

## 练习题

71117201 姜子玥

1、单片机和我们通常所用的微型计算机有什么区别和联系？

答：

区别：1) 组成不同：

单片机属于集成电路芯片，主要包含中央处理器 CPU、随机存储器 RAM、只读存储器 ROM、多种 I/O 口和中断系统、定时器/计数器等功能（可能还包括显示驱动电路、脉宽调制电路、模拟多路转换器、A/D 转换器等电路）等。

微型计算机系统包括硬件系统和软件系统两大部分。硬件系统由运算器、控制器、存储器（含内存、外存和缓存）、各种输入输出设备组成，采用“指令驱动”方式工作。软件系统可分为系统软件和应用软件。

2) 特性不同：

单片机：体积小，结构简单，功能完善，使用方便，可靠性比较强；应用时低电压、低能耗；对数据的处理能力和运算能力较强，有较强的控制能力；拥有简易携带等优势，同时性价比较高。

微型计算机：执行结果精确、处理速度快捷、性价比高、轻便小巧；且技术不断更新、产品快速换代，从单纯的计算工具发展成为能够处理数字、符号、文字、语言、图形、图像、音频、视频等多种信息的强大多媒体工具。

3) 应用不同：

单品机：主要用于自动化办公、机电一体化、尖端武器和国防军事领域、航空航天领域、汽车电子设备、医用设备领域、商业营销设备、计算机通讯、家电领域、日常生活和实时控制领域等。

微型计算机：主要用于机械制造技术，信息处理、加工、传输技术，自动控制技术，伺服驱动技术，传感器技术，软件技术等。

联系：

单片机是微型计算机小型化发展的结果，它将计算机的 CPU、RAM、ROM 半导体存储器等电路集成在一块芯片上，在功能上二者没有大的不同，都可以进行算术运算和逻辑运算，都可以上网互联。只是大小不同，应用不同而已。

2、简述哈佛结构与冯诺依曼结构的优缺点。

答：哈佛结构采用指令和数据独立编址，使用独立的两条独立的总线传输，CPU 读取指令和数据的操作可以重叠，优点在于更加可靠，更加适合于那些程序固化、任务相对简单的控制系统，缺点在于在计算机系统中，应用程序的多样性使得计算机要不断变化所执行的代码内容，并且频繁地对数据与代码战友的存储器进行重新分配的情况下，哈佛结构会对存储器的资源产生极大的浪费，初次之外结构也较为复杂、成本较高。

冯诺依曼结构采用指令和数据统一编址，使用同一条总线传输，CPU 读取指令和数据的操作无法重叠，优点在于对存储器中的代码和数据频繁修改统一编址时，有利于节约资源，结构简单，成本较低，缺点则是不够可靠，本身没有属性标志，会导致语义的差距，束缚了现代计算机的进一步发展。

3、简述 CISC 与 RISC 的优缺点。

答：CISC 优点：能够有效缩短新指令的伪代码设计时间，允许设计师实现 CISC 体系机器的

向上相容。新的系统可以使用一个包含早期系统的指令超集合，也就可以使用较早电脑上使用的相同软体，另外微程式指令的格式与高阶语言相匹配，因而编译器并不一定要重新编写。缺点：指令集以及晶片的设计比上一代产品更复杂，不同的指令需要不同的时钟来完成，执行较慢的指令将影响整台机器的执行效率。

**RISC 优点：**使用相同的晶片技术和相同运行时钟下，RISC 运行速度更快。可以比 CISC 处理器应用更多先进的技术，开发更快的下一代处理器。

**缺点：**程式开发者编写的代码量会非常大，需要选用合适的编译器。需要更快记忆体，集成于处理器内部。

#### 4、MSP430F6638 单片机 CPU 的寄存器有哪些？简述其功能和作用

答：四个专用寄存器（R0、R1、R2、R3）和 12 个通用寄存器（R4~R15）

1) R0 (PC)：程序计数器，指示下一跳将要执行的指令地址功能取完指令后 CPU 根据该指令的字节数自动增量 PC

2) R1 (SP)：堆栈指针，CPU 使用功能 20 位堆栈指针来存储子例程调用和中断的返回地址，作用在于保护现场和恢复现场

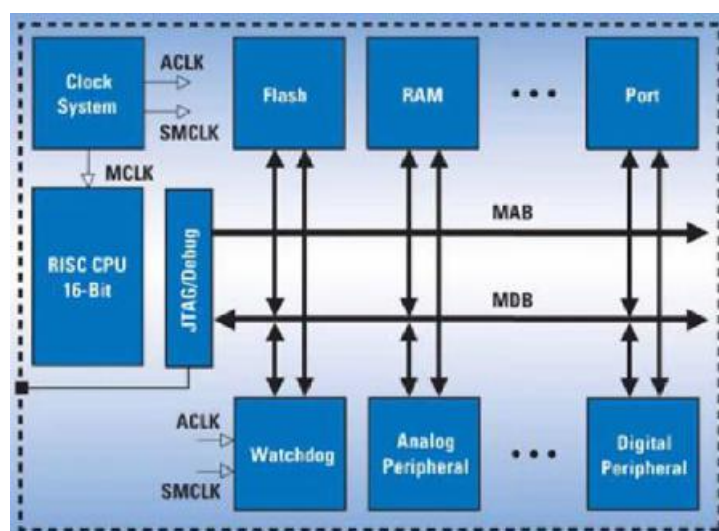
3) R2 (SR)：状态寄存器，用作源或目标寄存器的 16 位状态寄存器（SR，也称为 R2）只能用于通过字指令寻址的寄存器模式。寻址模式的其余组合用于支持常数发生器。

