|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **指令类型** | | | **助记符**  **(带\*为特权指令)** | **对标志寄存器的影响** | | | | | | | | | **备注** | **说明** | **举例** |
| **ZF** | **CF** | **PF** | **SF** | **OF** | **AF** | **DF** | **IF** | **TF** |
| 数据传送类 | 数据传送 | | MOV | 不影响标志位 | | | | | | | | |  | Move | MOV r/m32,imm32 |
| MOV**\*** |  | Move to/from Control Registers | MOV CR0,r32 |
| MOV**\*** |  | Move to/from Debug Registers | MOV r32, DR0-DR7 |
| 符号位扩展传送 | | MOVZX |  | Move with Zero-Extend | MOVZX r32,r/m16 |
| MOVSX |  | Move with Sign-Extension | MOVSX r32,r/m16 |
| 数据交换 | | XCHG |  | Exchange Register/Memory with Register | XCHG r32,r/m32 |
| CMPXCHG | √ | √ | √ | √ |  | √ |  |  |  |  | Compare and Exchange | CMPXCHG r/m32,r32 |
| CMPXCHG8B | √ |  |  |  |  |  |  |  |  |  | Compare and Exchange 8 Bytes | CMPXCHG8B m64 |
| 堆栈传送 | | PUSH | 不影响标志位 | | | | | | | | |  | Push Word or Doubleword Onto the Stack | Push r/m32 |
| POP |  | Pop a Value from the Stack | POP r.m32 |
| PUSHA |  | Push All General-Purpose Registers | PUSHA |
| POPA |  | Pop All General-Purpose Registers | POPA |
| PUSHAD |  | Push All General-Purpose Registers | PUSHAD |
| POPAD |  | Pop All General-Purpose Registers | POPAD |
| 地址传送 | | LEA |  | Load Effective Address | LEA r32,m |
| LDS |  | Load Far Pointer | LDS r32,m16:32 |
| LES |  | Load Far Pointer | LES r32,m16:32 |
| LFS |  | Load Far Pointer | LFS r32,m16:32 |
| LGS |  | Load Far Pointer | LGS r32,m16:32 |
| LSS |  | Load Far Pointer | LSS r32,m16:32 |
| 标志寄存器传送 | | LAHF | 不影响标志位 | | | | | | | | |  | Load Status Flags into AH Register | LAHF |
| SAHF | **标志寄存器低八位的内容由AH的值决定** | | | | | | | | |  | Store AH into Flags | SAHF |
| PUSHF | 不影响标志位 | | | | | | | | |  | Push EFLAGS Register onto the Stack | PUSHF |
| POPF | **标志寄存器的内容由装入的具体值决定** | | | | | | | | |  | Pop Stack into EFLAGS Register | POPF |
| PUSHFD | 不影响标志位 | | | | | | | | |  | Push EFLAGS Register onto the Stack | PUSHFD |
| POPFD | **标志寄存器的内容由装入的具体值决定** | | | | | | | | |  | Pop Stack into EFLAGS Register | POPFD |
| 查表 | | XLAT | 不影响标志位 | | | | | | | | |  | Table Look-up Translation | XLAT m8 |
| XLATB |  | Table Look-up Translation | XLATB |
| 输入输出 | | IN\* | 不影响标志位 | | | | | | | | |  | Input from Port | IN EAX,imm8 |
| OUT\* |  | Output to Port | OUT imm8, EAX |
|  | 输入 | INS\* | 不影响标志位 | | | | | | | | |  | Input from Port to String | INS m32, DX |
| INSB\* |  | Input from Port to String | INSB |
| INSW\* |  | Input from Port to String | INSW |
| INSD\* |  | Input from Port to String | INSD |
| 输出 | OUTS\* |  | Output String to Port | OUTS DX, m32 |
| OUTSB\* |  | Output String to Port | OUTSB |
| OUTSW\* |  | Output String to Port | OUTSW |
