المعهد التقاني للهندسة الميكانيكية والكهربائية بدمشق

سنة ثانية – اختصاص تحكم وأتمتة

مقـــدمـــة في متحكمـات الصغرية

PIC 16F84

PIC (Peripheral interface Controller)

2019-2020

اعداد م.ماهر معروف

الفصل الثاني: المتحكم الصغري PIC16F84

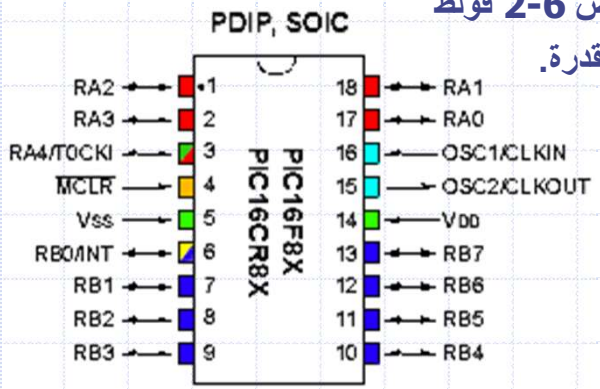
CHAPTER II: MICROCONTROLLER PIC16F84

2-1مقـدمة:

ينتمي المتحكم PIC16F84 إلى عائلة المتحكمات ذات بنية RISC ثمانية الخانات.

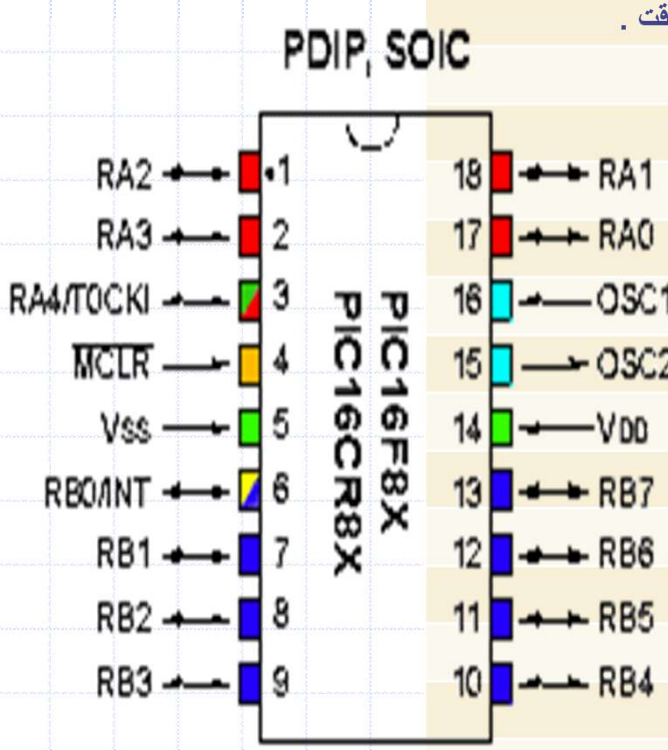
ويتمتع بالمواصفات والميزات التالية :

- خمس وثلاثون تعليمة فقط.
- تردد الساعة الاعظمي 10MHz.
- ذاكرة البرنامج 1024 تعليمة.
- ذاكرة المعطيات (RAM) 68 بايت.
- ذاكرة المعطيات (EEPROM) 64 بايت.
- عرض التعليمة 14 خانة.
- خط المعطيات 8 خانات.
- عدد السجلات الخاصة 15.
- ثمانية مستويات للمكّدس.
- ثلاث وضعيات عنونة: مباشرة وغير مباشرة ونسبية.
- أربع مصادر للمقاطعة.
- قابلية البرمجة والمحي 1000 مرة.
- قابلية الكتابة على EEPROM عشرة ملايين مرة.
- القدرة على الاحتفاظ بالمعطيات مخزَّنة أربعون عاماً.
- 13 خط دخل/خرج.
- تيار خرج عالي : السحب (25mA) الإعطاء (20mA).
- مؤقت/عدّاد 8 خانات مع مقسم داخلي 8 درجات.
- برمجة المتحكم ضمن الدارة باستخدام قطبين فقط.
- تصفيرعند بدء التغذية POR وعن طريق المؤقت PWRT.
- مؤقت إقلاع المذبذب OST.
- مذبذب داخلي مستقل لمؤقت المراقبة.
- إمكانية حماية المعطيات والبرنامج.
- وضعية توفير الطاقة SLEEP.
- مجال تغذية عريض 2-6 فولط
- استهلاك ضئيل للقدرة.



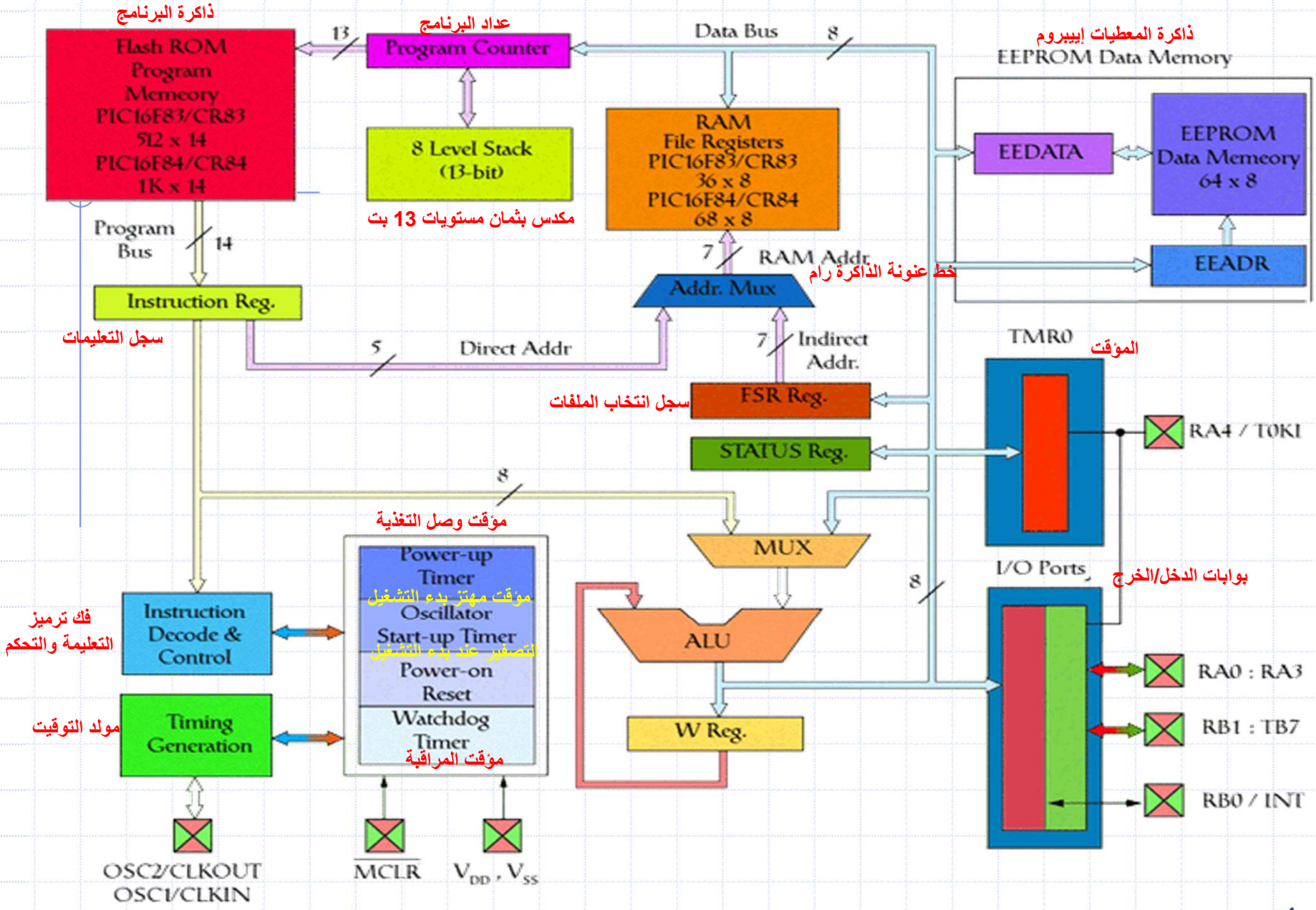
توزع أطراف المتحكم PIC16F84:

يملك المتحكم **PIC16F84** ثماني عشرة طرفاً, ويتوفر بالشكل الإعتيادي **DIP** أو بالحجم الصغير **SMD**.



