

RFID技术与MATLAB仿真基础

教师:聂云丽

时间: 2026.09



目录

CONTENTS

01 RFID技术概述

02 MATLAB仿真基础

03 项目实施

01

RFID技术概述

RFID技术简介

RFID的定义与别称



RFID（Radio Frequency Identification，RFID）射频识别技术，又称无线射频识别，是一种通信技术，可通过无线电信号识别特定目标并读写相关数据，而无需识别系统与特定目标之间建立机械或光学接触。它的标签常称为感应式电子晶片或近接卡、感应卡、非接触卡、电子标签、电子条码等。

核心特征

RFID的核心特征在于其无线通信能力。它能够通过无线电波在读写器和电子标签之间传输数据，突破了传统识别技术的限制，适用于多种复杂环境，为物联网的发展提供了强大的技术支持。

在物联网中的重要性

RFID技术是物联网中实现物品唯一标识和信息交互的关键技术之一。它能够为每一个物品赋予独特的身份标识，并通过网络实现信息的实时传输和共享，推动了物联网的广泛应用和发展。

面临的问题与发展趋势

尽管RFID技术具有诸多优势，但仍面临一些问题，如成本较高、隐私保护等。未来，随着标准化的推进、技术的小型化和成本降低，RFID技术将朝着更加高效、安全和广泛应用的方向发展。

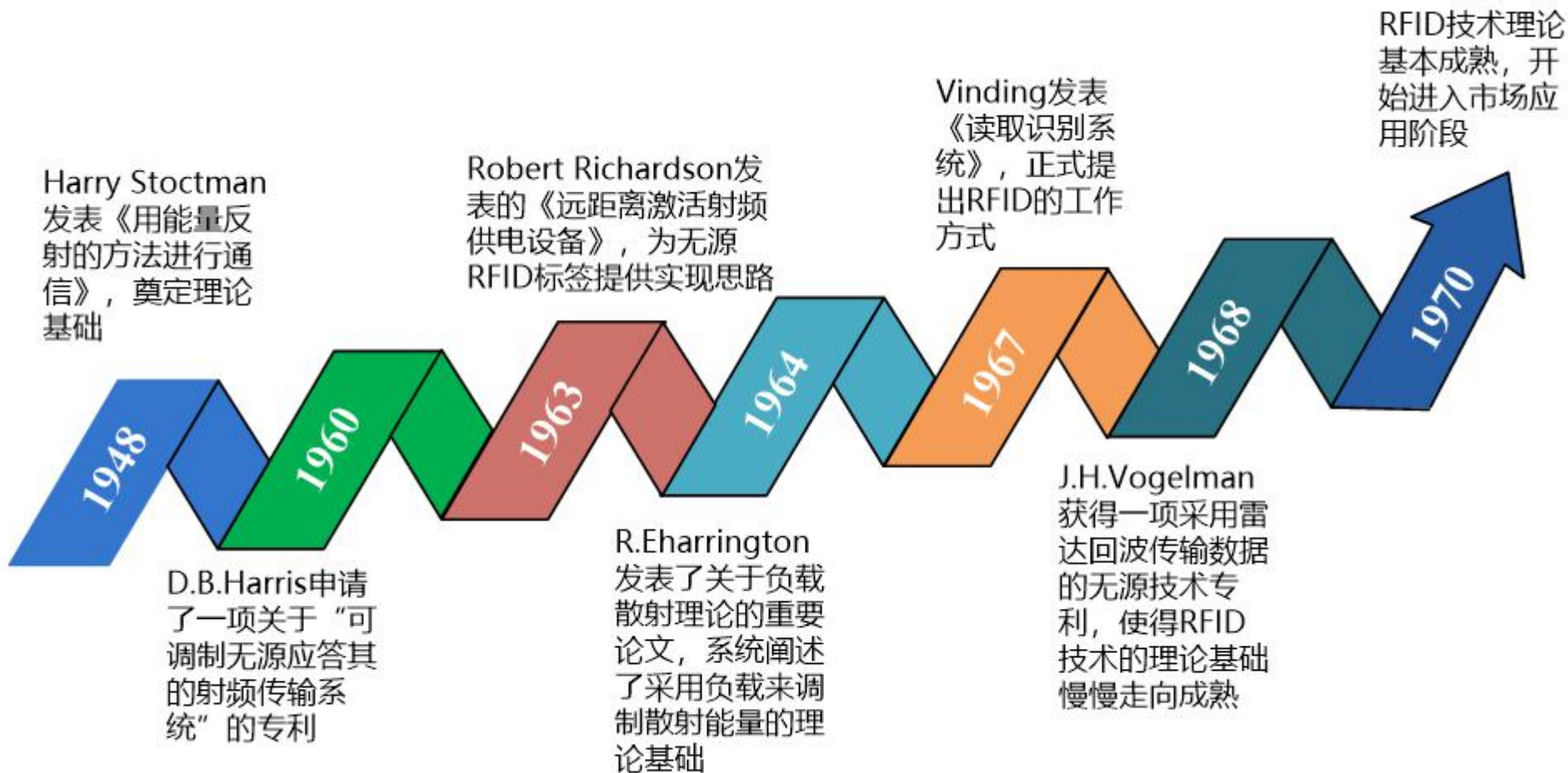
RFID技术的发展历程



发展历程概述

RFID技术的发展历程可以追溯到二战期间英国的无线电敌我识别系统。1948年，哈里·斯托克曼的论文为其奠定了理论基础。20世纪60年代，商品电子监视器成为其首个商业应用。90年代，公路自动收费系统推动了技术发展。2003年，沃尔玛等巨头的采用使RFID电子标签开始普及。进入21世纪后，随着标准化的推进和成本降低，RFID技术进入快速增长期，广泛应用于各行业。

