

2. 实验[2] A2000TM4 扩展板使用方法与 SYSTICK 中断

2.1 实验目的

熟悉 LED 显示和键盘管理芯片 TM1638 的功能，学会通过 TM4C1294 芯片程控该芯片的方法；
初步掌握简易的人机操作界面的设计技巧（利用数码管、LED 指示灯、按键阵列等）；
熟悉 SYSTICK 中断调用方式，掌握相应的编程技能；
学习和掌握在无操作系统支持的条件下，模拟多任务切换式处理的程序设计思想和方法。

2.2 实验主要器材和设备

电脑，TM4C1294NCPDT 实验板卡，A2000TM4 扩展板，数字示波器。

2.3 关于例程 exp2_0.c 的说明

关于 A2000TM4 扩展板的使用方法，建议认真学习和参考例程 exp2_0.c 的代码结构和指令写法。
按照本实验的任务要求，改写该例程，即可获得能实现有关功能的程序代码。

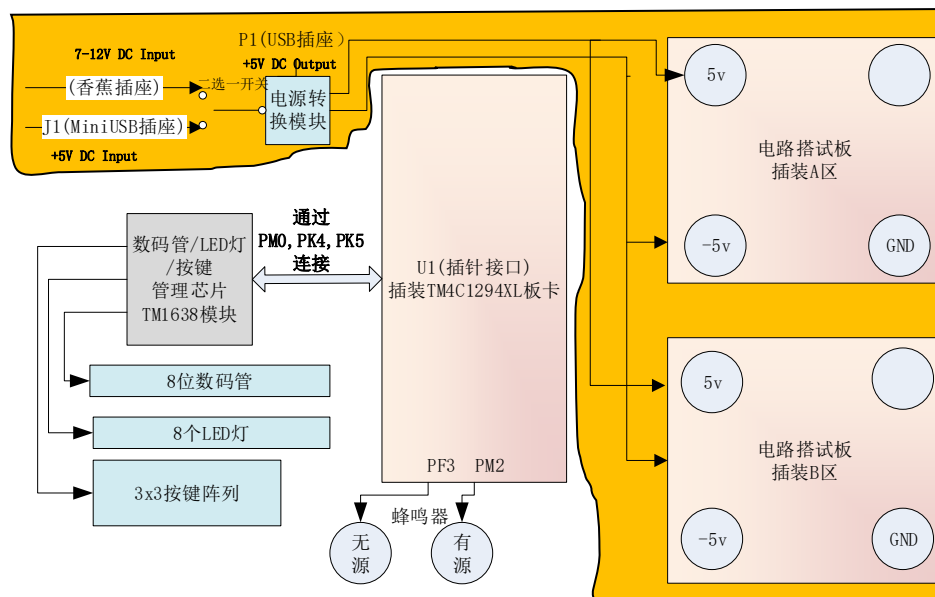


图 2-1 A2000TM4 扩展板电路组成框图

图 2-1 展示了 A2000TM4 扩展板电路的组成框图。从电路组成结构看，大致可以分为供电及电源转换电路（橙色底色区域）和 TM4C1294XL 板卡扩展电路（其他区域）两个部分。

TM4C1294XL 板卡扩展电路提供了必要的外设扩展，包括 8 个七段数码管、8 个 LED 指示灯和 9 个按键（排列成 3x3 阵列），还有 1 个无源蜂鸣器和 1 个有源蜂鸣器。通过适当编程和使用这些板载装置资源，可以为基于 A2000TM4 扩展板设计的实验作品提供人机交互方式，即简易的人机界面 (UI)。

扩展电路以 TM1638 集成芯片为核心器件。在微处理器程控下，TM1638 可以管理数码管显示，控制 LED 灯的亮灭，扫描检测按键操作情况。硬件电路中，TM4C1294 微处理器通过自身 GPIO 引脚 PM0、PK4、PK5 与 TM1638 芯片相连，传递程序控制信号。

鉴于多数学习者是初学者，课程的示例程序以开源方式提供了关于 TM1638 的控制程序段（C 语言函数子程序），被封装为头文件 tm1638.h 和源文件 tm1638.c，供学习者调用和参照学习。