رقم الطرف	الرمز	الوظيفة
1	**RA2**	القطب الثاني للبوابة **A**. ليس له وظيفة إضافية.
2	**RA3**	القطب الثالث للبوابة **A**. ليس له وظيفة إضافية.
3	**RA4**	القطب الرابع للبوابة **A** / مدخل **TOCK1** يعمل كمدخل ساعة للمؤقت .
4	**MCLR**	مدخل التصفير/ مدخل جهد البرمجة **Vpp** للمتحكم.
5	**VSS**	مدخل أرضي التغذية.
6	**RB0**	القطب صفر للبوابة **B**/ مدخل المقاطعة الخارجية.
7	**RB1**	القطب واحد للبوابة **B**. ليس له وظيفة إضافية.
8	**RB2**	القطب إثنان للبوابة **B**. ليس له وظيفة إضافية.
9	**RB3**	القطب ثلاثة للبوابة **B**. ليس له وظيفة إضافية.
10	**RB4**	القطب أربعة للبوابة **B**. ليس له وظيفة إضافية.
11	**RB5**	القطب خمسة للبوابة **B**. ليس له وظيفة إضافية.
12	**RB6**	القطب ستة للبوابة **B** / مدخل نبضات الساعة في وضعية البرمجة.
13	**RB7**	القطب سبعة للبوابة **B** / مدخل المعطيات في وضعية البرمجة.
14	**Vdd**	مدخل التغذية الموجب.
15	**OSC2**	للتوصيل مع المذبذب الخارجي
16	**OSC1**	للتوصيل مع المذبذب الخارجي
17	**RA2**	القطب الثاني للبوابة **A**. ليس له وظيفة إضافية.
18	**RA1**	القطب الأول للبوابة **A**. ليس له وظيفة إضافية.

المخطط الصندوقي لمكونات المتحكم PIC16F84



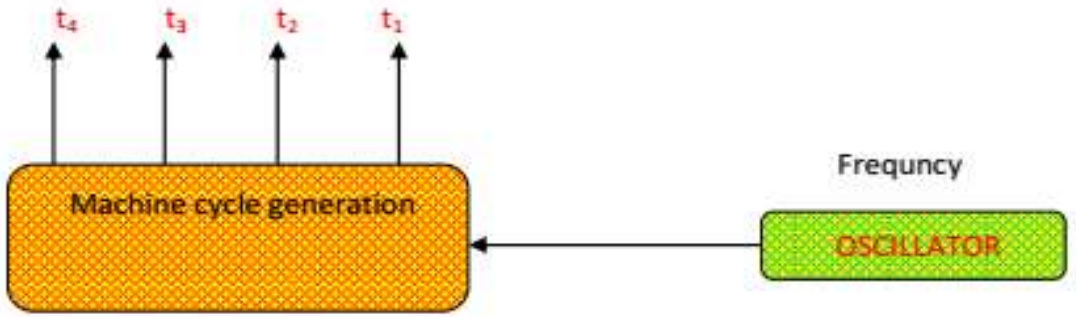
Control unit

وحدة السيطرة

السؤال الذي يتبادر الى الذهن ما هي الدائرة المسؤلة عن ؟

- نقل محتويات العداد PC الى السجل الخاص بتحديد عنوان الذاكرى memory والذي هو السجل MAR(memory address register)
- نقل محتويات الذاكرة Memory الى سجل التعليمة IR
- زيادة محتوى العداد PC وذلك لكي يؤشر الى الموقع التالي من الذاكرة next location of memory
- فك شفرة التعلمية وتنفيذها execution and decoding في وحدة الحساب والمنطق او أي وحدة اخرى ،

في الحقيقة الوحدة المسؤلة عن هذه العمليات هي وحدة السيطرة Control Unit ، واحياناً تسمى وحدة توليد دورة الماكنة Machine cycle generation ، وحدة السيطرة ترسل اشارات سيطرة Control Siginal الى باقي وحدات وحدة المعالجة المركزية CPU ، وذلك لكي تضمن التزامن Synchronization في عمل هذه الوحدات الشكل Figure 2.5 يبين مخطط لوحدة توليد دورة الماكنة،



وحدة السيطرة(وحدة توليد دورة الماكنة) تحتي على اربع اشارات Siginal أو أكثر حسب نوع المعالج Processor ، هذه الاشارات تسمى t_1,t_2,t_3,t_4 ،عند اول نبظة للمذبذب Oscillator، فأن وحدة السيطرة ترسل اشارة فقط عبر t_1 اما بقية الاشارات تكون غير مفعلة ، تستخدم هذه الاشارة لنقل محتويات عداد البرنامج PC الى وحدة الذاكرة Memory ، عندما تأتي النبظة الثانية من المذبذب فأن وحدة السيطرة ترسل اشارة فقط عبر t_2 كذلك بقية الاشارات تكون غير مفعلة ،تستخدم هذه الاشارة لنقل محتويات وحدة الذاكرة memory الى سجل التعليمة ، عندما تأتي النبظة الثالثة من المذبذب فأن وحدة السيطرة ترسل اشارة فقط عبر t_3 وتكون بقية الاشارات غير مفعلة ،تستخدم هذه الاشارة لزيادة محتويات عداد البرنامج pc بواحد ولذلك لكي يؤشر الى الموقع التالي من الذاكرة ، وسبب زيادة عداد البرنامج خلال الفترة t_3 هو عندما تأتي t_1 خلال الدورة التالية الجديدة ، فانة العداد PC سيكون مؤشر الى الموقع التالي و حيث أن t_1 تقوم بنقل محتويات العداد PC الى موقع الذاكرة memory لكي يتم تحديد موقع الذاكرة الذي عنوانة هو محتويات العداد PC، عندما تأتي النبظة الرابعة من المذبذب فأن وحدة السيطرة ترسل اشارة فقط عبر t_4 وتكون بقية الاشارات غير مفعلة ،تستخدم هذه الفترة لفك شفرة التعليمة وتنفيذها وحدة السيطرة هذه مبسطة في الحقيقة تحتاج فك شفرة التعليمة وتنفيذها اكثر من فترة زمنية t ، عندما تأتي النبظة الخامسة من المذبذب فأن وحدة السيطرة ترسل اشارة فقط عبر t_1 وبقية الاشارات غير مفعلة ، أي تعاد الدورة من جديد ، أذن نستنتج من ذلك ان كل اربع نبظات من المذبذب تنفذ تعليمة كاملة ، وكلما زادت سرعة المذبذب زادت سرعة تعاقب الفترات الزمنية t ، الجدول Table 2.1 يوضح فترات time وحدة السيطرة،

الحالة	العملية خلال الفترة الزمنية المفردة T	T_4	T_3	T_2	T_1	المذبذب
دورة ماكنة الاولى ، نتيجتها تنفيذ تعليمة كاملة	نقل محتويات عداد البرنامج الى وحدة الذاكرة	0	0	0	1	النبضة ١
	نقل نقل محتويات الذاكرة الى سجل التعليمه	0	0	1	0	النبضة ٢
	زيادة محتويات عداد البرنامج بواحد	0	1	0	0	النبضة ٣
	فك شفرة التعليمة وتنفيذها	1	0	0	0	النبضة ٤
دورة ماكنة الثانية ، نتيجتها تنفيذ تعليمة كاملة	نقل محتويات عداد البرنامج الى وحدة الذاكرة	0	0	0	1	النبضة ٥
	نقل نقل محتويات الذاكرة الى سجل التعليمه	0	0	1	0	النبضة ٦
	زيادة محتويات عداد البرنامج بواحد	0	1	0	0	النبضة ٧
	فك شفرة التعليمة وتنفيذها	1	0	0	0	النبضة ٨
						ما لا نهاية

