

Registers

- Control: dictates current state of the machine
- Status: indicates status of operation (error, overflow, etc.)
- MAR: Memory Address Register (holds address of memory location currently referenced)
- MDR: Memory Data Register: holds data being sent to or retrieved from the memory address in the MAR
- IP: Instruction Pointer (holds memory address of next instruction)
- IR: Instruction Register (holds current machine instruction)
- Operand_1, Operand_2, Result: ALU registers (for calculations and comparisons)
- General: fast temporary storage eax,ebx,ecx,edx

Instruction Execution Cycle

1. **Fetch** next instruction (at address in IP) into IR.
2. **Increment** IP to point to next instruction.
3. **Decode** instruction in IR
4. If instruction requires memory access,
 - A. Determine memory address.
 - B. **Fetch operand** from memory into a CPU register, or send operand from a CPU register to memory.
5. **Execute** micro-program for instruction
6. Go to step 1 (unless the "halt" instruction has been executed)

$$K_i = 2^{10}, M_i = 2^{20}, G_i = 2^{30}$$

- **ReadString** - Reads a string from keyboard, terminated by the Enter key.
 - Preconditions: **OFFSET** of memory destination in **EDX** , **Size** of memory destination in **ECX**
 - Postconditions: String entered is in memory
 - Return : **Length** of string entered is in **EAX**
- **ReadDec**, unsigned int EAX, **ReadInt** signed int EAX

push decrements esp then copies
pop copies and then increments esp
ESP always points to a value not an empty spot

- The **CALL** instruction calls a procedure
- **Pushes** the offset of the next instruction onto the stack
- **copies** the address of the called procedure into EIP

LOOP first decrements ECX then Jumps if !=0 [n-1]

CMP dest, source JE d=s JNE d !=s

UNSIGNED	SIGNED
JA, JNBE dest > source	JG jump if destination > source JNLE
JAE, JNE >=	JGE jump if destination >= source JNL
JB, JNAE <	JL jump if destination < source JNGE
JBE, JNA <=	JLE jump if destination <= source

0000	0001	0010	0011	0100	0101	0110	0111
0	1	2	3	4	5	6	7
F	E	D	C	B	A	9	8
1111	1110	1101	1100	1011	1010	1001	1000

Type	Used for:
BYTE	1-byte unsigned integers [0 ... 255], ASCII characters
SBYTE	1-byte signed integers [-128 ... 127]
WORD	2-byte unsigned integers [0 ... 65535], address
SWORD	2-byte signed integers [-32768 ... 32767]
DWORD	4-byte unsigned integers [0 ... 4294967295], address
SDWORD	4-byte signed integers [-2147483648 ... 2147483647]
QWORD	8-byte integer
TBYTE	10-byte integer
REAL4	4-byte floating-point
REAL8	8-byte floating-point
REAL10	10-byte floating-point