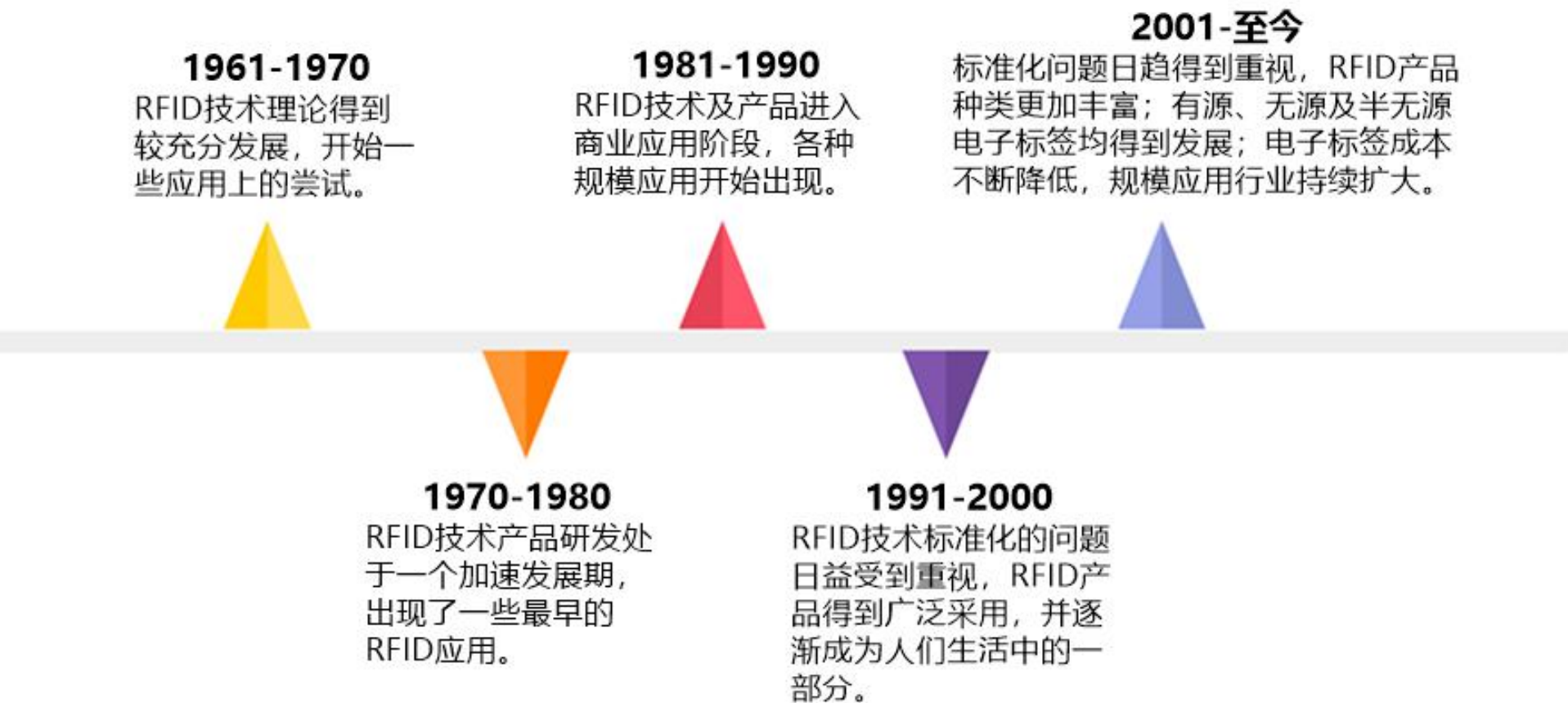
RFID技术的发展历程



RFID技术理论发展史
如图1-1所示，从该图可以看出，RFID技术理论从奠定基础到基本成熟后，RFID技术才开始进入的市场应用。

图 1-1. RFID 技术理论发展史

RFID技术的发展历程



RFID市场应用发展史
如图1-2所示，对照图1-1，从该图可以看出RFID技术理论得到充分发展并基本成熟后，RFID技术产品研发就进入加速发展，并开始进入市场应用阶段。

图 1-2. RFID 市场应用发展史。

RFID系统的工作原理

系统构成

RFID系统由电子标签、读写器和计算机网络构成。电子标签存储被识别物体的信息，读写器负责读取或写入标签信息，计算机网络则用于数据的处理和传输。



工作过程

读写器发射特定频率的射频信号，电子标签进入工作区域后被激活，将自身编码信息通过天线发射出去。读写器接收并处理信号，主机系统根据逻辑运算识别标签身份并做出相应处理。

通信方式

RFID系统有两种通信方式：电感耦合和电磁反向散射耦合。电感耦合适用于低频和高频系统，作用距离较短，常用于门禁卡等；电磁反向散射耦合适用于超高频和微波系统，作用距离较长，适用于物流和交通领域。

RFID系统的工作原理



系统构成

RFID系统由电子标签、读写器和计算机网络构成。电子标签存储被识别物体的信息，读写器负责读取或写入标签信息，计算机网络则用于数据的处理和传输。

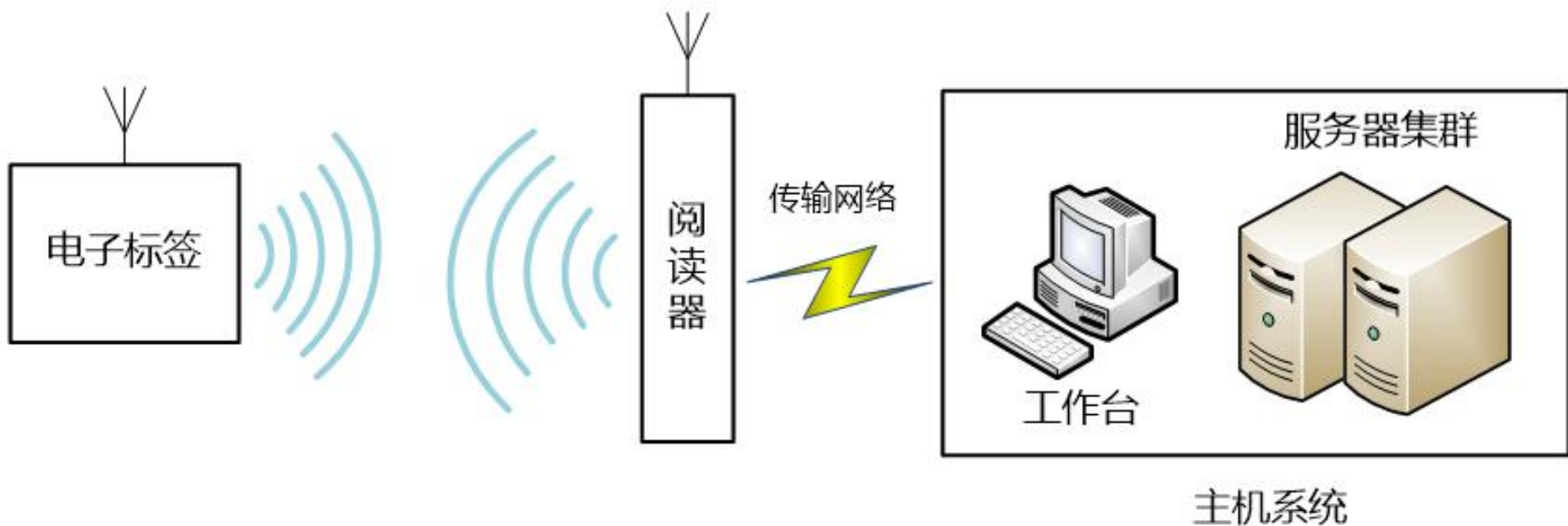


图 1-3. RFID 系统的基本工作原理如图。

RFID系统的组成结构



硬件组件

RFID系统的硬件组件包括电子标签、读写器、控制器、读写器天线和通信设施。电子标签存储信息，读写器负责读写操作，控制器协调各组件工作，天线用于信号传输，通信设施实现数据的有线或无线传输。