الشكل Figure 2.6 يوضح المخطط الصندوقي نموذجي مبسط للمعالج CPU ، أما الشكل Figure 2.7 يوضح المخطط الصندوقي مفصل

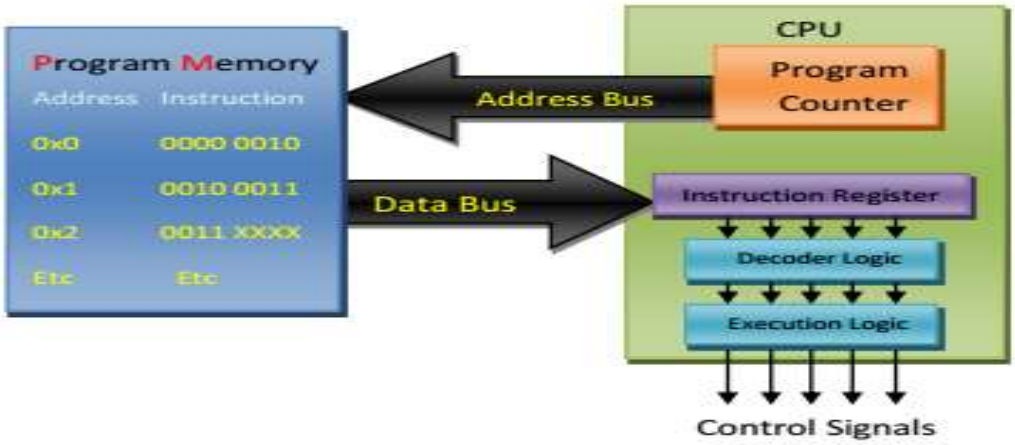


Figure 2.6

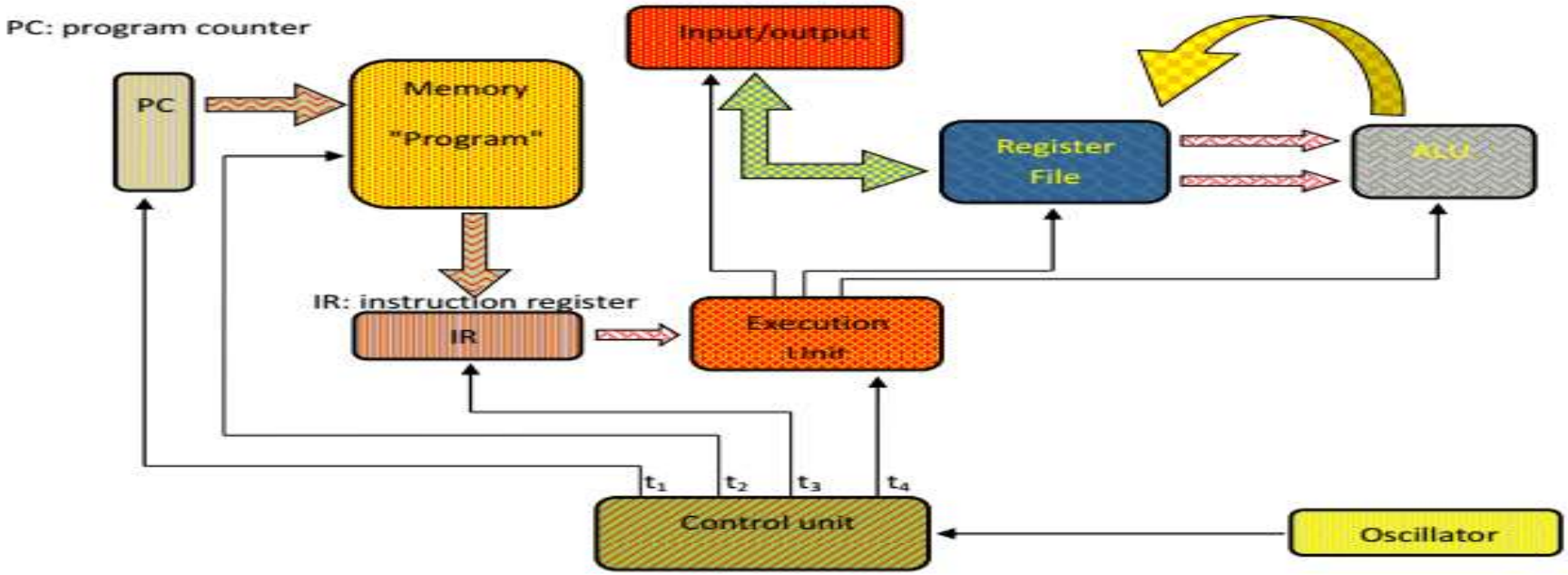
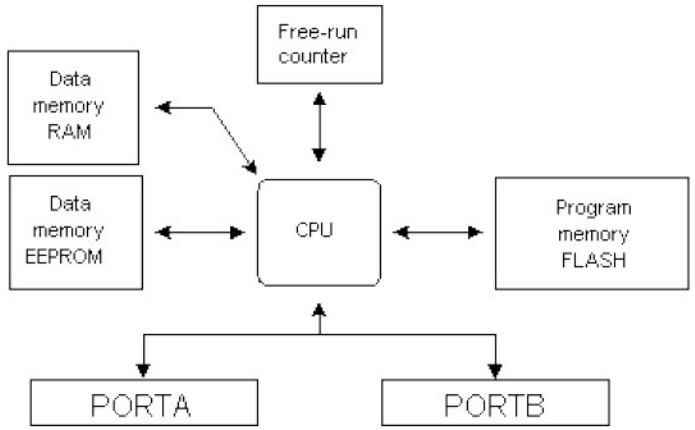


Figure 2.7

يبين الشكل التالي الكتل الأساسية المكونة له وهي :



PIC16F84 microcontroller outline

ذاكرة البرنامج (فلاش)Program Memory:
من أجل تخزين البرنامج المكتوب. وحيث أنها مصنعة بتقنية الفلاش فإنه يمكن برمجتها ومحيها لمرات كثيرة مما يجعل المتحكم مناسب من أجل تطوير الجهاز المركب ضمنه.

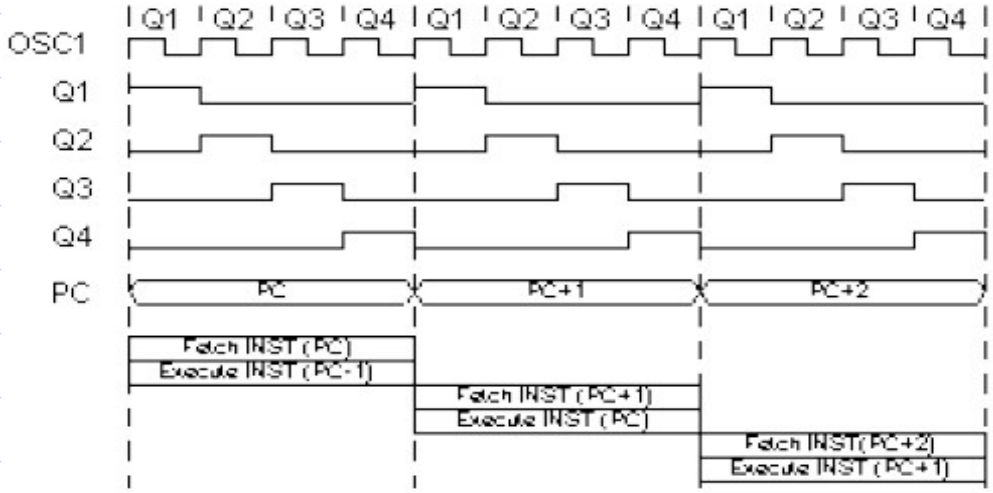
ذاكرة المعطيات EEPROM : تستخدم لتخزين المعطيات الهامة التي يجب أن نحتفظ بها عند انقطاع التغذية (على سبيل المثال: درجة الحرارة المرجعية في جهاز تنظيم الحرارة, فإذا فقدت فإنه يتوجب علينا إعادة ضبطها يدوياً عند عودة التغذية وبالتالي فقد الجهاز اعتماده الذاتي على نفسه.

ذاكرة RAM : وهي ذاكرة المعطيات التي تستخدم من قبل البرنامج أثناء تنفيذه. يتم تخزين جميع نواتج العمليات والمعلومات المؤقتة أثناء العمل.