| OUTSD\* |  | Output String to Port | OUTSD |
| 串  操  作 | 传送 | MOVS |  | Move Data from String to String | MOVS m32, m32 |
| MOVSB |  | Move Data from String to String | MOVSB |
| MOVSW |  | Move Data from String to String | MOVSW |
| MOVSD |  | Move Data from String to String | MOVSW |
| 比较 | CMPS | **由计算结果确定标志位的值** | | | | | | | | | 不会影响DF,IF,TF | Compare String Operands | CMPS m32, m32 |
| CMPSB |  | Compare String Operands | CMPSB |
| CMPSW |  | Compare String Operands | CMPSW |
| CMPSD |  | Compare String Operands | CMPSD |
| 搜索 | SCAS | 不会影响DF,IF,TF | Scan String | SCAS m32 |
| SCASB |  | Scan String | SCASB |
| SCASW |  | Scan String | SCASW |
| SCASD |  | Scan String | SCASD |
| 装入 | LODS | 不影响标志位 | | | | | | | | |  | Load String | LODS m32 |
| LODSB |  | Load String | LODSB |
| LODSW |  | Load String | LODSW |
| LODSD |  | Load String | LODSD |
| 填充 | STOS |  | Store String | STOS m32 |
| STOSB |  | Store String | STOSB |
| STOSW |  | Store String | STOSW |
| STOSD |  | Store String | STOSD |
| 循环前缀 | REP |  | Repeat String Operation Prefix | REP MOVS m32,m32 |
| REPZ |  | Repeat String Operation Prefix | REPZ LODS EAX |
| REPNZ |  | Repeat String Operation Prefix | REPNZ STOS m32 |
| REPE |  | Repeat String Operation Prefix | REPE CMPS m32,m32 |
| REPNE |  | Repeat String Operation Prefix | REPNE SCAS m32 |
| 条件传送 | 标志位测试 | CMOVZ/CMOVNZ | 1/0 |  |  |  |  |  |  |  |  | 指出测试的标志位，下同 | Move if zero |  |
| CMOVE/CMOVNE | 1/0 |  |  |  |  |  |  |  |  |  | Move if equal |  |
| CMOVC/CMOVNC |  | 1/0 |  |  |  |  |  |  |  |  | Move if carry |  |
| CMOVS/CMOVNS |  |  |  | 1/0 |  |  |  |  |  |  | Move if sign |  |
| CMOVO/CMOVNO |  |  |  |  | 1/0 |  |  |  |  |  | Move if overflow |  |
| CMOVP/CMOVNP |  |  | 1/0 |  |  |  |  |  |  |  | Move if parity |  |
| CMOVPE/CMOVPO |  |  | 1/0 |  |  |  |  |  |  |  | Move if parity even/odd |  |
| 无符号数比较 | CMOVA/CMOVNBE | 0& | &0 |  |  |  |  |  |  |  | ZF=0 AND CF=0 | Move if above |  |
| CMOVNA/CMOVBE | 1| | |1 |  |  |  |  |  |  |  | ZF=1 OR CF=1 | Move if below or equal |  |
| CMOVB/CMOVNAE | 0& | &1 |  |  |  |  |  |  |  | ZF=0 AND CF=1 | Move if below |  |
| CMOVNB/CMOVAE | 1| | |0 |  |  |  |  |  |  |  | ZF=1 OR CF=0 | Move if above or equal |  |
| 有符号数比较 | CMOVG/CMOVNLE | 0& |  |  | &= | = |  |  |  |  | ZF=0 AND SF=OF | Move if greater |  |
| CMOVNG/CMOVLE | 1| |  |  | |< | > |  |  |  |  | ZF=1 OR SF<>OF | Move if less or equal |  |
| CMOVL/CMOVNGE |  |  |  | < | > |  |  |  |  | SF<>OF | Move if less |  |
| CMOVNL/CMOVGE |  |  |  | = | = |  |  |  |  | SF=OF | Move if greater or equal |  |
| 条件设置 | 标志位测试 | SETZ/SETNZ | 1/0 |  |  |  |  |  |  |  |  | 指出测试的标志位，下同 | Set byte if zero |  |
