

## 第2章 TMS320C55x的硬件结构

### 内容提要

本章详细介绍TMS320C55x的硬件结构，包括C55x处理器的CPU体系结构、指令流水线、存储空间结构及TMS320VC5509A的主要特性等。

## 第2章 TMS320C55x的硬件结构

### 知识要点

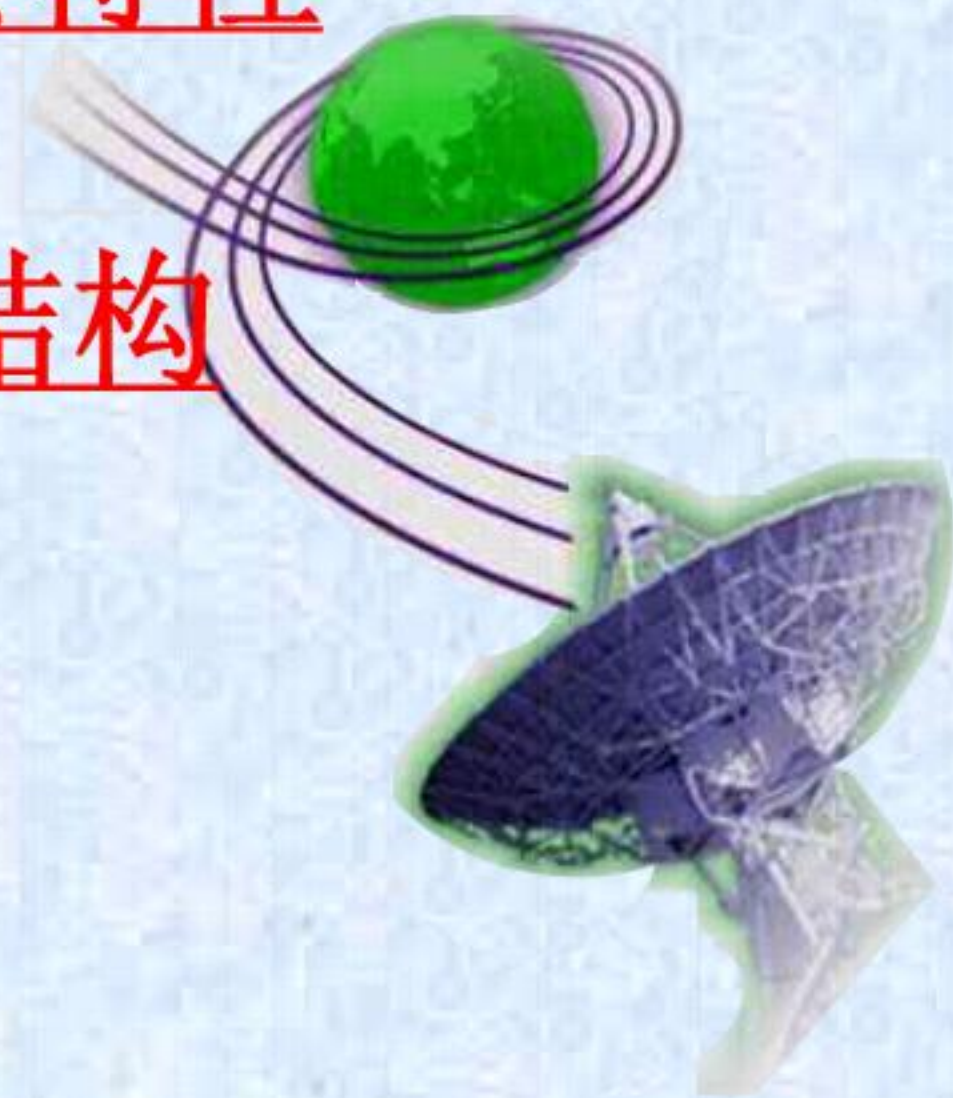
- TMS320C55x DSP的基本结构
- TMS320VC5509A的主要特性
- TMS320C55x 存储空间结构

## 第2章 TMS320C55x的硬件结构

### 2.1 TMS320C55x DSP的基本结构

### 2.2 TMS320VC5509A的主要特性

### 2.3 TMS320C55x 存储空间结构



### 2.1 TMS320C55x DSP的基本结构

TMS320C55x数字信号处理器是在C54x的基础上发展起来的新一代低功耗、高性能数字信号处理器，其软件具有C54兼容模式，极大地节省了C54x向C55x的转化时间。C55x采用了新的半导体工艺，其工作时钟大大超过了C54x系列处理器，CPU内部通过增加功能单元增强了DSP的运算能力，与C54x相比具有更高的性能和更低的功耗。这些特点使之在无线通信、便携式个人数字系统及高效率的多通道数字压缩语音电话系统中得到广泛应用。

## 第2章 TMS320C55x的硬件结构

C55x与C54x相比，C55x在硬件方面做了许多扩展，具体如表2-1所示。

表2-1 C55x与C54x的比较

内 容	C54x	C55x
乘法累加器 (MAC)	1	2
累加器 (ACC)	2	4
读总线	2	3
写总线	1	2
地址总线	4	6
指令字长	16位	8/16/24/32/40/48位
数据字长	16位	16位
算术逻辑单元 (ALU)	1 (40位)	1 (16位) (40位)
辅助寄存器字长	2字节 (16位)	3字节 (24位)
辅助寄存器	8	8
存储空间	独立的程序/数据空间	统一的程序/数据空间
数据寄存器	0	4