البوابتان PORT A و PORT B: وهي التوصيلات الفيزيائية بين المتحكم والعالم الخارجي. البوابة A تملك خمسة أطراف والبوابة B تملك ثمانية أطراف.

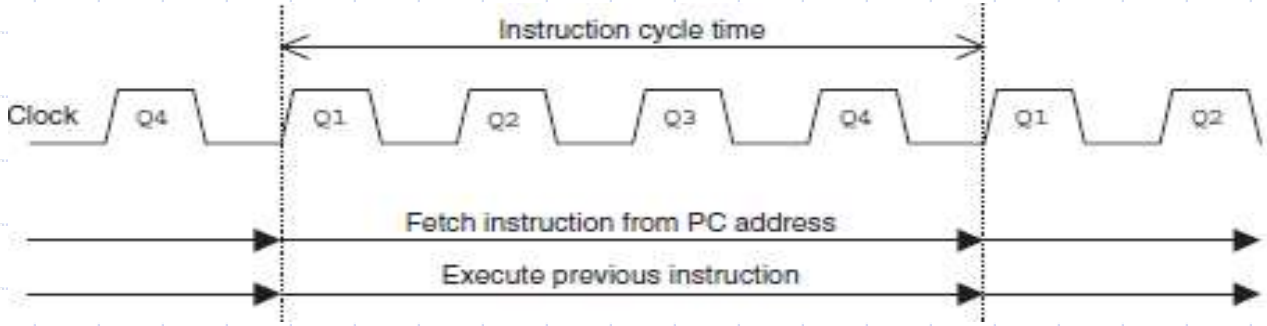
نبضات الساعة / دورة التعليمة:

نبضات الساعة هي المحرك الأساس للمتحكم, ويتم الحصول عليها من عنصر خارجي يدعى المهتز. تدخل نبضات الساعة من المهتز إلى المتحكم عبر القطب OSC1, حيث تقوم دارة داخلية بتقسيم هذه النبضات إلى أربع نبضات متساوية Q1 و Q2 وQ3 وQ4 لاتتداخل بينها.تشكل هذه النبضات الأربع دورة تنفيذ تعليمة واحدة (تسمى أيضاً دورة ألة واحدة) والتي خلالها يتم تنفيذ تعليمة واحدة فقط.



يبدأ تنفيذ التعليمة باستدعاء التعليمة التالية في السلسلة. تستدعى التعليمة من ذاكرة البرنامج عند كل Q1وتكتب في سجل التعليمات عند Q4. فك ترميز التعليمة وتنفيذها يتم بين النبضتين Q1 و Q4 التاليتين. نلاحظ من الشكل العلاقة بين دورة التعليمة وتردد المهتز OSC1 وكذلك النبضات الأربع. يمسك عداد البرنامج PC بعنوان التعليمة التالية.

Clock/Insruction Cycle

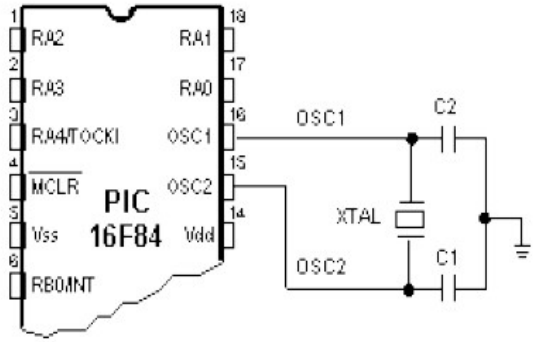


2-2 توليد نبضات الساعة – المذبذب :

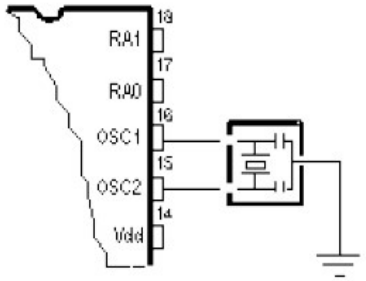
تستخدم دارة المذبذب لتزويد المتحكم بنبضات الساعة. وهو بحاجة لها لتنفيذ تعليمات البرنامج.

هناك أربعة أنواع من المذبذبات التي يمكن للمتحكم **PIC16F84** أن يتعامل معها. سنشرح منها النوعين الأكثر استخداماً مع المتحكم في الحياة العملية.

المذبذب الأول هو المذبذب الكريستالي **XT** والثاني المذبذب **RC** (المقاومة – المكثف).



Connecting the quartz oscillator to give clock to a microcontroller



Connecting a resonator onto a microcontroller

المذبذب XT:

يتواجد المذبذب الكريستالي ضمن حافظة معدنية وله قطبان ومكتوب عليه التردد الذي يهتز عنده.

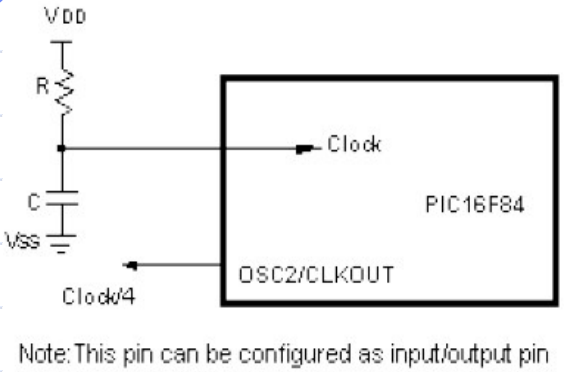
يوصل المذبذب كما في الشكل الى الأرجل **15** و **16** ويوصل الطرفان من ناحية أخرى إلى الأرض عن طريق مكثفين عدسيين بسعة **30PF** إلى الأرض.

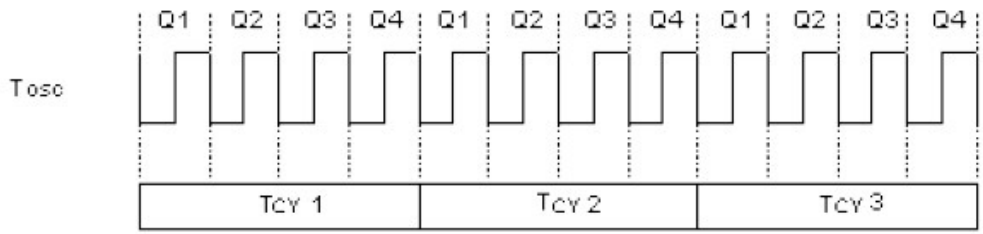
يمكن أن يتوفر في السوق عنصراً واحداً يجمع المذبذب مع المكثفات ضمن تغليفة واحدة من ثلاث أرجل تدعى المرنان السيراميكي **Ceramic Resonator**.

عند استخدام المذبذب في دارة عملية مع المتحكم, يجب إبقاؤه قريباً من أرجل المتحكم لتفادي أي تشويش على الأطراف التي يستقبل عندها المتحكم نبضات الساعة.

المذبذب RC:

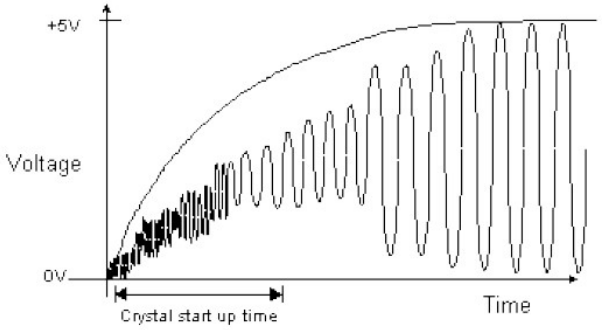
في التطبيقات التي لاتكون لدقة التوقيت فيها أهمية ضرورية, توفر المذبذبات RC بعض الكلفة.
يعتمد تردد الرنين للمهتز RC على معدَّل جهد التغذية وعلى قيم المقاومة R و المكثف C وعلى درجة الحرارة المحيطة.
يجب أن نذكر هنا أن تردد العمل يتأثر أيضاً بنسب الخطأ في قيم المقاومة والمكثف.
يوضح الشكل توصيل المذبذب مع المتحكم PIC16F84. إذا كانت قيمة المقاومة أقل من 2.2K يمكن أن يصبح المذبذب غير مستقر أو حتى أن يتوقف عن الاهتزاز.
أما إذا كانت ذات قيمة عالية جداً, مثلاً : 1 ميغا أوم, فقد يتأثر المذبذب بالضجيج.
لذا ينصح بإستخدام قيمة للمقاومة بين 3-100KΩ . يجب استخدام مكثف بسعة تزيد عن 20pF أيضاً لتحسين المناعة ضد الضجيج ولاستقرارية أعلى.
لايهم أي من المذبذبين تم إستخدامه , فإنه للحصول على تردد يمكن للمتحكم أن يعمل عليه, يجب تقسيم تردد المذبذب على أربعة. يمكن الحصول على التردد المقسم من القطب OSC2/CLKOUT ويمكن استخدامها لاختبار أو مزامنة دورات منطقية أخرى.