软件组件

软件组件包括边沿接口、中间件、企业应用接口和应用软件。它们在数据存储、管理和读写控制中发挥重要作用，确保RFID系统高效运行。

RFID系统的组成结构



典型的RFID系统主要由RFID读写器、电子标签、RFID中间件和应用系统软件4部分构成，一般把中间件和应用软件统称为应用系统，其系统构成如图1-5所示。

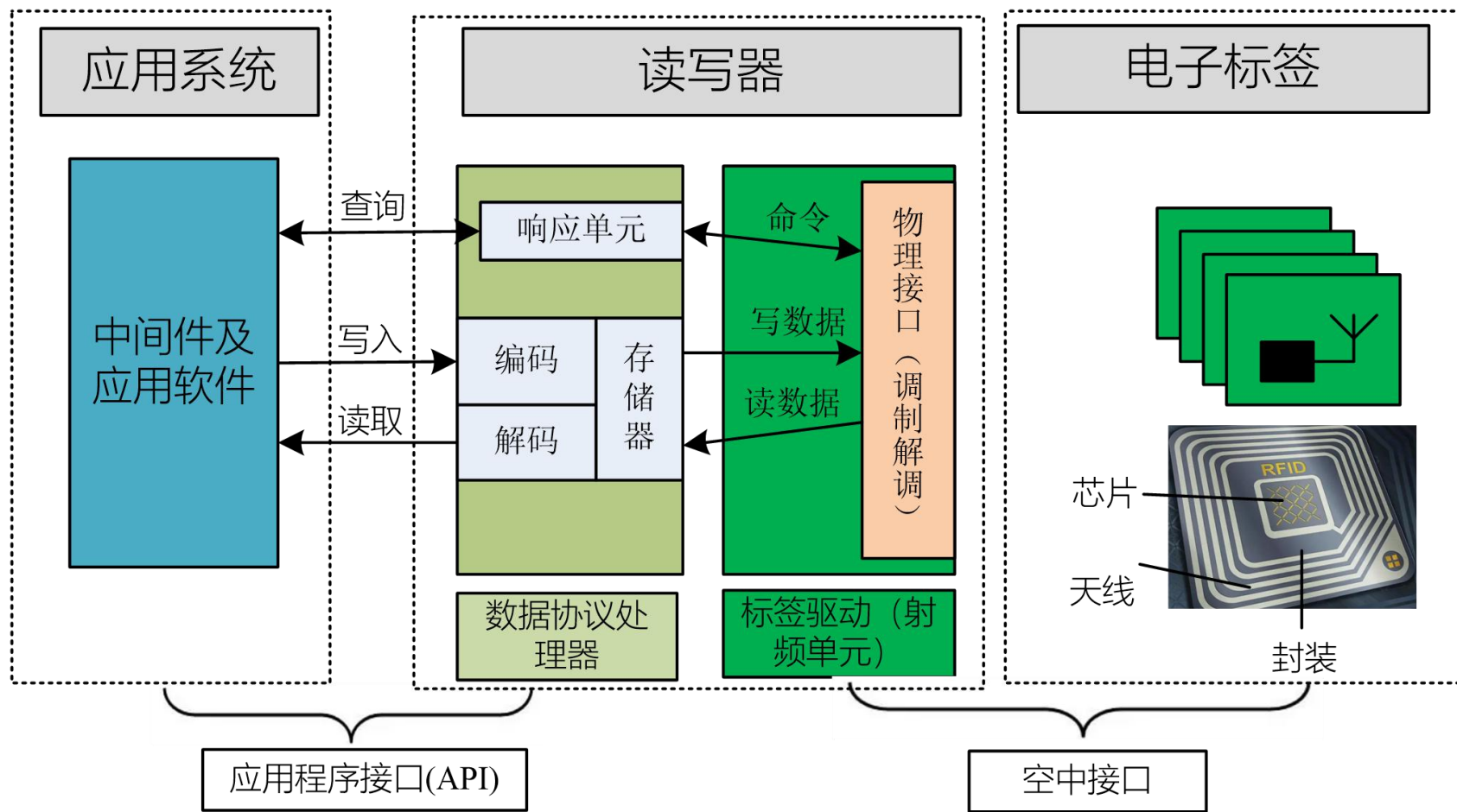
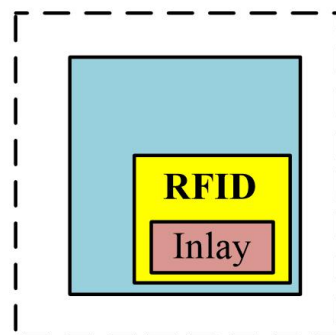


图1-5. 典型的RFID系统

RFID系统的组成结构

电子标签 (Electronic Tag) 也称应答器或智能标签 (Smart Label)，是一个微型的无线收发装置，主要由内置天线和芯片组成，如图1-6所示。



电子标签

- Inlay = 芯片+天线
- RFID = 加上保护层的Inlay
- 有不同的形状和大小尺寸
 - IC卡、标签(贴纸)、纽扣
- 标签的能量
 - 无源标签：由阅读器的电磁场提供
 - 有源标签：自带电源

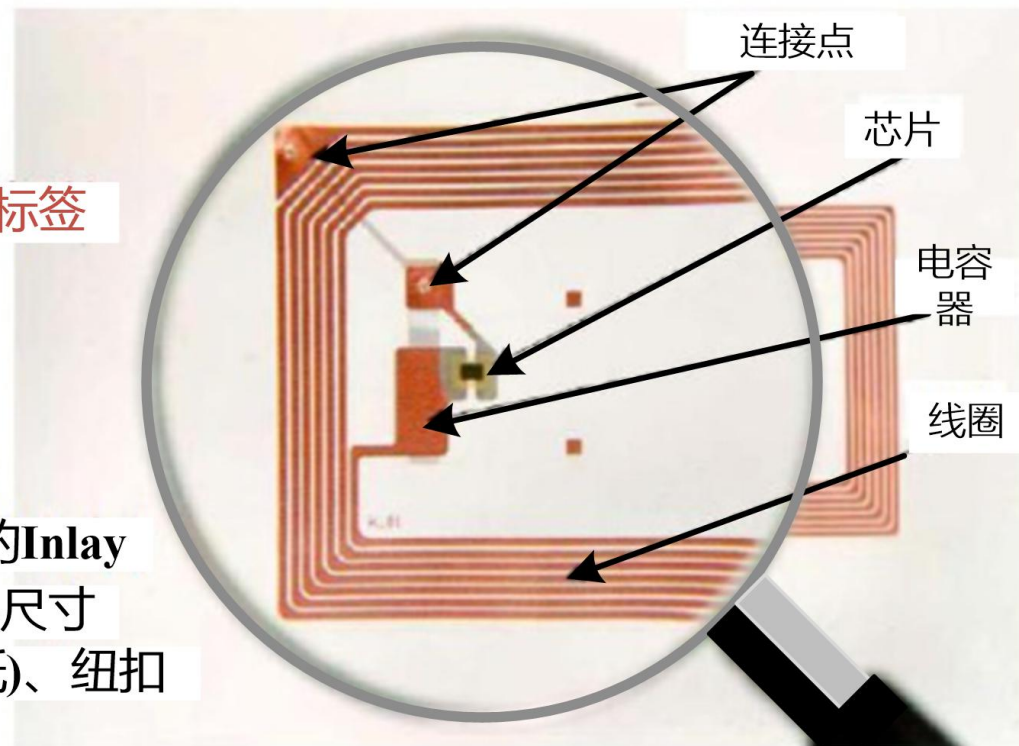


图1-6. 电子标签

RFID系统的组成结构

②、读写器

读写器是一个捕捉和处理RFID标签数据的设备，它可以是单独的个体，也可以嵌入到其他系统之中。读写器也是构成RFID系统的重要部件之一，由于它能够将数据写到RFID标签中，因此称为读写器。

