简单的叙述一下dsp芯片的发展概况

第一阶段，DSP的雏形阶段（1980年前后）代表产品：S2811 主要用途：军事或航空航天部门

第二阶段，DSP的成熟阶段（1990年前后）代表产品：TI公司的TMS320C20 主要用途：通信、计算机领域

第三阶段，DSP的完善阶段（2000年以后）代表产品：TI公司的TMS320C54 主要用途：各个行业领域

dsp技术的发展趋势主要体现在什么方面

1）DSP的内核结构将进一步改善

2）DSP 和微处理器的融合

3）DSP 和高档CPU的融合

4）DSP 和SOC的融合

5）DSP 和FPGA的融合

6）实时操作系统RTOS与DSP的结合

7）DSP的并行处理结构

8）功耗越来越低

简述TI公司C2000, C5000, C6000系列处理器的应用范围

C2x、C24x称为C2000系列，主要用于数字控制系统；

C54x、C55x称为C5000系列，主要用于功耗低、便于携带的通信终端；

C62x、C64x和C67x称为C6000系列，主要用于高性能复杂的通信系统，如移动通信基站。

简述DSP系统的基本构成

典型的DSP系统包括数字信号处理器DSP、存储器、A/D和D/A转换器、模拟控制和处理电路、各种控制口与通信口、电源管理、为并行处理或协处理提供的同步电路等。

浮点DSP芯片与定点DSP芯片 各有什么优缺点

按数据的定点格式工作的DSP芯片称为定点DSP 价格便宜 功耗低 但运算精度低

按数据的浮点格式工作的DSP芯片称为浮点DSP C语言编程调试方便 运算精度高 但价格较高

dsp系统有哪些特点

1）接口方便 2）编程方便 3）具有高速性 4）稳定性好 5）精度高

6）可重复性好 7）集成方便

在进行dsp系统设计时，应如何选择合理的dsp芯片

1）dsp的运算速度

2）dsp芯片价格

3）dsp芯片运算精度

4）dsp芯片的硬件资源

5）dsp芯片的开发工具

6）dsp芯片的功耗

7）其他因素

dsp芯片有哪些特点

1）哈佛结构 2）多总线 3）流水线技术 4）硬件配置强 5）多处理单元 6）低功耗

TSM320C45x的CPU主要包括哪些部分 他们的功能

40位算术逻辑运算单元（ALU）(完成宽范围的算术逻辑运算)

40位累加器A和B（用于存放从ALU或乘/加单元输出的数据，也能输出数据到ALU或乘/加单元）

移位-16~30位的桶行移位寄存器（对输入数据进行0到31位左移和0到16位的右移）

乘法器/加法器单元（可在一个指令周期里完成17\*17位的进制补码乘法运算，也可在一个流水线状态周期内完成一个乘法累加运算）

比较和选择及存储单元（专门用于快速执行Viterbi算法的特殊硬件）

指数编码器（求累加器中数的指数值）、CPU状态和控制寄存器

40位累加器A和B异同

同：存放运算前后数据

异：累计器A的31～16位可以用作乘法器的一个输入

桶形移位器

功能：1）在ALU运算前，对来自数据存储器的操作数或者累加器的值进行定标

2）对累加器的值进行算术或逻辑移位

3）对累加器归一化处理

4）对累加器的值存储到数据存储器之前进行定标

ST0、ST1、PMST的作用？它们是如何影响DSP工作的

ST0主要反映处理器的寻址要求和计算机的运行状态。

ST1主要反映处理器的寻址要求、计算初始状态的设置、I/O及中断的控制等。

PMST 主要设定和控制处理器的工作方式和存储器的配置，反映处理器的工作状态

TSM320C54x芯片的总线有哪些 他们各自的作用和区别是什么

8条16位主总线：

1）1条程序总线（PB） 传送取自程序存储器的指令代码和立即数操作数

2）3条数据总线（CB、DB和EB） 将内部各单元连接在一起

3）4条地址总线（PAB、CAB、DAB和EAB） 传送执行指令所需的地址

**填空** 在读程序存储器时，可能用到的总线有：地址总线PAB和程序总线PB；

在写数据存储器时，可能用到的总线有：地址总线EAB和数据总线EB。

什么是哈佛结构和冯诺依曼结构 它们有什么区别

哈佛结构：采用双存储空间，程序存储器和数据存储器分开，有各自独立的程序总线和数据总线，可独立编址和独立访问，可对程序和数据进行独立传输，使取指令操作、指令执行操作、数据吞吐并行完成，大大地提高了数据处理能力和指令的执行速度，非常适合于实时的数字信号处理。