运行示例程序 exp2_0.c，可观察的实验板上各部分功能如下，各部分同时工作。

- 功能 1：开机或复位后，底板上右四位数码管对应加计时功能，自动显示十进制计时数

值，最低位的时间单位是 0.1 秒；每计满 1000.0 秒归零，重新开始加计时。且功能 1 会受功能 3 影响。

- 功能 2：开机或复位后，底板上 8 个 LED 灯以循环跑马灯形式由左向右，约 0.5 秒变换 1 次。
- 功能 3：利用左数第二位数码管显示“键值”（按键编号）。当没有按键按下时，显示数码“0”；当人工按下某键，数码管显示该键的编号（1 到 9），且此刻，功能 1 的四位加计时数码管暂停变化（停止计时），直到按键被放开后自动继续计时。

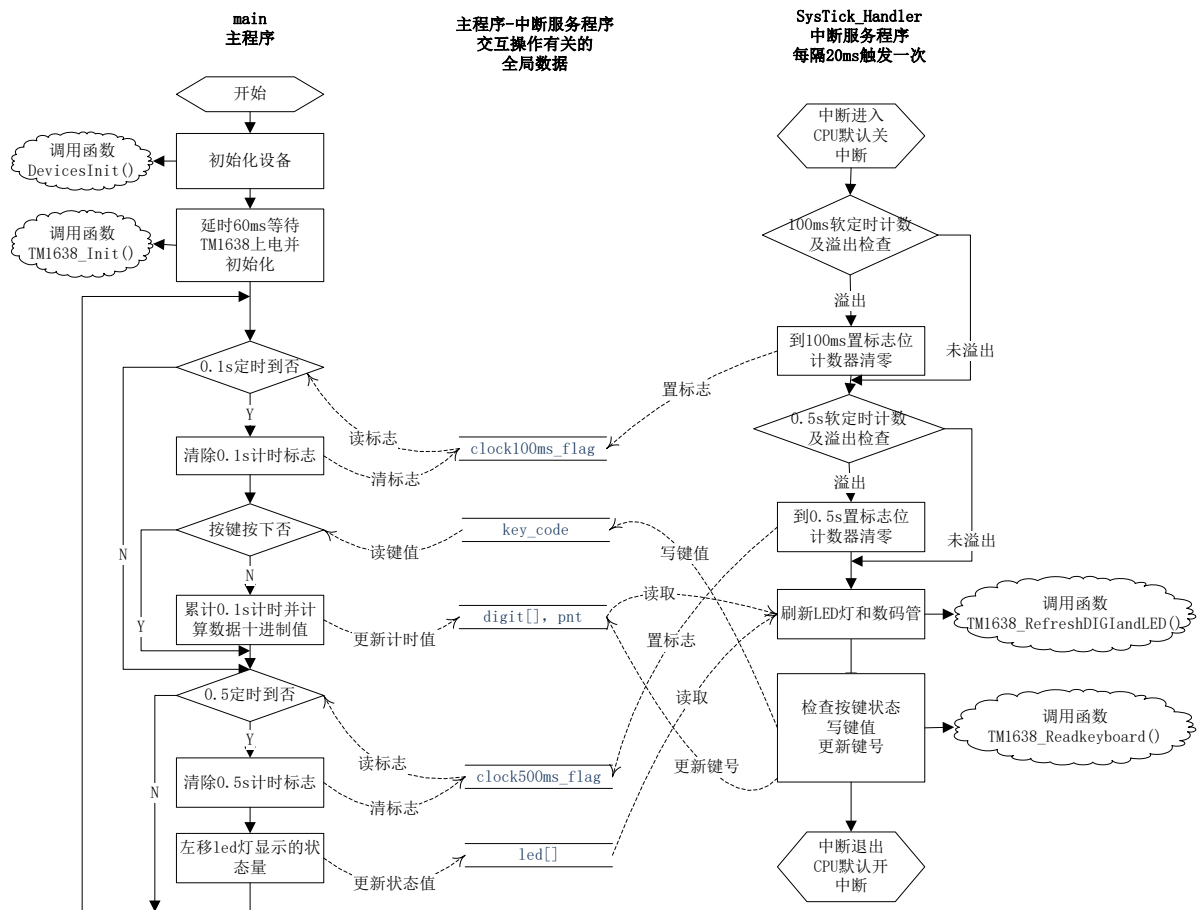


图 2-2 程序流程示意图

例程 exp2_0.c 的简明流程图如图 2-2。其中，主程序和 SysTick 中断服务程序被绘制成两条独立的流程线，在系统运行时两者受处理器硬件机制调度，由硬件机制决定当前应该由谁占用处理器并被运行。两者通过操作（读或写）专门的全局变量，来实现彼此之间的逻辑数据交互。比如，当中断服务程序运行时，读取到用户的按键操作情况，会把信息数据写入全局变量 key_code；然后，当主程序被调度运行时，有相应的代码会读取 key_code 数据。

