

第一章

1. 什么是可编程 DSP 芯片？它有什么特点？

答：可编程 DSP 芯片即通用数字信号处理器，具有完整的指令系统，可通过编程来实现各种数字信号处理功能，易于软件更新与系统升级。

主要特点有：

- (1)快速的指令周期，支持在一个指令周期内可完成一次乘法和一次加法运算；
- (2)采用改进的哈佛总线结构，可以同时完成获取指令和数据读取操作；
- (3)片内具有快速 RAM，可通过独立的总线对多个存储器块并行访问；
- (4)硬件支持低开销或无开销的循环及跳转指令，使得 FFT、卷积等运算速度大大提高； (5)快速的中断处理和硬件 I/O 支持，保证了实时响应能力；
- (6)专用寻址单元，具有在单周期内操作的多个硬件地址产生器；
- (7)采用流水线操作，使取指、译码、取操作数和执行指令等操作可以重叠执行；
- (8)片内集成了丰富的外设模块，简化了系统硬件设计。

2. 什么是定点 DSP 芯片和浮点 DSP 芯片？各有什么特点？

答：定点 DSP 芯片：数据以定点格式工作的 DSP 芯片，DSP 里面运算的数据的小数点位置是固定的。主频高、速度快、成本低、功耗小，计算复杂度不高。

浮点 DSP 芯片：数据以浮点格式工作的 DSP 芯片，DSP 里面运算的数据的小数点位置是浮动的。处理速度比定点 DSP 低，成本和功耗比定点 DSP 高，其动态范围和处理精度远高于定点 DSP 芯片，运算复杂度高，精度要求高。

3. 简述 TMS320C2000、TMS320C5000、TMS320C6000 的特点和应用领域。

答：(1) TMS320C2000 系列面向控制应用领域进行了优化，基于 32 位处理器内核，主频 300MHz，具有强大运算和控制功能，片内集成了丰富的控制外设模块，便于构成高效能的工业测控系统，片内集成了 Flash 存储器，简化了用户硬件电路；集成了异步串行通信接口，易于通过标准通信接口实现与计算机或其他微处理器间的远距离通信。主要用于大存储设备管理、高效能的实时控制等场合，如电机驱动控制、大容量开关电源、医疗仪器等。

(2) TMS320C5000 系列针对消费数字产品进行了优化，兼顾了低功耗、低成本和高效能，功耗低至 0.05mW/MIPS。基于 16 位定点 DSP，主频 300MHz，适用于 3G 手机及基站、无线调制解调器等对功耗要求苛刻的个人和便携式产品。

(3) TMS320C6000 系列针对高性能的复杂应用系统进行了优化，其中 C64x 指令执行速度 4800MIPS，C67x，主频 350MHz，C6000 系列片内可集成最多 8 个 CPU，适用于机器视觉、高性能计算、视频处理和高端成像设备。

4. 试比较 TMS320F281x 与你熟悉的一种 MCU 在运算能力和片内集成外设方面的差异。
8051 单片机。

答：运算能力：8051 单片机拥有 8 位数据总线，16 位地址总线的 CPU，采用哈佛结构，0~8KB 片内程序存储器，两个 16 位定时/计数器。而 TMS320F281x 时钟频率高达 150MHz，集成 32 位 C28x 内核和高性能外设，其优化的内核允许正在高达 10KHz 的采样频率下执行多种复杂的控制算法，TI 还提供了定点和浮点微处理器。

片内集成：8051 单片机片内集成存储器、CPU I/O 端口、定时器/计数器、中断系统。而 TMS320F281x 片内集成了大容量的 Flash 存储器与 RAM 功能强大的事件管理器、12 位高速 A/D 转换器及外部扩展接口、通用数字 I/O、McBSP、SCI、SPI、CAN 等接口，F2823x 还进一步提供了 DMA、I2C 接口及高分辨率的 PEM 通道。