4) R2/R3 (CG2)：常数发生器（CG1/CG2）如果操作数是 6 个常量之一，则寄存器是自动选择的；所以，在常量模式下，寄存器 R2 和 R3 不能作为源寄存器

5) R4~R15 通用寄存器：保存参加运算的数据及运算的中间结构，也可以用来存放地址。

#### 5、简述 MSP430F6638 单片机 CPU 的结构及其主要特性

答：结构框图如下：



- 1) 16 位精简指令集 CPU 通过地址总线和数据总线直接与存储器和片上外设相连
- 2) 单片机内部包含嵌入式仿真系统，具有 JTAG/SBW 接口
- 3) 智能时钟系统支持多种时钟，能够最大限度地降低功耗

主要特性：1) 低电源电压范围在 1.8V 至 3.6V；功耗超低 2) 在典型值内从待机模式唤醒；16 位 RISC 架构，扩展内存，高达 20-MHz 系统时钟；灵活电源管理系统 3) 统一时钟系统 4) 四个具有 3,5 或者 7 个剥啄寄存器的 16 位定时器 5) 两个通用串行通信接口

6、什么是中断嵌套，MSP430 单片机允许中断嵌套吗？

答：中断嵌套是指中断系统正在执行一个中断服务时，有另一个优先级更高的中断提出中断请求，这时会暂时终止当前正在执行的级别较低的中断源的服务程序，去处理级别更高的中断源，待处理完毕，再返回到被中断了的中断服务程序继续执行。MSP430 单片机允许中断嵌套。

7、单片机的复位的作用是什么？

答：单片机的复位，是为了把单片机初始化到一个确定的初始状态，比如把单片机内部的一些专用寄存器设置为一些特定的初值，同时，把 PC 设置为程序存储器的中第一条要执行的指令地址（不同单片机的这个首地址不一定一样）。

8、MSP430 时钟系统模块具有哪 5 个时钟来源？ 哪 3 个时钟信号？请分别解释它们的含义；

答：时钟来源：LFXT1CLK、XT2CLK、DCO、TX2CLK、FLXT1CLK

时钟信号：1) MCLK：主时钟信号，用于 MCU 和相关系统模块作时钟使用，同样可设置相关寄存器来决定分频因子及相关的设置； 2) SMCLK：子系统时钟，由 2 个时钟源信号所提供，可设置相关寄存器来决定分频因子及相关设置； 3) ACLK：辅助时钟信号，用于提供 CPU 外围功能模块作时钟信号使用

9、MSP430 系列单片机最显著的特点是什么？采用什么措施才能具备这样的特点？

答：显著特点：超低功耗

措施：1) MSP430 系列单片机的电源电压采用的是 1.8~3.6V 电压。因而可使其在 1MHz 的时钟条件下运行时，芯片的电流会在 200~400uA 左右，时钟关断模式的最低功耗只有 0.1uA

2) 独特的时钟系统设计。在 MSP430 系列中有两个不同的系统时钟系统：基本时钟系统和锁频环（FLL 和 FLL+）时钟系统或 DCO 数字振荡器时钟系统。有的使用一个晶体振荡器（32768Hz），有的使用两个晶体振荡器）。由系统时钟系统产生 CP [J 和各功能所需的时钟。并且这些时钟可以在指令的控制下，打开和关闭，从而实现对总体功耗的控制。

3) 系统运行时打开的功能模块不同，即采用不同的工作模式，芯片的功耗有着显著的不同。在系统中共有一种活动模式（AM）和五种低功耗模式（LPM0~LPM4）。在等待方式下，耗电为 0.7uA，在节电方式下，最低可达 0.1uA。

10、多个按键控制多个 LED（分别用查询方式、中断方式编程）

J51~S5 按键与 MSP430F6638 P4.0~P4.4 口连接：



图3.1.2 按键模块原理图

答：

//查询方式：

```
#include <reg52.h>
```

```
void delay(unsigned int n)//延时函数
```

```
{
```

```

        unsigned char i;
        while(--n)
            for(i=0;i<125;i++);
    }
    void main()
    {
        unsigned char i = 0;
        unsigned int temp;
        P0=0xff; //p0 端口制一关闭 LED
        P3=0xff; //设置 p3 端口初始值
        while(1) //循环
        {
            if(P3!=0xff) //如果无按键反应，则跳出
            {
                temp=(P3^P0)^0xff; //将上一状况与本次状况合并
                P0=temp;           //送至 p0 口
            }
            delay(500); //延时
        }
    }
}

