

# كيفية القرائه والكتابه على الذاكره الدائمه للمتحم الصغير ٢ اف ٨٤ أ

بقلم : حسن علي .

## ١ الكتبه :

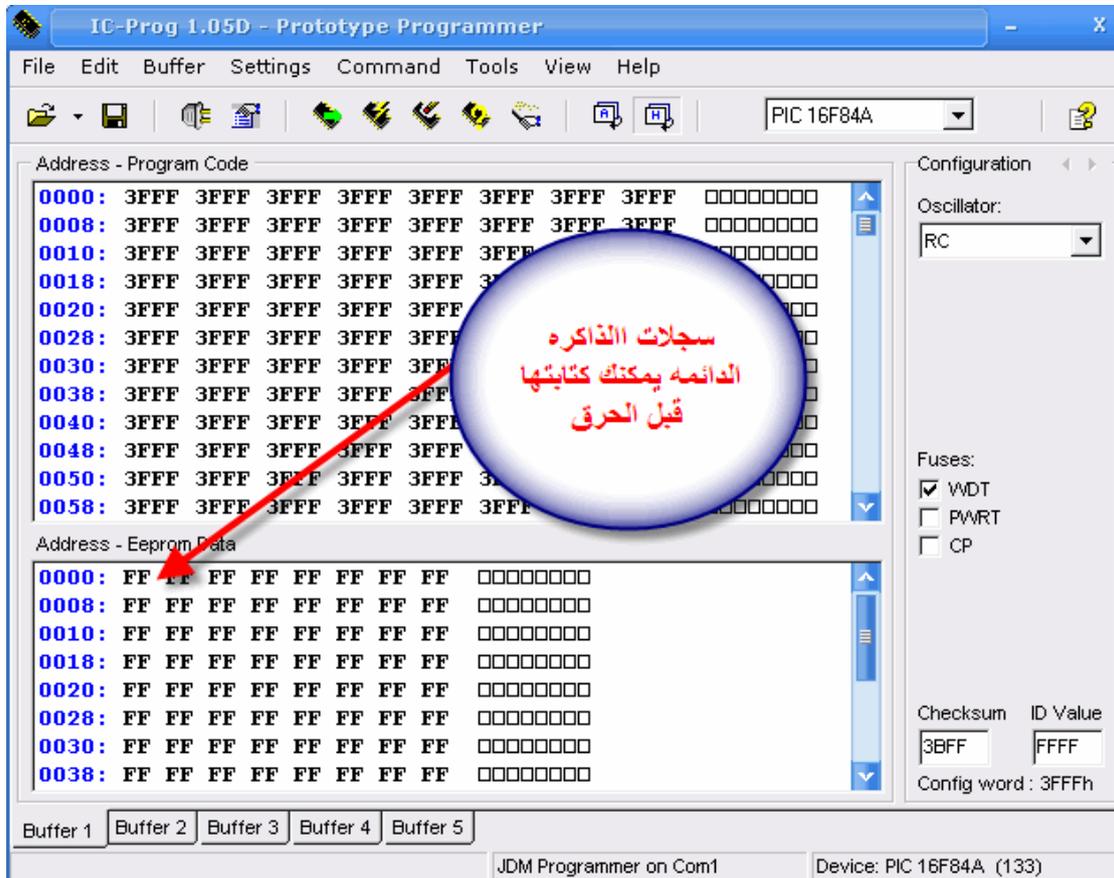
- بواسطه البرنامج الحارق .
- بعد تركيب متحكمك في الدائره التي صمم لأجلها .

## ٢ القرائه :

- بواسطه البرنامج الحارق
- بعد تركيب متحكمك في الدائره التي صمم لأجلها .

# الكتابة بواسطة برنامج الحرق

المقصود هنا أنك كتبت مشروعك ، وأصبح بصيغة هيكس Hex وتحاول حرقه بأي من البرامج المخصصة لذلك . فلوجة المتحكم -Picall- ICprog كلها تظهر مربع خاص لسجلات الذاكرة الدائمه ، يمكنك تعبأته قبل إعطاء أمر الحرق . مثلا ....



غير كما تريد ثم إحرق ..

## الكتابة بعد تركيب متحكمك في الدائره التي صمم لأجلها

المقصود هنا أن المتحكم وضع في الدائره النهائيه ، ونريده أن يحفظ في الذاكره الدائمه أرقام ... يقرأها أو تدخل عليه من الأطراف الخارجيه ... ككلمة سر مثلا ...

الذاكره الدائمه: مكونه من 64 سجل ، "مسجل" ، أو بايت .... ولكل واحد عنوان من صفر ولغاية 63 عشريا .... أو من صفر ولغاية 3F سادس عشري . . . . .  
 كتابة أو قراءة أي بايت منهم ، يستدعي إستخدام سجلات العمل الخاصه " SFR المخصصه لذلك وهي أربعه .

TABLE 3-1: REGISTERS/BITS ASSOCIATED WITH DATA EEPROM

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
08h	EEDATA	EEPROM Data Register							
09h	EEADR	EEPROM Address Register							
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD
89h	EECON2	EEPROM Control Register 2							

هذه السجلات موجوده في الذاكره المؤقته " RAM ووضيفتها إدارة الذاكره الدائمه "EEProm" ....

١. EEDATA "المعلومه"
٢. EEADR "العنوان"
٣. EECON1 "تحكم 1"
٤. EECON2 "تحكم 2"

سجل المعلومه "EEDATA" نضع به "المعلومه" أو البايث أو الرقم ، الذي نريد كتابته على الدائمه .