## 第2章 TMS320C55x的硬件结构

C55x的一系列特征使它具有处理效率高、低功耗和使用方便的等优点

表2-2 C55x的特征及优点

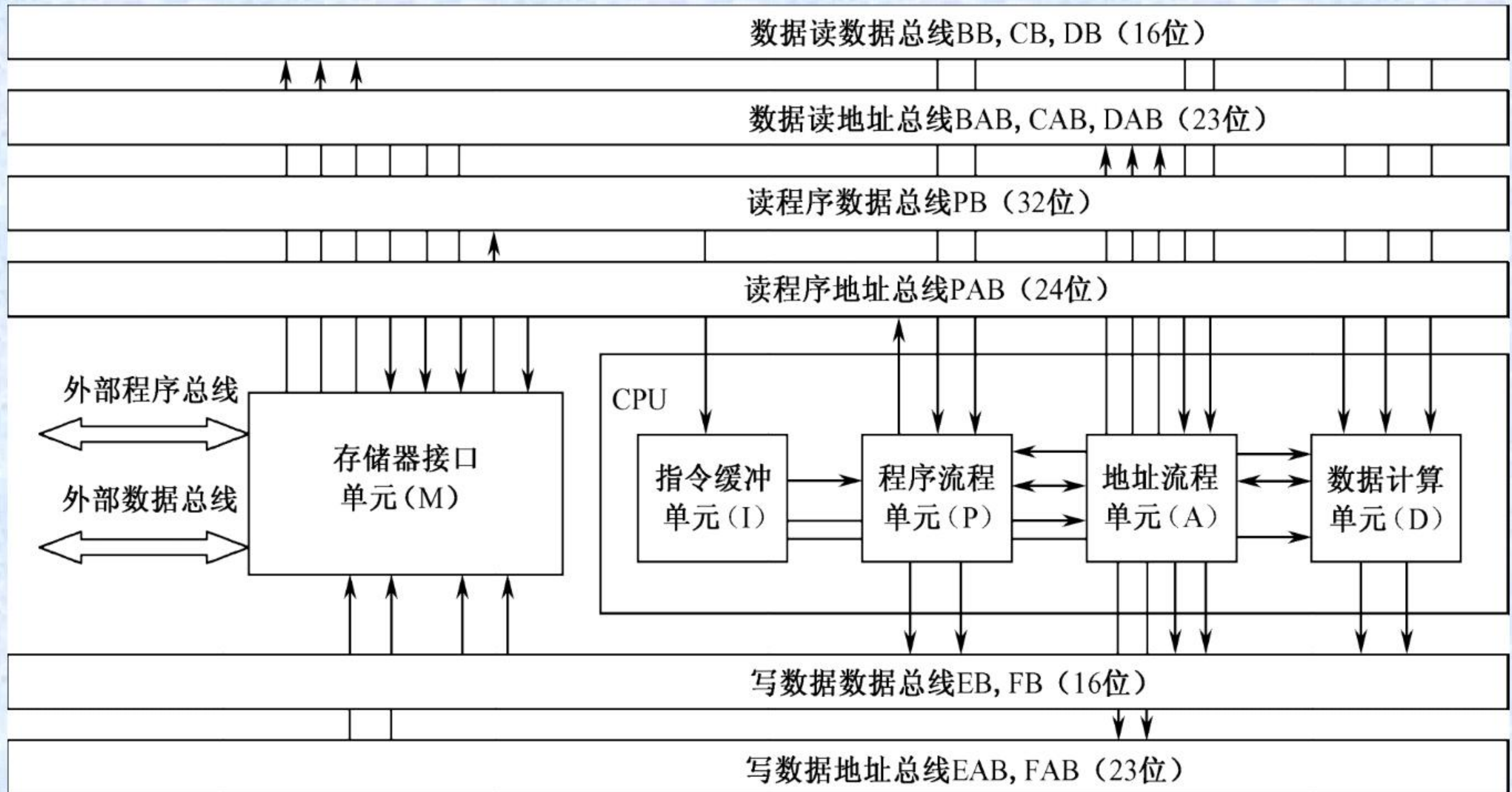
特 征	优 点
一个32位×16指令缓冲队列	缓冲变长指令并完成有效的块重复操作
两个17位×17位的乘法累加器	在一个单周期执行双乘法累加操作
一个40位算术逻辑单元 (ALU)	实现高精度算术和逻辑操作
一个40位桶形移位寄存器	能够将一个40位的计算结果最高向左移31位或向右移32位
一个16位算术逻辑单元 (ALU)	对主ALU并行完成简单的算术操作
4个40位的累加器	保留计算结果，减少对存储单元的访问
12条独立总线，其中包括 3条读数据总线 2条写数据总线 5条数据地址总线 1条读程序总线 1条程序地址总线	为各种计算单元并行地提供将要处理的指令和操作数——利用C55x的并行机制的优点
用户可配置IDLE域	改进了低功耗电源管理的灵活性

### 2.1.1 C55x 的CPU体系结构

C55x有1条32位的程序数据总线（PB），5条16位数据总线（BB、CB、DB、EB、FB）和1条24位的程序地址总线及5条23位的数据地址总线，这些总线分别与CPU相连。总线通过存储器接口单元（M）与外部程序总线 and 数据总线相连，实现CPU对外部存储器的访问。这种并行的多总线结构，使CPU能在一个CPU周期内完成1次32位程序代码读、3次16位数据读和两次16位数据写。C55x根据功能的不同将CPU分为4个单元，即指令缓冲单元（I）、程序流程单元（P）、地址流程单元（A）和数据计算单元（D）。

## 第2章 TMS320C55x的硬件结构

### TMS320C55x CPU结构图





## 第2章 TMS320C55x的硬件结构

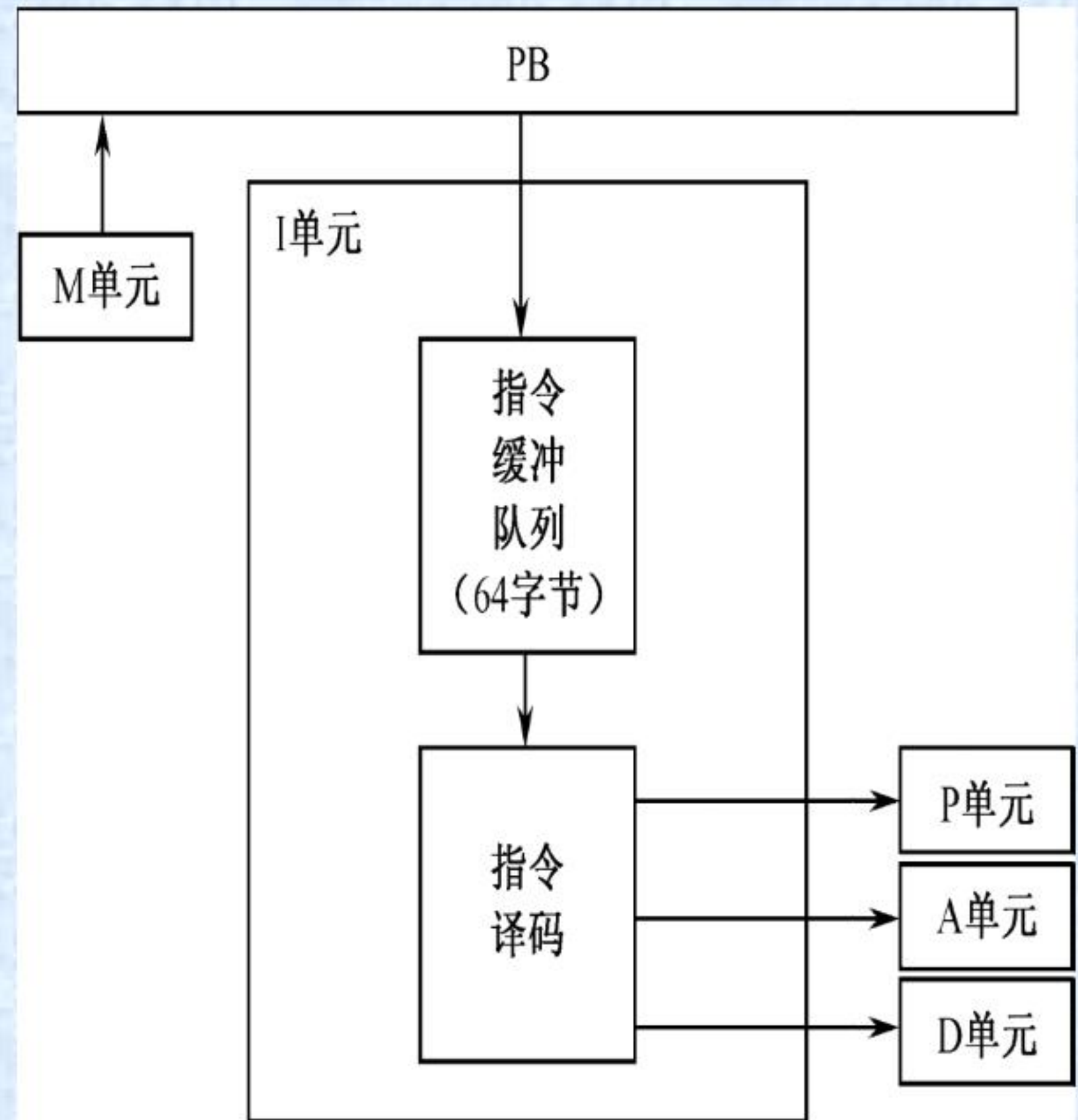
读程序地址总线（**PAB**）上传送24位的程序代码地址，由读程序数据总线（**PB**）将32位的程序代码送入指令缓冲单元**I**进行译码。

