CPU

（1）8086CPU两部分：总线接口单元BIU，负责CPU与存储器I/O设备之间的数据传输，读指令，读操作数，写操作数。执行单元EU不与系统外部直接相连，负责执行指令，执行的指令从BIU的指令缓冲器取得，执行指令若需要从存储器或I/O端口读写数据，由EU向BIU发出请求，再由BIU对存储器或I/O端口进行访问。

BIU读指令时，不影响EU执行指令，EU可以连续不断地直接从指令队列中获取指令，从而减少了CPU为取指令而等待的时间。所以一个指令周期=取指时间+执指时间=取指时间+访问内存时间。

（2）CPU读/写内存/IO外设总线周期：

T1：M/IO引脚，决定CPU读写内存还是外设；地址寄存器内容压入地址总线，激活将要读写存储单元。

地址寄存器

内存

T2：RD（读）引脚信号送到M/IO的所有存储单元，但只有T1激活的地址单元才会把数据压入数据总线。或WR（写），CPU将寄存器数据压入数据总线。

T3：检查READY引脚，高电平说明M/IO已经准备就绪，CPU进入T4；低电平说明M/IO还未准备就绪（存储单元的数据还未压入数据总线或者存储单元还未准备好接收CPU压入到数据总线的数据），插入TW（空闲时钟周期）。READY引脚可以避免高速CPU使用低速存储器造成数据丢失。

T4：数据总线的数据写入存储单元或者CPU接收数据总线的数据，让出总线。

综上，一个读/写总线周期，至少4个时钟周期。

（3）每一条指令可完成一个独立的算术运算或逻辑运算。所有指令的集合叫指令系统。指令系统的每个指令对应一个硬件直接实现，指令系统的设计决定计算机硬件结构。乘除运算指令，浮点运算指令可直接用硬件来实现，也可用最基本的指令编写程序来实现，采用硬件指令额目的是提高程序的执行速度，便于用户编写程序。但指令系统的指令太多，会采用大量使用频率很低的指令而使硬件资源浪费。

指令格式：操作码+地址码。地址码是操作数的地址，操作数来自内存或寄存器，RR，RS，SS。RR型速度最快。

机器字长指计算机能直接处理的二进制数据的位数，等于主存位数。指令字长度指操作码长，长了，指令种类多，太长取一条指令需要访问主存多次，降低了CPU的效率。

001 0x9A76CF40 AC (RS型) 操作码替换成助记符即汇编 ADD 0x9A76CF40 AC 所以每条汇编指令对应一条机器语言。汇编语言依赖于计算机的硬件结构和指令系统。不同机器有不同的指令，所以用汇编语言编写的程序不能在其他类型的机器上执行。高级语言与指令系统无关，所以高级语言无法编写直接访问硬件资源的软件！（如修改某个寄存器或存储单元）。所以高级语言存在调用汇编语言的接口，需要直接操作硬件时，用汇编实现，再生成目标文件，通过链接器生成可执行文件。

存储器可存数据或指令。寻址方式：指令寻址和操作数寻址。内存的地址通过编号识别，寄存器通过名字识别。

常用指令：

1. LAD 存数 CPU->存储器
2. STO 取数 存储器->CPU
3. EXC 交换 源和目标交换值
4. JMP 无条件转 将指定指令地址写入PC计数器
5. EOR 异或
6. INC 操作数+1
7. SUB 两个操作数减法

CPU：程序装入主存后，取指令和执行指令。

|  |  |  |
| --- | --- | --- |
| 指令地址 | 指令的操作码 | 指令的地址码 |
| 101 | LAD | R1,6 |
| 102 | ADD | R1，R2 |

程序计数器PC = 101

PC ——> 指令地址总线

从101读出LAD指令装入指令寄存器IR

PC++

对IR的操作码译码，解释出LAD指令。取指过程结束

一个CPU周期结束。

下面控制器OC执行LAD。

启动读总线周期，取出地址码6写入数据地址寄存器。消耗一个CPU周期

启动读总线周期，读出6存储单元数据，把该数据写入R1。消耗一个CPU周期

数据总线是16位时，一个总线周期可以读写一个字（2字节）。

寄存器：暂存指令和数据（从内存取出的）

控制器：控制从内存读写入寄存器，并根据指令运算的结果控制计算机。对操作码进行译码，产生相应的操作控制信号，以便启动规定的动作。指挥并控制CPU，数据cache和输入/输出设备之间数据流动的方向。

运算器：执行所有的算术运算，执行所有的逻辑运算，产生一个运算结果或判决。

时钟：指令周期指取出一条指令并执行这条指令的时间。

累加寄存器：存储执行运算的数据和运算后的数据。

程序计数器：执行指令时，CPU自动更改PC内容，使其总是存储指向下一条指令的地址。

指令寄存器：保存当前正在执行的指令。指令cache从PC计数器读出指令，指令寄存器从指令cache取出指令，指令寄存器的输出就是指令译码器的输入，将二进制的操作码译码，向控制器发出具体操作的特定信号。

基址寄存器：存储数据内存的起始地址。

变址寄存器：存储基址寄存器的相对地址。

通用寄存器：存储任意数据。

标志寄存器：进位标志，溢出标志，运算结果为0标志，中断和系统工作状态等标志。

顺序执行：程序计数器自动加1.

