

第 3 章 嵌入式系统入门实验

实验一 MC908GP32 入门实验

一、实验目的

1. 熟悉 MT-IDE 嵌入式开发系统的软硬件环境,掌握其使用方法及调试方式。
2. 掌握 GP32 的 I/O 口的使用方法及编程方法。
3. 熟悉嵌入式应用的常用元器件。
4. 熟悉 GP32 单片机汇编程序或 C 语言程序编程框架。
5. 理解课本中的第一个汇编程序或 C 语言程序代码。

二、预习要求

1. 仔细阅读本实验指导书。
2. 预习课本中有关嵌入式应用的常用元器件。
3. 复习有关的嵌入式开发系统介绍的章节。
4. 复习课本有关 I/O 口的知识。
5. 根据实验内容要求编写好程序。

三、实验设备及其连接

- | | |
|-------------------|----|
| 1. PC | 一台 |
| 2. MT-IDE 嵌入式开发系统 | 一套 |
| 3. 串行通信线 | 一根 |
| 4. 万用表 | 一只 |

四、实验内容

1. 使用 MT-IDE 嵌入式开发系统的软硬件环境,掌握其使用方法及调试方式。
2. 根据课本的样例程序编写主程序。
3. 运行课本中的第一个汇编程序或 C 语言程序代码。
4. 运行课本中的样例程序,观察小灯的闪烁;调整延时的长短,继续观察小灯的闪烁。
5. 编写一个程序,用 4 个开关分别接在 PTA4~PTA7 口,并通过 MCU 控制分别接在 PTA0~PTA3 口上的 4 个小灯,开关拨在 ON 位置时,对应小灯亮,否则为暗,运行过程中用万用表测试正在使用的 I/O 口电压变化。

五、MT-IDE 嵌入式开发系统的使用方法及编程提示

(一) 运用 MT-IDE 嵌入式开发系统进行开发

1. 运行 MT-IDE 嵌入式开发平台,单击“文件”|“新建”命令,弹出一个对话框,在其中的“新建工程”选项卡中,通过“浏览”按钮选择所要创建工程的路径或直接输入所要创建工程的路径,在下一个文本框中输入工程名,选择属于 C 工程还是汇编工程(本实验选“C 工程”),接着选择对应单片机的型号(本实验选 MC908GP32,以下简称 GP32),对应工程属性自动确定,即 RAM 起始地址、RAM 终点地址、Flash 起始地址和堆栈起始地址,单击“确定”按钮,退出对话框。

2. 开发平台自动创建 4 个原始文件(除 2 个工程管理文件,即 prj 文件和 src 文件,这 2 个文件不在界面文件列表中出现),即 3 个源程序文件(.c)和 1 个头文件(.h)。

3. 在此基础上,保留 SetUp.c、GP32C.h 和 vectors08.c 文件文件不变动,这里以建立 PTA 的 I/O 口程序举例说明,添加 IO.c、IOInit.c 和 IO.h 文件,根据需要在 main.c、IO.c、IOInit.c 和 IO.h 填写相应的代码。

4. 程序代码编写结束,先进行编译,检查程序语法是否能通过,单击“编译”|“编译源程序”命令,或者单击工具栏的“开始编译(F7)”按钮。如果有问题,在输出窗口中根据问题的提示,找到对应的行,检查语法错误的原因。

5. 检查逻辑问题,即进行通信测试,首先将连线接好,再将程序写入 GP32 芯片中,然后调试或运行。

6. 连线方式:实验箱的电源开关先不用打开,扩展核心板插在实验箱对应接口上,通过编程调试器的接线一头接在该扩展板的 J3,注意该接线的红边靠实验者的身体这边,另一头接 PC 的 USB 接口,然后单击工具栏的“USB 擦除/下载程序”按钮,通过对话框,将 GP32 芯片中原来的程序先擦除,再写入当前的程序至该芯片中,然后调试或运行查找逻辑问题。

