

(٤٠)
الباب الثالث

برمجة المعالج 8086

في لغة التجميع تستخدم الكلمات المختزلة mnemonic من أجل التعليمات فمثلاً بالنسبة للمعالج 8086 فالكلب المختزلة في لغة التجميع هي MOV و SUB و ADD و الترتيب على النقل هي على الترتيب ADD و SUB و MOV. المتحولات فتحدد المعطيات التي ستعالج من قبل المعالج بواسطة رمز العملية للتعليمات فمثلاً في التعليمات التي تصب محتويات مسجل القاعدة إلى محتويات المراكم فإن BX و AX هي المتحولات و تكتب التعليمات على الشكل التالي ADD AX, BX ففي هذا المثال تُضاف محتويات BX إلى AX و يوضع ناتج الجمع في AX و لذلك يُعتبر X متحول المصدر و AX متحول الهدف.

طاقم تعليمات المعالج 8086

يتوزع المعالج 8086 بمجموعة تعليمات مؤلفة من ما يقارب 117 تعليمة أساسية و كذلك إن المجال الواسع للمتحولات و أنظمة العنوان المسموحة للاستعمال مع هذه التعليمات يوسع مجموعة التعليمات إلى تعليمات أكثر، فمثلاً تعليمة MOV الأساسية تمتد إلى 28 تعليمة مختلفة و قابلة للتنفيذ على مستوى لغة الآلة.

أوامر نقل المعطيات :

يملك المعالج مجموعة تعليمات وظيفتها نقل المعطيات و ذلك إما بين مسجلات المعالج الداخلية أو بين مسجل داخلي و مكان تخزين في الذاكرة و هي:

1- تعليمة MOV

تستخدم هذه التعليمة لنقل بايت أو كلمة معطيات من متحول المصدر إلى متحول الهدف و لها الشكل التالي:

الكلمة المختزلة	المعنى	الصيغة	العملية	الاعلام المتأثرة
MOV	نقل	MOV D, S	$S \rightarrow D$	لا يوجد

شكل (1-3)

إن S, D لهذه التعليمة يمكن أن تكون مسجلات داخلية أو حجلات تخزين في الذاكرة و يبين الجدول التالي مختلف

الاشكال المختلفة لامر MOV

1. MOV AL, 02H (حمل البايت الاولى من المرمك بقيمة 02 بالنظام السداسي عشر)

الباب الثالث

برمجة المعالجات الدقيقة

أوامر الانتقال

أوامر الحساب

أوامر القفز

أوامر المنطق

أوامر الادخال والاخراج

(٤١)

MOV AX,20BFH (حمل المرمك البايت الاولى ب (BF) والبايت الثانية ب (20) بالنظام السداسى عشر)

MOV AL, B1 (حمل البايت الاولى من المرمك من البايت الاولى من المسجل B)

MOV AX,B: (حمل المسجل AX من المسجل Bx)

MOV AL,[1400H] (حمل البايت الاولى من المرمك من البايت فى الذاكرة التى عنوانها [1400H])

MOV [E000H],B (حمل بايت الذاكرة التى عنوانها [E000H] من البايت الثانية فى المسجل B)

MOV [E000H],F0H (حمل بايت الذاكرة التى عنوانها [E000H] بالقيمة الفورية F0H)

مستثانة من تعليمة MOV

نطبع تعليمة MOV أن تنقل المعطيات بشكل مباشر بين حجرتي ذاكرة لذلك لا نرى في الجدول المجاور : Mem → Mem و لحل هذه المشكلة فإن المعطيات المرغوب بنقلها يجب نقلها أولاً في مسجل داخلي تعليمة MOV ، و من ثم تنقل ، محتويات هذا المسجل إلى حجرة جديدة في الذاكرة بواسطة تعليمة MOV

، وضع قيمة فورية في مسجل مقطع مباشرة. أي أن التعليمة التالية غير مسموح بها MOV DS,1000 و لمشكلة نستخدم التعليمتين التاليتين :

MOV AX,1000

MOV DS,AX

نقل محتويات أحد مسجلات المقاطع إلى مسجل مقطع آخر مباشرة، أي أن التعليمة التالية غير مسموح بها MOV و لحل هذه المشكلة نقوم بـ

MOV AX,ES

MOV DS,AX

المطلوب عمل برنامج لتحميل المسجلات AL, BL, CL, DL بالمعلومات الفورية 02,04,06,08 وبعد عمل ازالة دورانية لبذرة المحتويات بداية البرنامج من العنوان E100 .