5. 简述 TMS320F281x 系列 DSP 芯片的特点，F2812 和 F2810 的主要区别有哪些？

答：

a) 高性能 CPU

(1)32 位定点 CPU，支持 16x16 位和 32x32 位的乘和累加操作；

(2)时钟频率高达 150MHz；

(3)增强型哈佛总线结构，4MB 的程序/数据寻址空间；

(4)快速的中断响应和处理；

(5)低功耗设计；

(6)支持空闲、后备和挂起三种低功耗工作模式；

(7)支持 JTAG 仿真接口。

b) 片内存储器

(1)128Kx16 位 Flash 存储器，方便固化用户应用软件及代码升级；

(2)1Kx16 位的一次可编程存储器 (OTP)；

(3)18Kx16 位单周期访问随机存储器 (SARAM)，分为 M0，M1，L0，L1 和 H0 五个 RAM 块；

(4)4Kx16 位的引导 ROM，提供多种上电引导模式；

(5)128 位密钥可用于保护 Flash/OTP/L0/L1 寻址空间，防止用户代码被非法访问。

c) 外部扩展接口

(1)1Mx16 位的寻址空间，分为 5 块区域，便于开发复杂的应用系统；

(2)支持可编程的等待状态和读写选通时序，便于灵活配置 DSP与扩展芯片间的时序；

(3)提供 3 个独立的片选信号，简化了系统硬件设计。

d) 两个事件管理模块

(1)8 通道 16 位 PWM 输出；

(2)灵活的死区产生和配置单元；

(3)外部可屏蔽的功率 /驱动保护中断；

(4)正交脉冲编码电路 (QEP)；

(5)三个捕捉单元，可精确捕捉外部引脚电平发生跳变的时刻。

e) 串行接口外设

(1)一个高速同步串行外设接口 (SPI)；

(2)两个串行通信接口 (SCIA和 SCIB)；

(3)一个兼容 CAN2.0B 标准的增强型控制局域网接口 (eCAN)；

(4)一个多通道缓冲串口 (McBSP)。

f) A/D 转换模块

(1)12 位分辨率的 A/D 转换器；

(2)2 个 8 通道输入多路开关；

(3)2 个采样保持器；

(4)借助排序器，可编程实现多个通道的自动转换；

(5)A/D 转换速率可达 12.5MSPS，输入电压范围为 0-3V。

g) 其他外设模块

(1)锁相环控制的 CPU时钟倍频系数；

(2)看门狗定时模块；

(3)3 个外部中断源；

- (4)外设中断扩展模块；
- (5)3 个 32 位的 CPU通用定时器；
- (6)56 个可编程的通用 I/O 引脚。

h) 软硬件开发工具

- (1)ANSI C/C++编译器 / 汇编器 / 链接器；
- (2)兼容 TMS320F24x/20x 处理器的源代码；
- (3)通用的集成开发环境；
- (4)提供 DSP BIOS支持；
- (5)具有并口、 USB、 PCI/ISA等接口的硬件仿真器；
- (6)提供初学者开发套件、评估模板以及广泛的第三方支持。

2810/2812 主要区别：

- (1)片内 Flash 容量不同， 2810 为 64K， 2812 为 128K；
- (2)2812 有外部扩展接口， 2810 无；
- (3)2810 芯片封装 128 引脚 LQFP， 2812 芯片封装 176 引脚 LQFP/PBGA

6. TMS320F281x系列 DSP芯片有哪些外部接口？

答：

- (1)XINTF， 1Mx16 位的寻址空间，分为 5 块区域，便于开发复杂的应用系统；
- (2)串行外设接口 (SPI)；
- (3)串行通信接口 (SCI)；
- (4)兼容 CAN2.0B 标准的增强型控制局域网接口 (eCAN)；
- (5)多通道缓冲串口 (McBSP)；
- (6)事件管理模块 (EV)
- (7)A/D 转换模块