读写器的硬件部分通常由收发机、微处理器、存储器、外部传感器/执行器，报警器的输入/输出接口、通信接口及电源等部件组成，如图1-7所示。

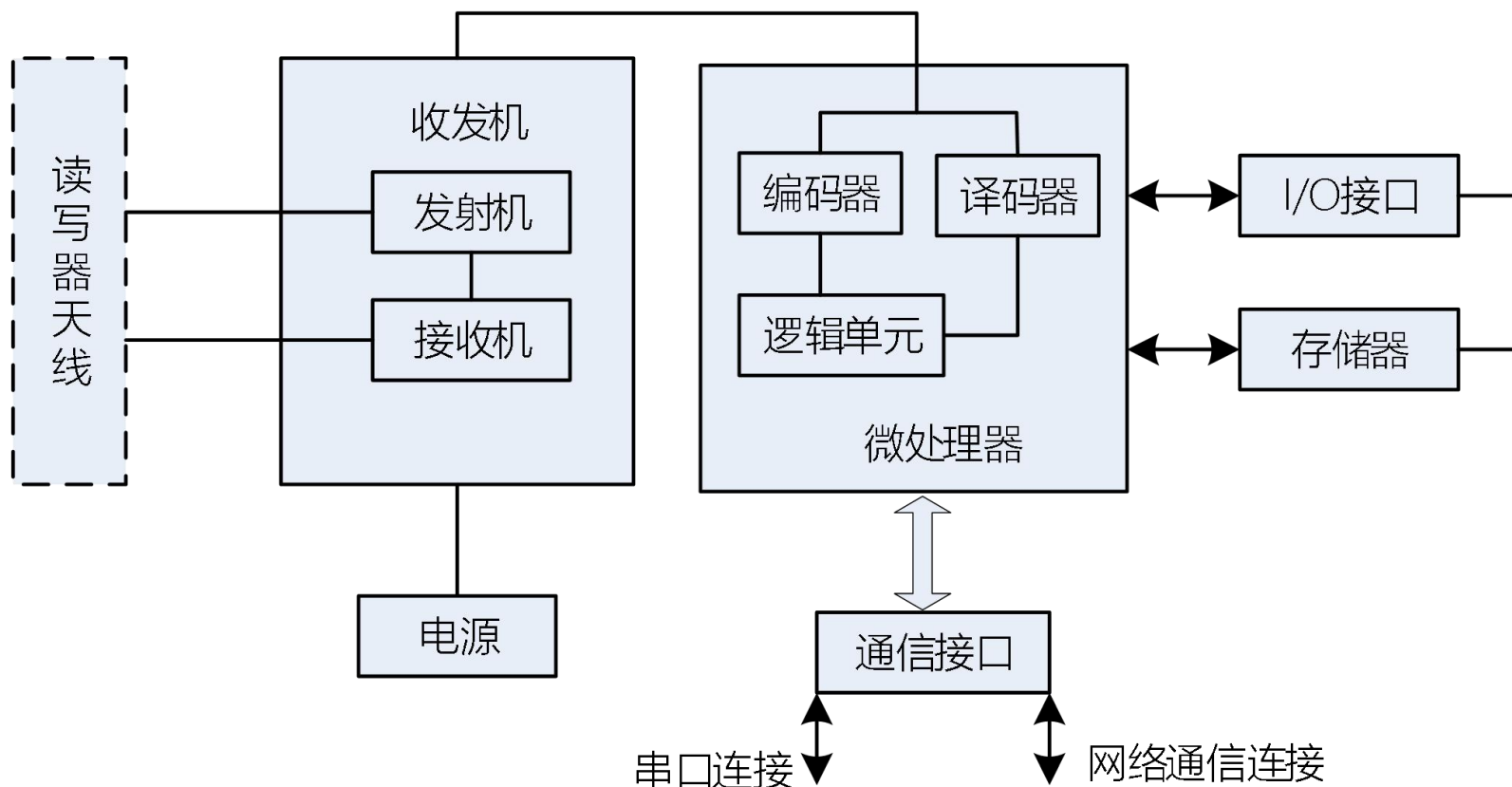


图1-7. 读写器的硬件部分



02

MATLAB仿真基础



MATLAB概述

01

MATLAB简介

MATLAB是由美国MathWorks公司开发的商业数学软件，广泛应用于数据分析、无线通信和深度学习等领域。它以矩阵计算为核心，集数值分析、数据可视化等功能于一体，是科研和工程领域的重要工具。

