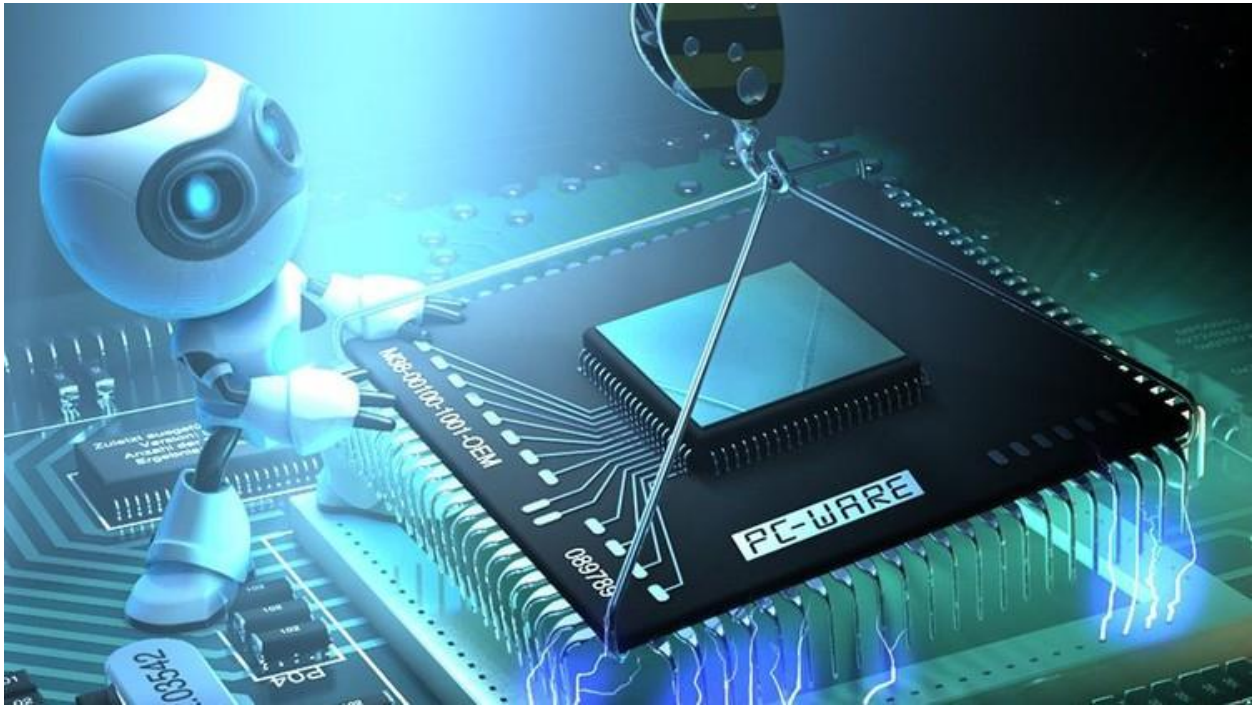
Logical Instructions

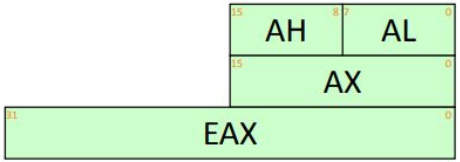
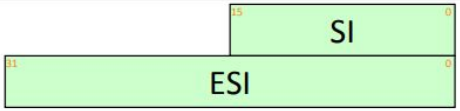


Logical Instructions

Instructions	Description	Notes	Example
ANDx op1, op2	op2 ← op2&op1	x = {L, W, B}	ANDL $13,%EAX
ORx op1, op2	op2 ← op2|op1	x = {L, W, B}	ORW %CX,%AX
XORx op1, op2	op2 ← op2^op1	x = {L, W, B}	XORL %EDX,%EAX
NOTx op1	op1 ← ~op1	x = {L, W, B}	NOTB %AH
SALx k,op1	op1 ← op1<<k (aritm.)	x = {L, W, B}, k: inm. o %CL	SALL $1,%EAX
SHLx k,op1	op1 ← op1<<k (log.)	x = {L, W, B}, k: inm. o %CL	SHLW %CL,%DX
SARx k,op1	op1 ← op1>>k (aritm.)	x = {L, W, B}, k: inm. o %CL	SARL $1,%EAX
SHRx k,op1	op1 ← op1>>k (log.)	x = {L, W, B}, k: inm. o %CL	SHRW %CL,%DX
CMPx op1, op2	op2-op1	x = {L, W, B}, activa flags	CMPL $13,%EAX
TESTx op1, op2	op2&op1	x = {L, W, B}, activa flags	TESTW %CX,%AX

Programmer Vision

Available registers

32 bits	16 bits	8 bits	
%eax	%ax	%ah,%al	
%ebx	%bx	%bh,%bl	
%ecx	%cx	%ch,%cl	
%edx	%dx	%dh,%dl	
%esi	%si		
%edi	%di		
%esp	%sp		Reserved for specific use of subroutines
%ebp	%bp		
%eip			Program counter
%eflags			Status word