7. 简述 TMS320F280x、 TMS320F281x、 TMS320F2833xDSP芯片各有什么特点。

答：

TMS320F280x

- (1)时钟频率 100MHz

(2)有引导 ROM、一次性可编程 ROM

(3)无外部扩展接口

(4)有看门狗定时器

(5)有 16 通道 ADC转换器

(6)有 SPI SCI eCAN 模块

(7)芯片封装 100 引脚

(8)内核电压 1.8V , I/O 电压 3.3V

TMS320F281x

(1)时钟频率 150MHz

(2)有引导 ROM、一次性可编程 ROM

(3)仅 2812 有外部扩展接口

(4)有看门狗定时器

(5)有 16 通道 ADC转换器

(6)有 SPI SCI eCAN 模块 , 多通道缓冲串口

(7)芯片封装 128 或 176 引脚

(8)内核电压 1.8/1.9V , I/O 电压 3.3V

(9)有事件管理器

TMS320F2833x

(1)时钟频率 150MHz

(2)增大了 Flash 和 SRAM存储器的容量

(3)提供 DMA 控制器

(4)有看门狗定时器

(5)有 16 通道 ADC转换器

(6)有 SPI SCI eCAN 模块 , 多通道缓冲串口

(7)芯片封装 128 或 176 引脚

(8)内核电压 1.9V , I/O 电压 3.3V

(9)有事件管理器

8. 简述 DSP控制系统的典型构成和特点。

DSP控制系统包括数字控制器、被控对象、执行机构和反馈元件 4 个部分。

特点：

- (1)在开发阶段易于进行各种控制策略的试验，便于对控制性能进行评价和改进，易于实现复杂的控制策略；
- (2)可承担设定值给定、数字滤波等额外任务；
- (3)传感器和其他参数标定更为容易；
- (4)可在线监测控制系统的运行工况，实现数据的显示、记录及远距离传输，便于在意外情况下分析故障原因；
- (5)系统的改进一般只涉及软件更改，便于更新升级；
- (6)实现的算法基本不受温度、时间等外界环境因素的影响，长时间工作稳定性、一致性好；
- (7)借助于网络易于实现复杂的控制系统结构。

第二章

1.简述在高速 DSP芯片内部配置 PLL的优点、

答：PLL 电路利用高稳定度的锁相环锁定时钟振荡频率，可提供稳定、高质量的时钟信号。

同时，可通过 PLL的 4 位倍频系数设置位来改变时钟频率，以使用户灵活设定需要的处理器速度。借助于 PLL 模块，允许用户选取较低的外部时钟频率，经过锁相环倍频后为 CPU提供较高的时钟频率。这种设计可有效地降低高速时钟信号电平切换时导致的高频噪声，保证时钟信号的波形质量，并简化硬件设计与电路板的布线。

2.若 CPU的时钟频率为 150MHz，试计算高速外设时钟和低速外设时钟的频率设定范围。

答：高速时钟 =SYSCLKOUT(HSPCLK \times 2)；低速时钟 = SYSCLKOUT(LSPCLK \times 2)。所以高速时钟范围是： 10.71MHz 到 150MHz；低速时钟频率范围是： 10.71MHz 到 150MHz。

3.简述看门狗模块的工作原理。假定时钟 OSCCLK的频率为 30MHz，试根据时钟分频系数取值，计算看门狗定时器的计数溢出周期。

答：工作原理：看门狗模块监视系统软件和硬件的运行，它可以按照用户设定的时间间隔产生中断或复位系统，如果软件进入非正常循环或运行到非法的程序空间，使得系统无法正常工作，那么看门狗定时器的计数器上溢，可以产生一个中断或复位信号，使系统进入用户预先设定的状态。

计数溢出周期：由 $WDCLK=OSCCLK/512/2^{(WDPS-1)}$ 得， $WDCLK=30/512MHz$ 到 $30/512/2^6$ ，所以 8 位看门狗计数器最大值为 256，溢出周期为 $256/(30/512)$ 到 $256/(30/512/2^6)MHz$ ，即

4.4ms 到 279.6ms。

4. 试将 CPU 定时器与你熟悉的一种单片机片内定时器或定时器接口芯片（如 8254）进行比较，简述二者的差异。

答：8051 的片内定时器最大为 16 位（也可配置成 8 位），它只能加计数，并在 0xffff 之后加 1 上溢，产生中断。所以，当需要获得一个定时周期值时，是先给它装载一个初值（=溢出值 0xffff - 需要的定时周期值），溢出后要重装这个初值。C28x 内核的 CPU 定时器则是 32 位，减计数，下溢中断，需要的周期值有对应的周期寄存器，中断后自动重装。C28x 的 32 位定时器，极大地提高了定时器的定时精度和定时长度。