سجل "العنوان" "EEADR" نضع به عنوان "السجل" حيث نريد تخزين المعلومه ، تذكر ، عندك من صفر لـ 63 .

سجل "تحكم 1" "EECON1" : مكون من خمس بيت على الشكل التالي :

- **RD** قرائه : عندما نريد قراءة موضع من الذاكره ، هنا نصدر الأمر بذلك ، بوضع واحد منطقي في البيت " RD ".... عندما تتم القرائه ، لست بحاجة لوضعه "صفر" برمجيا .... المتحكم يفعل ذلك تلقائيا .
- **WR** كتابه : عندما نريد كتابة موضع من الذاكره ، هنا نصدر الأمر بذلك ، بوضع واحد منطقي في البيت " WR ".... عندما تتم الكتابه، لست بحاجة لوضعه "صفر" برمجيا .... المتحكم يفعل ذلك تلقائيا .
- **WREN** تفعيل الكتابه : هنا نعطي تصريح أو نجيز أو نسمح بالكتابه على الذاكره الدائمه ... بوضعه واحد .... طيب.... ما الهدف من هذا "البيت" ؟ أن لا تحصل كتابات عفويه أو بالغلط . وعلينا وضعه "صفر" بعد الإنتهاء من الكتابه برمجيا .... لا يتغير تلقائيا .
- **WRERR** "علم خطأ" : يصبح "واحد" عندما تنتهي الكتابه قبل وقتها المحدد .... ( أي عملية إعادة إنظباط ...إخ). ويصبح صفر عندما تنتهي الكتابه بخير وكما يجب .
- **EEIF** "علم مقاطعة عملية الكتابه" ....يصبح "واحد" بعد أن تكتمل الكتابه ... ويجب محيه برمجيا .... و"صفر" لم تبدأ عملية الكتابه أو لم تنتهي بعد.

## كانت هذه معلومات أوليه والكتابه تبدأ بالخطوات التاليه :

١. تحميل الرقم "قيمة البايت" في سجل " المعلومه" EEDATA
٢. تحميل عنوان السجل الذي نريد الكتابه عليه في "العنوان" EEADR
٣. نبطل عمل جميع المقاطعات .
٤. نضع "واحد" في سجل "WREN" — وبذلك نسمح بالكتابه . نفعلها.
٥. نكتب الأسطر الإجباريه.
٦. نفعل المقاطعات مجددا .

وإنتهى الموضوع .

الأسطر الإجباريه وسجل "تحكم 2 " EECON 2 : هذا السجل ليس كباقي السجلات في المتحكم ، إذا حاولت قرائته فيظهر كـ "صفر" وله إستعمال واحد فقط إجباري عند الكتابه على الدائمه . ولأن هذه العملبه تستغرق 10 ميلي ثانيه . وسرعة المتحكم أكثر من ذلك . فهو الذي يحفظ التوازن بإدخال رقمين عليه ... خلال الكتابه ... وهذان السطرين البرمجين ...يسمان بالأسطر الإجباريه ....

لنرى ما تقوله Microchip عن الموضوع ....

### 3.2 Writing to the EEPROM Data Memory

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific sequence to initiate the write for each byte.

#### EXAMPLE 3-2: DATA EEPROM WRITE

	BSF	STATUS, RPO	; Bank 1
	BCF	INTCON, GIE	; Disable INTs.
	BSF	EECON1, WREN	; Enable Write
	MOVLW	55h	;
Required Sequence	MOVWF	EECON2	; Write 55h
	MOVLW	AAh	;
	MOVWF	EECON2	; Write AAh
	BSF	EECON1, WR	; Set WR bit
			; begin write
	BSF	INTCON, GIE	; Enable INTs.

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

١. تحميل الرقم "قيمة البايت" في سجل "المعلومة" EEDATA
٢. تحميل عنوان السجل الذي نريد الكتابة عليه في "العنوان" EEADR
٣. نبطل عمل جميع المقاطعات .
٤. نضع "واحد" في سجل "WREN" — وبذلك نسمح بالكتابة . نفعليها.
٥. نكتب الأسطر الإجباريه.
٦. نفعلي المقاطعات مجددا .

دعنا من الإنكليزيه ، وها هو حسن علي مجددا في قرينه يتسائل ... لو كنت تكتب برنامج تستخدم به الذاكره الدائمه كيف ستفعل ؟

الأمر سهل أولا : أستخدم سجلان عمل عام GPR الأول أسميه " نسخة معلومه" والثاني " نسخة عنوان " .

ثم أضع في برنامجي "مصغر" Subrutine أسميه كتابه .

احمل نسخة معلومه " و " نسخة عنوان " بما أريد ثم أستدعي مصغر كتابه  
وإنتهى الموضوع . ما أقوله لأصدقائي في الأم بي لاب يترجم ك :

**kitabah:**

```
bcf          STATUS,RP0
movlw       NaskahInwan
movwf      EEADR
movlw       NaskahMa3
movwf      EEDATA
bsf         STATUS,RP0
bcf         INTCON,GIE
bsf         EECON1,WREN
movlw       0x55
movwf      EECON2
movlw       0xaa ;
movwf      EECON2
bsf         EECON1,WR
```

**Intathir:**

```
Btfsc      EECON1,WR
goto       Intathir
bcf        EECON1,WREN ;
bsf        INTCON,GIE
bcf        STATUS,RP0
```

**Return**