3条读数据地址总线（**BAB**、**CAB**、**DAB**）与3条读数据数据总线（**BB**、**CB**、**DB**）配合使用，即**BAB**对应**BB**、**CAB**对应**CB**和**DAB**对应**DB**。地址总线指定数据空间或I/O空间地址，通过数据总线将16位数据传送到CPU的各个功能单元。其中，**BB**只与**D**单元相连，用于实现从存储器到**D**单元乘法累加器（**MAC**）的数据传送。特殊的指令也可以同时使用**BB**、**DB**和**CB**来读取三个操作数。

2条写数据地址总线（**EAB**、**FAB**）与两条写数据数据总线（**EB**、**FB**）配合使用，即**EAB**对应**EB**、**FAB**对应**FB**。地址总线指定数据空间或I/O空间地址，通过数据总线，将数据从CPU的功能单元传送到数据空间或I/O空间。所有数据空间地址由**A**单元产生。**EB**和**FB**从**P**单元、**A**单元和**D**单元接收数据，对于同时向存储器写两个16位数据的指令要使用**EB**和**FB**，而对于完成单写操作的指令只使用**EB**。

### 2.1.2 指令缓冲单元 (I)

C55x的指令缓冲单元由指令缓冲队列IBQ (Instruction Buffer Queue) 和指令译码器组成。在每个CPU周期内，I单元将从读程序数据总线接收的4B程序代码放入指令缓冲队列，指令译码器从队列中取6B程序代码，根据指令的长度可对8位、16位、24位、32位和48位的变长指令进行译码，然后把译码数据送入P单元、A单元和D单元去执行。



指令缓冲单元结构图

### 2.1.3 程序流程单元 (P)

程序流程单元由程序地址产生电路和寄存器组构成。程序流程单元产生所有程序空间的地址，并控制指令的读取顺序。

程序地址产生逻辑电路的任务是产生读取程序空间的24位地址。一般情况下，它产生的是连续地址，如果指令要求读取非连续地址的程序代码时，程序地址产生逻辑电路能够接收来自I单元的立即数和来自D单元的寄存器值，并将产生的地址传送到PAB。

在P单元中使用的寄存器分为5种类型。

(1) 程序流寄存器：包括程序计数器（PC）、返回地址寄存器（RETA）和控制流程关系寄存器（CFCT）。

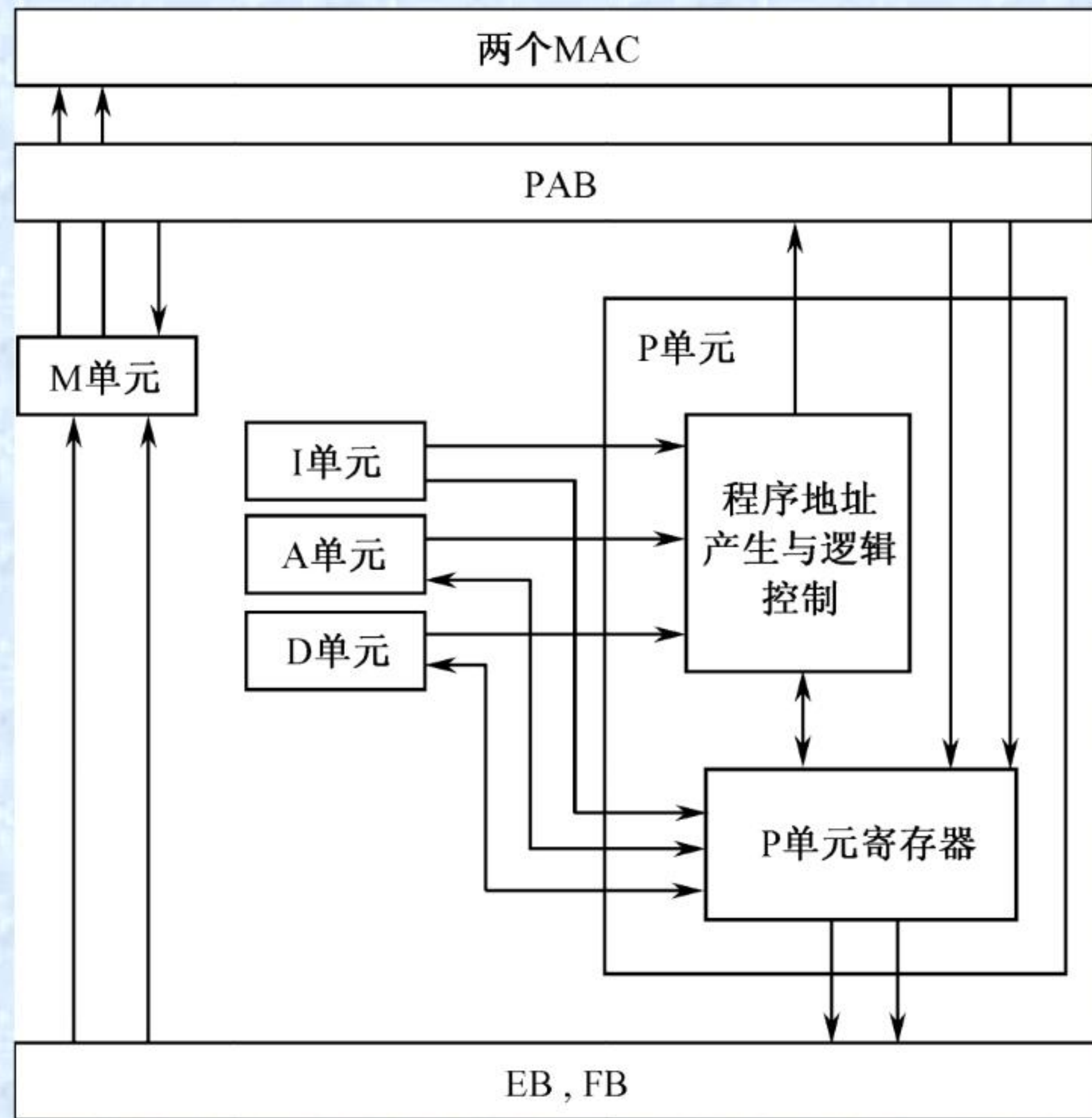
(2) 块重复寄存器：包括块重复寄存器0和1（BRC0, BRC1）、BRC1的保存寄存器（BRS1）、块重复起始地址寄存器0和1（RSA0, RSA1）以及块重复结束地址寄存器0和1（REA0, REA1）。

