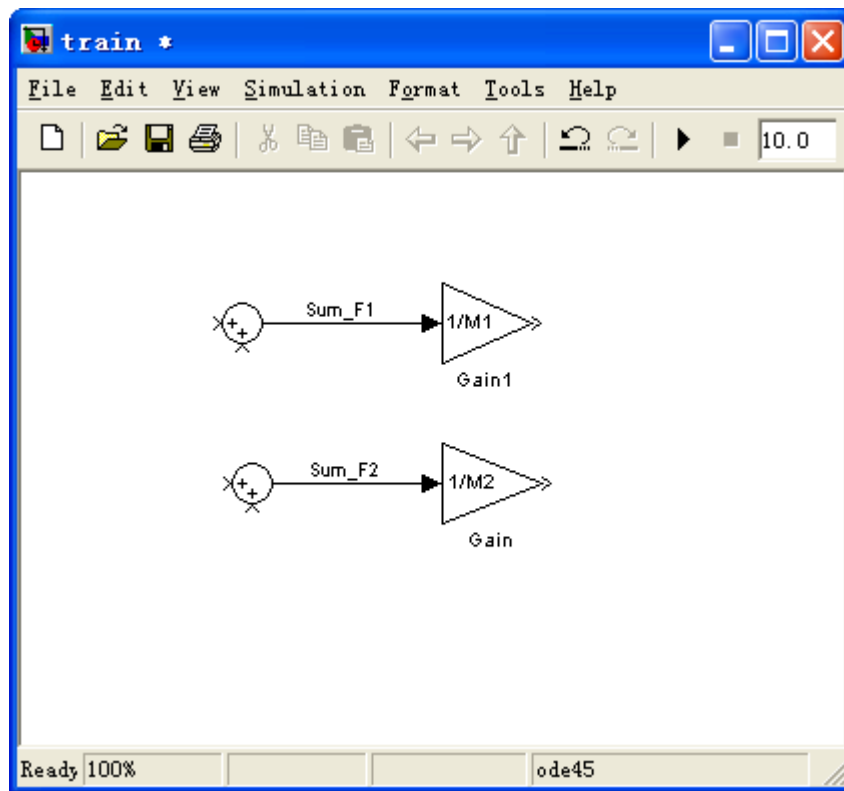
$$\Sigma F_2 = k(x_1 - x_2) - \mu M_2 g \dot{x}_2 = M_2 \ddot{x}_2$$