02

名称含义

MATLAB的名称来源于Matrix Laboratory，即矩阵工厂，反映了其强大的矩阵运算能力。它支持多种编程语言，能够满足不同领域的计算需求。

03

与其他软件的对比

MATLAB与Mathematica、Maple并称为三大数学软件。在数值计算方面，MATLAB具有显著优势，能够高效处理大规模数据和复杂算法。

04

应用范围

MATLAB在欧美高校和科研单位广泛应用，推动了科学研究和技术创新。在我国，MATLAB在科学研究和工程应用中也展现出巨大潜力，为相关领域的发展提供了有力支持。

MATLAB的特点

功能强大

MATLAB在数值运算、符号运算和数据可视化方面表现出色。它能够快速处理复杂数据，生成直观的图表，帮助用户更好地理解和分析结果。

算法自适应

MATLAB以数组为基本计算单元，提供了大量优化的功能函数。其编程语言简洁高效，能够自动适应不同类型的算法，提高开发效率。

语言简单

MATLAB的语法接近常规数学书写形式，易于学习和使用。它还配备了完备的帮助系统，用户可以快速查找和解决问题。

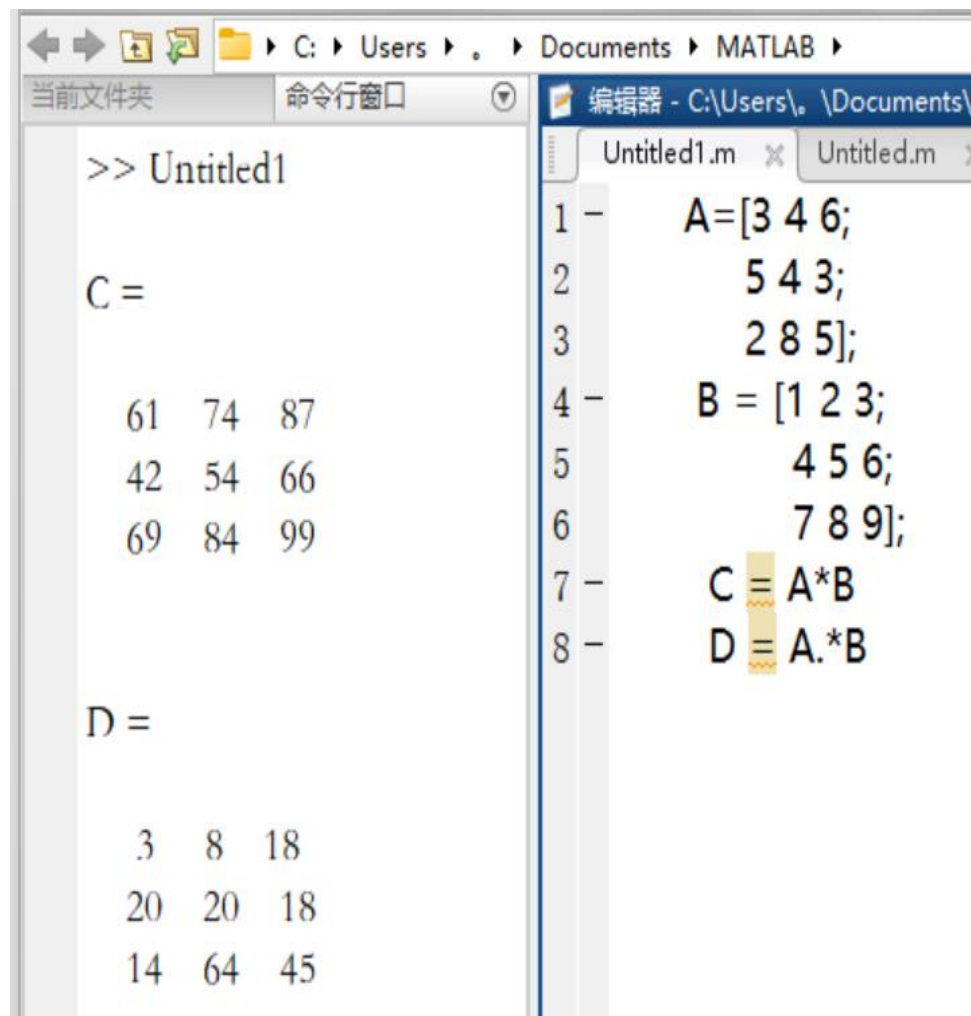
MATLAB的基本使用

(1) MATLAB简单计算

在没有定义变量名称时，数据暂存在ans中。定义变量之后，变量意义明显，计算过程一目了然，如图1-13所示。从该图可以看出，MATLAB对计算处理极其便捷。另外，它对于矩阵和函数的计算也是非常便捷的，因此，MATLAB在计算功能方面具有极其强大的功能。比如计算矩阵A和矩

阵B的乘积，即 $A \times B$ ，其中 $A = \begin{bmatrix} 3 & 4 & 6 \\ 5 & 4 & 3 \\ 2 & 8 & 5 \end{bmatrix}$ ，

$B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ ，计算如图1-14所示。



```
>> Untitled1

C =

    61    74    87
    42    54    66
    69    84    99

D =

     3     8    18
    20    20    18
    14    64    45
```

```
1 - A=[3 4 6;
2     5 4 3;
3     2 8 5];
4 - B = [1 2 3;
5         4 5 6;
6         7 8 9];
7 - C = A*B
8 - D = A.*B
```

MATLAB的基本使用



(2) 基本运算符号

Matlab进行数学处理时的优先级顺序和通常进行数学处理的优先级顺序一致，表现为指数优先，乘、除次之，加减再次之，逻辑运算最后；括号改变优先级，表达式从左到右依次运算，具体如下。

优先级	运算法则	优先级	运算法则	优先级	运算法则
1	()	5	:	9	&&
2	' , ^ , .^	6	> , >= , < , <= , = = , ~=	10	
3	+ , - , ~	7	&		
4	* , / , .* , ./	8	\		

MATLAB的基本使用



(3) 数值、变量及表达式

MATLAB的数只采用习惯的十进制表示，可以带小数点和负号，缺省了双精度浮点型数据，例如：3，-10，0.001， $1.3e10$ ， $1.13e-6$ 。

变量命名规则如下：

- ① 变量名、函数名对字母区分大小写；
- ② 变量名可以包含英文字母、下划线和数字，但必须以英文字母开头；
- ③ 变量名最多可以包含63个字符。

MATLAB预定义变量如下。

变量名	意义	变量名	意义
ans	最近计算结果	eps	正极小值 $2.2204e-16$
pi	圆周率	inf	无穷大
i或j	虚数单元	NaN	不是一个数字，0/0



MATLAB的基本使用



(4) 数组

① 数组生成

对于一维数组，有三种方法生成。第一是直接输入法，这种方法通过空格、逗号和分号来分隔数组元素，在数组中输入任意元素，生成一维数组，比如 $a = [1\ 2\ 3]$ ；第二是步长生成法，表现为 $x = a:step:b$ ，如 $a = 1:0.1:2$ ；第三是等间距线型生成法，表现为 $x = linspace(a,b,n)$ ，表示在表现为 a 到 b 区间内生成 n 个采样点数据。



Matlab的编程基础



流程控制

① for循环结构

for循环结构中，需要设定一定的循环条件，Matlab根据设定的循环次数执行循环体内的命令。其格式如下：

```
for x = array
    commands
end
```

其中，x是循环变量，array是条件数组，commands是要执行的循环代码。循环体的执行次数由array决定。

for循环结构示例如下

```
for n = 10 : -1 : 1
    x(n) = sin(n * pi / 5);      %执行的循环代码;
end
x
array = 1:10;                  %条件数组;
for n = array
    x(n) = sin(0.1*2*pi*n * pi / 5); %执行的循环代码;
end
x
```



Matlab的编程基础



② while循环结构

while循环结构对循环体进行无限次的循环运算，直到循环体满足循环结束条件，或达到一定的循环次数后终止。其格式如下：

```
while expression
    commands
end
```

其中，**expression**是条件表达式，一般情况下，**expression**的计算结果是一个标量，但也可以是一个数组表达式。当标量结果为true时，循环体一直被执行下去；当**expression**的结果是一个数组时，只有当数组中所有元素均为true时，循环体才会被执行。

```
clc;
n = 0;
for r=1:100
    while r+1>10    %条件表达式,判断r+1是否大于10;
        n= 10      %条件成立则n=10;
        break;
    end
end
```



Matlab的编程基础



③ if判断结构

if判断结构是根据某一给定的条件，来进行判断以执行不同的命令。其格式如下：

```
if expression
    commands
end
if expression
    commands1
else
    commands2
end
```

其中if-else-end判断结构是 处理判断条件为true或false时，分别执行不同的命令行。

当expression包含多条子逻辑表达式时，Matlab采用“避绕式”计算各表达式，例如（expression1 | expression2），当计算expression1为false才会计算expression2。示例如下：

```
clc;
clear all;
n = 0;
m = 0;
k = 0;
r = input('please input a number:'); %输入数字;
if r>10
    n = 10;
elseif r<10
    m = 5;
else
    k = 50;
end
```