C_{EXT}	R_{EXT}	FRECUENCIA
20 pF	5 k 10 k 100 k	4,61 MHz 2,66 MHz 311 kHz
100 pF	5 k 10 k 100 k	1,34 MHz 756 kHz 82,8 kHz
300 pF	5 k 10 k 100 k	428 kHz 243 kHz 26,2 kHz

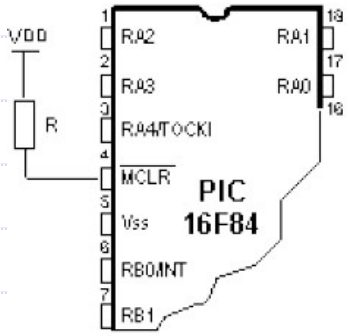
بعد تطبيق التغذية على دارة المتحكم فإن المذبذب الكريستالي يبدأ بالإهتزاز, حيث أنه كما يتبين من الشكل تكون هناك فترة ومطال غير مستقرين للاهتزاز, إلا أنها تستقر بعد فترة من الزمن.



لمنع تأثير مثل هذه الحالة على أداء المتحكم, نحتاج لإبقاء المتحكم في حالة تصفير خلال فترة استقرار المذبذب.

3-2 التصفير Reset:

يستخدم التصفير لوضع المتحكم في حالة معروفة. هذا يعني عملياً أن المتحكم قد يتصرف بشكل غير صحيح تحت بعض الظروف غير المناسبة. لذا من أجل أن يستمر المتحكم في أداءه الصحيح يجب أن يتم تصفيره, أي يجب وضع جميع سجلاته عند الحالة الأولية لها. يستخدم التصفير بالإضافة للحالة السابقة أيضاً عند اختبار جهاز كعنصر مقاطعة أثناء تنفيذ البرنامج, أو لجعل المتحكم جاهزاً عندما يتلقى برنامجاً.



لمنع وصول منطق منخفض (وهو الذي يؤثر على مدخل التصفير MCLR) بشكل غير مقصود, يجب وصل هذا المدخل إلى قطب التغذية الموجب عن طريق مقاومة رفع قيمتها بين 5-10KΩ .

يملك المتحكم PIC16F84 عدة مصادر للتصفير:

1) التصفير أثناء توصيل التغذية ويرمز له POR اختصاراً لـ Power-On Reset.
2) التصفير أثناء العمل الطبيعي بتطبيق منطق منخفض على مدخل التصفير MCLR .

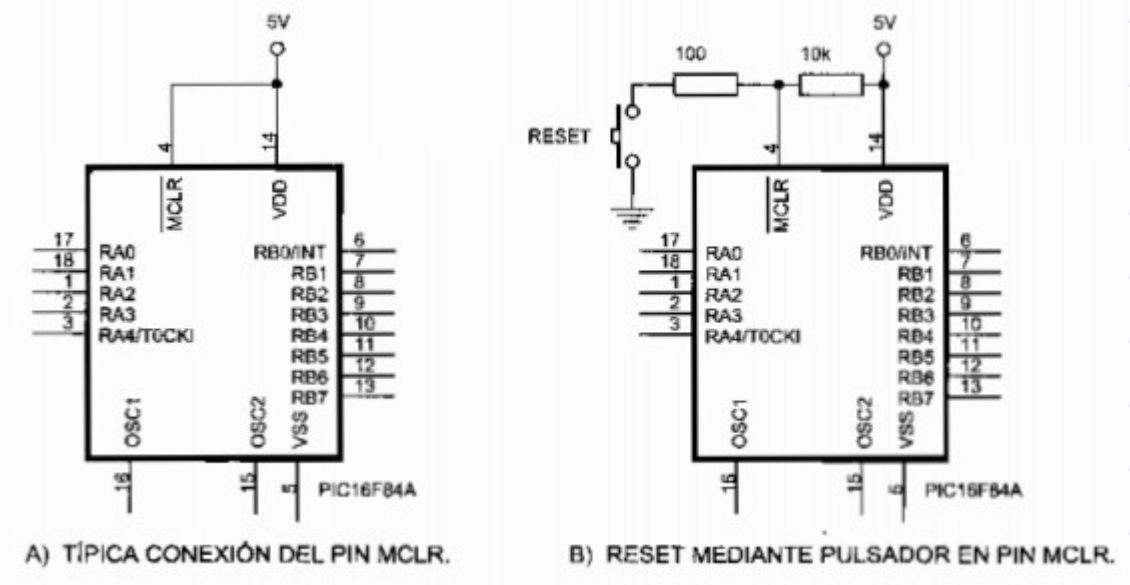
3) التصفير لإيقاظ المتحكم من وضعية السبات(تعليمة SLEEP).
4) التصفير عند طفح مؤقت المراقبة WDT.

أهم مصدرين للتصفير هما رقم (1) و(2). الأول يحدث في كل مرة يتم توصيل التغذية للمؤقت ويخدم في جعل كافة السجلات عند حالتها الأولية. والثاني ناتج عن التطبيق المقصود للمنطق المنخفض على مدخل التصفير أثناء العمل الطبيعي للمتحكم وهو غالباً ما يستخدم أثناء تطوير البرنامج.

لا يتم تصفير محتوى مواقع الذاكرة RAM أثناء التصفير, فهي لا تعرف حالتها عند وصل التغذية ولا تتغير عند أي تصفير. بعكس ذلك فإن سجلات الوظائف الخاصة SFR تعاد عند التصفير إلى حالتها الأولية. أحد أهم تأثيرات التصفير إعادة عداد البرنامج PC إلى الصفر وهذا يتيح إعادة تنفيذ البرنامج بدءاً من التعليمة الأولى.

هناك تصفير يحدث عند هبوط جهد التغذية تحت المستوى المسموح به ويسمى Brown-out Reset. تتولد نبضة تصفير أثناء الانخفاض في جهد التغذية عن طريق المتحكم نفسه عندما يكشف تناقصاً في جهد التغذية Vdd (في المجال بين 1.2V-1.8V). هذه النبضة تدوم لمدة 72ms, وهذا الزمن كاف لجعل المذبذب يستقر. هذه الفترة الزمنية يؤمنها مؤقت داخلي اسمه PWRT يملك عناصر التوقيت RC الخاصة به داخلياً. يكون المتحكم في حالة تصفير طالما كان المؤقت PWRT فعالاً.