IMUL 1,2,3 operand

Instruction Form Condition for Clearing CF and OF

r/m8 AL := sign-extend of AL to 16 bits

r/m16 AX := sign-extend of AX to 32 bits

r/m32 EDX:EAX := sign-extend of EAX to 32 bits

r16,r/m16 Result exactly fits within r16

r/32,r/m32 Result exactly fits within r32

r16,r/m16,imm16 Result exactly fits within r16

r32,r/m32,imm32 Result exactly fits within r32

Dividend / Divisor = Quotient remainder

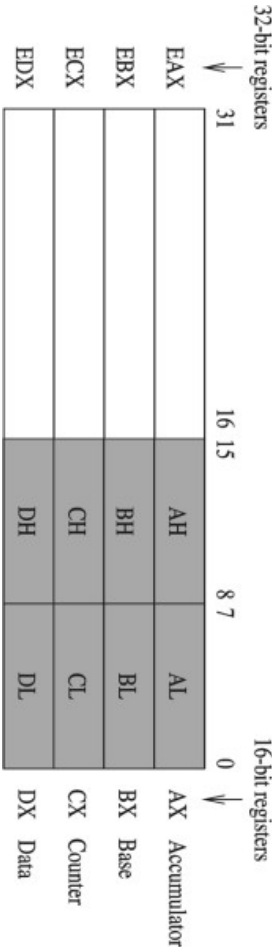
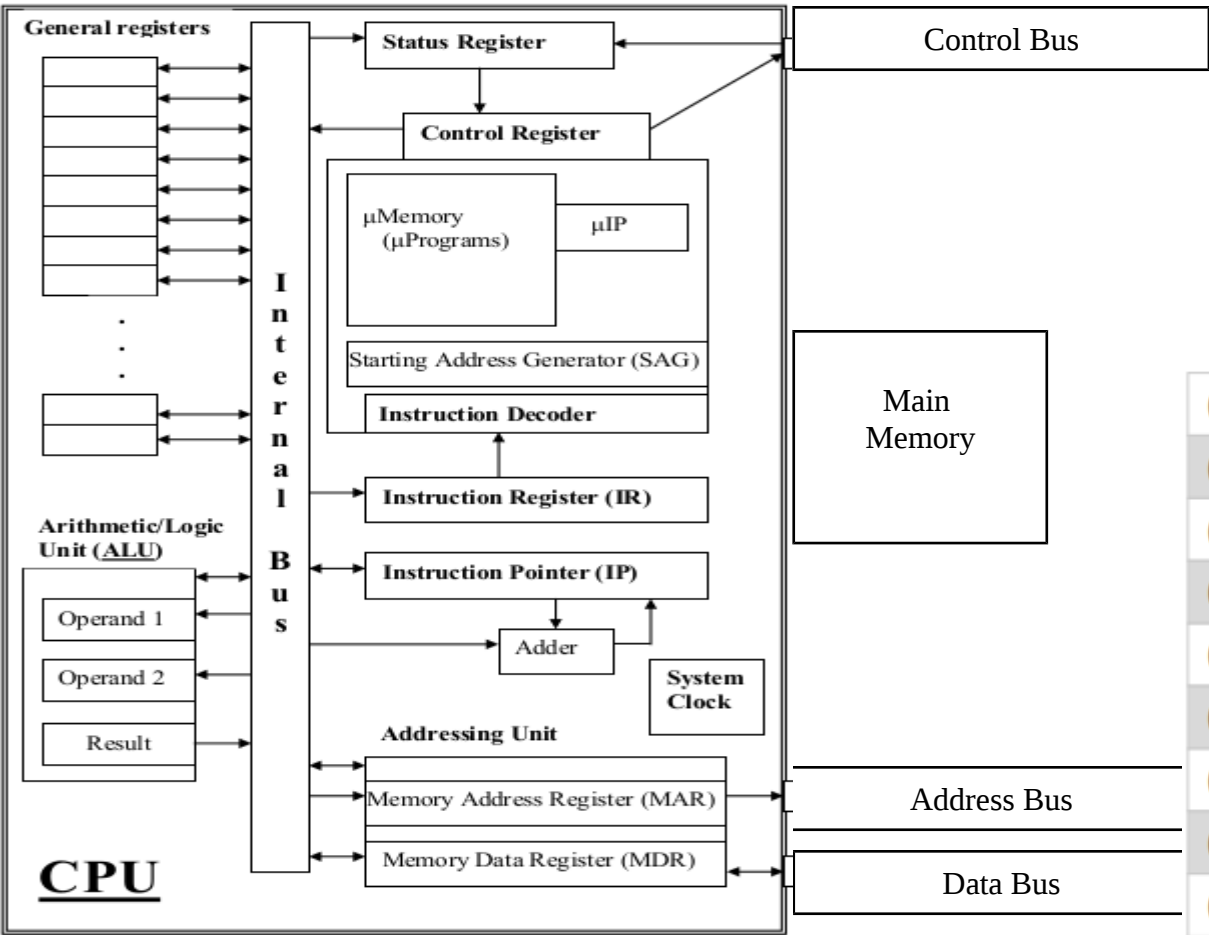
edx:eax div reg/mem EAX EDX

SUB sets carry if a-b a<b

0100 + 0100 = 1000 (overflow flag is turned on)

1000 + 1000 = 0000 (overflow flag is turned on)

OF overflow flag is XORED with Carry Flag



0.0000	0.0000
0.0001	0.0625
0.0010	0.1250
0.0011	0.1875
0.0100	0.2500
0.0101	0.3125
0.0110	0.3750
0.0111	0.4375
0.1000	0.5000
0.1001	0.5625
0.1010	0.6250
0.1011	0.6875
0.1100	0.7500
0.1101	0.8125
0.1110	0.8750
0.1111	0.9375

Table 12-1 IEEE Floating-Point Binary Formats.

Single Precision	32 bits: 1 bit for the sign, 8 bits for the exponent, and 23 bits for the fractional part of the significand. Approximate normalized range: 2^{-126} to 2^{127} . Also called a <i>short real</i> .
Double Precision	64 bits: 1 bit for the sign, 11 bits for the exponent, and 52 bits for the fractional part of the significand. Approximate normalized range: 2^{-1022} to 2^{1023} . Also called a <i>long real</i> .
Double Extended Precision	80 bits: 1 bit for the sign, 16 bits for the exponent, and 63 bits for the fractional part of the significand. Approximate normalized range: 2^{-16382} to 2^{16383} . Also called an <i>extended real</i> .