Matlab的编程基础



控制命令

在编写Matlab的M文件时，常常用到各种流程控制结构，这些流程结构执行过程中，常需要提前终止循环、跳出子程序等操作，需要使用控制语句来实现，这里主要介绍常用的continue和break语句。

continue命令主要用在循环语句中，提前结束循环体的本次操作，将continue直接放在循环控制体内，以配合if判断语句使用。

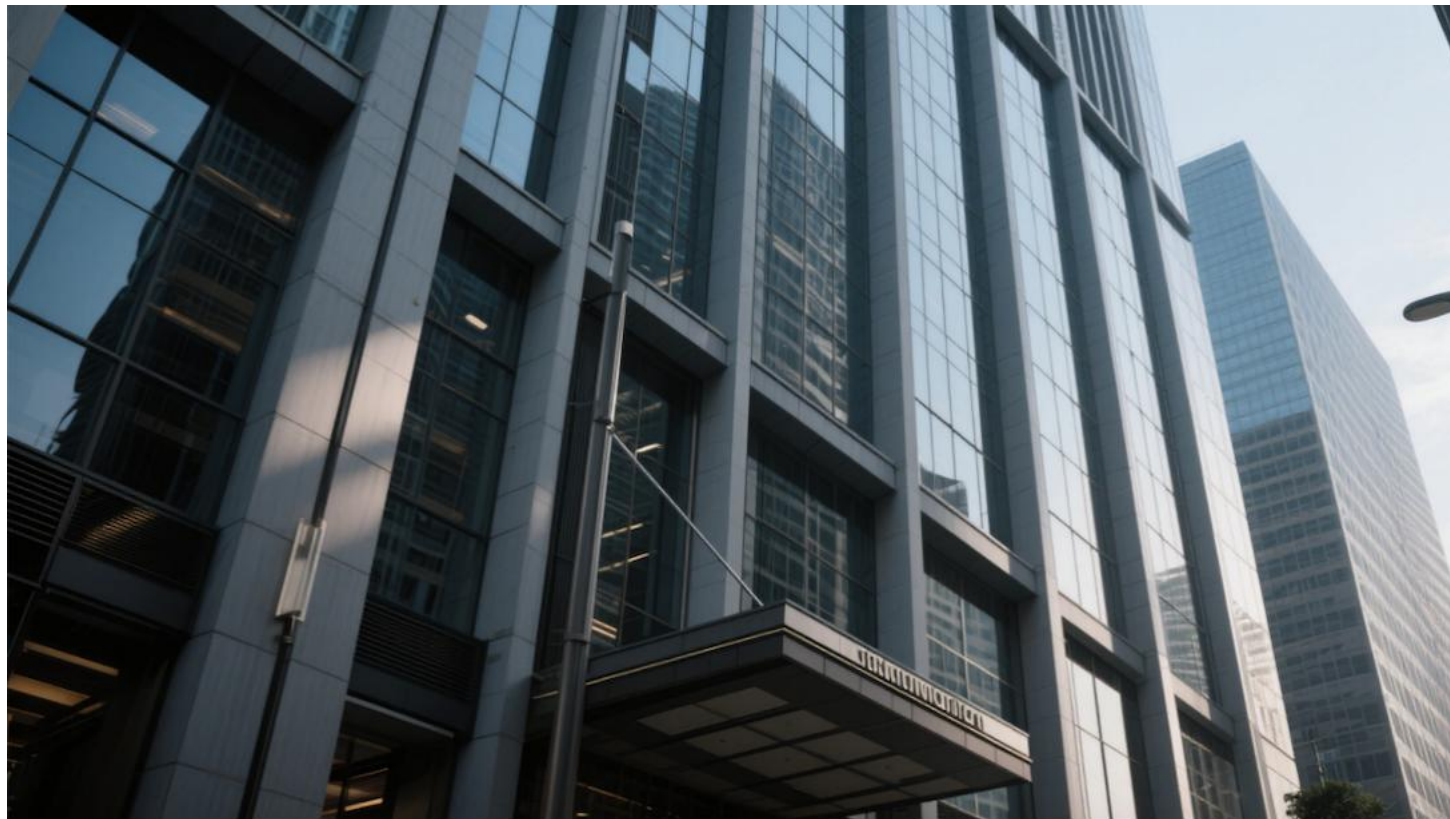
break命令和continue命令一样都用在循环结构中，当执行break命令时，程序跳出循环结构，转移到循环之外的下一条语句。

continue命令使程序跳转到循环的end语句处，结束一次循环，而break命令使程序跳出break所在的循环体。具体示例如下：

```
%计算浮点数eps的精度
EPS = 1;
for n = 1:1000
    EPS = EPS / 2;
    if (1 + EPS) > 1
        continue
    end
    EPS = EPS * 2;
    break;
end
EPS
```