分支和循环：跳转指令直接为程序计数器赋值（JMP（无条件转），JMPX（条件转））

函数调用：JMPC（把当前信息保存到已知的位置，转移到指定地址） RET（将已知位置内容代替PC和寄存器）

用于运算的数值存储在累加寄存器中，表示地址的数值存储在基址寄存器和变址寄存器中。

CPU主频，晶振，振荡频率的倒数称为振荡周期，也叫时钟周期，是MCU中最基本，最小的时间单位。在一个时钟周期CPU仅完成一个最基本的动作，时钟频率越大，MCU工作速度越快。MCS-51规定一个机器周期为6个状态，也就是12个时钟周期。执行一条指令所需要的时间叫指令周期，它是机器周期的整数倍，复杂程度不同的指令消耗的机器周期不同，最短是1个机器周期，最长是4个机器周期。所以，单片机的指令周期由振荡频率决定。

STO指令：把数据写入存储器的指定地址，RS型，源地址和目的地址一个是寄存器一个是存储器。

STO R3 0x9A76CF40，消耗3个CPU周期。

进制

N进制：（1）逢N进1 （2）N个符号（0---N-1） （3） 权值 … N-2 N-1 小数点 N0 N1 N2 …

（10．01）B = 1 × 2 + 0 × 1 + 0 × 2-1 + 1× 2

N进制的数 = 各权位的值×权位 的和

N进制的数×N 扩大N倍，小数点向左移动1位。除N同理。类比十进制即可得出结论。

进制转换

1. 任意进制转换为十进制

权位展开。（516.3）7 = 3×7-1+6+1×7+5×72

1. 十进制转换成二进制

整数部分除2取商继续除2直到商为0，逆向取余。小数部分乘2取整，直到小数部分为0。

1. 二进制与十六进制之间的转换

凑4位，整数部分高位补0，小数部分低位补0。4位合1或1展4

有符号数

最高位是符号位，0正1负，其他位是数值位。

机器数是指机器表示真值的方式：原码，反码，补码。

|  |  |  |
| --- | --- | --- |
| 真值 | +5 ——> +101 | -5 ——> -101 |
| 原码 | 0101 扩展8字长，补0 0000 0101 | 1101扩展8字长，补0 1000 0101 |
| 反码 | 和原码相同  扩展时，补0 | 符号位不变，数值位取反  1010（扩展时，补1）1111 1010 |
| 补码 | 和原码相同  扩展时，补0 | 符号位不变，数值位取反+1  1011(扩展时，补1) 1111 1010 |

已知补码，求真值：最高位是0，补码就是原码，直接求真值即可。最高位是1，求其补码得到原码，去掉最高位，计算出一个正数，取其相反数就是真值。

补码的补码是原码，任意两数的补码之和等于该两数之和的补码。

无符号数相加，最高位有进位，出错。

有符号数相加，结果超出（-2n-1—2n-1-1）,即字长不足以表示，则溢出出错。

判断溢出的方法：数据位的最高位和符号位都产生进位或者都没进位，则计算结果正确，否则溢出。

浮点数

任意数可以用N进制表示：M×Ne，M和e是N进制整数，e增加1或者减少1使小数点右移或左移一位。类比十进制，任意数都可以用科学计数法表示：M×10n，如8.7632×1063；同样，任意数也可以表示成M×2n。

double格式

|  |  |  |
| --- | --- | --- |
| 符号位（表示正负） | 尾数（决定精度）52bit | 阶码（决定表数范围）11bit |

double的小数位有52位，对应十进制最大值为4 503 599 627 370 496，这个数有16位，所以计算精度只能百分百保证十进制的15位运算。可以用十进制理解，例十进制3.1415926×1018，约定M只有5位，我们只能表示成3.1415×1018。

计算机无法精确表示0.1的；0.1= 0.0001100110011001100110011001100110011001100110011001101；

内存

8086CPU有16根数据引脚，可以一次存取8位或16位数据；有20根地址引脚，可以直接寻址1M（220）个存储单元和64k个I/O端口。地址总线宽度决定可用最大内存，数据总线宽度决定传输数据的速度。

虚拟内存是把磁盘的一部分当成假想的内存来使用。方法有两种：分页式和分段式。

windows采用分页式，在不考虑程序构造情况下，把运行的程序按照一定大小的页（page）进行分割，并以页为单位在内存和磁盘间进行置换。windows一页4k，把页从磁盘调入虚拟内存page in，从内存调入磁盘叫page out。为了实现虚拟内存，windows在磁盘上提供了虚拟内存用的页文件（page file）。该文件由windows自动做成和管理，文件大小就是虚拟内存的大小，一般与内存相等或其2倍。

分段式是把要运行的程序分割成以处理集合和数据集合等为单位的段落，然后再以分割后的段落为单位在内存和磁盘之间进行数据置换。