```

```

//中断方式：
#include "reg52.h"
typedef unsigned char u8;
typedef unsigned int u16;
sbit l1=P2^3;
sbit k1=P3^2;
sbit l2=P2^2;
sbit k2=P3^3;
void delay(u16 i)
{
    while(i--);
}
void anjian1()
{
    if(k1==0)//判断按键是否被按下
    {
        delay(1000);//消抖
        if(k1==0)//消抖之后再次判断是否被按下

        l1=~l1;//点亮 led 灯，led 灯默认高电平
        while(!k1);//判断是否断开
    }
}
}

```

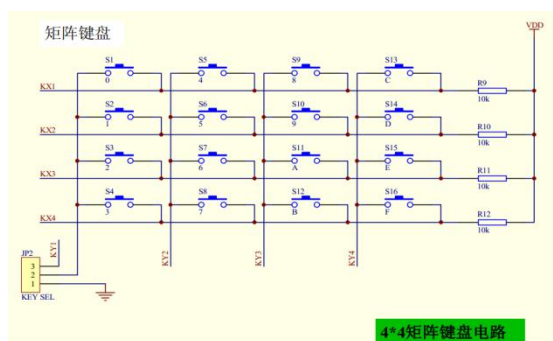
```

void anjian2()
{
    if(k2==0)//判断按键是否被按下
    {
        delay(1000);//消抖
        if(k2==0)//消抖之后再次判断是否被按下

        l2=~l2;//点亮 led 灯，led 灯默认高电'平
        while(!k2);//判断是否断开
    }
}
void main()
{
    while(1)
    {
        anjian1();
        anjian2();
    }
}

```

## 11、矩阵键盘处理程序



答:

```

#include <msp430f249.h>
// #include "io430.h"
char res_list[16] =
{0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,0x77,0x7c,0x39,0x5e,0x79,0x71};
void Output(int src)
{
    if (src>=0 && src<=15) P4OUT = ~res_list[src];
    else P4OUT = ~(0x40);
}
int scanline;
int location(void)
{
    int row=-1,col;

```

```

int _code = P5IN>>4;
switch(_code)
{
    case 0xf:{return -1;}           // no number
    case 0xe:{col = 0;break;}
    case 0xd:{col = 1;break;}
    case 0xb:{col = 2;break;}
    case 0x7:{col = 3;break;}
}
int _scan = scanline;
while(_scan){row+=1;_scan>>=1;}
return row+col*4;
}

int main( void )
{
    scanline = 1;
    P4DIR = 0x7f;
    P4OUT = 0xff;
    P5DIR = 0x0f;
    P5OUT = 0x00;
    Output(-1);
    WDTCTL = WDTPW + WDTHOLD;           // 关闭看门狗，使用计数器
    TAOCTL0 = CCIE;                     // CCR0 中断使能
    TA0CCR0 = 6;
    TAOCTL = TASSEL_2 + MC_1;           // SMCLK, 增计数模式, 清除 TAR 计数器
    __bis_SR_register(LPM0_bits + GIE); // 进入 LPM0,使能中断
}

// TAO 中断服务程序
#pragma vector=TIMER0_A0_VECTOR
__interrupt void TIMER0_A0_ISR(void)
{
    scanline = scanline<<1;
    if(scanline==16) scanline = 1;
    P5OUT = ~scanline & 0x0f;
    int res = location();
    if(res!=-1) Output(res);
}

```

12、编程作业： SMCLK 为 4MHZ，定时 1 分钟，通过 P1.0 交替输出“1”和“0”利用 TAO 定时器，使其工作在增计数模式，采用作为其计数参考时钟，并启用 TA0CCR0 计数中断，

在 TA0 中断服务程序中反转 P1.0 口状态，以便于用示波器进行观察。

答：

```
#include<msp430f6628.h>
void main(void){
    WDTCTL=WDTP+WDTHOLD;      // 关闭看门狗，使用计数器
    P1DIR |=0x01;
    TA0CCTL0 = CCLE;
    TA0CCR0 = 50000;
    TA0CTL=TASSEL_2+MC_1+TACLK;
    _bis_SR_register(LPM0_bits+GIE);
}

// TA0 中断服务程序
#pragma vector = TIMER0_A0_VECTOR_interrupt void TIMER0_A0_ISR(void){
    P1OUT ^=0x01;//反转 P1.0 输出口状态
}
```

13、编程作业：在 MSP430 单片机中，使用 SMCLK（1048576Hz）作为串口时钟源，波特率设置为 9600bps。写出相关初始化程序。

答：

```
UCA0CTL1 |=UCSSEL_2;      //SMCLK
UCA0BR0=6;                //整数分频系数为 6
UCA0BR1 =0;
UCA0MCTL = UCBRS_0+UCBRF_13+UCOS16; //调制器分频 UCBRFx=13，选择过采样模式
```