| SETE/SETNE | 1/0 |  |  |  |  |  |  |  |  |  | Set byte if equal |  |
| SETC/SETNC |  | 1/0 |  |  |  |  |  |  |  |  | Set byte if carry |  |
| SETS/SETNS |  |  |  | 1/0 |  |  |  |  |  |  | Set byte if sign |  |
| SETO/SETNO |  |  |  |  | 1/0 |  |  |  |  |  | Set byte if overflow |  |
| SETP/SETNP |  |  | 1/0 |  |  |  |  |  |  |  | Set byte if parity |  |
| SETPE/SETPO |  |  | 1/0 |  |  |  |  |  |  |  | Set byte if parity even/odd |  |
| 无符号数比较 | SETA/SETNBE | 0& | &0 |  |  |  |  |  |  |  | ZF=0 AND CF=0 | Set byte if above |  |
| SETNA/SETBE | 1| | |1 |  |  |  |  |  |  |  | ZF=1 OR CF=1 | Set byte if below or equal |  |
| SETB/SETNAE | 0& | &1 |  |  |  |  |  |  |  | ZF=0 AND CF=1 | Set byte if below |  |
| SETNB/SETAE | 1| | |0 |  |  |  |  |  |  |  | ZF=1 OR CF=0 | Set byte if above or equal |  |
| 有符号数比较 | SETG/SETNLE | 0& |  |  | &= | = |  |  |  |  | ZF=0 AND SF=OF | Set byte if greater |  |
| SETNG/SETLE | 1| |  |  | |< | > |  |  |  |  | ZF=1 OR SF<>OF | Set byte if less or equal |  |
| SETL/SETNGE |  |  |  | < | > |  |  |  |  | SF<>OF | Set byte if less |  |
| SETNL/SETGE |  |  |  | = | = |  |  |  |  | SF=OF | Set byte if greater or equal |  |
| 特  权  数  据  传  送 | 读取 | SGDT | 不影响标志位 | | | | | | | | |  | Store Global Descriptor Table Register | SGDT m |
| SIDT |  | Store Interrupt Descriptor Table Register | SIDT m |
| SLDT |  | Store Local Descriptor Table Register | SLDT r/m32 |
| SMSW |  | Store Machine Status Word | SMSW r32/m16 |
| STR |  | Store Task Register | STR r/m16 |
| RDMSR\* |  | Read from Model Specific Register | RDMSR |
| RDPMC\* |  | Read Performance-Monitoring Counters | RDPMC |
| 写入 | LGDT\* |  | Load Global Descriptor Table Register | LGDT m16&32 |
| LIDT\* |  | Load Interrupt Descriptor Table Register | LIDT m16&32 |
| LLDT\* |  | Load Local Descriptor Table Register | LLDT r/m16 |
| LMSW\* |  | Load Machine Status Word | LMSW r/m16 |
| LTR\* |  | Load Task Register | LTR r/m16 |
| WRMSR\* |  | Write to Model Specific Register | WRMSR |
| LAR(\*) | √ |  |  |  |  |  |  |  |  |  | Load Access Rights Byte | LAR r32,r/m32 |
| LSL(\*) | √ |  |  |  |  |  |  |  |  |  | Load Segment Limit | LSL r32,r/m32 |
| 算术运算类 | 加法 | | ADD | √ | √ | √ | √ | √ | √ |  |  |  |  | Add | ADD r/m32,imm32 |
| ADC | √ | √ | √ | √ | √ | √ |  |  |  |  | Add with Carry | ADC r/m32,imm32 |
| INC | √ |  | √ | √ | √ | √ |  |  |  |  | Increment by 1 | INC r/m32 |
| AAA | ？ | √ | ？ | ？ | ？ | √ |  |  |  | **？表示不确定或者未定义** | ASCII Adjust After Addition | AAA |
| DAA | √ | √ | √ | √ | ？ | √ |  |  |  |  | Decimal Adjust AL after Addition | DAA |
| XADD | √ | √ | √ | √ | √ | √ |  |  |  |  | Exchange and Add | XADD r/m32,r32 |
| 减法 | | SUB | √ | √ | √ | √ | √ | √ |  |  |  |  | Subtract | SUB r/m32,imm32 |