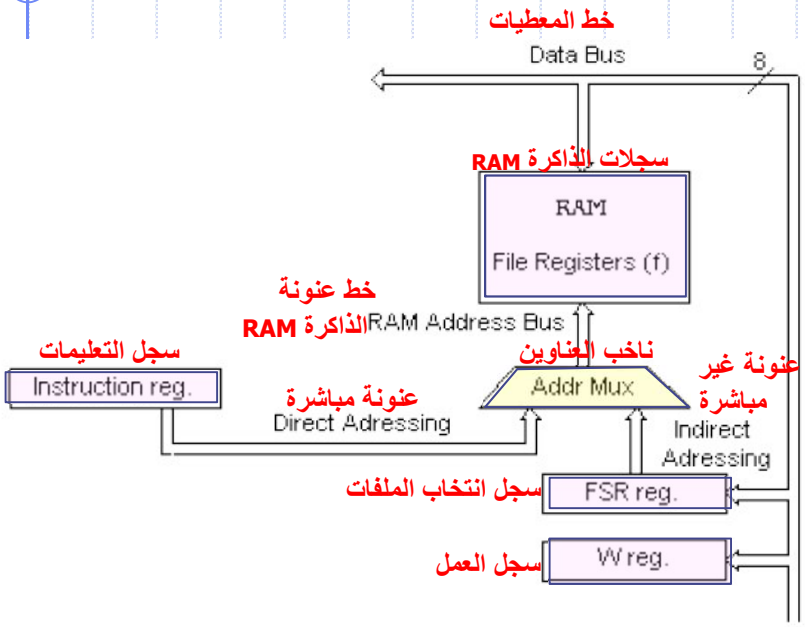
من ناحية ثانية وطالما أن الجهاز يعمل فإن المشكلة تظهر عندما لا يهبط جهد التغذية إلى الصفر وإنما يهبط تحت المستوى الذي تضمن العمل الصحيح للمتحكم. هذه الحالة تصادف كثيراً في الحياة العملية خاصة في البيئة الصناعية حيث يكون هناك سوء استقرار واضطرابات في التغذية بشكل يومي. لحل هذه المشكلة نحتاج التأكيد على أن يكون المتحكم في حالة تصفير في كل مرة ينخفض جهد التغذية فيها عن الحد المسموح به.



A) TÍPICA CONEXIÓN DEL PIN MCLR.

B) RESET MEDIANTE PULSADOR EN PIN MCLR.

3-2 وحدة المعالجة المركزية CPU:



Outline of the central processing unit-CPU

وحدة المعالجة المركزية هي دماغ المتحكم. وهي المسئولة عن إيجاد وجلب التعليمة الصحيحة التي يجب تنفيذها من أجل فك ترميزها (ترجمتها) ومن ثم تنفيذها.

تصل وحدة المعالجة المركزية جميع كتل المتحكم الداخلية بحيث تصبح واحدة.

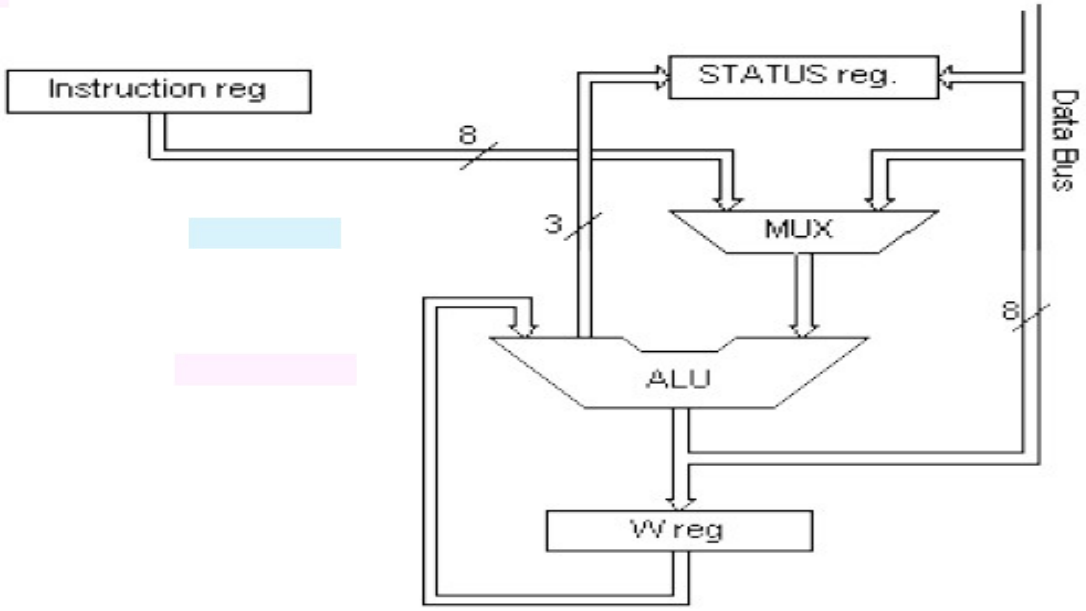
بالتأكيد فإن أهم وظيفة لها هي ترجمة تعليمات البرنامج.

بالتالي فإن التعليمات التي تجلب من ذاكرة البرنامج يجب أن تترجم من قبل وحدة المعالجة المركزية. يمكن بعدها الاختيار من ضمن قائمة تحوي جميع التعليمات مجموعة من الإجراءات التي تنفذ المهمة المحددة التي تعرّفها التعليمة. وحيث أن يمكن أن التعليمات نفسها تحتوي على مهام تحتاج لنقل المعطيات من موقع ذاكرة إلى آخر أو من الذاكرة إلى البوابات أو بعض الحسابات الأخرى, لذا يجب وصل وحدة المعالجة المركزية إلى جميع أطراف المتحكم وهذا يصبح ممكناً باستخدام خط العنونة وخط المعطيات Address & Data Bus.

وحدة الحساب والمنطق ALU:

هذه الوحدة مسئولة عن أداء عملية الجمع والطرح والنقل (يميناً ويساراً ضمن السجل) والعمليات المنطقية. نقل المعطيات داخل السجل يعرف أيضاً بالإزاحة.

يملك المتحكم PIC16F84 وحدة حساب ومنطق ثمانية الخانات وسجلات عمل ثمانية الخانات أيضاً.

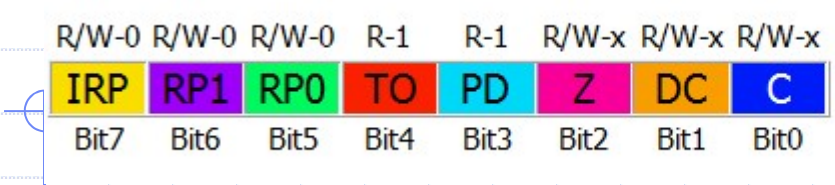


Arithmetic-logic unit and how it works

المسجلات الخاصة

1- مسجل الحالة

Status Register



-الخانة 0: "C" (الحامل Carry) النقل :

تتأثر هذه الخانة بعمليات الجمع والطرح والإزاحة.

1= يحدث النقل من أعلى خانة ناتجة

0= النقل لا يحدث

التعليمات التي تؤثر بهذه الخانة: ADDWF,ADDLW,SUBLW,SUBWF.

-الخانة 1: "DC" (حامل الخانة Digit Carry) خانة نصف الحامل:

تتأثر هذه الخانة بعمليات الجمع والطرح والإزاحة. بشكل مغاير لخانة الحامل, تمثل هذه الخانة عملية النقل من المكان الرابع الناتج. يتم وضعها Set بعملية الجمع عندما يظهر الحامل من الخانة الثالثة إلى الخانة الرابعة أو وضعها بواسطة عملية الطرح عند حدوث استعارة من الخانة الرابعة إلى الخانة الثالثة, أو عملية الإزاحة في كلا الاتجاهين.

1= يحدث النقل على الخانة الرابعة تبعاً لترتيب النتيجة.

0= النقل لا يحدث

التعليمات التي تؤثر بهذه الخانة: ADDWF,ADDLW,SUBLW,SUBWF.

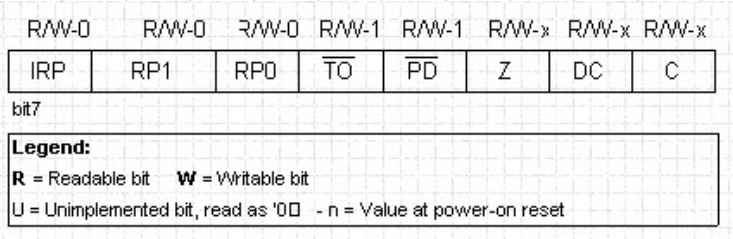
-الخانة 2: "Z" (خانة الصفر Zero bit) الإشارة إلى أن ناتج العملية صفر:

يتم وضع هذه الخانة عندما يكون ناتج عملية حسابية أو منطقية صفراً.

1= الناتج يساوي الصفر.

0= الناتج لا يساوي الصفر.

