

<<案例解说单片机C语言开发>>

图书基本信息

书名：<<案例解说单片机C语言开发>>

13位ISBN编号：9787121181719

10位ISBN编号：7121181711

出版时间：2012-9

出版时间：电子工业出版社

作者：程国钢

页数：416

字数：684000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<案例解说单片机C语言开发>>

### 内容概要

#### Keil

Vision是目前应用最广泛的51单片机软件开发环境，Proteus是目前应用最广泛的硬件仿真环境。

本书基于Keil

Vision和Proteus介绍了51单片机的体系结构、C51语言、内部资源，以及常用扩展器件的使用方法。

全书分为三部分：第1、2章是基础部分，介绍了51单片机的基础知识、Keil

Vision和Proteus的基础用法；第3~9章是基础应用部分，基于Proteus和Keil

Vision介绍了51单片机的内部资源和典型外部扩展器件的使用方法；第10章是综合应用部分，介绍了51单片机应用系统的基础设计方法，提供了包括频率计、波形发生器、密码保险柜、电子钟、RTX51操作系统在内的5个大型综合应用实例。

本书中提供了大量实例，它们都有详细的设计思路、典型器件列表、Proteus应用电路、Keil  $\mu$  Vision应用代码和仿真运行结果。

## <<案例解说单片机C语言开发>>

### 书籍目录

#### 第1章 51单片机的应用基础

##### 1.1 51单片机的体系结构介绍

###### 1.1.1 51单片机的8位处理器

###### 1.1.2 51单片机的存储器

###### 1.1.3 51单片机的外部引脚

###### 1.1.4 51单片机的时钟模块

###### 1.1.5 51单片机的其他资源

##### 1.2 51单片机的C51语言

###### 1.2.1 C51语言的数据类型、运算符和表达式

###### 1.2.2 C51语言的结构

###### 1.2.3 C51语言的函数

###### 1.2.4 C51语言的数组和指针

###### 1.2.5 C51语言的自构造类型

##### 1.3 Keil Vision软件开发环境的应用基础

###### 1.3.1 Keil Vision的界面

###### 1.3.2 Keil Vision的菜单

###### 1.3.3 Keil Vision的库函数

###### 1.3.4 使用Keil Vision

###### 1.3.5 Keil Vision的常见编译提示和错误

#### 第2章 Proteus硬件仿真环境

##### 2.1 Proteus的应用基础

###### 2.1.1 Proteus的界面

###### 2.1.2 Proteus支持的文件格式

###### 2.1.3 Proteus的菜单

###### 2.1.4 Proteus的快捷工具栏和工具箱

##### 2.2 使用Proteus

##### 2.3 Proteus和Keil Vision的联合调试应用实例

#### 第3章 Proteus中的51单片机的内部资源应用实例

##### 3.1 51单片机的定时/计数器应用实例

###### 3.1.1 定时/计数器基础

###### 3.1.2 定时/计数器输出方波应用实例

###### 3.1.3 定时/计数器输出PWM波形应用实例

###### 3.1.4 定时/计数器模拟外部中断信号应用实例

##### 3.2 51单片机的串口应用实例

###### 3.2.1 串口基础

###### 3.2.2 串口数据发送应用实例

###### 3.2.3 和PC进行串行通信应用实例

###### 3.2.4 串口模拟外部中断应用实例

##### 3.3 51单片机的函数库应用实例

###### 3.3.1 函数库基础

###### 3.3.2 右循环移位函数\_cror\_应用实例

###### 3.3.3 空操作函数\_nop\_应用实例

###### 3.3.4 串口字节发送函数putchar应用实例

###### 3.3.5 打印输出函数printf应用实例

###### 3.3.6 打印函数sprintf应用实例

## <<案例解说单片机C语言开发>>

- 3.3.7 内存复制函数memccpy应用实例
- 3.3.8 随机数函数rand应用实例
- 3.3.9 随机种子函数srand应用实例
- 3.3.10 建立并调用Send用户库函数应用实例
- 第4章 Proteus中的51单片机的人机交互通道应用实例
- 4.1 发光二极管应用实例
  - 4.1.1 器件基础
  - 4.1.2 应用实例的设计分析
  - 4.1.3 应用实例的代码
  - 4.1.4 应用实例的仿真结果和说明
- 4.2 单位数码管应用实例
  - 4.2.1 器件基础
  - 4.2.2 应用实例的设计分析
  - 4.2.3 应用实例的代码
  - 4.2.4 应用实例的仿真结果和说明
- 4.3 多位数码管应用实例
  - 4.3.1 器件基础
  - 4.3.2 应用实例的设计分析