المعالجات و المتحكمات الدقيقة

(٤٢)

شفرات اسمبلى

العنوان	شفرات اسمبلى
E100	MOV AL,02
E102	MOV BL,04
E104	MOV CL,06
E106	MOV DL,08
E108	MOV AH,AL
E109	MOV AL,BL
E10A	MOV BL,CL
E10B	MOV CL,DL
E10C	MOV DL,AH

نلاحظ ان طول الامر 2 بايت لان كود الامر يوضع فى بايت والقيمة الفورية فى بايت

اما طول الامر من مسجل الى مسجل يحتاج كود الامر فقط فى بايت

٢- تعليمة التبادل XCHG

تستخدم هذه التعليمة لاستبدال تحول المصدر بمحول الهدف و لاستبدال محول الهدف بمحول المصدر.

شكل (3-4)

الاعلام المتأثرة	العملية	المصيغة	المعنى	الكلمة المشتركة
لا يوجد	S → D D → S	XCHG D,S	تبادل	XCHG

(٤٣)

خاتمة لتعليمية XCHG :

1- XCHG AX,BX

2- XCHG [F03A H],BL

3- XCHG CX,BX

أبواب : Arithmetic instructions

جمع : و هي مشروحة بالجدول التالي:

العلامة المتأثرة	العملية	الصفة	المعنى	الكلمة المستخلصة
علام الحالة	$S + D \rightarrow D$ Carry \rightarrow CF	ADD D,S	جمع	ADD
علام الحالة	$S + D + CF \rightarrow D$ Carry \rightarrow CF	ADC D,S	جمع مع أخذ الانزياح بعين الاعتبار	ADC
علام الحالة	$D + 1 \rightarrow D$	INC D	الزيادة بمقدار واحد	INC

شكل (3-4)

ن AX = 4F3Dh و BX = FD81h و CF = 1 فما هي نتيجة تنفيذ التعليمة ADC AX,BX ؟ مبيناً الحالة بعد تنفيذ عملية الجمع هذه
تب الشفرة الثنائية للمتحويلات من أجل توضيح حالة الأعلام

$$\begin{array}{r}
 11111111 \\
 AX = 0100111100111101b \\
 BX = 1111110110000001b \\
 CF = 0001b + \\
 \hline
 1 \quad 0100110010111111b
 \end{array}$$

CF

ثم الحالة هي:

(٤٤)

PF = 0 لأن عند الواحدات فردي في البايت الأول من ناتج الجمع

AF = 0 لأنه لا يوجد انزياح من الخانة 3 إلى الخانة 4 في البايت الأول من ناتج الجمع (حيث يتم ترقيم الخانات بدءاً من الصفر)

SF = 0 و هي آخر خانة من نتيجة الجمع (الناتج موجب)

CF = 1 بسبب وجود ترحيل خارجي

OF = 1 لأنه يوجد ترحيل خارجي فقط

ملاحظة: الانزياح الداخلي من الداخل إلى الخانة ذات الأهمية العظمى MSB

ملاحظة: OF = 1 إذا وجد ترحيل داخلي فقط أو وجد ترحيل خارجي فقط

مثال : اكتب برنامج لجمع ارقمين h(F3) و h(98) ثم ضع الناتج في اماكن الذاكرة E300,E301
الحل:

```

MOV AL,F3 H
MOV BL,98 H
ADD AL,BL
MOV [E300],AL
MOV AL,00H
ADC AL,AL
MOV [E301],AL

```

ب تعليمات الطرح

هناك مجموعة واسعة من تعليمات الطرح كما هو واضح من الجدول التالي:

العلامة المتأثرة	العملية	الصفة	المعنى	الكلمة المستخلصة
أعلام الحالة	$D - S \rightarrow D$ Borrow \rightarrow CF	SUB D,S	طرح	SUB
أعلام الحالة	$D - S - CF \rightarrow D$ Carry \rightarrow CF	SBB D,S	الطرح مع الاستعارة	SBB
أعلام الحالة	$D - 1 \rightarrow D$	DEC D	الإنقاص بمقدار واحد	DEC

(٤٦)

ببساطة : النقطة تعنى عملية الضرب العادية، و الرمز S8 يعنى متحول مصدر عبارة عن بايت أما الرمز R فيعنى باقى القسمة و الرمز Q ما هو إلا حاصل قسمة.

ملاحظة: إذا كانت قيمة Q في الحالة الأولى (حالة بايت) مساوية لـ FF أو كانت قيمة Q في الحالة الثانية (حالة كلمة) مساوية إلى FFFF فتحدث مقاطعة من النوع صفر، و تُعرف هذه المقاطعة بخطأ التقسيم.

أوامر المنطق Logical instruction تنجز عمليات المنطقية خانة بخانة على متحولاتها. ولا يوجد فيها ترحيل و الجدول التالي يبين التعليمات المنطقية:

الكلمة المختارة	المتنى	الصيغة	العملية	الأعلام المتأثرة
AND	AND كل خانة أو بايت في المصدر والغرض	AND D,S	$S.D \rightarrow D$	أعلام الحالة
OR	OR كل خانة أو بايت في المصدر والغرض	OR D,S	$S \cup D \rightarrow D$	أعلام الحالة
XOR	XOR كل بت أو بايت في المصدر والغرض	XOR D,S	$S + D \rightarrow D$	أعلام الحالة
NOT	NOT تغيير كل خانة (0 إلى 1 والعكس) في البايت	NOT D	$D \rightarrow D$	لا يوجد

إن المتحولات المسموحة من أجل تعليمة AND, OR, XOR مبيّنة في الجدول التالي:

****التعليمة AND**

أمثال 1:

MOV AX, 05h

MOV BX, 06h

AND AX, BX

النتيجة تخزن في المسجل AX، بعد تنفيذ الكود AX ستحتوي على 04h و ذلك لأن:

05h = 101b

06h = 110b

04h = 100b

مثال 2:

MOV AX, 05h

(٤٥)

NEG	المتمم الثنائي	NEG D	$0 - D \rightarrow D$ $1 \rightarrow CF$	أعلام الحالة
-----	----------------	-------	---	--------------

تذكّر بالمتمم الثنائي :

إذا أردت الحصول على المتمم الثنائي للعدد 03F8h فاعمل ما يلي:

(1) تحويل هذا العدد إلى النظام الثنائي فيصبح 0000 0011 1111 1000

(2) أقلب الأصفار والوحيدات و الواحدات أصفاراً فينتج 1111 1100 0000 0111

(3) أضف واحد إلى الرقم الناتج فتحصل على المتمم الثنائي 1111 1100 = 03F8h -

أوامر الضرب و القسمة :

يتم تطبيق هذه التعليمات على الأعداد الثنائية و هذه التعليمات مبيّنة في الجدول التالي:

الكلمة المختارة	المتنى	الصيغة	الحماية	الأعلام المتأثرة
MUL	ضرب بدون إشارة	MUL S	$AL.S8 \rightarrow AX$ $AX.S16 \rightarrow DX, AX$	أعلام الحالة
DIV	تقسيم بدون إشارة	DIV S	$Q[AX/S8] \rightarrow AL$ $R[AX/S8] \rightarrow AH$ $Q[(DX,AX)/S16] \rightarrow AX$ $R[(DX,AX)/S16] \rightarrow DX$	أعلام الحالة

شكل (3-5)

(٤٨)

NOT:التعليمة

هذه أسهل تعليمة في العمليات المنطقية, يكون استعمالها كالتالي:

NOT REGISTER

هذه التعليمة تقوم بكل بساطة بعكس قيم البتات

مثال

MOV AX,F0h

NOT AX

النتيجة هي 0 Fh و تخزين في المسجل:AX:

F0h = 11110000b

00001111b = 0Fh

تعليقات مسجل الأعلام

و هي مبينة في الجدول التالي :

الكلمة المختارة	المعنى	الصيغة	العملية	الأعلام المتأثرة
LAHF	تحميل AH من مسجل الأعلام	LAHF	Flags ← AH النصف الأول من مسجل الأعلام يوضع في AH	لا يوجد
SAHF	تخزين قيمة AH في مسجل الأعلام	SAHF	AH → Flags يوضع AH في النصف الأول من مسجل الأعلام	أعلام الحالة عدا OF

(٤٧)

AND AX, 06h

ج سيكون مثل المثال الأول و النتيجة ستخزن في المسجل AX

OR:التعليمة

تعمل:

OR REGISTER1, REGISTER2

OR REGISTER, VAL

التعليمة ترجع 1 (TRUE) إلا عندما يكون بت من البيتين يحتوي على 1

مثال:

MOV AX, 05h

MOV BX, 06h

OR AX,BX

يجة تخزين في AX, وفي هذا المثال ستكون AX=07h و ذلك لأن:

05h = 10

06h = 11

XOR:التعليمة

تعمل:

XOR REGISTER1, REGISTER2

XOR REGISTER, VAL

مثال:

MOV AX,05h

MOV BX,06h

XOR AX,BX

د تنفيذ التعليمة XOR في المثال سيحتوي المسجل AX على 03 h و ذلك لأن:

05h = 101

06h = 110

011b = 03

(٥٠)

أوامر المكسدة

أوامر الدفع و السحب PUSH و Pull
إن التعليمة المستخدمة لحفظ البيانات في المكسدة هي تعليمة الدفع PUSH و التعليمة المستخدمة لاسترجاعها هي تعليمة Pull .

هاتين التعليمتين هم أغلب التعليمات استعمالاً في عمليات المكسدة، و يكون استعمالهم كالتالي:

PUSH REGISTER
Pull REGISTER

هذه التعليمة تأخذ معامل واحد، هذا المعامل يجب أن يكون حجمه 16 Bits لنقل أننا نريد تخزين قيمة المسجل X بشكل مؤقت في المكسدة، ننفذ التالي:

SH AX

الآن نغير قيمة المسجل AX كما نشاء و عندما نريد استرجاع القيمة الأصلية ننفذ التالي:

I AX

SH AL

الكود التالي خاطئ:

الكود خاطئ لأن معامل التعليمة PUSH يجب أن يكون حجمه 16 Bit .

Jump instructions أوامر القفز

*** تعليمات القفز

الغاية من تعليمة القفز هي تعديل طريق تنفيذ التعليمات في البرنامج. و هناك نوعان من تعليمات القفز، وهي :
المشروط و القفز غير المشروط. في القفز غير المشروط لا يوجد أي شروط من أجل حدوث القفز أما في القفز المشروط فإن الحالات الشرطية الموجودة في لحظة تنفيذ تعليمة القفز تتخذ القرار فيما إذا سيحدث القفز أم لا، و حال تحقق الحالات الشرطية فإنه يتم القفز، و إلا يتابع التنفيذ بالتعليمة التي تلي تعليمة القفز في البرنامج.

(٤٩)

تعليمات تتعامل مع BIT في مسجل الالام

CLC	تنظيف الـ CF	CLC	$0 \rightarrow CF$	CF
STC	توضيع الـ CF	STC	$1 \rightarrow CF$	CF
CMC	متعم أحادي لـ CF	CMC	$CF \rightarrow CF$	CF
CLI	تنظيف الـ IF	CLI	$0 \rightarrow IF$	IF
STI	توضيع الـ IF	STI	$1 \rightarrow IF$	IF

شكل (3-36)

تعليمات المقارنة

تسمح تعليمة المقارنة CMP بمقارنة عددين بـ 8 بت أو 16 بت و هي مشروحة بالجدول التالي:

