Veriyolu genişliği → 4,8,16 bit gibi;

Adres ucu sayısı →4096-8192 vb. $2^n = 4096$ $n=12$

Temporal locality→ Aynı değişken birden çok kullanılabilir.
diğe cache'e atıyoruz.

Spatial locality → diziyi toplu okuyoruz

İşaretli; gösterim 1101 0010 →-82

1'st complement 1010 1101 →-82

2's '' → 1010 01 01 1111 →-82

BCD → 255 → 0010 0101 0101 8101

Little endian → 1234 H →
1204 H → 12 H
1004 H → 34 H

16 bitlik 20 bitlik adres segment+offset+16 ile saklanır

AX: DIV,MUL gizli operand DIV BX

$$\frac{DX:AX}{DX \mid AX} \Big| BX$$

CS ile BX,DI,SI kullanılır,

BX: Base register

SI,DI →adres yazmaçları, offset ve CS

CX: Döngü yazmacı CL,CH

DX: Data register DIV,MUL vs. DL,DH

SP: Stack pointer PUSH,POP vs.

BP: Base '' değiştirilemez

Flagler

Parity Flag: PF → Çift bit -tek bit

AF: Toplama işleminde taşma var mı? varsa 1

Zero flag: En son yapılan işlemin sonucu 0 ise 1

Sign flag: Sonuç — ise 1

Ⓞ Overflow flag: Taşma varsa 1 (işaretlide)

Basında E varsa 32 bit R varsa 64 bit

reg8 → 8 bitlik yazmaç.

acc → Ax, AL, AH

regw → 16 bitlik "

sreg → segment yazmacı CS, SS, ES, DS

mem → adres

i data → sabit

dest/src: hedef lkaynak

Komutlar

MOV → sreg-sreg olmaz, sreg-idata olmaz mem-mem olmaz
ve memory idata/da word PTR ya da byte PTR gerekli

LEA → tanımla LEA SI, A sayisi

LDS → load data segment register
LDS reg, mem
reg ← [mem]
DS ← [mem+2]

LES → load extra segment register
LES reg, mem
reg ← [mem]
ES ← [mem+2]

XCHG → değiştirme
XCHG dest, src

XLAT / XLATB
AL ← DS [BX+AL]

Aritmetik komutlar

ADD

Toplama

ADC → Elldeli toplama

SUB → Çıkarma

SBB → Elldeli ç.

INC → +1

DEC → -1

NEG → 2S complement

CMP → Çıkarma → Bayrak değişir

MUL → Çarpma

SMUL → işaretli ç.

DIV → Bölme

IDIV → işaretli B.