(3) 单重复寄存器：包括单重复计数器（RPTC）和计算单重复寄存器（CSR）。

(4) 中断寄存器：包括中断标志寄存器0和1（IFR0, IFR1）、中断使能寄存器0和1（IER0, IER1）以及调试中断使能寄存器0和1（DBIER0, DBIER1）；

(5) 状态寄存器：包括状态寄存器0, 1, 2和3（ST0-55, ST1-55, ST2-55和ST3-55）。

## 第2章 TMS320C55x的硬件结构



程序流程单元结构图

### 2.1.4 地址流程单元 (A)

地址流程单元包括数据地址产生电路、算术逻辑电路和寄存器构成。

数据地址产生电路 (DAGEN) 能够接收来自I单元的立即数和来自A单元的寄存器产生读取数据空间的地址。对于使用间接寻址模式的指令，由P单元向DAGEN说明采用的寻址模式。

A单元包括一个16位的算术逻辑电路 (ALU)，它既可以接收来自I单元的立即数，也可以与存储器、I/O空间、A单元寄存器、D单元寄存器和P单元寄存器进行双向通信。ALU可以完成算术运算、逻辑运算、位操作、移位、测试等操作。

A单元包括的寄存器有以下几种类型。

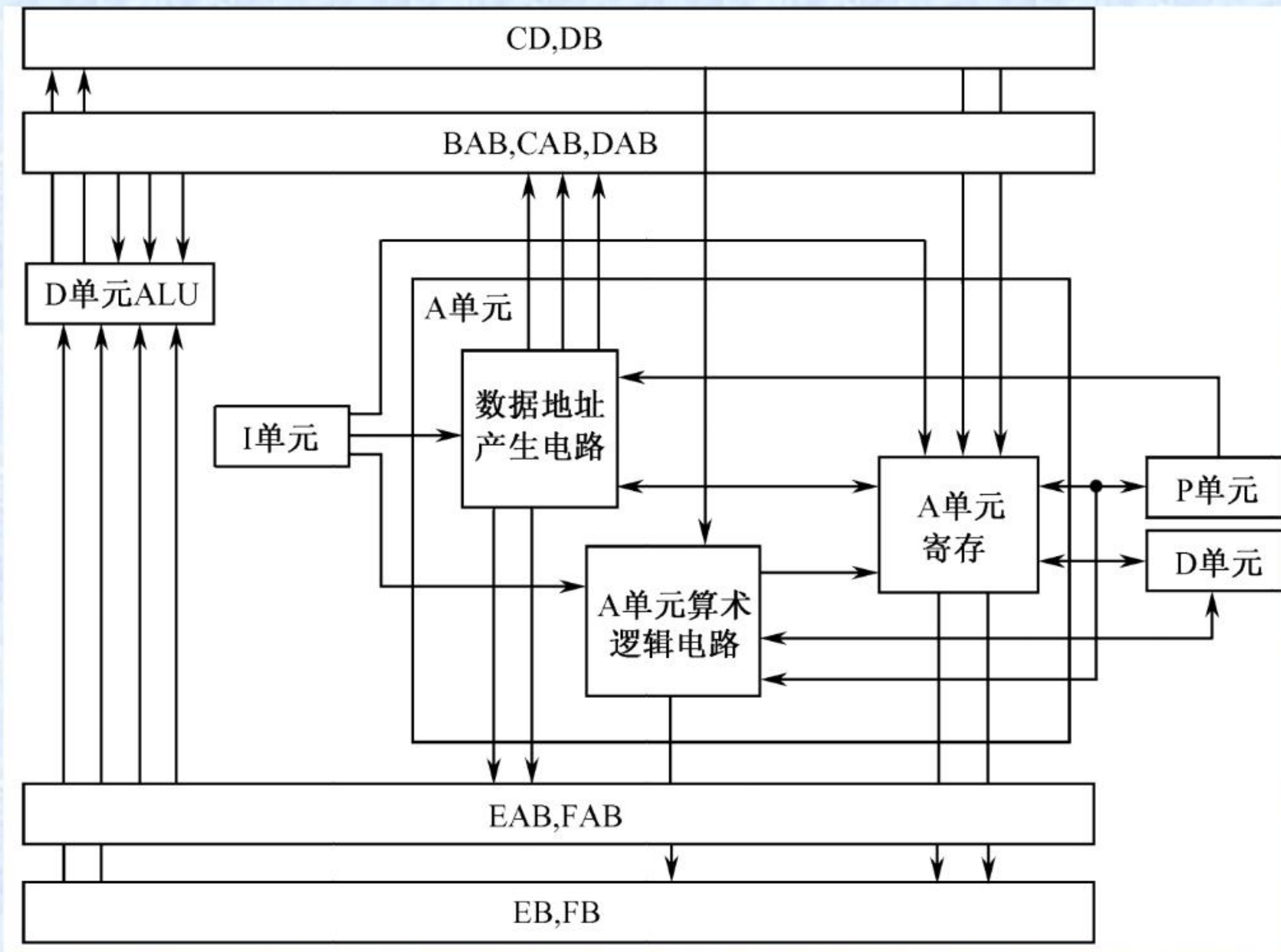
(1) 数据页寄存器：包括数据页寄存器（DPH, DP）和接口数据页寄存器（PDP）；

(2) 指针：包括系数数据指针寄存器（CDPH, CDP）、栈指针寄存器（SPH, SP, SSP）和8个辅助寄存器（XAR0~XAR7）；

(3) 循环缓冲寄存器：包括循环缓冲大小寄存器（BK03, BK47, BKC）、循环缓冲起始地址寄存器（BSA01, BSA23, BSA45, BSA67, BSAC）；

(4) 临时寄存器：包括临时寄存器（T0~T3）。

## 第2章 TMS320C55x的硬件结构



地址流程单元结构图



### 2.1.5 数据计算单元 (D)

数据计算单元由移位器、算术逻辑电路、乘法累加器和寄存器组构成。D单元包含了CPU的主要运算部件。

单元移位器能够接收来自I单元的立即数，能够与存储器、I/O空间、A单元寄存器、D单元寄存器和P单元寄存器进行双向通信，此外，还可以向D单元的ALU和A单元的ALU提供移位后的数据。移位器可完成以下操作：