العلام المتأثرة	العملية	الصفة	المعنى	الكلمة المختزلة
العلام الحالة	D - S تأثر الأعلام	CMP D,S	مقارنة عددين	CMP

شكل (3-37)

تجري عملية الطرح ضمناً دون تخزين نتيجتها في متحول الهدف D (أي تبقى كلاً من محتويات المصدر S و محتويات الهدف D على حالها) و تستعمل هذه التعليمة لجعل أعلام الحالة تأخذ قيمة واحد منطقي أو صفر منطقي. ويتأثر العلم Z علامة الصفر

Input/output instructions أوامر الإدخال والإخراج

الامر	وظيفته
IN	لنقل البايت أو الكلمة من مدخل معين إلى المرمك
OUT	لنقل البايت أو الكلمة من المرمك إلى المخرج

شكل (3-38)

(٥١)

١) تعليمة القفز غير المشروط :

و هي مشروحة في الجدول التالي:

الأعلام المتأثرة	العملية	الصيغة	المعنى	الكلمة المختارة
لا يوجد	القفز إلى العنوان المحدد بواسطة المتحول operand	JMP operand	قفز غير مشروط	JMP

شكل (3-9)

٢) تعليمة القفز المشروط

و هي مشروحة في الجدول التالي:

الأعلام المتأثرة	العملية	الصيغة	المعنى	الكلمة المختارة
لا يوجد	فإنه يتم القفز إلى العنوان cc إذا تحقق الشرط المحدد بواسطة المتحول و إلا فيتم تنفيذ التعليمة التالية لتعليمة القفز	Jcc متحول	قفز مشروط	Jcc

شكل (3-10)

بعض تعليمات القفز المشروط و هي مشروحة في الجدول التالي:

وصف الأمر	أمر القفز
اقفز إذا كانت النتيجة فوق الصفر	JA
اقفز إذا كانت النتيجة فوق الصفر أو تساوى صفر	JAE
اقفز إذا كانت النتيجة تحت الصفر	JB
اقفز إذا كانت النتيجة تحت الصفر أو تساويه	JBE
اقفز إذا كانت النتيجة تساوى الصفر	JE/JZ
اقفز إذا لم يكن هناك حمل	JNC
اقفز إذا كانت النتيجة لا تساوى الصفر	JNE

(٥٢)

مثال:

اكتب برنامج يقرأ محتويات البايت (A830) باستمرار ثم يختبر هذه المحتويات بحيث اذا كانت صفرا يضع واحد في المسجل B وإذا كانت سالبة يضع اثنين في المسجل B وإذا كانت موجبة يضع أربعة في نفس المسجل

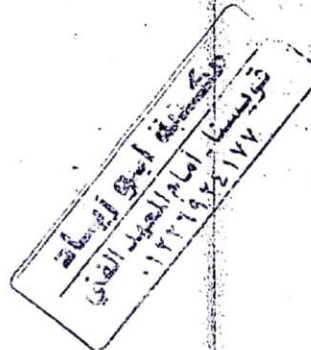
اولا يجب جمع محتويات هذه البايت على المرمك المحمل بالصفر لتغيير الاعلام Sf&Zf

نفرض بداية البرنامج من العنوان E001H

شفرة الاسمبلى

العناوين

E001	E002	E003	MOV Ax, 0000H
E004	E005	E006	ADD Ax, (A830H)
E007	E008	E009	JPNZ (E00FH)
E00A	E00B		MOV B,01 H
E00C	E00D	E00E	JP (E001 H)
E00F	E010	E011	JPP (E017H)
E012	E013		MOV B,02 H
E014	E015	E016	JP (E001H)
E017	E018		MOV B,04 H
E019	E01A	E01B	JP (E001H)



(٥١)

تعليمية القفز غير المشروط :

هي مشروحة في الجدول التالي:

الأعلام المتأثرة	العملية	الصيغة	المعنى	الكلمة المختزلة
لا يوجد	القفز إلى العنوان المحدد بواسطة المتحول operand	JMP operand	قفز غير مشروط	JMP

