

Computer instruction

يملك ال computer بصورة عامة ثلاث صيغ من صيغ الايعازات "instruction format" كل صيغة من هذه الصيغ تمتلك 16 bit .:

*الجزء الأول من صيغة الإيعاز يمثل "operation code" جزء العملية " opcode " وهو يتكون من "3 bit" .

*أما ال 13 bits المتبقية تعتمد في معناها على ال opcode .

* ١- memory-reference ins.: يستخدم 12 bit للعنوانه address mode وفيه $ii=0$ لنوع العنوانه الغير مباشر addr. Mode و $ii=0$ للعنوانه الغير مباشرة indirect address .mode

* ٢- the register-reference inst.: ويمكن التعرف عليه من خلال ال operation code وهي ١١١ والبت الأخير من جهة اليسار يساوى "٠" أي البت ال ١٥ من الإيعاز. في هذا النوع من الصيغة فان الإيعاز يحدد العملية او يقوم بفحص "Ac register" .

* ٣- اما ال input-output instruct.: فانه لا يحتاج إلى إشارات الذاكرة memory reference ويمكن معرفته من خلال ١١١ وال 12 bits يستخدم لتحديد نوع عملية الإدخال والإخراج أو لغرض فحص الأداء.

- الغرض من الإيعاز هو تحديد وظيفة عملية معينه لكي يقوم processor بتنفيذها وكذلك تحديد ال operands التي سوف يتم تطبيق العملية عليها.
- حيث ان ال operands تحتوي على ال input data ويوضح اين سوف يتم خزن النتائج.
- حيث ان operand field يحتوي على عناوين في الذاكرة الرئيسية هو مواقع في ال processor وهي ال registers.
- اغلب الايعازات تتضمن عمليات register transfer بالصيغة التالية:.

$$X1=f(X1,X2,\dots,Xn)$$

حيث ان X1 يقوم بخزن النتائج.

حيث ان ال F هي ال opcode، ال n تمثل عدد ال operands مع اليعاز.من اجل تقليل حجم اليعاز وبالتالي تقليل حجم البرنامجمن الشائع ان نشير في اليعلاز ال m من ال operands بشكل واضح وصريح (explicit) اما البقية فيكون ضمنى (implicit) بحيث ان $(m < n)$.
*عادة فان explicit address تشير الى main memory اما implicit address تشير الى register .

طرق العنونة في المعالج 80386DX (Addressing mode)

**addressing mode: is a method of specifying an operands.

و هي الطريقة التي يتم من خلالها تحديد المعامل. وتقسم طرق العنونة الى ثلاث فئات:.

1-Register operand addressing mode.

2-Immediate operand addressing mode.

3-Memory operand addressing mode.

1- Register Addressing Mode

فيها يمكن الوصول للoperands بواسطة تحديد المسجل الداخلي للمعالج والذي يكون حاويا لل(operands). اما المسجلات التي يمكن ان تكون مصدر للبيانات او للمعاملات هي المسجلات الداخلية للمعالج كما في الجدول في الشكل رقم (3.7).مثال Accumulator
AL,BH

MOV AX,BX

-يعني نقل محتويات المسجل BX الى المسجل AX .

نلاحظ في الشكل (3.8 (a) ان المسجل BX يحتوي على (ABCD) وان المسجل AX مهمل المحتويات. اما (CS,IP) فتشير الى الموقع الذي يحتوي على اليعاز. كما نلاحظ في الشكل.

-بعد تنفيذ اليعاز يكون محتوى المسجل (AX) هي (ABCD) والمسجلين (IP,CS) يشيران الى موقع اليعاز التالي في التنفيذ، كما في الشكل (3.8(b)).

٢- Immediate operand addressing mode :

إذا كان ال(operands) جزء من الإيعاز بدلا من محتويات المسجل او موقع الذاكرة فهذا يشير الى طريقة العنوان المباشرة، مثال

MOV AL, 15H

-إذا كان "X" تمثل قيمة ثابتة (const)، فهنا نستطيع وضع قيمتها في الإيعاز مباشرة في جزء ال operand. مثال كما في الشكل (3.10(a)) قبل تنفيذ الإيعاز اما بعد تنفيذ الإيعاز فيكون شكل المعالج كما في الشكل (3.10(b)).

MOV ECX, 12345

3- 16-Bit Memory Operand Addressing Mod:

للإشارة إلى ال(operand) موجودة بالذاكرة، فان المعالج يحتاج الى حساب العنوان الحقيقي لل(physical address(ph)) الخاص بالoperand وبعدها يبدأ عملية القراءة او الكتابة في موقع الخزن.

*ان العنوان الحقيقي (ph) يتكون من segment based address (SBA) عنوان معتمد على المقطع والeffective address (EA) العنوان الفعال.
* (SBA) تحدد بداية موقع الsegment في الذاكرة.
* (EA) تحدد بعد ال operand عن بداية ال segment.