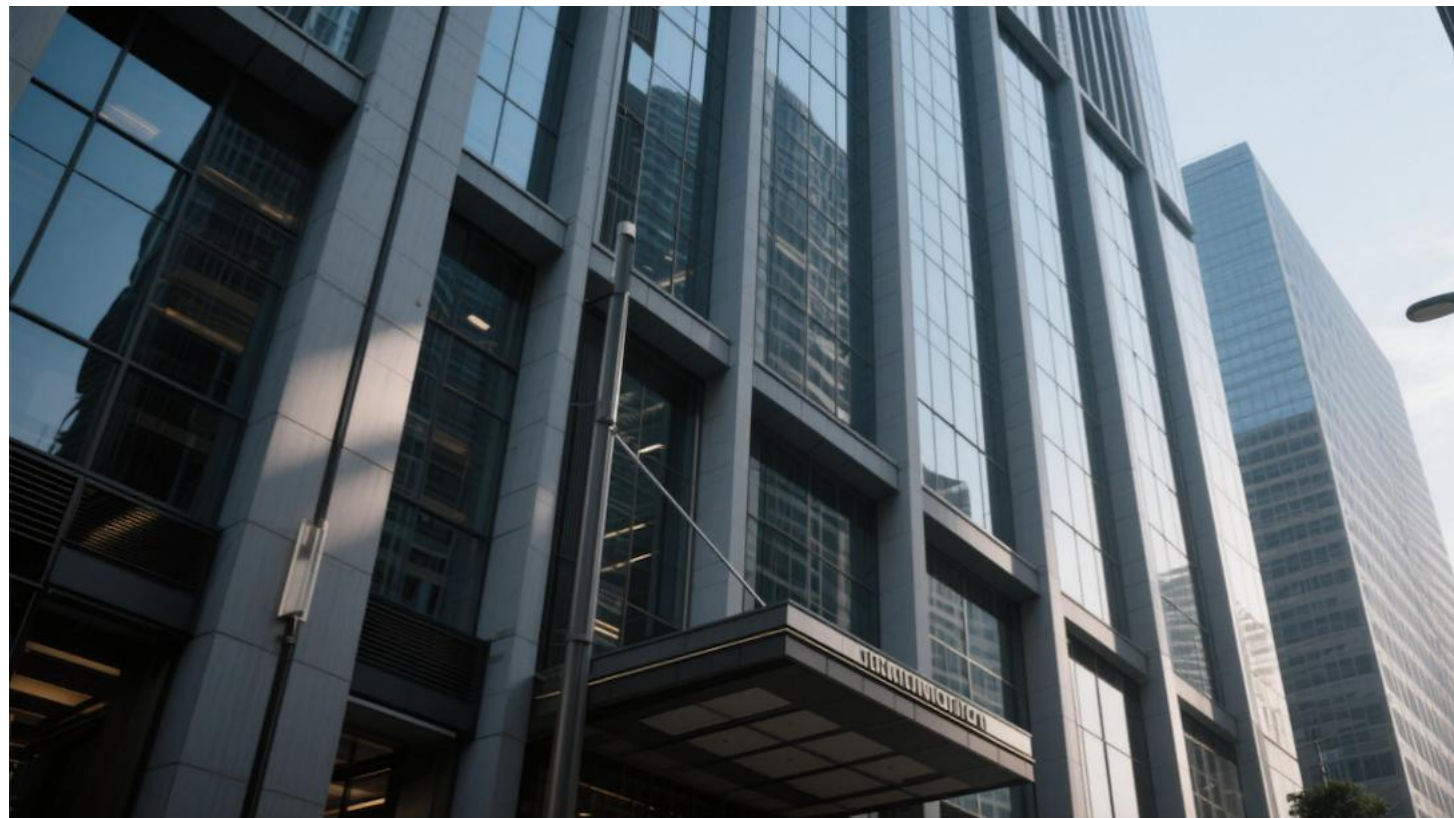
Matlab的simulink基础



Simulink是MATLAB的重要组成部分，用它来进行RFID系统仿真有利于读者对RFID有更直观的认识。它通过可视化的设计，非常方便系统仿真的构建。



Matlab的simulink基础



(1) Simulink的启动与退出

在MATLAB的命令行窗口输入Simulink命令，或选择MATLAB主窗口“主页”选项卡，单击SIMULINK 命令组中的 Simulink 命令按钮，或选择 MATLAB 主窗口 “主页” 选项卡，单击“新建” 命令按钮，再从下拉菜单中选择 Simulink Model 命令，这时进入 Simulink 起始页。接着在Simulink起始页单击Blank Model按钮，打开一个名为untitled的模型编辑窗口，如下图所示。利用模型编辑窗口，可以通过鼠标的拖放操作创建一个仿真模型。

Matlab的simulink基础

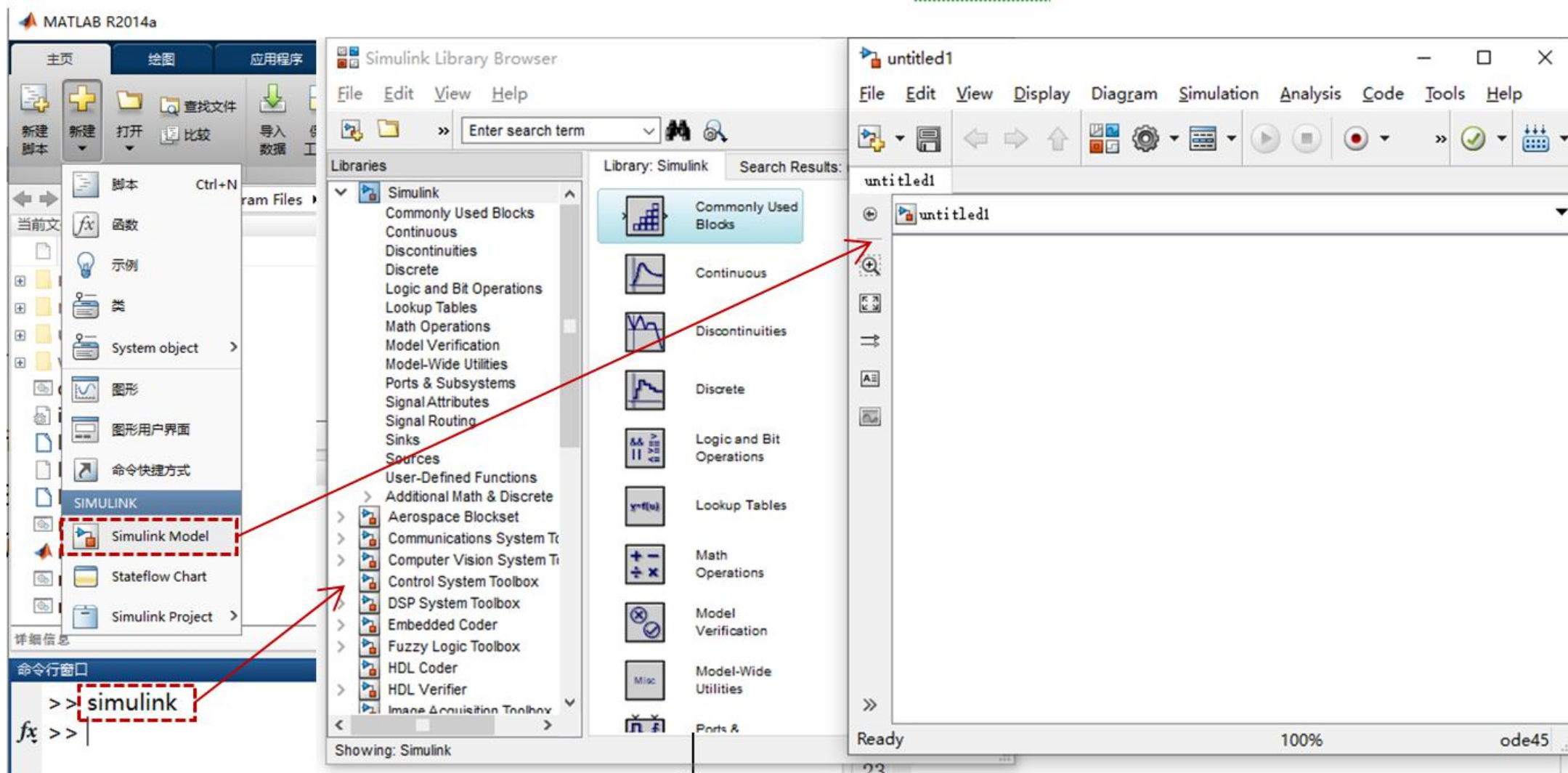
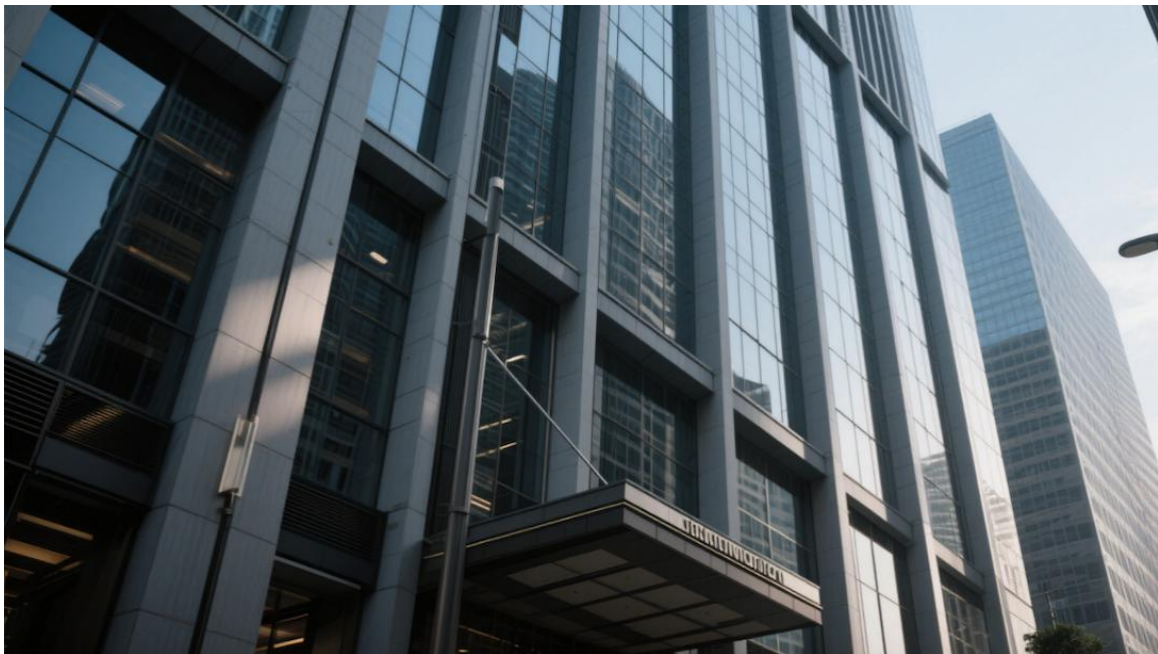