شكل (3-9)

(تعليمية القفز المشروط :

هي مشروحة في الجدول التالي:

الأعلام المتأثرة	العملية	الصيغة	المعنى	الكلمة المختزلة
لا يوجد	فإنه يتم القفز إلى العنوان cc إذا تحقق الشرط المحدد بواسطة المتحول و إلا فيتم تنفيذ التعليمية التالية لتعليمية القفز	Jcc متحول	قفز مشروط	Jcc

شكل (3-10)

بعض تعليمات القفز المشروط و هي مشروحة في الجدول التالي:

وصف الأمر	أمر القفز
اقفز إذا كانت النتيجة فوق الصفر	JA
اقفز إذا كانت النتيجة فوق الصفر أو تساوى صفر	JAE
اقفز إذا كانت النتيجة تحت الصفر	JB
اقفز إذا كانت النتيجة تحت الصفر أو تساويه	JBE
اقفز إذا كانت النتيجة تساوى الصفر	JE/JZ
اقفز إذا لم يكن هناك حمل	JNC
اقفز إذا كانت النتيجة لا تساوى الصفر	JNE

(٥٢)

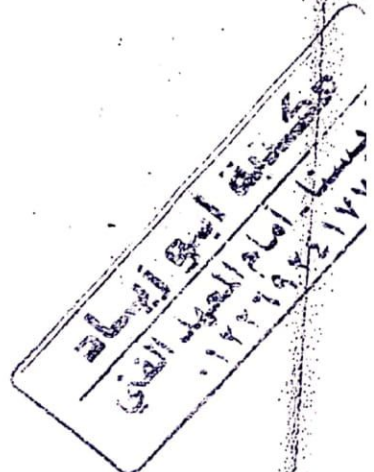
مثال:

اكتب برنامج يقرأ محتويات البايت (A830) باستمرار ثم يختبر هذه المحتويات بحيث اذا كانت صفرا يضع واحد في المسجل B واذا كانت سالبة يضع اثنين في المسجل B واذا كانت موجبة يضع اربعة في نفس المسجل

اولا يجب جمع محتويات هذه البايت على المرمك المحمل بالصفر لتغيير الاعلام Sf&Zf

نفرض بداية البرنامج من العنوان E001H

العناوين	شفرة الاسمبلى
E001 E002 E003	MOV Ax , 0000H
E004 E005 E006	ADD Ax, (A830H)
E007 E008 E009	JPNZ (E00FH)
E00A E00B	MOV B,01 H
E00C E00D E00E	JP (E 001 H)
E00F E010 E011	JPP (E017H)
E012 E013	MOV 'B,02 H
E014 E015 E016	JP (E00 1H)
E017 E018	MOV B,04 H
E019 E01A E01B	JP (E00 1H)



(٥٤)

SUB AL, CL
HLT

MOV AH, 4
MOV AL, 4
CMP AH, AL
HLT

MOV AH, 4
MOV AL, 3
CMP AH, AL
HLT

يتغير العلم Z إذا كان الرقمين متساويين يحمل ب (1) وإذا كانا مختلفين يحمل ب (0)

مثال 5: أكتب برنامج لتحميل النواوين الآتية ومن ثم جمع محتوياتهم

[E10A], [E108], [E106], [E104], [E102] مع العلم أن محتويات العناوين بالترتيب هي:

(01, 02, 03, 04, 05) ثم يتم وضع الناتج النهائي للجمع في العنوان [E10B H]

MOV [E102H], 01H
MOV [E104H], 02H
MOV [E106H], 03H
MOV [E108H], 04H
MOV [E10AH], 05H
MOV AL, [E102H]
ADD AL, [E104H]

(٥٣)

أمثلة مختلفة للبرامج

حاول كتابة هذه البرامج بأكثر من طريقة

مثال 1: أكتب برنامج لجمع الرقمين 5 و 10 ويطرح من الناتج 1

الحل:

MOV AL, 5
MOV BL, 10
ADD AL, BL
SUB AL, 1
HLT

مثال 2: أكتب برنامج لجمع الرقمين 50 و 100 و يضع الناتج في العنوان 512

mov AL, 100
mov BL, 50
add AL, BL
mov [512H], AL
hlt



مثال 3: أكتب برنامج لتحميل الأرقام 00001010b و 0Ah و 10o بالمسجلات CL, BL, AL بالترتيب ثم قم بجمع محتويات BL, AL و طرَح الناتج من محتويات CL

MOV AL, 00001010 B
MOV BL, 0A h
MOV CL, 10 o
ADD AL, BL

(٥٦)
اسئلة على الباب الثالث

س1: اذكر ثلاثة من اواخر المنطق مع شرح وظيفتهم ؟

س2: اكتب الصيغ المختلفة لأوامر الانتقال ؟

س3: اذكر ثلاثة صيغ مختلفة من أوامر الجمع مع شرح وظيفتهم ؟ وكذلك اذكر ثلاثة صيغ مختلفة من أوامر الطرح مع شرح وظيفتهم ؟

س5: اذكر اثنين من أوامر الضرب مع شرح وظيفتهم ؟

س6: اذكر اثنين من أوامر القفز مع شرح وظيفتهم ؟

س7: اكتب برنامج لجمع العددين 23F0H و B080H مع وضع النتائج في الامكن التالية بالذاكرة بالترتيب E008, E004, E000 وذلك للمعالج 8086

س8: قارن بين تلمتي

أ- ADD&ADC

ب- SUB&SBB

ج- القفز المشروط والقفز الغير مشروط

س9: المطلوب عمل برنامج لتحميل المسجلات Ax, Bx, Cx بالمعلومات الفورية 1104H, 0096H, 08A3H وبعد ذلك يتم عمل اراحة دورانية ليذء المحتويات بداية البرنامج من العنوان E1A0

س10: اكتب برنامج يقرأ محتويات البايت (4000 H) باستمرار ثم يختبر هذه المحتويات بحيث اذا كانت صفرًا يضع اثنين في المسجل C واذا كانت سالبة يضع اربعة في المسجل B بداية البرنامج من العنوان (0200H)

(٥٥)

ADD AL, [E106H]

ADD AL, [E108H]

ADD AL, [E10AH]

MOV [E10B H], AL

HLT

نح 0FH

س6: اكتب برنامج لجمع العددين 1A50H و H5242 مع وضع النتائج في الامكن التالية بالذاكرة بالترتيب 512, 1024, 20, 8086

mov AL, 50H

mov AH, 1AH

mov BL, 42H

mov BH, 52H

ADD AL, BL

mov [2048H], AL

ADDC AH, BH

mov [1024H], AH

mov AL, 00H

ADDC AL, AL

mov [512H], AL

hlt

ملحظة هامة جدا: على الطلاب ان يدرك الفرق بين العناوين والبيانات التي تبدأ برقم او التي تبدأ بحرف عند كتابة برنامج

الباب الرابع

اساسيات مواجهة المعالج

فصل خطوط المعالج

بوابة ثنائية المنطق

بوابة ثلاثية المنطق

1: اكمل الجدول الاتي:-

الاعلام التي تتاثر	الصيغة	المعنى	الكلمة المختزلة
			ADD
			JP
		مقارنة عددين	
	AND D,S		
		تنظيف ال CF	

12: ماهي الحالات المستثناة من امر MOV

13: اكتب برنامج يجمع محتويات العناوين الاتية :

B000H, B002H, B004H, B006H ويضع الناتج في البايت التي عنوانها B00AH على اعتبار ان النتيجة تزيد عن بايت واحدة (لا يوجد مرحل على الاطلاق)

14: اكتب برنامج لجمع رقمين (34) و (87) العشري وخذن الناتج في مسجل C_x

15: كم عدد البايت التي يحتلها كود الاوامر الاتية (طول كل امر)

1. MOV AX, BX
2. MOV BL, CL
3. ADD AX, 3A40 H
4. SUB DL, 3FH
5. CL CF