7. 这里有两种方式:一种是在线调试方式,另一种是直接运行方式。

8. 在线调试方式是:实验箱中核心扩展板的开关键 J5 仍处在“写入器”位置,串行线也接好,电源开关打开。单击“开始调试”按钮,然后单击 VB 开发的串行口收发测试界面的“发送”按钮,即可看到运行效果。如果没有出现预计的结果说明逻辑上是有问题,需要进行查找。为了便于查找问题所在,可以用调试功能的单步进行的方式,先在可能的位置设置断点,接着单击工具栏上的“开始调试”按钮,之后可以用单步调试功能,即 F10 或 F11 功能键进行一步一步检查程序每一步的运行结果是否准确,细心判断即可查找出问题所在。

9. 直接运行方式是:不设断点的情况下,单击“开始调试”按钮;或者实验箱中核心板上的开关键 J5 改在“扩展板”位置;或者编程调试器与实验箱 MONO8 断开,串行口接好(若与计算机通信时才使用,但这里暂不使用),电源开关打开,然后按一下实验箱中的复位(reset)按钮,即可看到运行效果。不过这样操作比较麻烦,若有问题又要重新接线,恢复前面的状态。另外应注意,在调试时,在适当位置插入停止语句,观察结果判断对错(具体参考第 1 章)。

(二) 编程提示

1. 按照结构要求写好编程代码和注释。

2. 08 汇编源程序通常由 7 个部分组成:程序描述、头文件与有关常量命名、内存变量、主程序、内部直接调用子程序、外部子程序、中断向量。

二、预习要求

1. 仔细阅读本实验指导书。
2. 复习有关的串行通信接口(SCI)的章节。
3. 熟悉 GP32 串行通信接口(SCI)的工作方法及编程要求。
4. 根据实验内容要求编写好程序,为实验做充分的准备。

三、实验设备及其连接

- | | |
|-------------------|----|
| 1. PC | 一台 |
| 2. MT-IDE 嵌入式开发系统 | 一套 |
| 3. 串行通信线 | 一根 |
| 4. 万用表 | 一只 |

四、实验内容

1. 理解串行通信接口(SCI)原理。
2. 运行与理解各子程序。
3. 主程序运行课本的样例程序。
4. 编制一个查询方式通信程序。

通过 PC 发送一个命令以控制接在 PTA0~PTA7 口的 8 个小灯亮暗状态,具体是:各个小灯初始状态都为暗,首先进行通信测试,PC 发送“A”(65),若单片机接收到则 8 个小灯全亮,并回送字符“A”(65),若单片机接收到但不是“A”(65),说明有误码,则 4 个小灯全亮,并回送字符“0”(48),否则 8 个小灯仍然全暗无反应,PC 在 1 分钟后自己判断线路不通,并显示“线路不通”。成功后进入控制状态,PC 发送字符串“11”,PTA0 口灯亮;发送“10”,PTA0 口灯暗;发送“21”,PTA1 口灯亮;发送“20”,PTA1 口灯暗;以此类推,当给 PTA 口发送“01”,8 个小灯全亮,发送“00”,全暗。PC 方程序已知。

5. 编制一个中断方式通信程序。

编程要求同上一条。

五、编程提示

1. 按照结构要求写好编程代码和注释。

2. 计算波特率,SCIInit.h(SCI 初始化子程序)设置 SCI 比特率寄存器(SCBR),设置允许 SCI、正常码输出、8 位数据、无校验,即设置 SCI 控制寄存器 1(SCC1)的相应位,设置是否允许发送与接收、是中断接收还是查询接收,即设置 SCI 控制寄存器 2(SCC2)的相应位。

3. 查询方式通信程序的主程序主体是一个死循环,循环体中是不断检测来自 PC 的数据是否可以接收,即检测 SCI 状态寄存器 1(SCS1)第 5 位是否为 1,成立时可以接收数据;并发送接收到的数据,即不断检测 SCI 状态寄存器 1(SCS1)第 7 位是否为 1,成立时可以发送数据。