(1) 对40位的累加器可完成向左最多31位和向右最多32位的移位操作，移位数可从临时寄存器（T0~T3）读取或由指令中的常数提供；

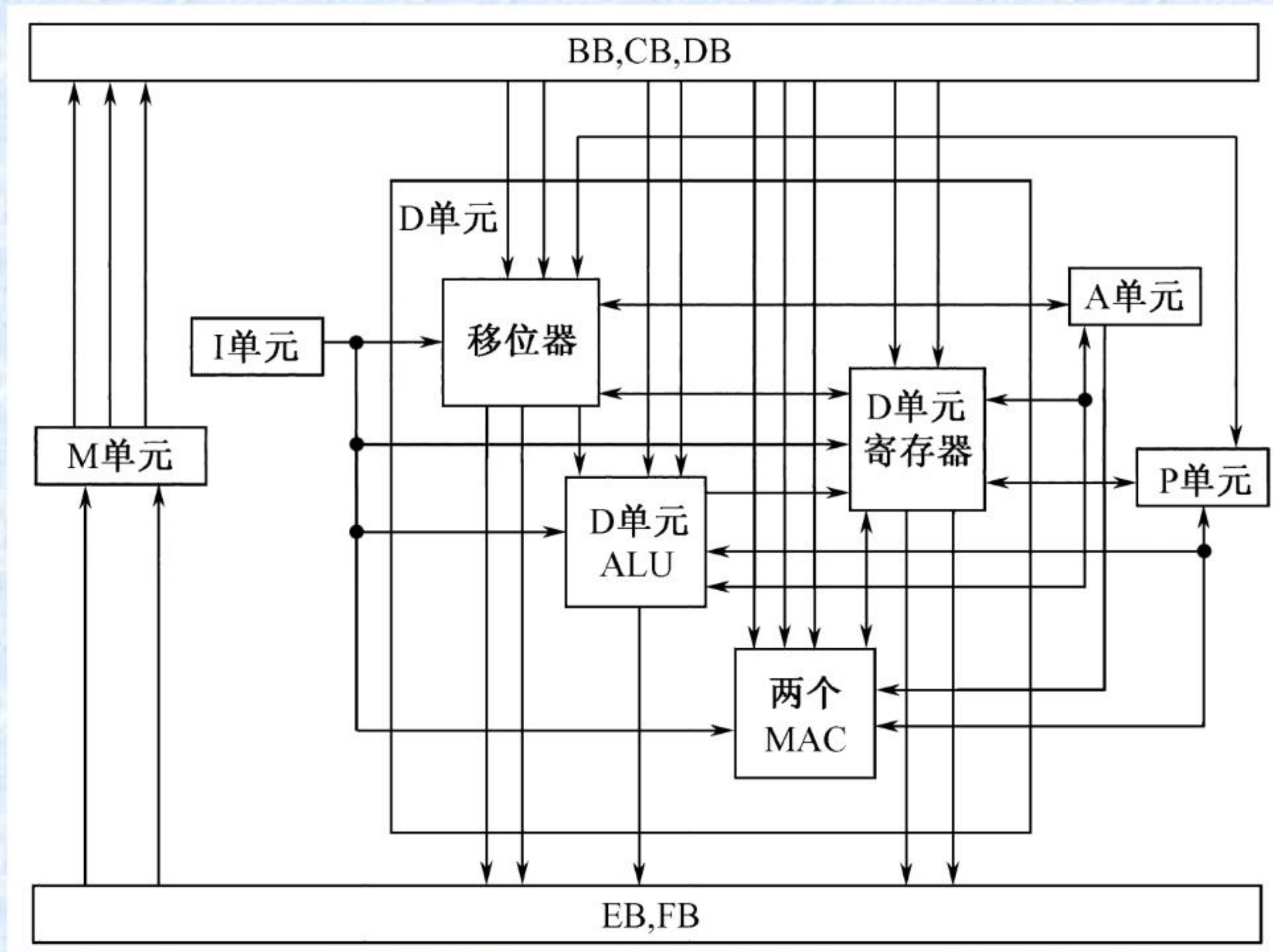
(2) 对于16位寄存器、存储器或I/O空间数据可完成左移31位或右移32位的移位操作；

(3) 对于16位立即数可完成向左最多15位的移位操作。

**D单元的40位算术逻辑电路可完成以下操作：**

- (1) 完成加、减、比较、布尔逻辑运算和绝对值运算等操作；**
- (2) 能够在执行一个双16位算术指令时同时完成两个算术操作；**
- (3) 能够对D单元的寄存器进行设置、清除等位操作。**

## 第2章 TMS320C55x的硬件结构



数据计数单元结构图

### 2.1.6 指令流水线

C55x CPU采用指令流水线工作方式，C55x的指令流水线包括两个阶段：

第一阶段是取流水线，即从内存中取出32位的指令包，放入指令缓冲队（IBQ）中，然后为流水线的第二阶段提供48位的指令包。

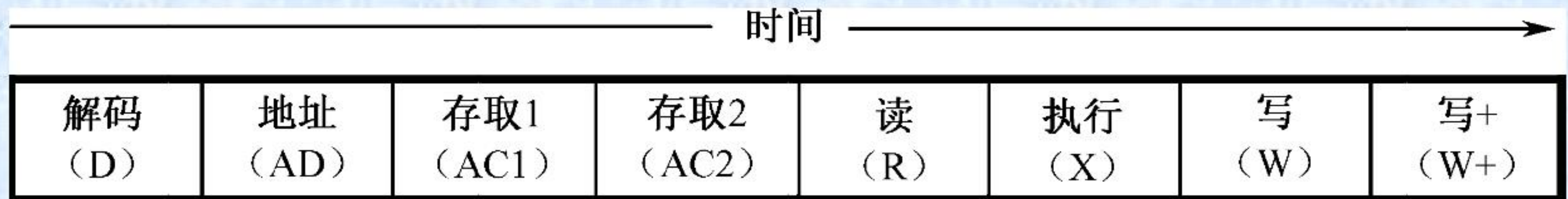


#### 流水线的第一阶段（取流水线）

其中PF1表示向存储器提供的程序地址，PF2表示等待存储器的响应，F表示从存储器取一个指令包并放入指令缓冲队列中，PD表示对指令缓冲队列中的指令预解码（确定指令的起始和结束位置；确定并行指令）。

## 第2章 TMS320C55x的硬件结构

第二阶段是指执行流水线，这部分的功能是对指令进行解码，完成数据的存取和计算。



流水线的第二阶段（执行流水线）

## 第2章 TMS320C55x的硬件结构

### 流水线第二分段各阶段的执行情况

流水线节拍	内 容
<b>D</b>	从指令缓冲队列中读6个字节的指令 对一个指令对或一个单指令进行解码 给对应的CPU功能单元分配指令 读取ST <sub>x</sub> 中55种与数据地址产生相关的位 ST1_55(CPL) ST2_55(ARnLC) ST2_55(ARMS) ST2_55(CDPLC)
<b>AD</b>	读/修改与数据地址产生有关的寄存器 例如： - *AR <sub>x</sub> +(T0)中的AR <sub>x</sub> 和T0 - BK03（如果AR2LC = 1） - SP（pushes和 pops过程中） - SSP, 在32位栈模式中 与对SP的操作一致 在A单元的ALU中完成操作，例如： - 使用AADD指令进行算术运算 - 用SWAP指令交换A单元中的寄存器 - 向A单元的寄存器写入常量（BK <sub>xx</sub> , BS <sub>Axx</sub> , BRC <sub>x</sub> , CSR, 等） 在条件分支指令中AR <sub>x</sub> 如果不等于0, AR <sub>x</sub> -1