  - 4.3.3 应用实例的代码
  - 4.3.4 应用实例的仿真结果和说明
- 4.4 MAX7219应用实例
  - 4.4.1 器件基础
  - 4.4.2 应用实例的设计分析
  - 4.4.3 应用实例的代码
  - 4.4.4 应用实例的仿真结果和说明
- 4.5 1602液晶应用实例
  - 4.5.1 器件基础
  - 4.5.2 应用实例的设计分析
  - 4.5.3 应用实例的代码
  - 4.5.4 应用实例的仿真结果和说明
- 4.6 12864液晶应用实例
  - 4.6.1 器件基础
  - 4.6.2 应用实例的设计分析
  - 4.6.3 应用实例的代码
  - 4.6.4 应用实例的仿真结果和说明
- 4.7 独立按键应用实例
  - 4.7.1 器件基础
  - 4.7.2 应用实例的设计分析
  - 4.7.3 应用实例的代码
  - 4.7.4 应用实例的仿真结果和说明
- 4.8 行列扫描键盘应用实例
  - 4.8.1 器件基础
  - 4.8.2 应用实例的设计分析
  - 4.8.3 应用实例的代码
  - 4.8.4 应用实例的仿真结果和说明
- 4.9 拨码开关应用实例
  - 4.9.1 器件基础

## <<案例解说单片机C语言开发>>

4.9.2 应用实例的设计分析

4.9.3 应用实例的代码

4.9.4 应用实例的仿真结果和说明

第5章 Proteus中的51单片机的信号采集通道应用实例

5.1 ADC0809应用实例

5.1.1 器件基础

5.1.2 应用实例的设计分析

5.1.3 应用实例的代码

5.1.4 应用实例的仿真结果和说明

5.2 TLC2543应用实例

5.2.1 器件基础

5.2.2 应用实例的设计分析

5.2.3 应用实例的代码

5.2.4 应用实例的仿真结果和说明

5.3 DS1302应用实例

5.3.1 器件基础

5.3.2 应用实例的设计分析

5.3.3 应用实例的代码

5.3.4 应用实例的仿真结果和说明

5.4 DS18B20应用实例

5.4.1 器件基础

5.4.2 应用实例的设计分析

5.4.3 应用实例的代码

5.4.4 应用实例的仿真结果和说明

5.5 SHT11应用实例

5.5.1 器件基础

5.5.2 应用实例的设计分析

5.5.3 应用实例的代码

5.5.4 应用实例的仿真结果和说明

第6章 Proteus中的51单片机的信号输出通道应用实例

6.1 DAC0832应用实例

6.1.1 器件基础

6.1.2 应用实例的设计分析

6.1.3 应用实例的代码

6.1.4 应用实例的仿真结果和说明

6.2 MAX517应用实例

6.2.1 器件基础

6.2.2 应用实例的设计分析

6.2.3 应用实例的代码

6.2.4 应用实例的仿真结果和说明

6.3 74HC138应用实例

6.3.1 器件基础

6.3.2 应用实例的设计分析

6.3.3 应用实例的代码

6.3.4 应用实例的仿真结果和说明

6.4 74HC273应用实例

6.4.1 器件基础

## <<案例解说单片机C语言开发>>

- 6.4.2 应用实例的设计分析
- 6.4.3 应用实例的代码
- 6.4.4 应用实例的仿真结果和说明
- 6.5 74HC244应用实例
  - 6.5.1 器件基础
  - 6.5.2 应用实例的设计分析
  - 6.5.3 应用实例的代码
  - 6.5.4 应用实例的仿真结果和说明
- 6.6 74HC164应用实例
  - 6.6.1 器件基础
  - 6.6.2 应用实例的设计分析
  - 6.6.3 应用实例的代码
  - 6.6.4 应用实例的仿真结果和说明
- 6.7 74HC165应用实例
  - 6.7.1 器件基础
  - 6.7.2 应用实例的设计分析
  - 6.7.3 应用实例的代码
  - 6.7.4 应用实例的仿真结果和说明
- 6.8 CD4094应用实例
  - 6.8.1 器件基础
  - 6.8.2 应用实例的设计分析
  - 6.8.3 应用实例的代码
  - 6.8.4 应用实例的仿真结果和说明
- 6.9 74HC595应用实例
  - 6.9.1 器件基础
  - 6.9.2 应用实例的设计分析
  - 6.9.3 应用实例的代码
  - 6.9.4 应用实例的仿真结果和说明
- 6.10 8255A应用实例
  - 6.10.1 器件基础
  - 6.10.2 应用实例1的设计分析
  - 6.10.3 应用实例1的代码
  - 6.10.4 应用实例1的仿真结果和说明
  - 6.10.5 应用实例2的设计分析
  - 6.10.6 应用实例2的代码
  - 6.10.7 应用实例2的仿真结果和说明
- 第7章 Proteus中的51单片机的存储器应用实例
  - 7.1 62256应用实例
    - 7.1.1 器件基础
    - 7.1.2 应用实例的设计分析
    - 7.1.3 应用实例的代码
    - 7.1.4 应用实例的仿真结果和说明
  - 7.2 AT24C04A应用实例
    - 7.2.1 器件基础
    - 7.2.2 应用实例的设计分析
    - 7.2.3 应用实例的代码
    - 7.2.4 应用实例的仿真结果和说明