# 火车系统仿真

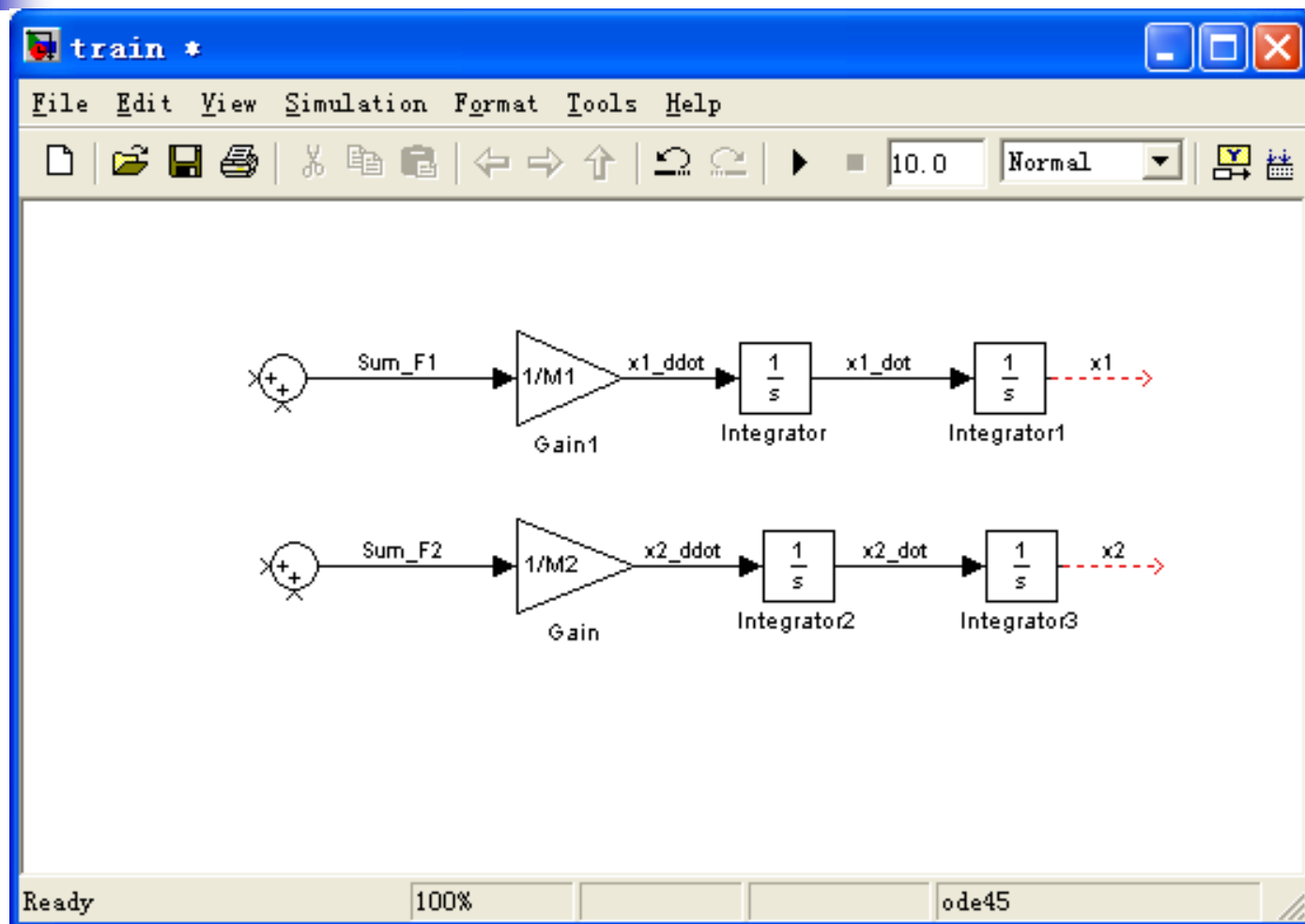
- 构建simulink模型



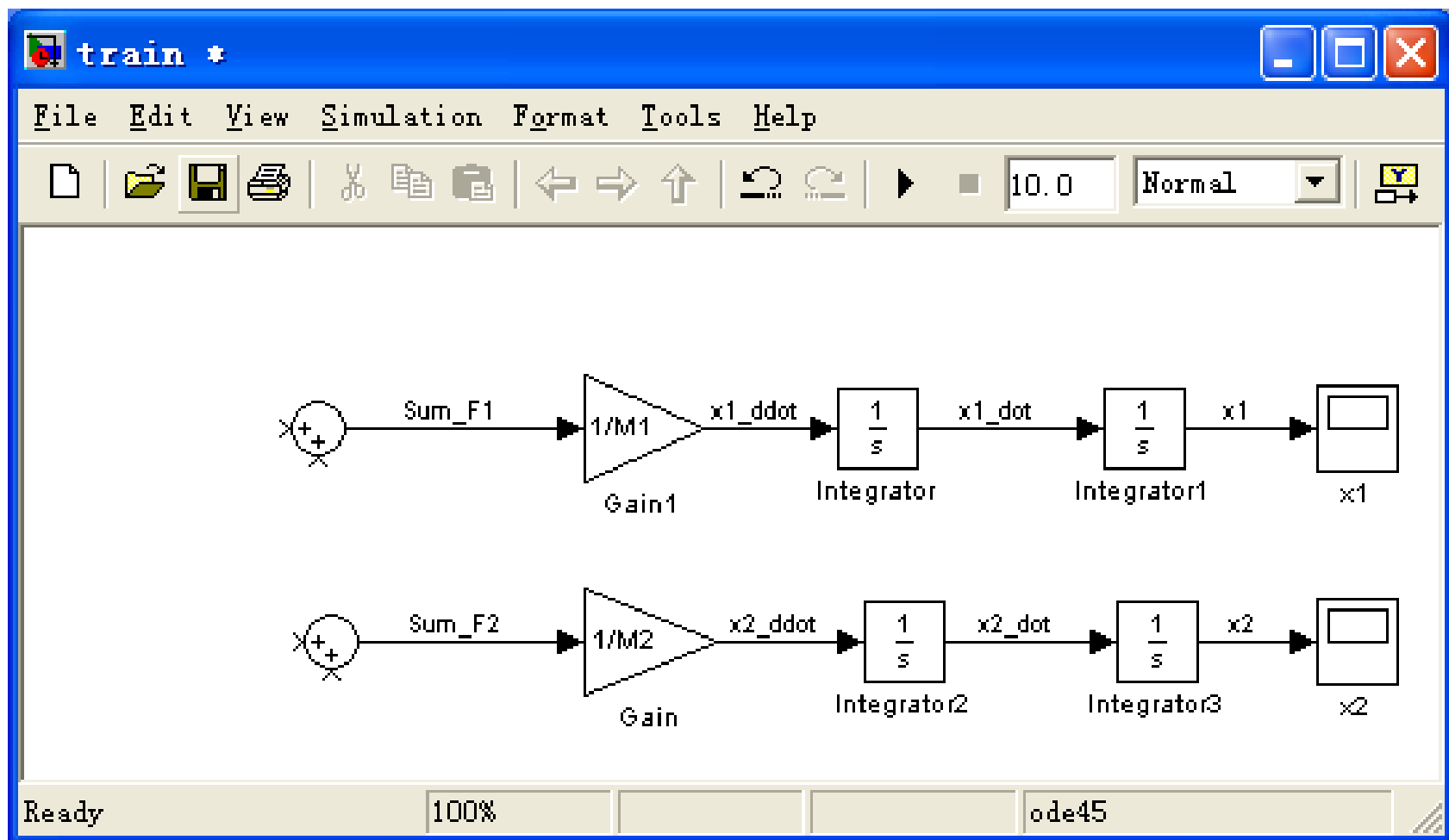