## 第2章 TMS320C55x的硬件结构

### 流水线第二分段各阶段的执行情况（续）

流水线节拍	内 容
AC1	在存储器读操作中，在相应的CPU地址总线上传送地址
AC2	允许存储器对请求的响应是一个周期
R	从存储器和通过映射方式寻址的寄存器中读数据 在R节拍执行D单元的预取A单元寄存器指令时，读A单元的寄存器 在R节拍判断条件指令的条件
X	读/修改不通过映射方式寻址的寄存器 读/修改寄存器中的单个位 设置条件 如果指令不是向存储器中写，就判断XCCPART的条件 判断RPTCC指令的条件
W	向存储器映射方式寻址的寄存器或I/O空间写数据 向存储器写数据，从CPU来看，写操作在该节拍完成
W+	向存储器写数据，从存储器来看，写操作在该节拍完成

下面通过举例来说明流水线的工作方式。

**AMOV #k23, XAR<sub>x</sub>** :在AD节拍用一个立即数对XAR<sub>x</sub>初始化;

**MOV #k, AR<sub>x</sub>** :AR<sub>x</sub>不是通过存储器映射方式寻址的, 在X节拍用一个立即数初始化AR<sub>x</sub>;

**MOV #k, mmap(AR<sub>x</sub>)**: AR<sub>x</sub>是通过存储器映射方式寻址的, 在W节拍用一个立即数初始化AR<sub>x</sub>;

**AADD #k, AR<sub>x</sub>**: 对于这个特殊指令, 在AD节拍用一个立即数对AR<sub>x</sub>初始化;

**MOV #k, \*AR<sub>x</sub>+**: 在W+节拍对存储器进行写操作;

**MOV \*AR<sub>x</sub>+, AC0**: 在AD节拍对AR<sub>x</sub>进行读和更新操作, 在X节拍载入AC0;

**ADD #k, AR<sub>x</sub>**: 在X节拍的开始时刻读AR<sub>x</sub>, 在X节拍的结束时刻修改AR<sub>x</sub>;

**ADD AC<sub>y</sub>, AC<sub>x</sub>**: 在X节拍读/写AC<sub>x</sub>和AC<sub>y</sub>;



## 第2章 TMS320C55x的硬件结构

**MOV mmap(ARx), ACx:** ARx是通过存储器映射方式寻址的, 在R节拍读取ARx, 在X节拍修改ACx;

**MOV ARx, ACx:** ARx不是通过存储器映射方式寻址的, 在X节拍读取ARx, 在X节拍修改ACx;

**BSET CPL:** 在X节拍设置CPL位;

**PUSH, POP, RET**

**或AADD #K8, SP:** 在AD节拍读取和修改SP, 如果选择32位栈模式, SSP会发生变化;

**XCCPART overflow(ACx):** 在X节拍判断条件, 但是不管条件是否满足, AR1都会加1;

**|| MOV \*AR1+, AC1**

**XCCPART overflow(ACx):** 在R节拍判断条件, 满足条件向存储器完成写操作, 但是不**|| MOV AC1, \*AR1+** 管条件是否满足, AR1都会加1

**XCC overflow(ACx):** 在AD节拍判断条件, 只有满足条件时, AR1加1

**|| MOV \*AR1+, AC1**

### 2.2 TMS320VC5509A的主要特性

#### 2.2.1 VC5509A的主要特性

VC5509A是C55x系列一款典型的处理器，在VC5509A中集成了一个C55x内核，128KB×16位片上RAM存储器，并具有最大8MB×16位的外部寻址空间，片上还集成了USB总线、McBSP和I2C等外部接口。

### 1. CPU

- 两个乘法累加单元（MAC）：每个乘法累加单元可在一个周期内处理17位×17位的乘法运算。
- 40位的算术逻辑单元（ALU）和一个16位的算术逻辑单元：算术逻辑单元具有并行处理能力，采用并行处理可以降低系统处理时钟从而降低功耗。这些资源由CPU中的地址单元（AU）和数据单元（DU）进行管理。
- 采用先进的多总线结构：通过三条内部数据/操作数来读总线和两条内部数据/操作数来写总线，从而完成对指令和数据的访问。

### 2. 存储器

- 有128KB×16位的片上RAM，其中包括64KB的双存取RAM（DARAM）和192KB的单存取RAM（SARAM）。
- 8MB×16位的最大可访问外部寻址空间（同步DRAM）。
- 外部存储器接口（EMIF）与通用输入/输出（GPIO）共用引脚，当不使用外部存储器接口时，这些引脚可以当作通用输入/输出接口来使用，EMIF接口可以实现与异步静态RAM（SRAM）、异步（EPROM）和同步DRAM（SDRAM）的无缝连接。

### 3. 片上外设

- 两个20位的定时器。
- 一个看门狗定时器。
- 6通道直接存储器存取控制器（DMA），DMA控制器在不需要CPU干预的情况下可以提供6路独立的通道用于数据传输，并且可达每周期两个16位数据的吞吐量。
- 外部存储器接口（EMIF），它提供与异步存储器如EPROM、SRAM及高密度存储器如同步DRAM的无缝连接。
- 三个串口支持最多三个多通道缓冲串口（McBSP）或最多两个多媒体/安全数字卡接口。三个全双工多通道缓冲串口（McBSP）提供了与各种工业级串行设备的无缝接口，其多通道通信最多可以实现128个独立通道。
- 增强型主机接口（EHPI）是一个16位的并行接口，主机能够提供HPI接口访问5509A上的32KB片上存储器。
- 可编程锁相环（DPLL）时钟发生器。
- USB全速（12Mbps）从端口。
- I2C主从接口。
- 一个实时时钟。

### 2.2.2 VC5509A的引脚功能

**TMS320VC5509APGE**采用塑料四方扁平封装形式（**LQFP**），共有**144**个引脚，按其功能可分为并行总线引脚、中断和复位引脚、位输入/输出信号引脚、**I2C**引脚、多通道缓冲串口信号引脚、**USB**引脚、**A/D**引脚、测试引脚和电源引脚等几部分。

### 1. 并行总线引脚

并行总线A13~A0直接与外部引脚相连，这14个引脚可以完成以下三个功能：HPI地址总线（HPI.HA[13:0]）、EMIF地址总线（EMIF.A[13:0]）或通用输入/输出（GPIO.A[13:0]）。这三个功能可以通过外部总线选择寄存器（EBSR）中的并行端口模式字段来设置，这些引脚的初始状态由GPIO0引脚决定。