| SBB | √ | √ | √ | √ | √ | √ |  |  |  |  | Integer Subtraction with Borrow | SBB r/m32,imm32 |
| DEC | √ |  | √ | √ | √ | √ |  |  |  |  | Decrement by 1 | DEC r/m32 |
| AAS | ？ | √ | ？ | ？ | ？ | √ |  |  |  |  | ASCII Adjust AL After Subtraction | AAS |
| DAS | √ | √ | √ | √ | ？ | √ |  |  |  |  | Decimal Adjust AL after Subtraction | DAS |
| 乘法 | | MUL | ？ | √ | ？ | ？ | √ | ？ |  |  |  |  | Unsigned Multiply | MUL r/m32 |
| IMUL | ？ | √ | ？ | ？ | √ | ？ |  |  |  |  | Signed Multiply | IMUL r32,r/m32,imm32 |
| AAM | √ | ？ | √ | √ | ？ | ？ |  |  |  |  | ASCII Adjust AX After Multiply | AAM |
| 除法 | | DIV | ？ | ？ | ？ | ？ | ？ | ？ |  | √ | √ | 此指令可能会产生中断，故影响IF,TF | Unsigned Divide | DIV r/m32 |
| IDIV | ？ | ？ | ？ | ？ | ？ | ？ |  | √ | √ | 同上 | Signed Divide | IDIV r/m32 |
| AAD | √ | ？ | √ | √ | ？ | ？ |  |  |  |  | ASCII Adjust AX Before Division | AAD |
| 求负 | | NEG | √ | √ | √ | √ | √ | √ |  |  |  | CF同源操作数的真或假 | Two's Complement Negation | NEG r/m32 |
| 符号位扩展 | | CBW | 不影响标志位 | | | | | | | | |  | Convert Byte to Word | CBW |
| CWDE |  | Convert Word to Doubleword | CWDE |
| CWD |  | Convert Word to Doubleword | CWD |
| CDQ |  | Convert Double to Quadword | CDQ |
| 比较 | | CMP | **由计算结果确定标志位的值** | | | | | | | | | 不会影响到DF,IF,TF | Compare Two Operands | CMP r/m32,imm32 |
| 逻辑运算类 | 单  操  作  数 | 移位 | SHL | √ | √ | √ | √ | 见  备  注 | ？ |  |  |  | OF标志位只在移位次数是1时有效 | Shift left imm8/CL times | SHL r/m32,imm8 |
| SHR | √ | √ | √ | √ | ？ |  |  |  | Shift right imm8/CL times | SHR r/m32,imm8 |
| SAL | √ | √ | √ | √ | ？ |  |  |  | Shift left without sign bit imm8/CL times | SAL r/m32,imm8 |
| SAR | √ | √ | √ | √ | ？ |  |  |  | Shift right without sign bit mm8/CL times | SAR r/m32,imm8 |
| SHLD | √ | √ | √ | √ | ？ |  |  |  | Double Precision Shift Left | SHLD r/m32,r32,imm8 |
| SHRD | √ | √ | √ | √ | ？ |  |  |  | Double Precision Shift Right | SHRD r/m32,r32,imm8 |
| 循环  移位 | ROL |  | √ |  |  | √ |  |  |  |  |  | Rotate left imm8/CL times | ROL r/m32,imm8 |
| ROR |  | √ |  |  | √ |  |  |  |  |  | Rotate right imm8/CL times | ROR r/m32,imm8 |
| RCL |  | √ |  |  | √ |  |  |  |  |  | Rotate left with CF imm8/CL times | RCL r/m32,imm8 |
| RCR |  | √ |  |  | √ |  |  |  |  |  | Rotate right with CF imm8/CL times | RCR r/m32,imm8 |
| 调整字节序 | BSWAP | 不影响标志位 | | | | | | | | |  | Byte Swap | BSWAP r32 |
| 求反 | NOT |  | One's Complement Negation | NOT r/m32 |
| 双  操  作  数 | 与 | AND | √ | 0 | √ | √ | 0 | ？ |  |  |  |  | Logical AND | AND r/m32,r/m32 |
| 或 | OR | √ | 0 | √ | √ | 0 | ？ |  |  |  |  | Logical Inclusive OR | OR r/m32,r/m32 |
| 异或 | XOR | √ | 0 | √ | √ | 0 | ？ |  |  |  |  | Logical Exclusive OR | XOR r/m32,r/m32 |