تتمة سجل الحالة:



R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit7							

Legend:
R = Readable bit **W** = Writable bit
U = Unimplemented bit, read as '0' - n = Value at power-on reset

-الخانة 3 "PD" (خانة انقطاع التغذية Power Down):

يتم وضع هذه الخانة كلما وصلت التغذية إلى المتحكم عند بدء تشغيله أو عند التصفير الاعتيادي أو بعد تنفيذ التعليمة CLRWDT.
التعليمة SLEEP تقوم بتصفيره عند دخول المتحكم ضمن نظام الاستهلاك الضئيل للطاقة. يمكن تكرار وضعه عبر التصفير أو عن طريق وصل أو فصل التغذية. يمكن أن يتم وضعه أيضاً بتطبيق إشارة على المدخل RB0/INT أو بحدوث تغير على البوابة RB أو عند الانتهاء من الكتابة على الذاكرة EEPROM الداخلية أو عن طريق مؤقت المراقبة.

1= بعد وصل التغذية.

0= عند تنفيذ تعليمة SLEEP.

-الخانة 4 "TO" (خانة المهلة Time-Out أو طفح مؤقت المراقبة). يتم وضع هذه الخانة بعد وصل التغذية وتنفيذ تعليمتي CLRWDT و SLEEP. يتم تصفير الخانة عندما يعطي مؤقت المراقبة مؤشراً نهائياً أن هناك أمراً غير صحيحاً.

1= الطفح لم يحدث.

0= الطفح يحدث.

-الخانتين 6:5 "RP1:RP0" (خانات انتقاء بنك السجلات).

تشكل هاتين الخانتين الجزء الأعلى من العنوان عند العنونة المباشرة. بما أن التعليمات التي تعنون الذاكرة مباشرة تملك سبع خانات فقط فإنها تحتاج إلى خانة أخرى من أجل عنونة المواقع الـ 256 كلها التي يملكها المتحكم PIC16F84. الخانة RP1 غير مستخدمة ولكنها تركت لبعض التوسعات المستقبلية لهذا المتحكم.

01= بنك السجلات الأول

00= بنك السجلات الثاني

تتمة سجل الحالة:

R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit7							

Legend:
R = Readable bit **W** = Writable bit
U = Unimplemented bit, read as '0' - n = Value at power-on reset

-الخانة 7 "IRP" (خانة إنتقاء بنك السجلات).

مهمة هذه الخانة العمل كخانة ثامنة للعنونة الغير مباشرة للذاكرة RAM الداخلية.

1= البنكان 2و 3 .

0= البنكان 0 و 1.

يحتوي سجل الحالة على الحالة الحسابية لوحدة الحساب والمنطق ALU (الخانات C,DC,Z) وعلى حالة التصفير (TO,PD) وعلى خانات اختيار بنك السجلات (IRP,RP1,RP0).باعتبار أن التحكم باختيار بنك الذاكرة يتم عبر هذا السجل لذا من المفترض وجوده في كل بنك. سوف نشرح بنك السجلات بتفصيل أكبر في فصل تنظيم الذاكرة. يمكن لسجل الحالة أن يكون وجهة لأي تعليمة مع أي سجل آخر. إذا كان هذا السجل وجهة للتعليمات التي تؤثر على الخانات Z أو DC أو C بعدها تكون الكتابة إلى الخانات الثلاث غير ممكنة.

2- مسجل الخيارات

Option Register

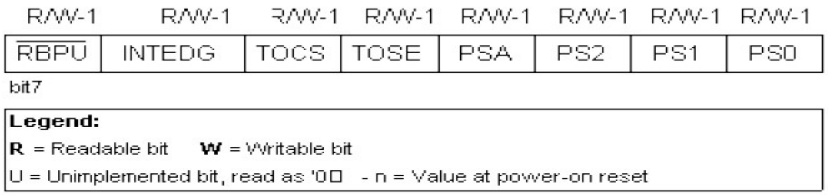
R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{RBPU}$	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit7							

Legend:
R = Readable bit **W** = Writable bit
U = Unimplemented bit, read as '0' - n = Value at power-on reset

-الخانات PS0,PS1,PS2 0:2 (خانة انتخاب معدل التقسيم).

تحدد هذه الخانات الثلاث خانة اختيار معدل التقسيم. ما نوع المقسِّم وكيف يمكن لهذه الخانات أن تؤثر على عمل المتحكم سيتم شرحه في قسم خاص من المؤقت TMR0.

Bits	TMR0	WDT
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128



R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit7							

Legend:
R = Readable bit **W** = Writable bit
U = Unimplemented bit, read as '0' - n = Value at power-on reset

تتمة سجل الخيارات:

-الخانة 3 "PSA" (خانة تعيين المقسِّم Prescaler Assignment Bit).

تقوم بتعيين المقسِّم بين المؤقت صفر ومؤقت المراقبة.

1= المقسم معين لمؤقت المراقبة.

0= المقسم معين للمؤقت حر الجريان TMR0.

-الخانة 4 "T0SE" (خانة اختيار حافة المصدر للمؤقت صفر TMR0 Source Edge Select Bit).

يمكن قدح المؤقت صفر بتطبيق نبضات على المدخل RA4/T0CKI, هذه الخانة تحدد فيما إذا كان القدح بواسطة الحافة الهابطة أو الصاعدة للإشارة.

1= حافة هابطة.

0= حافة صاعدة.

-الخانة 5 "T0CS" (خانة اختيار مصدر نبضات الساعة TMR0 Clock Source Select bit).

هذه الخانة تتيح زيادة المؤقت الحر الجريان لقيمته إما من المهتز الداخلي عند كل ربع 1/4دور المهتز, أو عبر نبضات خارجية مطبقة على المدخل RA4/T0CKI.

1= نبضات خارجية.

0= ربع تردد الساعة.

-الخانة 6 "INTEDG" (خانة اختيار حافة المقاطعة Interrupt Edge Select bit).

مع إمكانية تفعيل المقاطعة, تحدد هذه الخانة الحافة التي يتم تفعيل هذه المقاطعة فيها على المدخل RB0/INT.

1= حافة صاعدة.

0= حافة هابطة.

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{RBPU}$	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
bit7							

Legend:
R = Readable bit **W** = Writable bit
U = Unimplemented bit, read as '0' - n = Value at power-on reset

تتمة سجل الخيارات:

-الخانة 7 "RBPU" (خانة تأهيل الرفع للأعلى للبوابة PORTB Pull-up Enable bit B).

تقوم هذه الخانة بوصل أو فصل مقاومات الرفع الداخلية للبوابة B.

1= مقاومات الرفع مفصولة.

2= مقاومات الرفع موصولة.

3- مسجل البوابات PORTS:

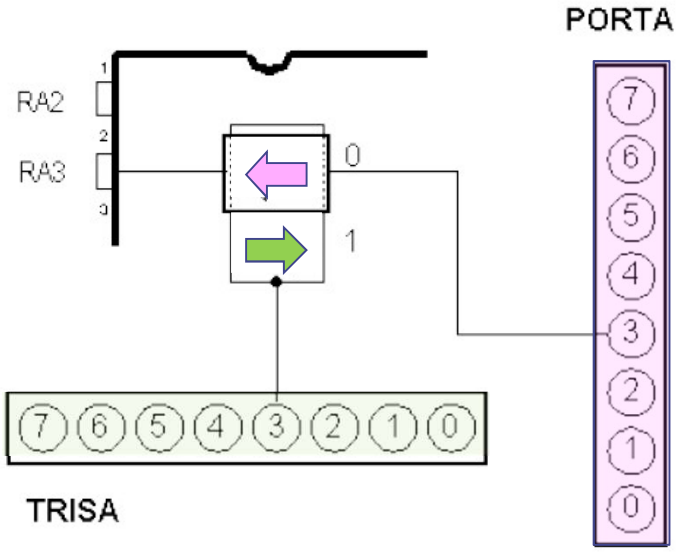
تشير البوابات إلى مجموعة من أطراف المتحكم التي يمكن الوصول إليها لحظياً أو يمكن أن نضع عليها أصفاراً أو واحدات أو أن يقرأ منها الحالة المتواجدة.

البوابة فيزيائياً هي سجل موجود داخل المتحكم موصول بأسلاك إلى أطرافه الخارجية.

تمثل البوابات التوصيلات الفيزيائية لوحدة المعالجة المركزية مع العالم الخارجي.