# 火车系统仿真



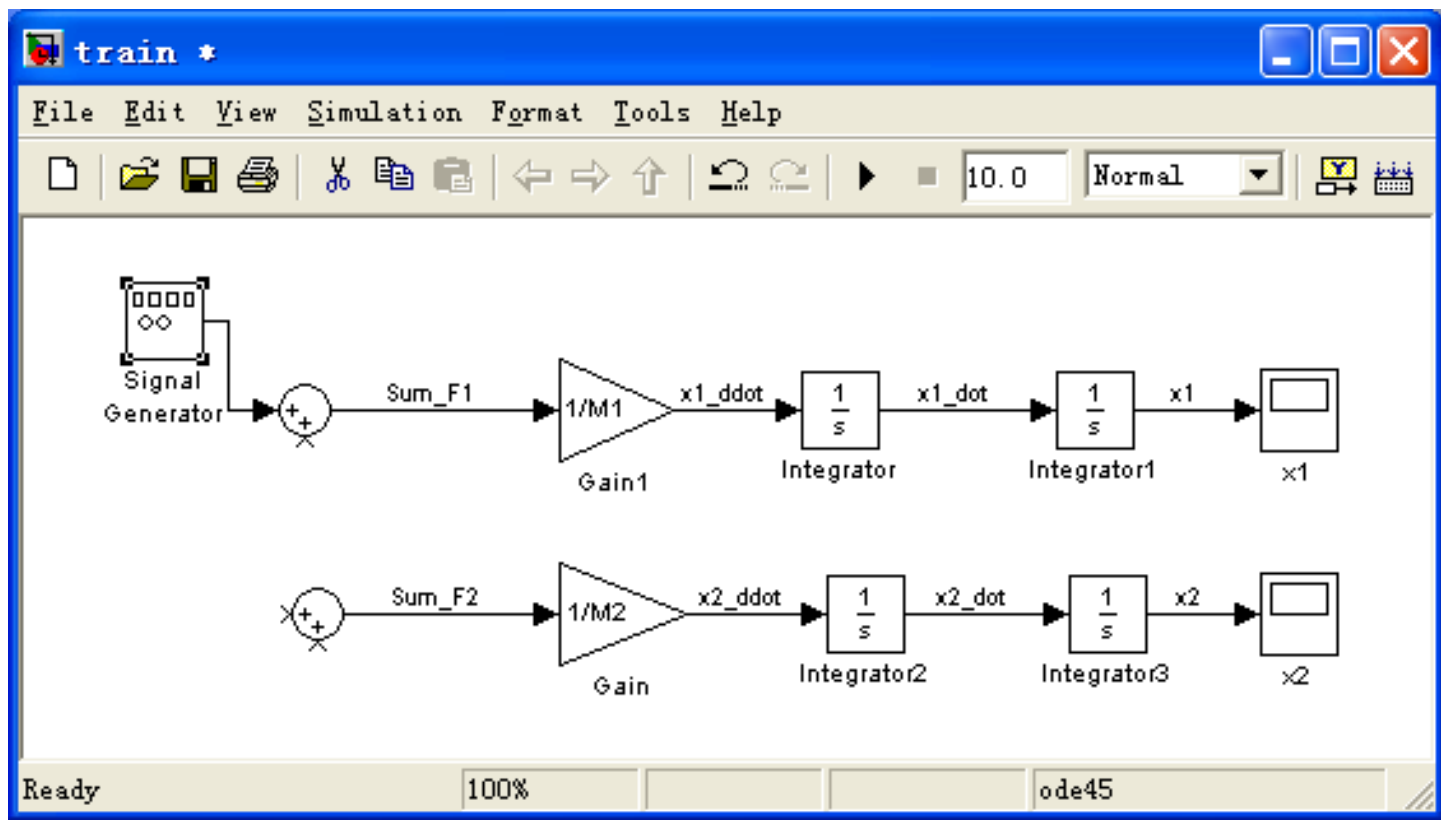
# 火车系统仿真



# 火车系统仿真

