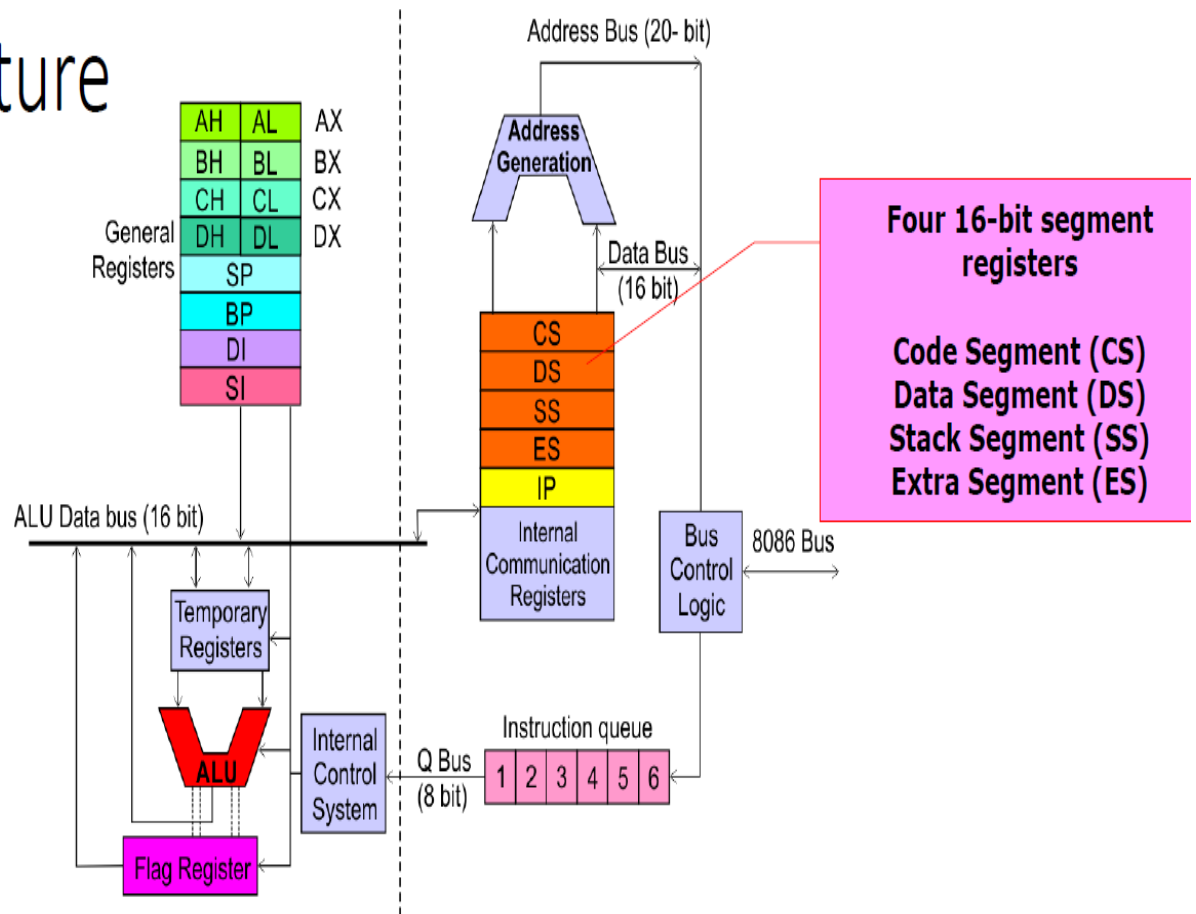


Architecture



تنظيم الحاسوب عملي

قبل البدء يجب معرفة انواع الرجسترات (Register) وكيفية تقسيمها

ملاحظة:- الرجسترات تستقبل كحد اقصى 16 بت باينري اي 4 بت من النظام السادس عشر

(1234H)

1 - (AX , BX , CX , DX) :- هذا النوع يستقبل 16 بت باينري اي 4 بت هكسا ولكن يقسمها الى نوعين :-

الاول Low ويرمز له L ويستقبل 8 بت باينري اي 2 بت هكسا التي على اليمين (AL او BL او CL او DL)

الثاني High ويرمز له H ويستقبل 8 بت باينري اي 2 بت هكسا التي على اليسار (AH او BH او CH او DH)

2 - (SI , DI , BP , SP) :- هذا النوع يسمى رجسترات الفهرسة ولا يتم تقسيمه

3 - (CS , DS , ES , SS) :- هذا النوع هو SEGMENT ولا يتم تقسيمه ايضاً

ملاحظة :- هنالك رجستر IP يشير الى التنفيذ الحالي و رجستر Flags تحديد حالة الحالية للمعالج

ملاحظة :- يجب كتابة نوع النظام للرقم مثلاً AH او 1010B في حالة عدم الكتابة فسيعتبر الرقم عشري decimal