并行双向数据总线D31~D0可以完成两个功能：EMIF数据总线（EMIF.D[15:0]）或HPI数据总线（HPI.HD[15:0]）。同样，这两个功能可以通过外部总线选择寄存器中的并行端口模式位域来设置，这些引脚的初始状态由GPIO0引脚决定。

### 2. 中断引脚和复位引脚

中断引脚INT[4:0]作为低电平有效的外部中断输入引脚，由中断使能寄存器（IER）和中断模式位来屏蔽和区分优先次序。

引脚低电平有效，当该信号有效时，DSP将终止任务的执行并使程序指针指向FF 8000h，当变为高电平时，DSP从程序存储器FF 8000h的位置开始执行。



### 3. 位输入/输出信号引脚

**GPIO[7:6,4:0]**共7个输入/输出线，可以单独配置成输入或输出引脚，作为输出时又可以单独被设置或清除。当DSP复位时，这7个引脚首先会被配置为输入线；复位后，会采集**GPIO[3:0]**的电平来确定DSP的引导模式。

**XF**引脚作为外部标志，由**BSET XF**指令设置为高电平，有三种方式来设置**XF**为低电平：通过**BCLR XF**指令来设置、在多处理器协同工作时给其他处理器发信号而载入**ST1.XF**，或当**XF**作为通用输出引脚时。

### 4. 时钟信号引脚

**CLKOUT**是DSP时钟输出信号引脚。其周期为CPU的机器周期。当为低电平时，该引脚呈高阻状态。

**X2/CLKIN**是晶振连接到内部振荡器的输入引脚，若使用外部时钟时，该引脚作为外部时钟的输入引脚。

**X1**是内部振荡器连接到外部晶振的输出引脚，如果不使用内部时钟，该引脚悬空。

**TIN/TOUT0**是定时器0输入/输出引脚。作为输出引脚时，当片内定时器减到0时，该引脚发出一个脉冲或变化的状态。作为输入引脚时，该引脚为内部定时器模块系统时钟源。复位后，该引脚是输入状态。

**RTCINX1**是实时时钟振荡器的输入引脚。

**RTCINX2**是实时时钟振荡器的输出引脚。

### 5. I2C引脚

**SDA**是I2C（双向）数据线。复位后，该引脚呈高阻状态。

**SCL**是I2C（双向）时钟引脚。复位后，该引脚呈高阻状态。

### 6. McBSP信号引脚

**TMS320C55x DSP**提供了高速多通道缓冲串口（**Multi-channel Buffered Serial Ports, McBSP**），**DSP**可以通过**McBSP**与其他**DSP**、编码器 etc 器件相连。

### 7. USB引脚

在TMS320C55xDSP产品中，TMS320VC5507、TMS320VC5509和TMS320VC5509A提供了USB模块。

**DP**引脚是差分（正）接收/发送引脚。

**DN**引脚是差分（负）接收/发送引脚。

**PU**引脚是上拉引脚，用于上拉检测电阻。

### 8. A/D引脚

VC5509A提供了一个10位的A/D转换器。AIN0和AIN1分别是模拟输入通道0和模拟输入通道1。

### 9. 测试引脚

VC5509A与其他C5000产品一样具有符合IEEE1149.1标准的测试/仿真接口，其引脚有：

- TCK引脚是IEEE标准1149.1测试时钟输入引脚。通常是一个占空比为50%的方波信号。在TCK的上升沿，将输入信号TMS和TDI在测试访问端口（Test Access Port, TAP）的变化记录到TAP控制器、指令寄存器或选定的测试数据寄存器中。TAP输出信号TDO在TCK的下降沿发生变化。
- TDI引脚是IEEE标准1149.1测试数据输入引脚，在TCK的上升沿将TDI记录到选定的指令或数据寄存器中。
- TDO引脚是IEEE标准1149.1测试数据输出引脚，在TCK的下降沿将选定的指令或数据寄存器的内容从TDO输出。
- TMS引脚是IEEE标准1149.1测试方式选择引脚，在TCK的上升沿将串行控制输入信号记录到TAP控制器中。

- 引脚是IEEE标准1149.1测试复位引脚，当该引脚为高电平时，DSP芯片由IEEE标准1149.1扫描系统控制工作；若该引脚悬空或为低电平，则芯片正常工作。
- EMU0引脚是仿真器中断0引脚。当为低电平时，为了保证的有效性，EMU0必须为高电平。当为高电平时，EMU0是仿真系统的中断信号，并由IEEE标准1149.1扫描系统来定义是输入还是输出。
- EMU1/引脚是仿真器中断1引脚/关断所有输出引脚。当为高电平时，EMU1/是仿真系统的中断信号，并由IEEE标准1149.1扫描系统来定义是输入还是输出。当为低电平时，EMU1/被设置为的有效性，将所有的输出设置为高阻状态。

### 10. 电源引脚

VC5509A有内核电源和外设电源两种。

CVDD是数字电源，对于时钟为108MHz、144MHz和200MHz的DSP对应的CVDD分别为+1.2V、+1.35V和+1.6V，为CPU内核提供专用电源。

DVDD是数字电源，+3.3V，为I/O引脚提供专用电源。

USBVDD是数字电源，+3.3V，为USB模块的I/O引脚提供专用电源。

RDVDD是数字电源，对于时钟为108MHz、144MHz和200MHz的DSP对应的RDVDD分别为+1.2V、+1.35V和+1.6V，为RTC模块的I/O引脚提供专用电源。

## 第2章 TMS320C55x的硬件结构

**RCVDD** 是数字电源，对于时钟为108MHz、144MHz和200MHz的DSP对应的RCVDD分别为+1.2V、+1.35V和+1.6V，为RTC模块提供专用电源。

**AVDD** 是模拟电源，+3.3V，为10位的A/D提供专用电源。

**ADVDD** ， +3.3V，为10位A/D数字部分提供专用电源。

**USBPLLVDD** 是数字电源，对于时钟为108MHz、144MHz和200MHz的DSP对应的USBPLLVDD分别为+1.2V、+1.35V和+1.6V，为USB的PLL提供专用电源。

**VSS**是数字地，为I/O和内核引脚接地。

**AVSS**是模拟地，为10位A/D接地。

**ADVSS**为10位A/D的数字部分接地。

**USBPLLVSS**是数字地，为USB的PLL接地。



### 2.3 TMS320C55x 存储空间结构

**C55x DSP的存储空间包括统一的数据/程序空间和I/O空间。数据空间用于访问存储器和内存映射寄存器，程序空间用于CPU从存储器中读取指令，而I/O空间用于CPU与外设之间的双向通信。**

### 2.3.1 存储器映射

**C55x的寻址空间为16MB，当CPU从程序空间读取程序代码时，使用24位地址，当访问数据空间时，使用23位的地址。但是在访问数据空间时，将23位地址左移一位，并将地址总线上的最低有效位（LSB）置0，使得在对数据空间或程序空间寻址时，地址总线都传送24位地址。**