5. 假设 CPU 的时钟频率为 150MHz，试根据周期寄存器和分频寄存器的取值范围，计算 CPU 定时器可以实现的定时最大周期值。

答：已知计数器共 32 位，分频寄存器共 16 位，所以 $2^{16} \times 2^{32} / 150\text{MHz} = 1876499.8\text{s} = 21.7$ 天，即最大定时周期为 21.7 天。

6. 试分析改变 PLLCR 寄存器的值时，对定时器的中断周期有什么影响。

答：首先，改变 PLLCR 寄存器的值能改变 DIV 的值，进而改变 CLKIN 的值，即当 PLLCR 为 0000 时， $\text{CLKIN} = \text{OSCCLK} / 2$ ；当 PLLCR 为 0001~1010 时， $\text{CLKIN} = (\text{OSCCLK} \times \text{DIV}) / 2$ ；当 PLLCR 为 1011~1111 时，CLKIN 状态维持原状。因为 CLKIN 与 SYSCLKOUT 同频率，所以改变 PLLCR 能改变 SYSCLKOUT 的值。SYSCLKOUT 变化将使得定时器产生中断的时间不同，即计时周期发生变化。所以改变 PLLCR 寄存器中的值会影响定时器的中断周期的大小，且当 PLLCR 数值增大时，SYSCLKOUT 增大，定时器中断周期减小。

7. F281x 芯片的很多引脚是复用的，结合芯片封装尺寸、引脚利用效率、功能配置等方面，讨论这些复用引脚有哪些优缺点。

答：F281x 集成了众多的外设，每个外设或多或少都会有与片外的输入输出联系，即需要配置引脚。在每个应用中，实际使用的外设又会有不同，将不用的外设的引脚用于基本的简单输入输出引脚功能，即 GPIO 功能，就比较合理。这样，有些引脚就需要复用。F281x 的引脚复用规则是：将一种外设的功能与简单的输入输出功能复用在一起。这样提高了引脚的利用效率，进而减小了芯片的封装尺寸，这是优点。缺点就是使用中需要配置引脚。

8.简述 CPU中断和外设级中断的相应流程，二者的主要区别体现在什么方面？

答：

CPU级中断：

- (1)CPU收到来自 PIE的中断， IFR置位；
- (2)CPU 清零 IFR 和 IER中的相应位，置位 INTM 禁止全局中断
- (3)刷新流水线并保存返回的地址，自动现场保护
- (4)要执行的中断服务程序地址直接取自 PIE 中断向量表
- (5)取出中断向量后自动将中断标志位清零

外设级中断：

- (1)外设中断事件发生， IF 置 1
- (2)若 IE 已置 1，则外设向 PIE 控制器产生一个中断请求
- (3)中断服务程序中由用户软件清零 IF，准备接受下一次中断

主要区别：CPU级中断是取出中断向量后自动将中断标志位清零，外设级中断是中断服务程序中由用户软件清零 IF。

9.参考 2.4.5 节的例程，假设改为由 CPU定时器 1 产生中断，试编写中断初始化和中断服务程序。

答：程序如下所示：

```
Void main(void)
{
    InitSysCtrl();

    DINT;

    IER=0x0000;

    IFR=0x0000;

    InitPieCtrl();

    InitPieVectTable();

    InitPeripherals();

    #if(0)

    EALLOW;

    PieVectTable.TINT=&ISRTimer0;

    EDIS;
```



```

ConfigCpuTimer(&CpuTimer0,150,1000000);

    CpuTimer0Regs.TCR.bit.TSS=0;

    PieCtrl.PIEIER1.bit.INTx7=1;

    IER|= M_INT1;

    EINT;

    #else

    EALLOW;

    PieVectTable.XINT13=&ISRTimer1;

    EDIS;

ConfigCpuTimer(&CpuTimer1,100,100000);

    CpuTimer1Regs.TCR.bit.TSS=0;

    IER|= M_INT13;

    #endif

