


实验二 键盘和显示仿真

- * 一、实验目的和要求
- * 二、实验设备
- * 三、实验内容
- * 四、实验原理和步骤
- * 五、思考题

一、实验目的和要求

- * 继续深入学习和掌握MCS-51的体系结构、指令系统和汇编语言开发
- * 掌握MCS-51的中断系统和定时器
- * 掌握多位7段数码管的显示方式
- * 初步理解键盘矩阵的输入方式

二、实验设备

- * Proteus仿真软件
 - * 实验二样例工程文件
- 

三、实验原理及预习

- * MCS-51的体系结构、指令系统和汇编语言
- * 多位7段数码管的显示原理
- * 键盘矩阵的工作原理
- * MCS-51中断系统的原理和编程
- * MCS-51定时器工作原理和编程

四、实验内容和步骤

- * 4.1 根据样例工程文件，分析多位7段数码管和键盘矩阵的工作流程
- * 4.2 根据要求建立仿真工程，绘制原理图
- * 4.3 按照要求和实验步骤开发程序，完成显示和键盘输入

4.1 分析

- * 打开“实验2-学生”样例工程文件
- * 分析多位7段数码管的接口方式和 workflows，使用简单程序进行实验
- * 分析键盘矩阵的接口方式和 workflows，使用简单程序进行实验
- * 分析结果整理后写入实验报告

4.2 绘制

- * 在Proteus中新建工程，选项如下：
 - * 1、建立DEFAULT原理图
 - * 2、不创建PCB Layout
 - * 3、选择“Create Firmware Project”，Family选择8051，Controller选择“AT89C51”，Compiler选择“ASEM-51”
 - * 这样建立的工程里面包括一个CPU，以及Source Code的基本框架

4.2 绘制

- * 在原理图中添加一个集成化的4位7段数码管组件（ 7SEG-MPX4* ），一个集成化的计算器键盘矩阵（ KEYPAD-SMALLCALC ）
 - * 上述两个部件的原理和4.1中分析的类似，集成化的部件使得用户可以用不用连接内部线路，只需要连接外部接口即可
- * CPU上选择适当的接口，与上述两个部件相连接

4.3 编程

- * 最终目标：
- * 编写程序，检测键盘按键，将其对应的数字或符号逐次在数码管上显示，按ON/C键显示清空，其他运算键自拟显示符号。
- * 为完成上述任务，可以分为以下几个步骤完成

4.3 编程

- * 步骤1：使用延时扫描方式，在数码管上显示不同的数字，如1234。
 - * 思路：每隔一定延时，在数码管特定位置显示对应的数字编码，然后修改要显示的位置下标
 - * 注意：建立每个数字（以及可能的其他符号）对应的编码表
 - * 建议：将要显示的数字放到一个显示缓冲区，显示时从缓冲区取，而不是直接计算。

4.3 编程

- * 步骤2：使用定时器中断，在中断处理程序中刷新数码管显示内容
 - * 思路：掌握中断的设置方式，掌握定时器的初始化和控制方式，设置合适的定时时间，将步骤1的显示代码移植到定时器中断中。
 - * 注意：留意中断处理程序中使用的寄存器，保证在发生中断时，不会和主程序发生冲突，必要时在中断处理程序中使用堆栈保存寄存器，特别是PSW和ACC

4.3 编程

- * 步骤3：检测键盘输入，每当有一次按键时，将当前显示的内容所有数字加1（如当前显示的是982F，则按键后，显示A930）
 - * 思路：在主程序中检测是否有键按下和抬起，然后形成结果，修改显示缓冲区。显示缓冲区中初值自己给定
 - * 注意：要检测按键的抬起操作，而不是键按下的状态，因为一次按键会持续很长时间。只有完整KEY-DOWN和KEY-UP才构成一次键盘输入

4.3 编程

- * 步骤4 (*)：读取真正的键盘按键，并将其转换为对应的功能，完成整体要求
 - * 思路：在键按下的操作中，扫描输入，得到行和列，设置一个转换表，完成键码到输入字符的转换。
 - * 注意：本步骤为附加要求，建议有能力的组完成。能够完成前面步骤3，即可请教师检查

五、思考题

- 1、分析键盘矩阵的接口方式和工作流程。
- 2、在步骤2，不同的定时器时间会对实验效果产生什么影响？
- 3、在实际使用中，键盘按键会产生抖动，如何在程序中去除抖动的影响？
- 4、当键盘中有多个键同时按下，如何得到全部的按键代码？