4. 中断方式通信程序的主程序主体也是一个死循环,但该循环体是一个空循环体,所有接收和发送数据程序代码放在中断程序中,这里没有检测代码,数据开始发送和接收的条

件就是中断的条件。

六、实验报告要求

回答下列问题：

1. 全部发送字节每位为 0,用万用表测量芯片的 TX 端电压是多少? 并说明其原理。
2. 全部发送字节每位为 1,用万用表测量芯片的 TX 端电压是多少? 并说明其原理。
3. 什么是空闲线,如何识别?
4. 串行 SCI 通信有哪些中断? 各在什么情况下发生,作用是什么?
5. 如何知道串行口 TX 发送了信号?

提示：

方法一,利用万用表测试在 TX 端发送 0 或者 255 持续 0.5s 时的电压值。

方法二,在 TX 端发送 0 或 255 持续 0.5s 时的小灯变化。

第 4 章 嵌入式系统基本应用实验

实验三 数码管 LED 实验

一、实验目的

1. 理解 8 段数码管的基本原理(如图 4-1 和图 4-2 所示)。

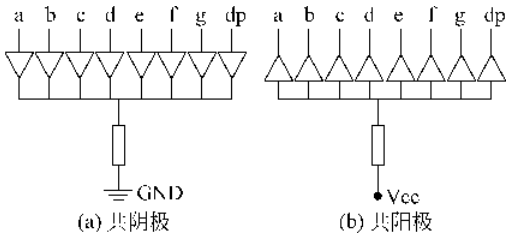


图 4-1 数码管

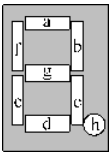


图 4-2 数码管外形

- 2. 理解 8 段数码管的显示和编程方法。
- 3. 理解 4 连排共阴极 8 段数码管 LG5641AH 与 MCU 的接线图(如图 2-23 所示)。

二、预习要求

- 1. 仔细阅读本实验指导书。
- 2. 复习有关的定时器溢出中断的章节。
- 3. 熟悉 LED 的工作方法及编程方法。
- 4. 熟悉 LG5641AH 的工作方法及编程。
- 5. 根据实验内容要求编写好程序。

三、实验设备及其连接

- | | |
|----------------------------|----|
| 1. PC | 一台 |
| 2. MT-IDE 嵌入式开发系统 | 一套 |
| 3. 串行通信线 | 一根 |
| 4. 4 连排共阴极 8 段数码管 LG5641AH | 一个 |

四、实验内容

- 1. 理解 8 段数码管原理。
- 2. 运行与理解各子程序。
- 3. 主程序运行课本的样例程序。
- 4. 编制一个 4 连排共阴极 8 段数码管程序。

MCU 的 PTC3~PTC0 接该 4 连排数码管的位选线(自左到右),PTB7~PTB0 接 LED

的数据线 hgfedcba,在 4 连排 8 段数码管显示 MCU 复位后的开始到现在的运行时间。由于只有 4 个数码管,所以只显示 MCU 运行到现在的分和秒,当计时达到一个小时,就重新从 00: 00 开始计时。另外,也可以通过 PC 方的串口通信程序,指定计时的开始值,该 PC 方的程序已知。