## 第2章 TMS320C55x的硬件结构

	数据空间	程序空间
第0主数据页	MMR 00 0000~00 005F	00 0000~00 00BF
	00 0060~00 FFFF	00 00C0~01 FFFF
第1主数据页	01 0000~01FFFF	02 0000~03 FFFF
第2主数据页	02 0000~02FFFF	04 0000~05 FFFF
	⋮	⋮
第127主数据页	7F 0000~7F FFFF	FE 0000~FF FFFF

### 存储器映射

## 第2章 TMS320C55x的硬件结构

数据空间被分成128个主数据页（第0页到第127页），每个主数据页的大小为64K字，指令通过7位的主数据页值和16位的偏移值共同来确定数据空间的任何一个地址。

在第0主数据页中，前96个地址（00 0000h~00 005Fh）为存储映射寄存器（MMR）保留，相对应在程序空间有192个地址（00 0000h~00 00BFh），这段存储区为系统保留区，用户不能使用该区。

### 2.3.2 程序空间

当CPU读取指令时，程序空间才被访问。CPU采用字节寻址来读取变长的指令，指令的读取要和32位的偶地址对齐（地址的低两位为零）。

#### 1. 字节寻址（24位）

当CPU从程序空间读取指令时，采用字节寻址，即按字节分配地址，且地址为24位。一个行宽为32位存储器的地址分配由下图说明，每个字节分配一个地址，例如字节0的地址是00 0100h，字节2的地址是00 0102h。



### 2. 程序空间的指令结构

DSP支持8位、16位、24位、32位和48位长度的指令。表2-5和图2-9说明了指令在程序空间如何存放。在32位宽的存储器中存放了5条指令，每一条指令的地址是指操作码最高有效字节的地址，阴影部分表示没有代码。

表2-5 指令长度及地址分配

指 令	长度（位）	地 址
A	24	00 0101h
B	16	00 0104h
C	32	00 0106h
D	8	00 010Ah
E	24	00 010Bh

## 第2章 TMS320C55x的硬件结构

图2-9 存储器中的指令

字节地址	字节0	字节1	字节2	字节3
00 0100h~00 0103h		A (23~16)	A (15~8)	A (7~0)
00 0104h~00 0107h	B (15~8)	B (7~0)	C (31~24)	C (23~16)
00 0108h~00 010Bh	C (15~8)	C (7~0)	D (7~0)	E (23~16)
00 010Ch~00 010Fh	E (15~8)	E (7~0)		

### 3. 程序空间的边界对齐

在程序空间存放指令时不需要边界对齐，当读取指令时要和32位的偶地址对齐。也就是说，在读取一条指令时，CPU要从最低两位是0的地址读取32位的代码，这样的地址其最低位应是0h，4h，8h和Ch。

不过，也会遇到写入到程序计数器PC中的地址值和程序空间的读取地址不一致的情况，例如，执行一个子程序B：

#### CALL B

假设子程序的第一条指令C的地址是00 0106h，PC的值是00 0106h，但是读程序地址总线（PAB）上的值是32位边界地址00 0104h，CPU在00 0104h地址开始读取4字节的代码，而第一个被执行的指令是C。



### 2.3.2 数据空间

C55x DSP采用字寻址来读/写数据空间的8位、16位或32位数据。

#### 1. 字寻址（23位）

当CPU访问数据空间时，采用字寻址，即为每个16位的字分配一个23位宽的地址，下面说明了一行32位宽的存储器的地址分配，字0的地址为00 0100h，字1的地址为00 0101h。



## 第2章 TMS320C55x的硬件结构

由于地址总线是24位宽，所以，当CPU读/写数据空间时，23位的地址左移一位，最低位补0。例如，一条指令在23位地址00 0102h上读一个字，读数据地址总线上传送的值是00 0204h，如下所示。

字地址：000 0000 0000 0001 0000 0010

读数据地址总线：0000 0000 0000 0010 0000 0100

### 2. 数据类型

**C55x DSP指令处理的数据类型有8位、16位和32位。**

数据空间是采用字寻址，但C55x有专门的指令可以选择字的高字节或低字节，进行8位数据的处理。字节装载指令将从数据空间读取的字节进行0扩展或符号扩展，然后装入寄存器中；字节存储指令可将寄存器中的低8位数据存储到数据空间指定的地方。

### 字节装载和字节存储指令

指 令	存取的字节	操 作
<b>MOV high_byte(Smem) , dst</b> <b>MOV low_byte(Smem) , dst</b> <b>MOV high_byte(Smem)&lt;&lt;#SHIFTW , ACx</b> <b>MOV low_byte(Smem) &lt;&lt;#SHIFTW , ACx</b>	<b>Smem(15~8)</b> <b>Smem(7~0)</b> <b>Smem(15~8)</b> <b>Smem(7~0)</b>	<b>字节装载</b>
<b>MOV src , high_byte(Smem)</b> <b>MOV src , low_byte(Smem)</b>	<b>Smem(15~8)</b> <b>Smem(7~0)</b>	<b>字节存储</b>

## 第2章 TMS320C55x的硬件结构

当CPU存取长字时，存取地址是指32位数据的高16位（MSW）的地址，而低16位（LSW）的地址取决于MSW的地址。具体说明如下所示。

如果MSW的地址是偶地址，则LSW的地址加1。



如果MSW的地址是奇地址，则LSW的地址减1。



对于已确定地址的MSW（LSW），将其地址的最低有效位取反，可得到LSW（MSW）的地址。

### 3. 数据空间的数据结构

下面通过实例来说明数据在数据空间是如何存储的。  
有7种变长的数据存储于32位宽的存储器中。

根据表2-7和图2-10可以看出，为了存取一个长字必须参考它的MSW，C的存取地址是00 0102h，D的存取地址是00 0105h；字地址也可以存取字节，如在地址00 0107h上，同时存放了数据F（高字节）和数据G（低字节）。利用表2-6中的专用指令可以进行字节的存取。

## 第2章 TMS320C55x的硬件结构

表2-7 数据长度及地址分配

数 据	数据类型	地 址
A	字节	00 0100h (低字节)
B	字	00 0101h
C	长字	00 0102h
D	长字	00 0105h
E	字	00 0106h
F	字节	00 0107h (高字节)
G	字节	00 0107h (低字节)

## 第2章 TMS320C55x的硬件结构

图2-10 存储器中的指令

字地址	字 0	字 1	
00 0100h~00 0101h	A	B	
00 0102h~00 0103h	C (31~16)	C (15~0)	
00 0104h~00 0105h	D (15~0)	D (31~16)	
00 0106h~00 0107h	E	F	G



## 第2章 TMS320C55x的硬件结构

### 2.3.4 I/O空间

**C55x DSP的I/O空间与数据/程序空间是分开的，采用16位宽的字寻址，即为每个字分配一个16位地址，其寻址范围为64K字，如下所示：**

地址	I/O空间
0000h~FFFFh	64K字

当CPU访问I/O空间时，用DAB读数据，用EAB写数据。由于DAB和EAB都是24位的，所以在16位地址前补0构成24位地址。例如，一个指令在地址0102h处读取一个字，DAB上传送的地址是00 0102h。