وبالنسبة للنظام السادس عشر يجب كتابة 0 قبل A B C D E F لأنه في حالة اردنا تحميل قيمة عشرة في النظام السادس عشر نكتب AH هنا سيفهمها البرنامج على انها رجستر لهذا نكتب 0

Memory

يرمز للذاكرة بأسم الموقع بين اقواس مربعة

مثال [BX]

عند ادخال 4 بت من النظام السادس عشر (H) يتم تقسيمها الى 2 بت اي انها تشبه النوع الاول من الرجسترات (AX , BX , CX , DX)

الجزء الاول يأخذ 2 بت التي على اليمين ويرمز له [BX]

اما الجزء الثاني يأخذ **2 بت** التي على اليسار ويرمز له **بموقع الذاكرة + 1** اي انه يكتب بالشكل التالي **[BX + 1]**

ملاحظة :- يمكن ان يشار للموقع **[BX+2]** ففي هذه الحالة سيكون الجزء الثاني **[BX+3]**

مثال آخر الموقع **[BP+SI+10H]** نفس الشيء سيكون الجزء الثاني **[BP+SI+10H+1]**

Mov and XCHG

| MOV | XCHG |
|--------------------|--------------|
| SREG, MEMORY | REG , MEMORY |
| MEMORY, SREG | MEMORY , REG |
| REG, SREG | REG , REG |
| SREG, REG | |
| MEMORY , IMMEDIATE | |
| REG , IMMEDIATE | |
| REG , REG | |

REG: AX, BX, CX, DX, AH, AL, BL, BH, CH, CL, DH, DL, DI, SI, BP, SP.

SREG: DS, ES, SS, and only as second operand: CS.

memory: [BX]

Immediate : 1234H

Physical Address and logical address

Ex: DS contains value 1234h, SI contains the value 7890h

sol.

Logical address: 1234:7890

Physical address = (Segment base*10H) + Offset Value.

$$= 1234h * 10h + 7890h = 19BD0h$$

Variables

تعريف المتغيرات يقسم الى :-

DB وتعني Define Byte ويتم فيها حفظ خلية واحدة فقط في الذاكرة لمتغير والحجم الكلي 1Byte

DW وتعني Define Word ويتم فيها حفظ خليتان في الذاكرة لمتغير والحجم الكلي 2Byte

و تكون طريقة التعريف بهذا الشكل ((القيمة DB اسم المتغير))

ملاحظة :- التعريفات تكون بعد كلمة **RET**

Array

تعريف المصفوفات يتم بالشكل الآتي القيمة , القيمة , القيمة DB اسم المصفوفة

ويتم الوصول الى عناصر المصفوفة عن طريق الاقواس المربعة اي مشابهة لطريقة البرمجة في السي شارب او البرامج الاخرى

مثال //

هنا تم تعريف مصفوفة وتم حفظ العناصر a DB 5h,2h,6h,1h

a[0]=5

a[1] =2

a[2]=6

a[3]=1

يمكن استخدام **DUP** لفتح مصفوفة كبيرة تحمل عناصر متكررة وتتم بالشكل الآتي

(الرقم المراد تكراره) **DUP** عدد التكرارات **DB** اسم المصفوفة

// مثال

a DB 4 DUP (8)

هنا سيكون الترتيب بالشكل التالي $a[0] = 8$ $a[1] = 8$ $a[2] = 8$ $a[3] = 8$

اي انه تم تكرار رقم 8 اربعة مرات

يمكن حفظ ارقام في متغيرات عن طريق امر **EQU**

مثال $C EQU 5$ وهنا يتم التعريف قبل **RET**

| الايعازات الحسابية |
|-----------------------------|
| ADD,SUB,CMP,AND,TEST,OR,XOR |
| REG , Memory |
| Memory , REG |
| REG , REG |
| Memory , immediate |
| REG , immediate |

هذه الايعازات تؤثر على الاعلام الآتية : CF , ZF , SF , OF , PF , AF

ADD

هذا الايعاز يؤثر على الاعلام الآتية : CF , ZF , SF , OF , AF

يستخدم هذا الايعاز للجمع

// مثال

MOV AL,23 تحميل قيمة **23H** في الرجستر **AL**

MOV DH,42H تحميل قيمة **42H** في **DH**

ADD DH,AL جمع قيمة AL و DH و تحميل الناتج وحفظه في DH

SUB

هذا الايعاز يؤثر على الاعلام الآتية: CF , ZF , SF , OF , AF

يستخدم هذا الايعاز للطرح

مثال //

MOV BX,44H تحميل قيمة 44H في الرجستر BX

MOV CX,42H تحميل قيمة 42H في الرجستر CX

SUB BX,CX طرح قيمة CX و BX وتحميل الناتج وحفظه في BX

ملاحظة : لايمكن جمع او طرح جزء رجستر من رجستر كلي اي مثلاً DH و CX او العكس الا اذا كان الاثنان مقسمين اي DL و CL او CH او الاثنان CX و BX او AX