五、编程提示

1. 按照结构要求写好编程代码和注释。
2. 计时的工作主要由定时器溢出中断来完成,LED 只是负责把到目前为止的时间显示出来。定时器溢出中断每一秒钟产生一次,用于计时,计时的初始值可以由 PC 方的串行通信程序指定。
3. 采用“位选线扫描法”来逐个显示选中的数码管。对于如图 2-23 所示的 4 连排数码管,利用 CS3、CS2、CS1、CS0 控制各个数码管的位选信号,每个时刻只能让一个数码管有效,即 CS3、CS2、CS1、CS0 只能有一个为 0,例如令 CS3=0,CS2、CS1、CS0=111,则数据线上的数据体现在第一个数码管上,其他则不受影响。要让各个数据管均显示需要的数字,则必须逐个使相应位选信号为 0,其他位选信号为 1,并将要显示的一位数字送到数据线上。这种方法叫“位选线扫描法”。虽然每个时刻只有一个数码管有效,但只要延时适当,由于人眼的“视觉暂留效应”(约 100ms 左右),看起来则是同时显示的。
4. 8 段数码管一般由 8 个发光二极管(light-emitting diode,LED)组成,每一个位段就是一个发光二极管。一个 8 段数码管分别由 a、b、c、d、e、f、g 位段,外加上一个小数点的位段 h(或记为 dp)组成。根据公共端所接电平的高低,可分为共阳极和共阴极两种,如图 4-1 所示。实物外形如图 4-2 所示。有时数码管不需要小数点,只有 7 个位段,称 7 段数码管。共阴极 8 段数码管的信号端高电平有效,只要在各个位段上加上相应的信号即可使相应的位段发光,比如:要使 a 段发光,则在 a 段加上高电平即可。共阳极的 8 段数码管则相反,在相应的位段加上低电平即可使该位段发光。因而,一个 8 段数码管就必须有 8 位(即 1 个字节)数据来控制各个位段的亮灭。
5. 7 段 LED 显示字型码。如表 4-1 所示。

表 4-1 7 段 LED 显示字型码

显示字符	h	g	f	e	d	c	b	a	字型码	
									共阴极	共阳极
0		0	1	1	1	1	1	1	3F	C0
1		0	0	0	0	1	1	0	06	F9
2		1	0	1	1	0	1	1	5B	A4
3		1	0	0	1	1	1	1	4F	B0
4		1	1	0	0	1	1	0	66	99
5		1	1	0	1	1	0	1	6D	92
6		1	1	1	1	1	0	1	7D	82
7		0	0	0	0	1	1	1	07	F8
8		1	1	1	1	1	1	1	7F	80
9		1	1	0	1	1	1	1	6F	90
A		1	1	1	0	1	1	1	77	88
B		1	1	1	1	1	0	0	7C	83
C		0	1	1	1	0	0	1	39	C6
D		1	0	1	1	1	1	0	5E	A1
E		1	1	1	1	0	0	1	79	86
F		1	1	1	0	0	0	1	71	8E

六、实验报告要求

回答下列问题：

- 1. 驱动 LED 数码管有哪些方法？
- 2. LED 静态显示和动态显示方式各有什么优缺点？
- 3. 在本自编程序中,让第二个数码管的小数点仿照电子时钟每秒钟闪烁一下,只要改一处,能行吗？
- 4. LED 要显示的数字如果较多,那么要占用 MCU 较多的 I/O 引脚。想一想,为了节约引脚,可以芯片扩充,那么 4 个引脚最多能扩充出几个引脚？