| 检测 | TSET | √ | 0 | √ | √ | 0 | ？ |  |  |  |  | Logical Compare | TEST r/m32,r/m32 |
| 位操作类 | 标志位操作 | | CLC |  | √ |  |  |  |  |  |  |  | 置CF=0 | Clear Carry Flag | CLC |
| CMC |  | √ |  |  |  |  |  |  |  | 使CF取反 | Complement Carry Flag | CMC |
| STC |  | √ |  |  |  |  |  |  |  | 置CF=1 | Set Carry Flag | STC |
| CLD |  |  |  |  |  |  | √ |  |  | 置DF=0 | Clear Direction Flag | CLD |
| STD |  |  |  |  |  |  | √ |  |  | 置DF=1 | Set Direction Flag | STD |
| CLI\* |  |  |  |  |  |  |  | √ |  | 置IF=0 | Clear Interrupt Flag | CLI |
| STI\* |  |  |  |  |  |  |  | √ |  | 置IF=1 | Set Interrupt Flag | STI |
| CLTS\* |  |  |  |  |  |  |  |  |  | 清空CR0中的TS标志 | Clear Task-Switched Flag in CR0 | CLTS |
| 位测试 | | BT | ？ | √ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Test | BT r/m32,imm8 |
| BTC | ？ | √ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Test and Complement | BTC r/m32,imm8 |
| BTR | ？ | √ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Test and Reset | BTR r/m32,imm8 |
| BTS | ？ | √ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Test and Set | BTS r/m32,imm8 |
| 位扫描 | | BSF | √ | ？ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Scan Forward | BSF r32,r/m32 |
| BSR | √ | ？ | ？ | ？ | ？ | ？ |  |  |  |  | Bit Scan Reverse | BSR r32,r/m32 |
| 程序控制类 | 无条件转移 | | JMP | 不影响标志位 | | | | | | | | |  | Jump |  |
| 条  件  转  移 | 寄存器测试 | JCXZ |  | Jump short if CX=0 |  |
| JECXZ |  | Jump short if ECX=0 |  |
| 标志位测试 | JZ/JNZ | 1/0 |  |  |  |  |  |  |  |  | 指出测试的标志位，下同 | Jump short if zero |  |
| JE/JNE | 1/0 |  |  |  |  |  |  |  |  |  | Jump short if equal |  |
| JC/JNC |  | 1/0 |  |  |  |  |  |  |  |  | Jump short if carry |  |
| JS/JNS |  |  |  | 1/0 |  |  |  |  |  |  | Jump short if sign |  |
| JO/JNO |  |  |  |  | 1/0 |  |  |  |  |  | Jump short if overflow |  |
| JP/JNP |  |  | 1/0 |  |  |  |  |  |  |  | Jump short if parity |  |
| JPE/JPO |  |  | 1/0 |  |  |  |  |  |  |  | Jump short if parity even/odd |  |
| 无符号数比较 | JA/JNBE | 0& | &0 |  |  |  |  |  |  |  | ZF=0 AND CF=0 | Jump short if above |  |
| JNA/JBE | 1| | |1 |  |  |  |  |  |  |  | ZF=1 OR CF=1 | Jump short if below or equal |  |
| JB/JNAE | 0& | &1 |  |  |  |  |  |  |  | ZF=0 AND CF=1 | Jump short if below |  |
| JNB/JAE | 1| | |0 |  |  |  |  |  |  |  | ZF=1 OR CF=0 | Jump short if above or equal |  |
| 有符号数比较 | JG/JNLE | 0& |  |  | &= | = |  |  |  |  | ZF=0 AND SF=OF | Jump short if greater |  |
| JNG/JLE | 1| |  |  | |< | > |  |  |  |  | ZF=1 OR SF<>OF | Jump short if less or equal |  |
| JL/JNGE |  |  |  | < | > |  |  |  |  | SF<>OF | Jump short if less |  |
| JNL/JGE |  |  |  | = | = |  |  |  |  | SF=OF | Jump short if greater or equal |  |
| 循环控制 | | LOOP | 不影响标志位 | | | | | | | | |  | Jump short if ECX=0 | LOOP rel8 |