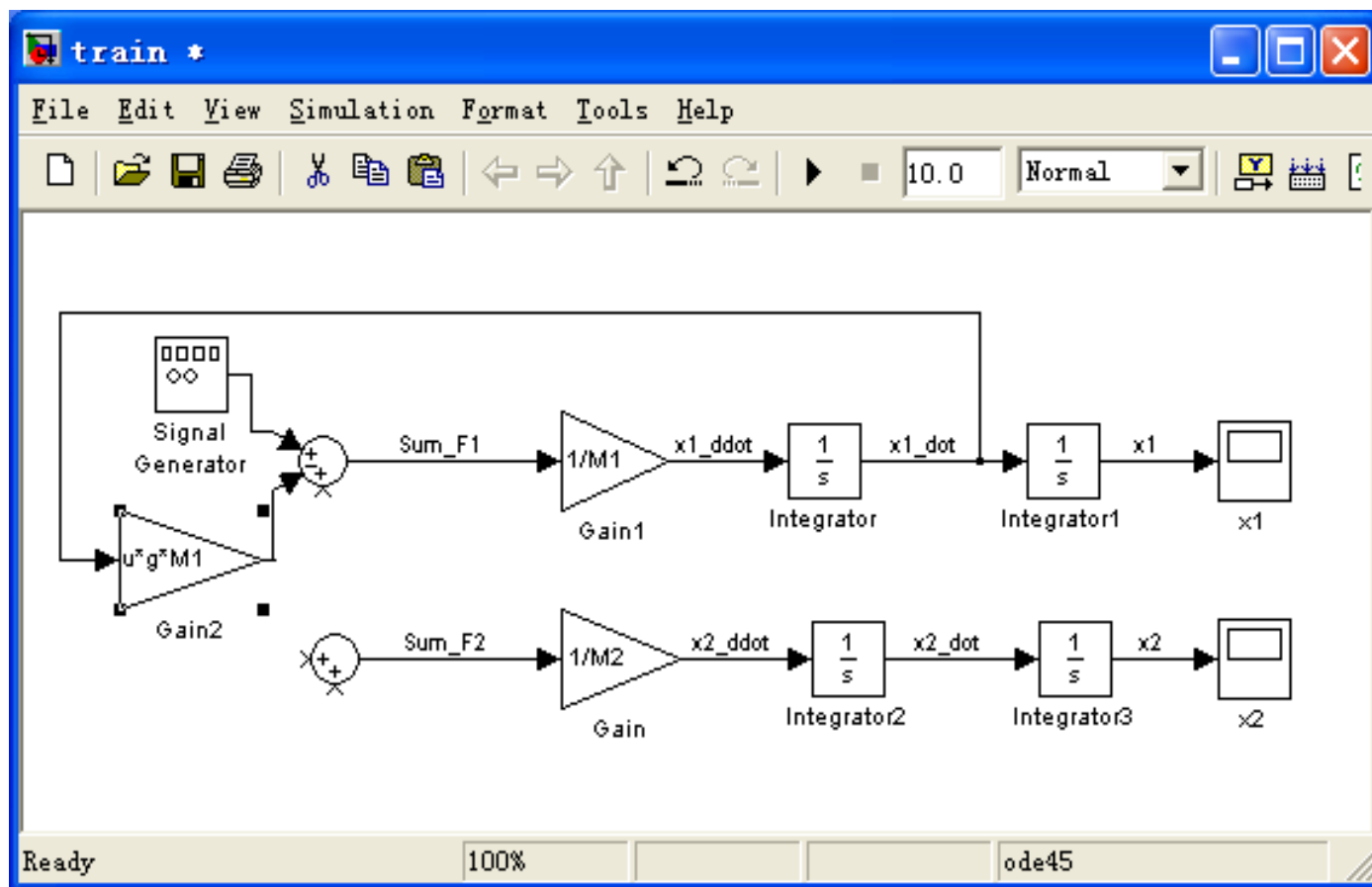


# 火车系统仿真

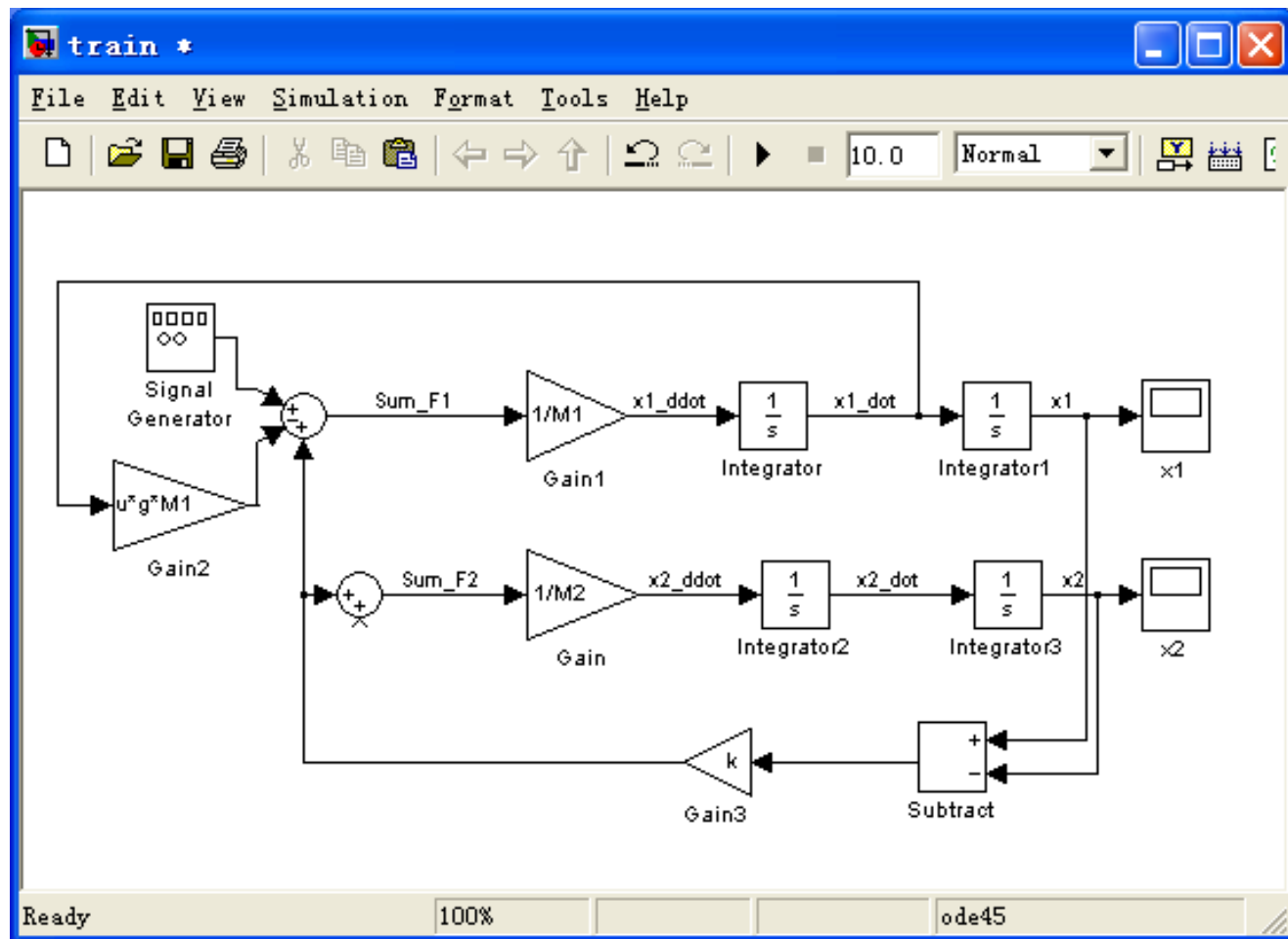