图1-15 Simulink的启动



Matlab的simulink基础



① 模型元素

典型的Simulink模型包括以下3种元素。

A.信号源（Source）。信号源可以是 Constant（常量）、Clock（时钟）、Sine Wave（正弦波）、Step（单位阶跃函数）等。

B.系统模块。例如，Math Operations模块（数学运算）、Continuous模块（连续系统）、Discrete 模块（离散系统）等。

C.信宿（Sink）。信号可以在 Scope（示波器）、XY Graph（图形记录仪）上显示，也可以存储到文件（To File），导出到工作空间（To Workspace）。

② 仿真步骤

利用 Simulink 进行系统仿真通常包括以下步骤。

A.建立系统仿真模型，包括添加模块，设置模块参数，进行模块连接等操作。

B.设置仿真参数。

C.启动仿真并分析仿真结果。



03

项目实施

项目实例1

简单实例

下面利用Simulink仿真曲线 $y(t) = \sin(10\pi t) \times \sin(20\pi t)$ ($0 \leq t \leq 4$) 说明利用Simulink建立仿真模并进行系统仿真的方法。

首先打开一个模型编辑窗口，创建仿真模型；其次打开Simulink Library Browser窗口，将所需模块添加到模型中，如图1-16所示，其中正弦信号由信号源模块库（Sources）中的Sine Wave 模块提供，求积用数学运算模块库（Math Operations）中的Product 模块实现，再用信号输出模块库

（Sinks）中的示波器模块（Scope）输出波形，把这些模块连线后，单击运行按钮，然后双击编辑界面的示波器模块。

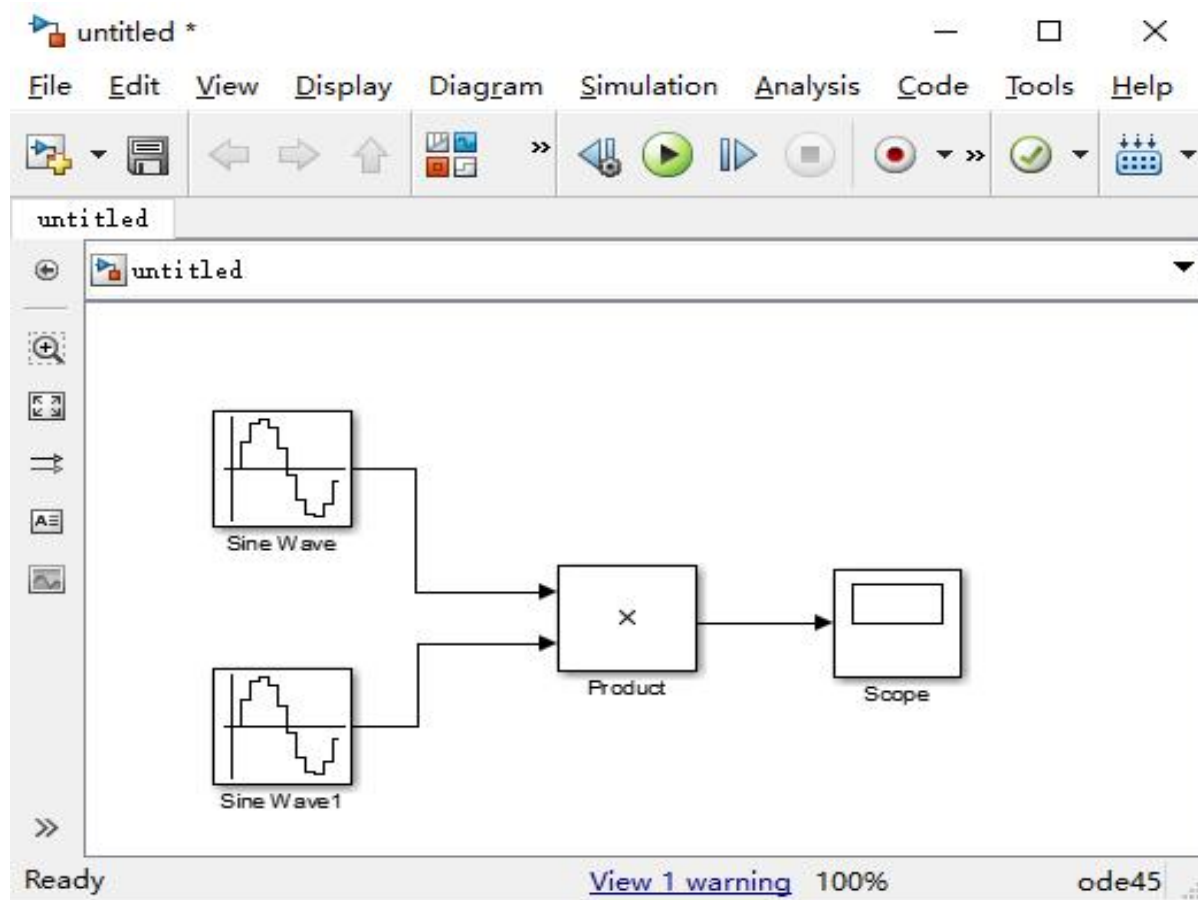


图1-16 仿真模型

项目实例2

用Matlab程序实现混合带噪信号的滤波

现用Matlab程序实现混合带噪信号的滤波。设定带噪的混合信号为 $y(t) = \sin(10\pi t) + \sin(30\pi t) + v(t)$ ，其中 $v(t)$ 是高斯白噪声。对于这个混合带噪信号，我们需要通过低通滤波去除噪声和其中的高频信号 $\sin(30\pi t)$ ，实现程序如下。

```
clc; %清屏;
clear all; %清除工作空间的所有变量,函数,和MEX文件;
pi = 3.1415926;
f1 = 5; %设置第一个信号的频率;
f2 = 15; %设置第二个信号的频率;
k = 16;
fs = k*f2; %采样频率;
ts = 1/fs; %采样时间;
t = 0:ts:1.5;
y = sin(2*pi*f1*t)+sin(2*pi*f2*t); %混合无噪信号;
yn = awgn(y,5); %添加5dB的高斯白噪声;
cutoff_freq = 6; %截止频率 (Hz) ;
order = 4; %滤波器阶数;
cfn= cutoff_freq / (fs / 2); %计算截止频率对应的正切值;
[b, a] = butter(order, cfn, 'low'); %设计低通滤波器;
filtered_signal = filter(b, a, yn); %得到滤波后的信号;
figure(1) %分别显示三个波形图;
plot(t, y);
figure(2)
plot(t,yn);
figure(3)
plot(t, filtered_signal);
```