| LOOPZ |  | Jump short if ECX=0 and ZF=1 | LOOPZ rel8 |
| LOOPNZ |  | Jump short if ECX=0 and ZF=0 | LOOPNZ rel8 |
| LOOPE |  | Jump short if ECX=0 and ZF=1 | LOOPE rel8 |
| LOOPNE |  | Jump short if ECX=0 and ZF=0 | LOOPNE rel8 |
| 中断与返回 | | INT | 根据不同情况影响标志位 | | | | | | | | |  | Call to Interrupt Procedure | INT imm8 |
| INT1 |  | Call to Interrupt Procedure | INT1 |
| INT3 |  | Call to Interrupt Procedure | INT3 |
| INTO |  | Call to Interrupt Procedure | INTO |
| IRET | **影响所有标志位** | | | | | | | | | 标志位到恢复中断以前的状态 | Interrupt Return | IRET |
| IRETW |  | Interrupt Return Word | IRETW |
| IRETD |  | Interrupt Return Doubleword | IRETD |
| 系统调用与返回 | | SYSENTER | 不影响标志位 | | | | | | | | |  | System Enter | SYSENTER |
| SYSLEAVE |  | System Leave | SYSLEAVE |
| 函数调用与返回 | | CALL | 如果发生任务切换则影响所有 | Call Procedure | CALL m16:32 |
| RET |  | Return from Procedure | RET imm16 |
| RETN |  | Return from Procedure to Near | RETN imm16 |
| RETF |  | Return from Procedure to Far | RETF imm16 |
| CPU控制类 | 空操作 | | NOP | 不影响标志位 | | | | | | | | |  | No Operation | NOP |
| 暂停 | | HLT\* |  | Stop instruction execution and Halt | HLT |
| 等待协处理器 | | WAIT |  | Check pending unmasked floating-point exceptions | WAIT |
| 交权给协处理器 | | ESC |  | Escape | ESC EXTOPRD,OPRD |
| 总线锁定前缀 | | LOCK |  | Assert LOCK# Signal Prefix | LOCK |
| 清除内部缓存 | | INVD\* |  | Invalidate Internal Caches | INVD |
| 刷新内部缓存 | | WBINVD\* |  | Write Back and Invalidate Cache | WBINVD |
| 清除旁路翻译缓存 | | INVLPG\* |  | Invalidate TLB Entry | INVLPG m |
| 脱离系统管理模式 | | RSM | √ | √ | √ | √ | √ | √ | √ | √ | √ |  | Resume from System Management Mode | RSM |
| 杂项 | 段寄存器读检测 | | VERR | √ |  |  |  |  |  |  |  |  |  | Verify a Segment for Reading | VERR r/m16 |
| 段寄存器写检测 | | VERW | √ |  |  |  |  |  |  |  |  |  | Verify a Segment for Writing | VERW r/m16 |
| 高级语言辅助 | | ENTER | 不影响标志位 | | | | | | | | |  | Make Stack Frame for Procedure Parameters | ENTER imm16,imm8 |
| LEAVE |  | High Level Procedure Exit | LEAVE |
| 检验并调整RPL | | ARPL |  | Adjust RPL Field of Segment Selector | ARPL r/m16,r16 |
| 数组边界检查 | | BOUND |  | Check Array Index Against Bounds | BOUND r32,m32&32 |
| 识别CPU | | CPUID |  | CPU Identification | CPUID |
| 读取时间戳 | | RDTSC(\*) |  | Read Time-Stamp Counter | RDTSC |
| 未定义 | | UD2 |  | Undefined Instruction | UD2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 协处理器类 | 传送 |  |  |  |  |  |
|  |  |  |  |  |
| 加法 |  |  |  |  |  |
|  |  |  |  |  |
| 减法 |  |  |  |  |  |
|  |  |  |  |  |
| 乘法 |  |  |  |  |  |
|  |  |  |  |  |
| 除法 |  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 等待 | FWAIT |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**浮点**