冯诺依曼结构：该结构采用单存储空间，即程序指令和数据共用一个存储空间，使用单一的地址和数据总线，取指令和取操作数都是通过一条总线分时进行。当进行高速运算时，不但不能同时进行取指令和取操作数，而且还会造成数据传输通道的瓶颈现象，其工作速度较慢。

区别

哈佛：该结构采用双存储空间，程序存储器和数据存储器分开，有各自独立的程序总线和数据总线，可独立编址和独立访问，可对程序和数据进行独立传输，使取指令操作、指令执行操作、数据吞吐并行完成，大大地提高了数据处理能力和指令的执行速度，非常适合于实时的数字信号处理。

冯：当进行高速运算时，不但不能同时进行取指令和取操作数，而且还会造成数据传输通道的瓶颈现象，其工作速度较慢。

试述三种存储器空间的各自作用及配置情况是什么

程序存储空间：存放要执行的指令和指令执行中所用的系数表 CPU自动通过总线P以及外部总线访问程序空间

数据存储空间：存放执行指令所要用的数据 CPU根据指令通过总线CB、DB、EB以及外部总线访问数据空间

I/O空间：I/O空间与存储器映像外围设备相接口 也可以作为附加的数据存储空间使用 CPU根据I/O读写指令通过总线C/D/E以及外部总线访问I/O空间

TMS320C54x存储空间分配及片内存储器类型

总存储空间为192K 由3个可选择的存储空间构成

1）64K的程序空间

2）64K的数据空间

3）64K的I/O空间

片内存储器：

ROM(只读存储器)

RAM(随机存储器) 其中RAM可分为两种SARAM(单寻址)和DARAM(双寻址)

定时器由哪些寄存器组成 它们是如何工作的

定时器寄存器（TIM） TIM是一减1计数器

定时器周期寄存器（PRD） PRD中存放时间常数

定时器控制寄存器（TCR） TCR中包含有定时器的控制位和状态位

都是存储器映像寄存器，它们在数据存储器中的地址分别为0024h、0025h、0026h。

哪些办法可以引起硬件中断或软件中断

软件中断由程序指令INTR、TRAP、RESET要求的中断

硬件中断由外围设备要求的中断

1）受外部中断信号触发的外部硬件中断；

2）受片内外围电路信号触发的内部硬件中断。

简述DSP处理器中断系统分为几种中断，如何清除中断标志

软件中断：软件中断是由程序指令产生的中断请求，有三种情况将清除中断标志

硬件中断：由片外管脚和片内外设引起的中断，又分为可屏蔽中断和非屏蔽中断

1）软件和硬件复位，即C54x的复位引脚RS=0；

2）相应的IFR标志位置1；

3）使用相应的中断号响应该中断，即使用INTR #K指令

CPU是如何响应中断的

对于软件中断和非屏蔽中断，CPU是立即响应

对于可屏蔽中断，只有满足以下条件才能响应

1）优先级别最高

2）状态寄存器ST1中的INTM位为0

3）中断屏蔽寄存器IMR中的相应位为1

时钟发生器由哪些部分组成 它们是如何工作的

由内部振荡器和锁相环（PLL）电路组成。

内部振荡器提供参考时钟输入

PLL兼有倍频和信号提纯的功能，用高稳定的参考振荡器锁定，可以提供高稳定的频率源。

TMS320C54x硬件复位地址为多少 如何确定中断向量地址 计算INT0（IPTR=001H）的中断向量地址

复位后，复位向量地址为：0080H

确定地址方式：IPTR+左移2位后的中断向量序列号

10H左移2位后成为40H，IPTR=001H，则中断向量地址为00C0H

C54x有哪些主要的基本数据寻址方式

1）立即寻址：由指令提供一个操作数

2）绝对寻址：由指令提供一个操作数地址

3）累加器寻址：以累加器的内容为地址访问程序空间某单元

4）直接寻址：寻址地址为DP或SP的值加上指令提供的偏移量

5）间接寻址：利用辅助寄存器访问存储器

6）存储器映像寄存器寻址：用来改变映像寄存器，但不影响DP或SP的值

7）堆栈寻址：用来管理系统堆栈中的操作

直接寻址有两种方式 它们是什么 如何控制；当SP=2000 偏移地址为25H时 寻址的是哪个存储空间的那个地址单元

直接寻址根据CPL的设置不同进行区分

CPL=0时，为DP寻址

CPL=1时，为SP寻址

此时寻址的是数据存储空间的2025H单元

什么是流水线技术

每条指令可通过片内多功能单元完成取指、译码、取操作数和执行等多个步骤，实现多条指令的并行执行，从而在不提高系统时钟频率的条件下减少每条指令的执行时间。利用这种流水线结构，加上执行重复操作，就能保证在单指令周期内完成数字信号处理中用得最多的乘法 - 累加运算。