```

第三章

1.如何通过软件判断内部 RAM 单元或者外部 RAM 芯片是否正常工作？对于 Flash 或 EPROM 等存储芯片应该如何诊断？

答：通过写入与读出值的比对来诊断内部 RAM 单元或外部 RAM 单元是否正常工作。判断 RAM 单元的工作是否正常，不论内部 RAM 单元或外部 RAM 单元，一般先写入一个值，再读出来与写入值校验。比如对同一个地址写入 0~0xFF，然后读出数据，看两者是否相同，再从地址 0 开始写入有规律的数据，如全 0 或全 1，还有 0 与 1 间隔如 0x55,0xAA,,然后读出看与原数据是否相同。独立芯片的 Flash 和 EPROM,需要在专门的编程器上来操作。有擦除、空检查、写入、校验等操作。

2.外部扩展接口 (XINTF) 适合扩展哪些外设芯片？

答：XINTF 适合扩展并行的外设芯片，如 ADC、DAC、RAM、USB、FIFO 等接口。

3.F2812 提供了 3 个供外设使用的片选信号，如果扩展的外设芯片超过 3 个如何产生这些外设芯片的片选信号？

答：提供的三个片选信号代表了很大的寻址空间，所以当外设芯片超过 3 个时，可以用高位

的地址线来译码出更多的次片选信号，通过次片选信号来跟外设匹配。同时还可以通过 XREADY信号来与外设的访问速度和时序匹配。

4.对于例 3.2，分析空间 2 的一个读周期包含的 XTIMCLK时钟周期数；如果 CPU时钟频率为 150MHz，则完成一个完整的读周期需要多长的时间？

答：根据例程可知，XRDLEAD位值为 3,XRDACTIVE位值为 7,XRDTRAIL位值为 3,同时 X2TIMING位值为 1，说明 XTIMCLK=SYSCLKOUT/2因为 SYSCLKOUT为 150MHz，所以一个完整的读周期需要时间为：（3+7+3+1）/150MHz=0.213 μs。

5.对于图 3.10，设要求通道 A、B、C、D 的输出电压分别为 -4V、0V、2V、4V，试编程实现四路 ADC的输出同步刷新。

答：程序如下：

```
Uint16 *ADD_DAC_A = (Uint16 *) 0x80000 ;
Uint16 *ADD_DAC_B = (Uint16 *) 0x80001 ;
Uint16 *ADD_DAC_C = (Uint16 *) 0x80002 ;
Uint16 *ADD_DAC_D = (Uint16 *) 0x80003 ;

main(void)
{
    InitSysCtrl();

    DINT;

    IER = 0x0000;

    IFR = 0x0000;

    EALLOW;

    GpioMuxRegs.GPAMUX.bit.GPIOA1=0;
    GpioMuxRegs.GPAMUX.bit.GPIOA2=0;
    GpioMuxRegs.GPADIR.bit.GPIOA1=1;
    GpioMuxRegs.GPADIR.bit.GPIOA2=1;

    EDIS;

    GpioDataRegs.GPADAT.bit.GPIOA1=0

    Delay();

    GpioDataRegs.GPADAT.bit.GPIOA1=1
```

```

GpioDataRegs.GPADAT.bit.GPIOA2=1

*ADD_DAC_A=(-4-(-5))*2^14/10 ;

*ADD_DAC_B=5*2^14/10 ;

*ADD_DAC_C=(2-(-5))*2^14/10 ;

*ADD_DAC_D=(4-(-5))*2^14/10 ;

GpioDataRegs.GPADAT.bit.GPIOA2=0

Delay();

GpioDataRegs.GPADAT.bit.GPIOA2=1

for(;;){

}

```

第七章

1.与 2.2 节介绍的 CPU 定时器相比，事件管理器的通用定时器有何特点？

答：事件管理器的 EVA 和 EVB 各有 2 个通用定时器。其特点为：16 位；包含计数器、比较寄存器（双缓冲）、周期寄存器；可连续增计数、连续增 / 减计数、定向增 / 减计数；可产生对称和非对称 PWM 波；中断事件有下溢、上溢、比较匹配和周期匹配。CPU 定时器：32 位，连续减计数，只在减计数到 0 时使中断标志置位。

2.对于例 7.2，当高速外设时钟设为 75MHz 时，试计算定时器 1 的定时周期。

答：由 T1CON.all=0x1742 可知其模式为连续增计数模式，分频系数为 128；根据周期寄存器值 0x0200 可知，周期为 $2 \times 16 \times 16 + 1$ ，所以定时周期为 $(2 \times 16 \times 16 + 1) \times 128 / 75\text{MHz} = 875.5 \mu\text{s}$ 。