## <<案例解说单片机C语言开发>>

### 7.3 MMC存储卡应用实例

#### 7.3.1 器件基础

#### 7.3.2 应用实例的设计分析

#### 7.3.3 应用实例的代码

#### 7.3.4 应用实例的仿真结果和说明

## &lt;&lt;案例解说单片机C语言开发&gt;&gt;

## 章节摘录

版权页：插图：2.View菜单 Proteus的View菜单主要用于设置Proteus的相关显示内容，包括图形刷新，坐标选择，放大缩小，是否显示快捷菜单栏等操作，如图2.6所示，其详细说明如下（括号中为对应的快捷键）。

Redraw (R)：刷新设计图纸，会去掉图纸上无效的相关图形。

Grid (G)：打开或关闭图纸上的参考坐标点。

Origin (O)：设置图纸的坐标原点。

X Cursor (X)：修改图纸的X坐标。

Snap 10th (Ctrl+F1)：选择坐标点密度为10th。

Snap 50th (F2)：选择坐标点密度为50th。

Snap 100th (F3)：选择坐标点密度为100th。

Snap 0.5in(F4)：选择坐标点密度为500th。

Pan (F5)：以当前鼠标位置为中心显示图纸。

Zoom In (F6)：放大图纸。

Zoom Out (F7)：缩小图纸。

Zoom All (F8)：将图纸缩小到显示全部。

Zoom to Area.显示某个区域，按住鼠标右键用于选择显示的区域。

Toolbars：用于打开或关闭对应的快捷菜单栏，如图2.7所示，分别提供了FileToolbar（文件相关快捷菜单栏）、View Toolbar（显示相关快捷菜单栏）、Edit Toolbar（编辑相关快捷菜单栏）、Design Toolbar（设计相关快捷菜单栏）。

3.Edit菜单 Proteus的Edit菜单通常用于对Proteus的设计图的全部或部分区域进行操作，包括取消刚刚完成的操作或重复刚刚取消的操作，剪切、复制等，如图2.8所示，其详细说明如下（括号中为对应的快捷键）。

Undo (Ctrl+Z)：取消刚刚完成的操作。

Redo (Ctrl+Y)：重做刚刚取消的操作。

Find and Edit Component (E)：查找和编辑器件。

EditObjectUnderCursor (Ctrl+E)：编辑鼠标选中的目标。

Cut to clipboard：将选中部分剪切到粘贴板。

Copy to clipboard：将选中部分复制到粘贴板。

Paste from clipboard：将粘贴板的内容复制到当前文件。

Send to back (Ctrl+B)：选中目标到后台，多层图形叠加时有效。

Bring to front (Ctrl+F)：选中目标到前台，多层图形叠加时有效。

Tidy：清理器件列表中没有使用的器件。

4.Tools菜单 Proteus的Tools菜单提供了对Proteus的电路图的某些自动操作，如自动添加器件标号，自动标注器件，自动生成图纸的材料清单，自动生成网络表等，如图2.9所示，其详细说明如下（括号中为对应的快捷键）。

Real Time Annotation (Ctrl+N)：实时标注，当该选项被选中时，表示在放置一个新的器件时，Proteus ISIS会自动给该器件加上编号。

Wire Auto Router (W)：自动连线，当该选项被选中时，如果将鼠标移动到一个引脚上，便会自动产生一个连线提示。

Search and Tag (T)：搜索标签。

Property Assignment Tool (A)：属性编辑工具。

Global Annotator：统一编号，用于对由多张图纸组成的工程文件中的器件使用统一的编号。

当选中该选项时，会弹出如图2.10所示的对话框，用于选择作用范围（Whole Design：整个工程；Current Sheet：单张图纸）、编号方式（Total：全部相同；Incremental：增量）、Initial Count（初始化数值设置）。



## <<案例解说单片机C语言开发>>

### 编辑推荐

《案例解说单片机C语言开发:基于8051+Proteus仿真》中提供了大量实例，它们都有详细的设计思路、典型器件列表、Proteus应用电路、Keil  $\mu$  Vision应用代码和仿真运行结果。

《案例解说单片机C语言开发:基于8051+Proteus仿真》适合于具有初步单片机基础的单片机工程师进阶学习，以及高等院校电子类专业的学生和单片机爱好者阅读，也可以作为工程设计的参考手册。

<<案例解说单片机C语言开发>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>