简述流水线操作的基本原理

流水线操作是各指令以机器周期为单位相差一个时钟周期，连续并行工作的情况。其本质是DSP多条总线彼此独立地同时工作，使得同一条指令在不同机器周期内占用不同总线资源

TMS320C54x芯片的流水线冲突是怎样产生的 有哪些方法可以避免流水线冲突

C54x的流水线结构，允许多条指令同时利用CPU的内部资源。由于CPU的资源有限，当多于一个流水线上的指令同时访问同一资源时，可能产生时序冲突。

解决办法 1）由CPU通过延时自动解决

2）通过程序解决，如重新安排指令或插入空操作指令。为了避免流水冲突，可以根据等待周期表来

选择插入的NOP指令的数量。

软件开发的环境有哪几种

1）集成的开发环境

2）非集成的开发环境 在非集成开发环境中 软件开发常采用编辑、汇编、链接、调试等功能

什么是COFF格式 它有什么特点

汇编器和链路器生成的目标文件，是一个可以由C54x器件执行的文件。这些目标文件的格式称为公共目标文件格式，即COFF。

特点：在编写汇编语言程序时，COFF采用代码段和数据段的形式，以便于模块化的编程，使编程和管理变得更加方便。

COFF包含三种形式的段：.text段、.data段和.bss段分别包含什么内容?

.text 段(文本段) 通常包含可执行代码

.data 段(数据段) 通常包含初始化数据；

.bss 段(保留空间段) 通常为未初始化变量保留存储空间。

链接器对段是如何处理的

链接器将一个或多个 COFF 目标文件中的各种段作为链接器的输入段，经过链接后在一个可执行的 COFF 输出模块中建立各个输出段，通过情况下是将不同目标文件中的同名段进行合并，并为各个输出段分配进具体的存储器中

什么是程序的重定位

将各个段配置到存储器中，使每个段都有一个合适的起始地址

将符号变量调整到相对于新的段地址的位置

将引用调整到重新定位后的符号，这些符号反映了调整后的新符号值

链接器能完成什么工作（链接命令文件功能）链接器命令文件中,MEMORY 命令和 SECTIONS命令的任务是什么

指定存储器的结构和段的定位，有MEMORY和SECTIONS两条指令。

MEMORY伪指令 用来定义目标系统的存储器配置空间 包括对存储器各部分命名 以及规定它们的起始地址和长度

SECTIONS伪指令 用来指定链接器将输入段组合成输出段方式 以及输出段在存储器中的位置 也可用于指定子段

链接器在处理段的时候,有2个主要任务:

1) 把一个或多个COFF目标文件中的各种段作为链接器的输入段,经链接后在一个可执行的COFF输出模块中建立各个输出段

2) 为各个输出段选定存储器地址

常用汇编命令有哪些 它们的作用是？

段定义命令 对各种段进行定义

常数初始化命令 对常数进行初始化

调整段程序计数器 调整SPC

对输出列表文件格式的命令 对输出列表文件格式化

引用其它文件的命令 引用其他文件

控制条件汇编的命令 控制条件汇编

在汇编时定义符号的命令 在汇编时定义符号

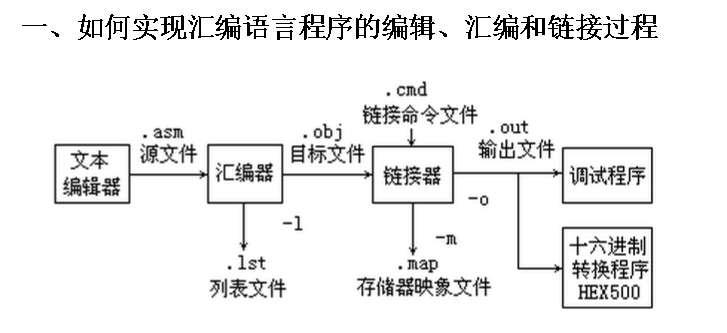
其他汇编命令 执行其他功能

宏指令与子程序有何异同

同：都是重复执行一段程序。

异：它们都可以被多次调用，但是子程序汇编成目代码的过程只进行一次，而在用到宏指令的每个地方都要对宏指令中的语句逐条地进行汇编；在调用前，由于子程序不使用参数，子程序所需要的寄存器等都必须事先设置好，对于宏指令来说，由于可以使用参数，调用时只要直接代入参数就行了

软件开发进程C语言模块流程



后面还有个框图

DSP应用系统设计流程