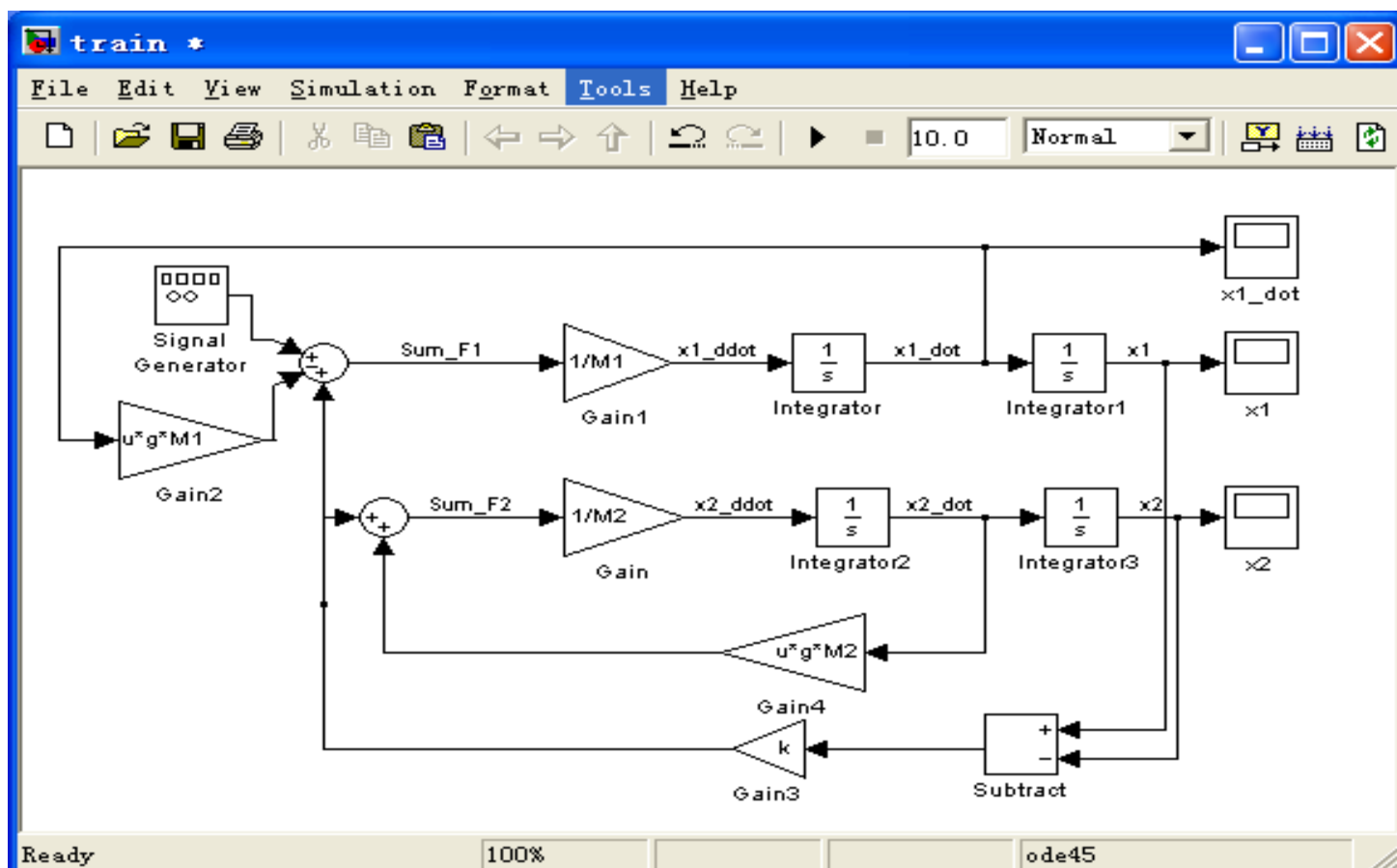
# 火车系统仿真



# 火车系统仿真



# 火车系统仿真





# 火车系统仿真

---

- 在命令行窗口输入相关参数

```
M1=1;