(SBA): Identifies the starting location of the segment in memory.

(EA): Represents the offset of the operands from the beginning of this segment of memory.

3-1- Direct Addressing Mode

يمثل طريقة Immediate A.M في إن المعلومات الموجودة في الإيعاز مباشرة. في هذا النوع من طريقة العنوان فان الopcode يتبعها الeffective address بدلا من البيانات هذا ال(EA) يستخدم مباشرة كمسافة بين الموقع الحالي وموقع الoperand. الشكل (3.13(a))
MOV CX , [1234H]

DS:PA

DS: data segment

$$PA = \begin{pmatrix} CS \\ DS \\ SS \\ ES \\ FS \\ GS \end{pmatrix} : \text{Direct address.}$$

يتم اضافة (0) الى قيمة المسجل (DS) أي تضرب (10)

$$PA = 02000_{16} + 1234_{16} \\ = 03234_{16}$$

بعدها يتم قراءة محتوى العنوان والقراءة تتم من بداية العنوان ويتم نقل المحتوى الى المسجل الموجود في الابعاز كما في الشكل (3.13(b)).

3-2- Register Indirect Addressing Mode

في هذا النوع من العنوان فان (EA) يجمع مع محتويات (DS) لغرض الحصول على (PA)، قيمة الـ (EA) موجودة في احد المسجلات اما (base register او index register)،

$$PA = \begin{pmatrix} CS \\ DS \\ SS \\ ES \\ FS \\ GS \end{pmatrix} : \begin{pmatrix} BX \\ BP \\ SI \\ DI \end{pmatrix}$$

مثال على هذا النوع من العنوان

MOV AX,[SI]

يتم جمع محتويات المسجل المذكور في الابعاز مع محتويات المسجل (DS) كما في الشكل

(3.15(a))

$$PA = DS + SI \\ PA = 02000_{16} + 1234_{16} \\ PA = 03234_{16}$$

اما الشكل (3.15(b)) فيمثل شكل المعالج بعد تنفيذ الابعاز .

مقارنة بين انواع العنونة الثلاثة:

<i>Immediate</i>	<i>direct</i>	<i>indirect</i>
* أي تعديل في القيم يتطلب إجراء تعديل في البرنامج (أي تغيير بشكل الإيعاز من اجل تغيير أُل data لذلك يكون اقل مرونة)	* مرونة أعلى فقط تغيير محتوى العنوان الموجود مع الايعازات.	* يوفر مرونة عالية لإمكانية تغيير العناوين بسهولة.
* تعتبر سريعة جدا لان أُل data موجودة مباشرة	* أبطأ حيث تحتاج إلى fetch إضافي من اجل جلب ال data.	* بطئ جدا لأنه يحتاج إلى 2 fetch إضافي

3-3- Based Addressing Mode

في هذا النوع من العنونة فان (EA) الخاص بال "operand" يتم الحصول عليه بواسطة جمع الازاحه (displacement) المباشرة أو غير المباشرة مع محتويات إما (base register) مع محتويات (BX) (base register) او محتويات (BP) (base pointer).

$$PA = \left\{ \begin{array}{c} CS \\ DS \\ SS \\ ES \\ FS \\ GS \end{array} \right\} : \left\{ \begin{array}{c} SI \\ DI \end{array} \right\} + \left\{ \begin{array}{c} 8\text{-bit displacement} \\ 16\text{-bit displacement} \end{array} \right\}$$

مثال على هذا النوع من العنونة

MOV [BX]+1234H, AL
 PA=02000H+1000H+1234H
 PA= 04234H

كما هو موضح في الشكل (3.17(a)) وبعد تنفيذ الايعاز يكون شكل المعالج كما في الشكل (3.17(b)).

3-4- Based-Index Addressing Mode

يمكن استخدام هذا النوع من العنونة لغرض الوصول الى هياكل البيانات المعقدة مثل مصفوفة ثنائية الابعاد.

$$PA = \left\{ \begin{array}{c} CS \\ DS \\ SS \\ ES \\ FS \\ GS \end{array} \right\} : \left\{ \begin{array}{c} BX \\ BP \end{array} \right\} + \left\{ \begin{array}{c} SI \\ DI \end{array} \right\} + \left\{ \begin{array}{c} 8\text{- bit displacement} \\ 16\text{- bit displacement} \end{array} \right\}$$

مثال على هذا النوع من العنونة

MOV AL, [BX],[SI],1234H

PA=DS:EA

EA=[BX]+[SI]+1234

PA=02000H+1000H+2000H+1234H

PA=6234H

تنفيذ الايعز كما في الشكل (3.17(a)) وبعد تنفيذ الاياز يكون شكل المعالج كما في الشكل

(3.17(b)).