实验四 液晶 LCD 显示实验

一、实验目的

- 1. 理解点阵字符型 LCD 的显示原理。
- 2. 理解 HD44780 的原理框图(如图 4-3 所示)。

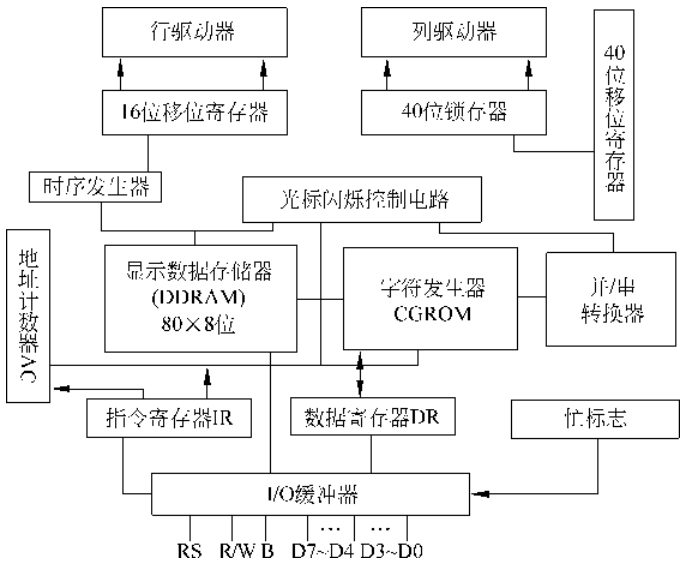


图 4-3 HD44780 原理框图

- 3. 了解字符型液晶显示模块的控制器 HD44780 的显示和编程方法。
- 4. 了解 HD44780 的引脚信号(如表 4-2 所示)。
- 5. 理解 HD44780 与 MCU 的接线图(如图 4-4 所示)。

表 4-2 HD44780 引脚信号

管脚号	符号	电平	方向	引脚含义说明
1	Vss			电源地
2	Vdd			电源(+5V)
3	V0			液晶驱动电源(0~5V)
4	RS	H/L	输入	寄存器选择：1——数据寄存器 0——数据寄存器
5	R/ $\overline{W}$	H/L	输入	读写操作选择：1——读操作 0——写操作
6	E	H/L H→L	输入	使能信号：R/ $\overline{W}$ =0,E 下降沿有效 R/ $\overline{W}$ =1,E=1 有效
7~10	DB0~DB3		三态	8 位数据总线的低 4 位,若与 MCU 进行 4 位传送时,此 4 位不用
11~14	DB4~DB7		三态	8 位数据总线的高 4 位,若与 MCU 进行 4 位传送时,只用此 4 位
15~16	E1~E2		输入	上下两行使能信号,只用于一些特殊型号

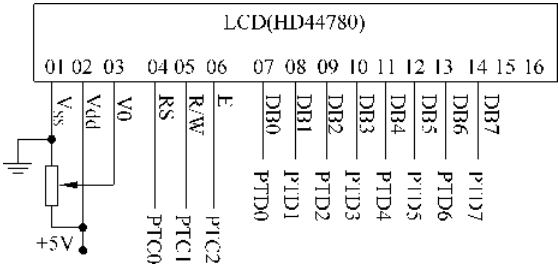


图 4-4 MCU 与 LCD 的连接

二、预习要求

- 1. 仔细阅读本实验指导书。
- 2. 复习有关的定时器溢出中断的章节。
- 3. 熟悉 HD44780 的工作方法及编程方法。
- 4. 根据实验内容要求编写好程序。

三、实验设备及其连接

- | | |
|-------------------|----|
| 1. PC | 一台 |
| 2. MT-IDE 嵌入式开发系统 | 一套 |
| 3. 串行通信线 | 一根 |
| 4. HD44780 兼容 LCD | 一个 |

四、实验内容

- 1. 理解点阵字符型 LCD 原理。
- 2. 运行与理解各子程序。
- 3. 主程序运行课本的样例程序。

4. 编制一个点阵字符型 LCD 程序。

MCU 的 PTD7-PTD0 接 LCD 数据线,PTC2-PTC0 接 LCD 控制线,在 LCD 上显示当前时间,格式形如“时:分:秒”。时间的初值可以通过串口通信取 PC 方的时间来设置。

五、编程提示

1. 按照结构要求写好编程代码和注释。

2. 计时的工作主要由定时器溢出中断来完成,LCD 只是负责把目前的时间显示出来。定时器溢出中断每秒钟产生一次,用于计时,时间的初始值可以由 PC 方的串行通信程序指定。