```

```
M2=0.5;
```

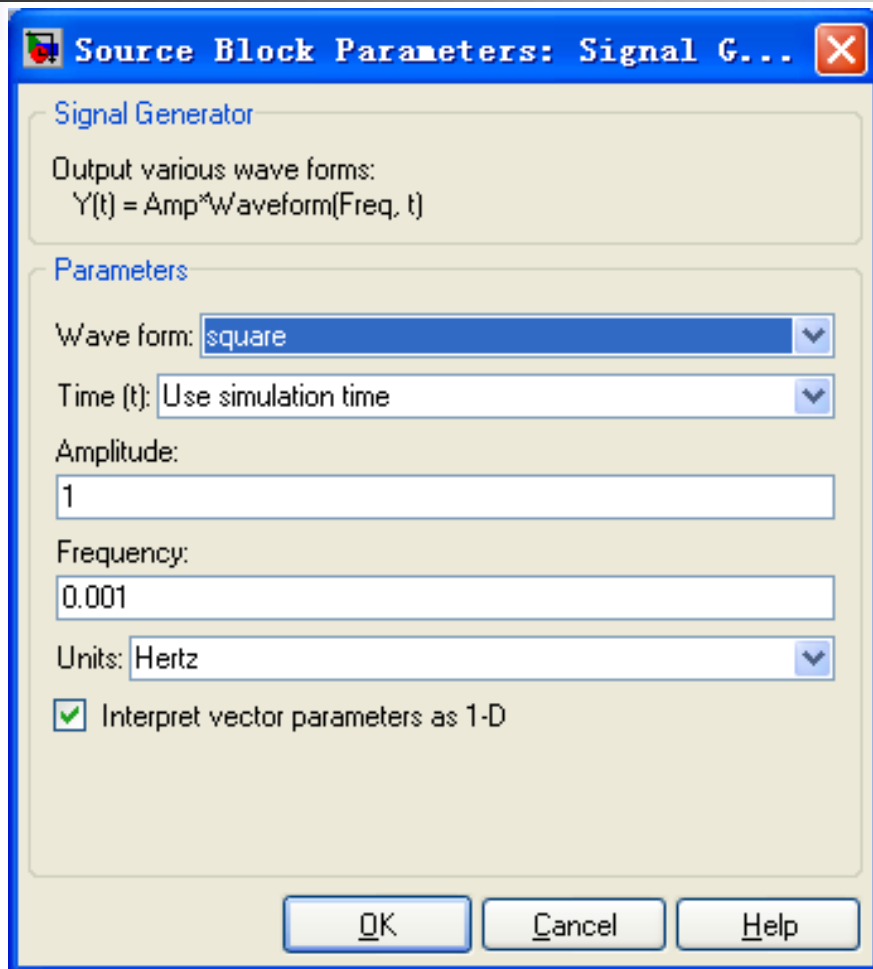
```
k=1;
```

```
F = 1;
```