在例程 exp2_0.c 中所定义的常量、变量（及初值）全部列出在表 2-1 和表 2-2。

表 2-1 示例程序的常量

常量名	含义	值
V_T100ms	100ms (0.1s) 定时值	5
V_T500ms	500ms (0.5s) 定时值	25

表 2-2 示例程序的变量

变量名	类型	含义
clock100ms	uint8_t	100ms(0.1s)软定时计数器，每隔 20ms 加 1；值为 V_T100ms 时归零，同时 clock100ms_flag 置 1。
clock500ms	uint8_t	500ms(0.5s)软定时计数器，每隔 20ms 加 1；值为 V_T500ms 时归零，同时 clock500ms_flag 置 1。
clock100ms_flag	uint8_t	100ms(0.1s)timer 计数溢出标志
clock500ms_flag	uint8_t	500ms(0.5s)timer 计数溢出标志
test_counter	uint32_t	测试用计数器，当 100ms 到，计数加 1；计满 10000 归零。
disp_digit[8]	uint8_t	数组变量 disp_digit[]用于缓存数码管的显示内容，数组长度为 8，对应板上数码位从左到右序号排列为 4、5、6、7、0、1、2、3；比如，disp_digit[4]对应最左一个数码管，disp_digit[2]对应左数第七个数码管。
disp_pnt	uint8_t	8 位变量 disp_pnt 用于缓存数码管小数点的亮灭状态，0 灭 1 亮。板上数码小数点从左到右对应 disp_pnt 的第 4、5、6、7、0、1、2、3 位；初值 0x04 会使左数第七个数码小数点被点亮。
disp_led[8]	uint8_t	数组变量 disp_led[]用于缓存 LED 指示灯的状态值，0 灭，1 亮。板上指示灯从左到右对应的下标序号为 7、6、5、4、3、2、1、0。初值 {1, 1, 1, 1, 1, 1, 1, 0} 会使八个指示灯从左到右显示为“亮亮亮亮亮亮亮灭”。
key_code	uint8_t	变量 key_code 用于缓存最近一次读到的键值。3x3 按键阵列从左向上右下编号 1-9。比如第二行第三个按键按下，键值为 6；无键按下，则键值用 0 表示。

程序中定义的函数体按逻辑联系，分布放置在适当的文件中。在文件 exp2_0.c 中，除了有主程序和 SYSTICK 中断服务程序外，还包括用于微处理器系统初始化的函数。而与 TM1638 芯片操作有关的函数定义放在文件 tm1638.c 中，对应的原型声明列入文件 tm1638.h 中。

表 2-3 列出了这些函数。图 2-3 示出了这些函数体彼此间的调用和被调用关系。

表 2-3 示例程序的函数

函数名	功能	位置
int main(void)	主程序	exp2_0.c
void SysTick_Handler(void)	SYSTICK 中断服务程序	exp2_0.c
void GPIOInit(void)	I/O 口初始化	exp2_0.c
void SysTickInit(void)	SysTick 计时器初始化	exp2_0.c
void DevicesInit(void)	初始化 I/O 口、SysTick 计时器等	exp2_0.c
uint8_t TM1638_DigiSegment (uint8_t digit)	数码管显示数据的译码	tm1638.c
void TM1638_Serial_input (uint8_t data)	程控 TM1638 芯片做串行数据输入	tm1638.c
uint8_t TM1638_Serial_output(void)	程控 TM1638 芯片做串行数据输出	tm1638.c
uint8_t TM1638_Readkeyboard (void)	读取 TM1638 获取按键状态	tm1638.c
void TM1638_RefreshDIGIandLED(刷新 8 位数码管（含小数点）和 8 组 LED 指示	tm1638.c

uint8_t digit_buf[8], uint8_t pnt_buf, uint8_t led_buf[8])	灯	
void TM1638_Init(void)	初始化 TM1638	tm1638.c