3.设置 PWM 输出引脚的输出方式为高有效和低有效时，对占空比的设定有什么区别？

答：PWM 输出引脚的波形是比较寄存器中的值与计数寄存器中值相比较的结果，使用高电平有效输出，引脚的占空比与比较寄存器中的值成反比关系，即设置的值越大，输出占空比越小，使用低电平有效输出，引脚的占空比与比较寄存器中的值成正比关系。对于同一个设定值，高有效和低有效的占空比之和恰好为 1。

4.与线性功率放大器相比，PWM 功率放大器有何优点？适用于哪些场合？

答：优点：通过调节占空比，利用负载的滤波作用，其功率损耗比较低、放大器的输出是一

串宽度可调的矩形脉冲； 适用场合： 数字音频、 各种 LED模块、 直流伺服系统、 扬声器等。

5.与通用定时器产生的 PWM 波形相比，应用全比较单元和 PWM 电路产生的 6 路 PWM 输出有何特点？

答：通过通用定时器产生的 PWM 波形是一路独立的 PWM 波，是通过简单比较产生的。全比较单元和 PWM 电路产生的 6 路 PWM 波形是三对互补输出的 PWM 波形。一路 PWM 适合启用 DAC转换器和 PWM 功率放大器，而 6 路 PWM 波形适合目前常用的运动控制功率放大电路的拓扑结构，且可以插入死区。

6.假设例 7.3 中的告诉外设时钟频率为 75MHz，试分析各路 PWM 信号的频率和占空比。

答：由 T1CON可知，输出为无分频系数、连续增计数模式，由 T1PR可知，周期为 FFFF, 比较值为 3C00，由 T1PIN可知，T1PWM 为低电平有效。

综上分析可得占空比为： $(15360/65535+1)=23.4\%$ ，频率为： $75\text{MHz}/65536=1.144\text{KHz}$ ；
同理 T2PWM 占空比为： $960/(4096+1)=76.6\%$ ，频率为： $75\text{MHz}/4097=18.31\text{KHz}$ 。

对于 3 对 PWM 信号来说，分析相同，由 ACTRA可得，死区为 0，所以 PWM1/3/5 高电平有效，占空比分别为： $1-(3072/65536)=95.3\%$ ， $1-(15360/65535+1)=76.6\%$ ， $1-(64512/65536)=1.6\%$ ，频率为： $75\text{MHz}/65536=1.144\text{KHz}$ ；

PWM2/4/6 的占空比分别为 4.7%，23.4%，98.4%，频率为 $75\text{MHz}/4097=18.31\text{KHz}$ 。

7.假定采样频率为 100Hz，试根据例 7.4 编程实现电动机的转角和转速测量与实时刷新。

答：

```
#include "DSP281x_Device.h"

Uint32 count_out=0;

Uint16 Count_angle;

interrupt void cpu_timer0_isr(void);

Void main(void)

{

InitSysCtrl();

DINT;

InitPieCtrl();

IER = 0x0000;
```

```

IFR = 0x0000;

InitPieVectTable();

EALLOW;

PieVectTable.TINT0 = &cpu_timer0_isr;

EDIS;

InitEvB();

InitCpuTimers();

ConfigCpuTimer(&CpuTimer0, 150, 10000);

StartCpuTimer0();

IER |= M_INT1;

PieCtrlRegs.PIEIER1.bit.INTx7 = 1;

EINT;

for(;;)

{

    Count_angle= EvaRegs.T4CNT *0.1+ count_out*18.2*360;

    LEDdisplay(Count_angle);

}

}

Void InitEvB(void)

{

    EALLOW;

    GPBMUX[0]=0xffff;

    EDIS;

    T4CON[0]=0x187c;

    CAPCONB[0]=0x70fc;

    T4PR[0]=0xffff;

    T4CNT[0]=0x8000;

}

interrupt void cpu_timer0_isr(void)

{

    If(EvaRegs.T4CNT>=0xffff)

```

```
{  
    count_out++;  
}  
PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;  
}
```