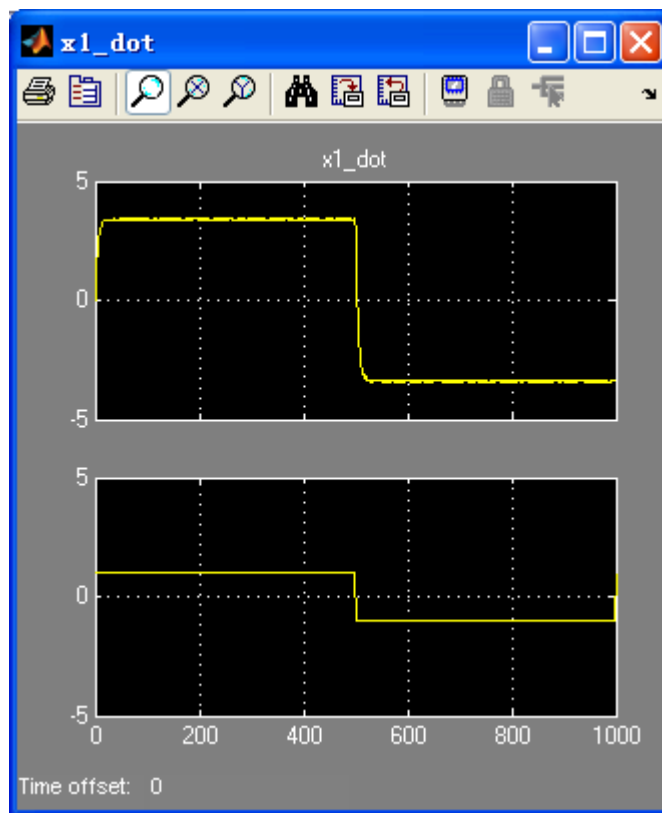
```
mu = 0.02;
```

```
g = 9.8;
```

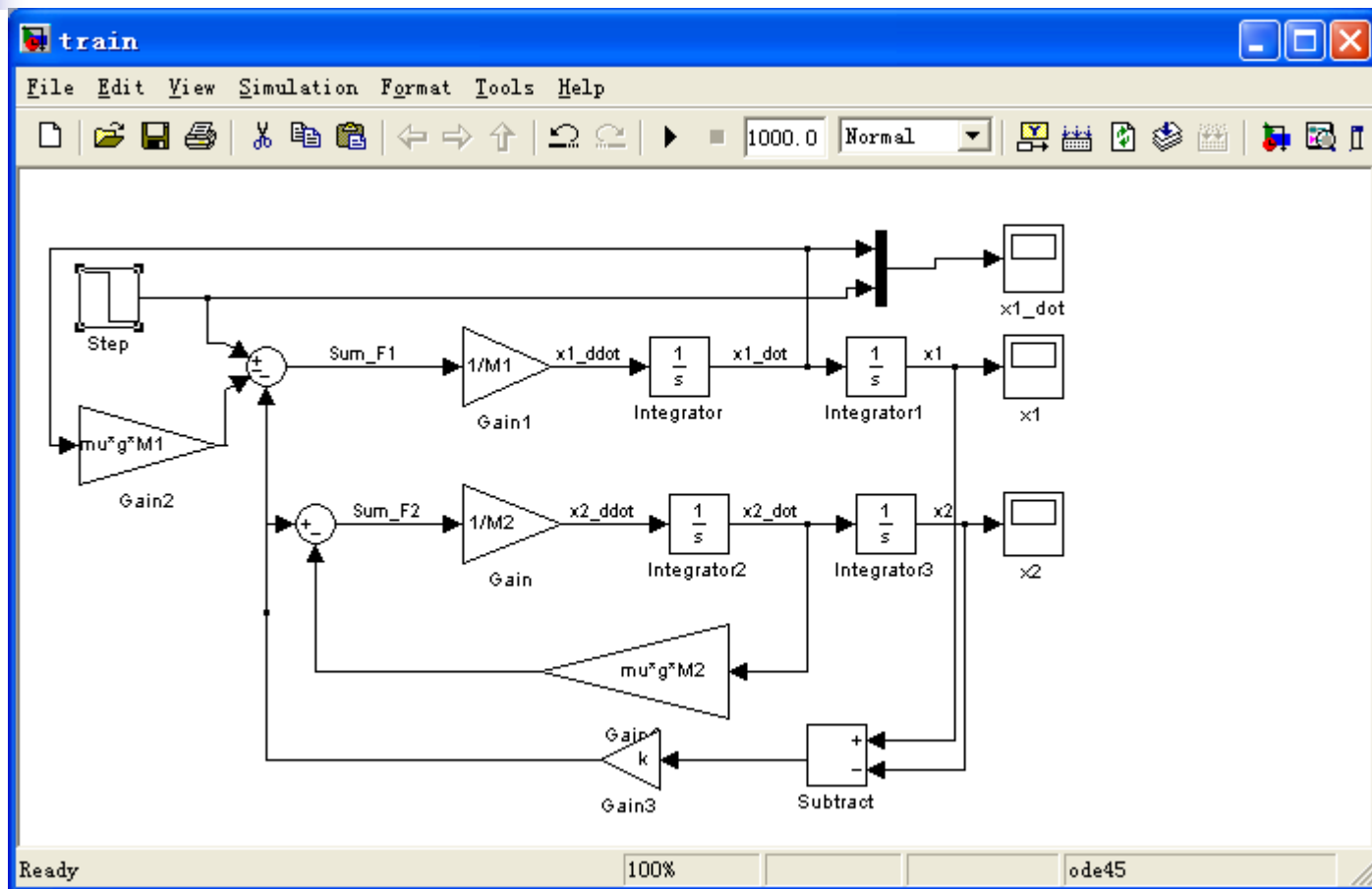
# 火车系统仿真

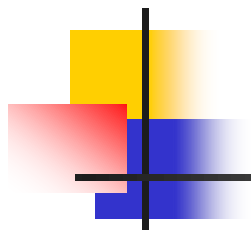


# 火车系统仿真



# 火车系统仿真





---

谢谢！