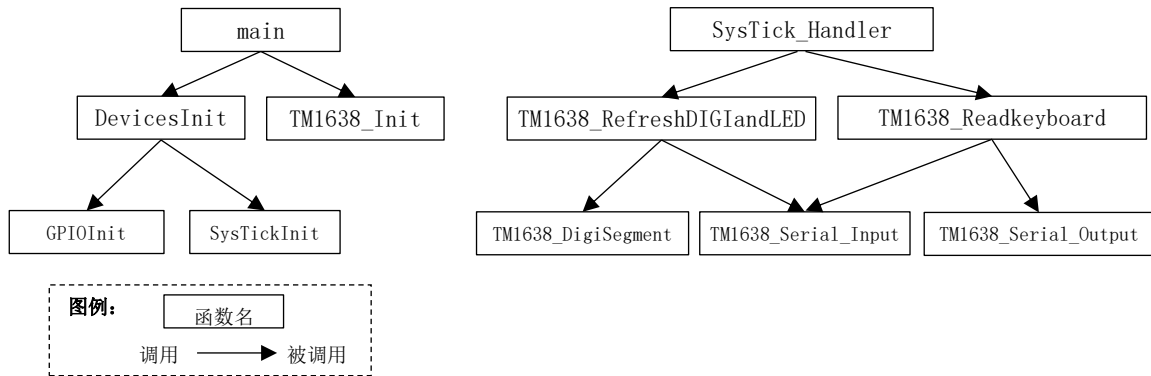


图 2-3 程序函数的调用和被调用关系图

2.4 实验任务要求

2.4.1 实验任务 2_1

将 exp2_0.c 另存命名为 exp2_1.c。以 exp2_0.c 的功能为基础，修改程序，完成实验任务 2_1。
在实验任务 2_1 中，要让电路呈现三种工作模式（见表 2-4）。通过人工按动一次任意键，使电路做一次模式切换，在三种工作模式中轮转。开机或复位运行，以模式 0 为默认工作模式。

表 2-4 工作模式列表

模式	功能描述
0	原功能 1 和功能 3 有效；功能 2 暂停（LED 灯静止不闪动）；
1	原功能 1 暂停（计时）；原功能 3 有效；8 个指示灯以走马灯方式，每 0.5 秒向右（循环）移动一格
2	原功能 1 暂停（计时）；原功能 3 有效；8 个指示灯以走马灯方式，每 0.5 秒向左（循环）移动一格

一次人工按键操作应包括按下和放开两个过程。任意按下一键，会触发一次模式转变，但按住不放不能连续改变工作模式，放开动作也不直接影响工作模式；直至放开之后的下一次按键，才能再次触发转变。

2.4.2 实验任务 2_2

将 exp2_0.c 另存命名为 exp2_2.c。以 exp2_0.c 的功能为基础，修改程序，完成实验任务 2_2。
exp2_2.c 的功能要点如下。

- 1) 原功能 1 继续有效，且受功能 3 影响。
- 2) 取消原功能 2；指定最左端 LED 灯作为一个状态标志，当功能 1 正常计数时最左端 LED 灯亮，功能 1 暂停计时则最左端 LED 灯灭；其余 LED 灯保持为全灭。
- 3) 原功能 3 改为：利用左数第二位数码管显示“键值”（按键编号）；当没有按键按下时，显示数码“0”；当人工按下某键，数码管显示该键的编号（1 到 9）。当按一次 1 号键，可循环切换功能 1 的计时工作状态，启动或暂停(start/pause) 计时（计数）功能；当按一次 2 号键，对功能 1

的计时值清零；当按下其余键时，对功能 1 没有影响。

2.4.3 实验任务 2_3

在实验任务 2_2 基础上，修改程序 exp2_2.c 代码并另存命名为 exp2_3.c，完成实验任务 2_3。
将功能 1 计时的加计数改为减计数，且计时初值为 9999（注：对应 999.9 秒）；每计满 1000.0 秒，重新开始。

2.4.4 实验任务 2_4

在实验任务 2_3 基础上，修改程序 exp2_3.c 代码并另存命名为 exp2_4.c，完成实验任务 2_4。
添加使用 3 号按键作为加减计数模式切换控制。开机或复位后，为减计时；之后每按一次 3 号键，切换一次加减计数模式。

取消用左数第二位数码管显示“键值”的功能。改以左侧 4 位数码管显示计时模式状态，即用“_UU_”表示加计数，显示“_AA_”表示减计数。

2.5 实验结果的考评

课程组织现场考评，检查学习者对实验任务的完成情况。

- ✧ 对实验任务规定的功能，学习者应能熟练操作和展示；
- ✧ 学习者应能熟练掌握实验步骤中的具体操作和设计能力，根据评测官现场要求（工作参数或功能设计要求，可能略微有别于已做内容）展示这些技能。

课程还对实验报告进行评价。

2.6 实验报告要求

根据自己的实验结果记录，编写实验报告。课程提供有报告模板作为参考，其中内容允许学习者根据自己情况灵活调整。