3. HD44780 的编程中可能涉及到的寄存器。

- 指令寄存器(IR): IR 用于 MCU 向 HD44780 写入指令码。
- 数据寄存器(DR): DR 用于寄存数据。忙标志(BF): $BF=1$,表示组件正在进行内部操作,不能接受外部指令或数据。
- 地址计数器(AC): AC 作为 DD RAM 或 CG RAM 的地址指针。
- 显示数据寄存器(DD RAM): DD RAM 用于存储显示数据,共有 80 个字符码。
- 字符发生器 ROM(CG ROM): CG ROM 由 8 位字符码生成 5×7 点阵字符 160 种和 5×10 点阵字符 32 种,其中大部分与 ASCII 码兼容。
- 字符发生器 RAM(CG RAM): CG RAM 是提供给用户自定义特殊字符用的,它的容量仅为 64B,编址为 00~3FH。

4. HD44780 的指令集

- 清屏(clear display): 清屏指令使 DD RAM 的内容全部被清除,屏幕光标回原位,地址计数器 $AC=0$ 。
- 归位(return home): 归位指令使光标和光标所在位的字符回原点(屏幕的左上角)。
- 输入方式设置(entry mode set): 该指令设置光标、画面的移动方式。
- 显示开关控制(display ON/OFF control): 该指令设置显示、光标及闪烁开、关。
- 光标或画面移位(cursor or display shift): 该指令使光标或画面在没有对 DD RAM 进行读写操作时被左移或右移,不影响 DD RAM。
- 功能设置(function set): 该指令为工作方式设置命令(初始化命令)。
- CG RAM 地址设置(CG RAM address set): 该指令设置 CG RAM 地址指针。
- DD RAM 地址设置(DD RAM address set): 该指令设置 DD RAM 地址指针。
- 读忙标志 BF 和 AC 值(read busy flag and address count): 该指令读取 BF 及 AC。
- 写数据到 DDRAM 或 CGRAM(write data to DDRAM or CG RAM): 该指令根据最近设置的地址,将数据写入 DD RAM 或 CG RAM 中。
- 读 DDRAM 或 CGRAM 数据(read data from DDRAM or CGRAM): 该指令根据最近设置的地址,从 DD RAM 或 CG RAM 读数据到总线 DB7~DB0 上。

六、实验报告要求

回答下列问题:

1. LCD 有哪些特点?
2. LCD 有哪些种类?

3. 数字可以直接在 LCD 上显示吗?
4. 本实验利用 HD44780 LCD 显示屏进行,MCU 向 LCD 输送指令和数据,是通过不同的数据线吗?
5. 该 LCD 存储指令和数据是用不同的寄存器吗?
6. 参考本 LCD,想一想自定义字符是怎样保存至 LCD 中的? 又是怎样调用的?

实验五 键盘中断实验

一、实验目的

1. 熟练运用 MT-IDE 嵌入式开发系统的环境、汇编、C 语言、调试方式。
2. 复习串行通信接口(SCI)的内容。
3. 加强键盘中断基本原理及编程原理的理解。
4. 理解运行课本中的程序代码。
5. 理解“行扫描”法的原理并能进行键值识别和键值编码。
6. 理解键盘接线原理图(如图 2-19 所示)。

实验箱提供一个 16 键键盘,用于键盘中断信号的输入。系统提供了两种接线方式:

- (1) 当将键盘接入上一排插孔时为固定接线,键盘接线原理图如图 2-19 所示。
- (2) 当将键盘接入下一排插孔时为手动接线,连线的位置在键盘的左边。

二、预习要求

1. 仔细阅读本实验指导书。
2. 复习有关的键盘中断和串行通信接口(SCI)的章节。
3. 熟悉 GP32 键盘模块的工作方法及编程方法。
4. 根据实验内容要求编写好程序,为实验做充分准备。

三、实验设备及其连接

- | | |
|-------------------|----|
| 1. PC | 一台 |
| 2. MT-IDE 嵌入式开发系统 | 一套 |
| 3. 串行通信线 | 一根 |
| 4. 小键盘 | 一个 |
| 5. 万用表 | 一个 |

四、实验内容

1. 理解键盘模块及键盘中断的原理。
2. 运行与理解各子程序。
3. 主程序运行课本的样例程序。
4. 编制一个中断方式的 16 键键盘程序,使用“行扫描”法识别按键。