**F2XM1 FABS FADD FADDP FBLD FBSTP FCHS FCLEX FCOM FCOMP FCOMPP**

**FCOS FDECSTP FDISI FDIV FDIVP FDIVR FDIVRP FENI FFREE FIADD**

**FICOM FICOMP FIDIV FIDIVR FILD FIMUL FINCSTP FINIT FIST FISTP**

**FISUB FISUBR FLD FLD1 FLDL2E FLDL2T FLDCW FLDENV FLDLG2 FLDLN2**

**FLDPI FLDZ FMUL FMULP FNCLEX FNDISI FNENI FNINIT FNOP FNSAVE**

**FNSTCW FNSTENV FNSTSW FPATAN FPREM FPREM1 FPTAN FRNDINT FRSTOR**

**FSAVE FSCALE FSETPM FSIN FSINCOS FSQRT FST FSTCW FSTENV FSTP**

**FSTSW FSUB FSUBP FSUBR FSUBRP FTST FUCOM FUCOMP FUCOMPP FWAIT**

**FXAM FXCH FXTRACT FYL2X FYL2XP1**

**CF**(Carry Flag):进位标志。当指令执行的结果（8位或16位）在最高位上产生了一个进位或借位时，CF =1。

**AF**(Auxiliary Carry Flag):辅助进位标志。当一个8位数（或16位数）的低四位向高四位（即b3向b4）有进位或借位时，AF＝1。常用于十进制算术运算指令。

**OF**(Overflow Flag): 溢出标志。在算术运算中，带符号数的运算结果超出了8位或16位符号数所能表示的范围时，OF＝1。

**ZF**(Zero Flag): 零标志。当运算结果为全零时，ZF=1。

**SF**(Sign Flag): 符号标志。当运算结果为正数，即结果的最高位为0时，SF=1。

**PF**(Parity Flag):奇偶标志。当算术逻辑运算的结果中1的个数为偶数时，PF＝1，为奇数时，PF=0。

**DF**(Direction Flag):方向标志。用于控制数据串操作指令的步进方向，当DF=1时，表示从高地址向低地址以递减的顺序对数据串中的数据进行处理。

IF(Interrupt-enable Flag):中断允许标志。当IF=1时，CPU可以响应外部可屏蔽中断请求。该标志可以用指令设置为1或0。

TF(Trap Flag):陷阱标志。当TF=1时，CPU进入单步工作方式，每执行完一条指令就自动产生一个内部中断，以便进行程序调试。当TF＝0时，正常执行程序。