يستخدمهم المتحكم لمراقبة أو التحكم عناصر أو أجهزة أخرى.

تبعاً لوظائفها فإن بعض أطراف البوابات لها وظيفة إضافية مثل القطب PA4/T0CKI على سبيل المثال, وهو يمثل القطب الرابع للبوابة A وبنفس الوقت هو المدخل الخارجي للعداد الحر الجريان. اختيار أحد هاتين الوظيفتين للقطب يتم في واحد من سجلات التهيئة Configuration registers.كتوضيح على هذا الخانة الخامسة T0CS من سجل الخيارات. باختيار أحد الوظائف فإنه يتوقف عمل الأخرى.



Relationship between TRISA and PORTA register

تتمة البوابات:

يمكن تعريف جميع أطراف البوابة كمداخل أو مخارج تبعاً لحاجة الجهاز الذي يتم تطويره. بغرض تعريف القطب كدخل أو خرج يجب كتابة المجموعة من الأصفار والواحدات في السجل TRIS. إذا تم كتابة المنطق واحد 1 في المكان المناسب من السجل TRIS فهذا يعني أن القطب تم تعريفه كدخل وإذا تم العكس فهو يصبح قطب خرج. كل بوابة لها سجل TRIS خاص بها. بهذا فالبوابة A تملك السجل TRISA عند العنوان 85h والبوابة B لها السجل TRISB عند العنوان 86h.

البوابة B:

تملك هذه البوابة ثمانية أطراف. السجل TRISB يعمل على توجيه المعطيات عند العنوان 86h. وضع أي خانة في السجل المذكور سيعرّف قطب البوابة المقابل على أنه قطب دخل بينما تصفير أي خانة فيه سيعرف القطب المقابل على أنه قطب دخل. كل واحد من أقطاب هذه البوابة يملك مقاومة رفع ضعيفة داخلية (وهي تعطي الخط المنطق واحد) والتي يمكن تفعيلها بتصفير الخانة السابعة RBPU من سجل الخيارات OPTION. تفصل هذه المقاومات آلياً عند تعريف أي كقطب كخرج. وعند بدء تشغيل المتحكم تفصل هذه المقاومات.

الأقطاب الأربعة RB4:RB7 يمكن أن تتسبب بحدوث مقاطعة عند تغير حالتها المنطقية من المنطق واحد إلى المنطق صفر أو بالعكس. فقط الأقطاب المعرَّفة كمداخل يمكنها فعل هذه المقاطعة (عند تعريف الأقطاب كمخارج لاتحدث مقاطعة عند تغير منطقها). خيار المقاطعة هذا بالعمل مع مقاومات الرفع الداخلية يجعلان من السهل حل بعض الحالات العملية مثل حالة الكيبورد المصفوفي. إذا وصلت الصفوف Rows إلى هذه الأقطاب فإن كل ضغطة على المفتاح ستتسبب بحدوث مقاطعة. سيحدد المتحكم عندها أي مفتاح تم ضغطه. لا يتوجب الرجوع إلى البوابة B بنفس الوقت الذي يتم فيه معالجة المقاطعة.

يبين المثال التالي كيف تم تحديد الأقطاب 0و 1 و2 و3 كمداخل والأقطاب الباقية كمخارج.

```
clrf   STATUS        ;Bank0
clrf   PORTB         ;PORTB=0
bsf    STATUS,RP0    ;Bank1
movlw  0x0F          ;Defining input and output  pins
movwf  TRISB         ;Writing to TRISB register
```

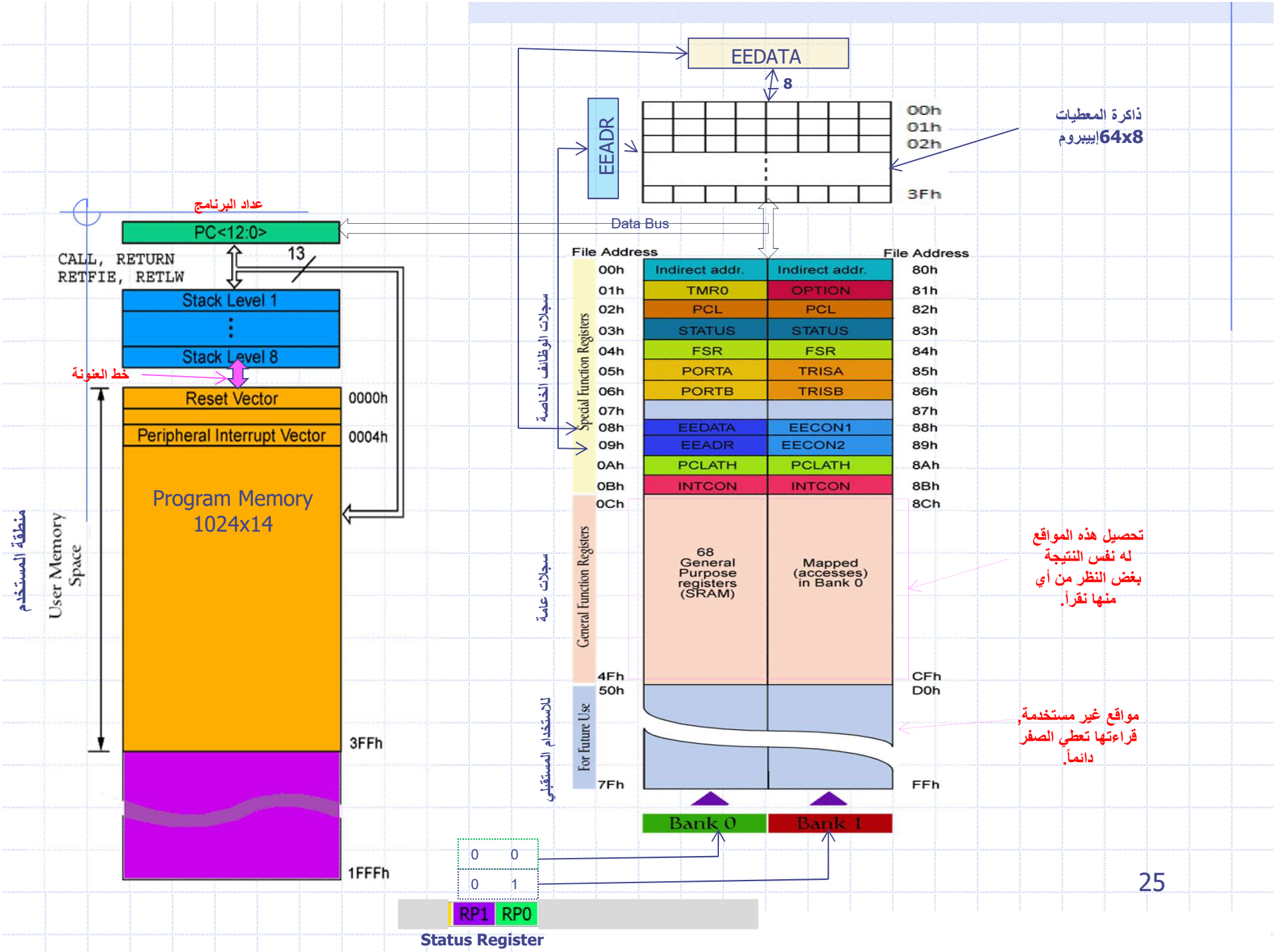
تتمة البوابات:

البوابة A:

تملك هذه البوابة خمسة أطراف. السجل TRISA يعمل على توجيه المعطيات عند العنوان 85h. وضع أي خانة في السجل المذكور سيعرِّف قطب البوابة المقابل على أنه قطب دخل بينما تصفير أي خانة فيه سيعرف القطب المقابل على أنه قطب خرج. يملك القطب الخامس من هذه البوابة وظيفة إضافية فهو يعمل كمدخل خارجي للمؤقت TMR0. يمكن انتقاء أحد هاتين الوظيفتين بوضع أو تصفير الخانة T0CS (خانة انتقاء مصدر نبضات ساعة المؤقت).يتيح هذا القطب ازدياد قيمة المؤقت سواء من المهتز الداخلي أو من النبضات الخارجية على المدخل RA4/T0CKI.

يبين المثال التالي كيف تم تحديد الأقطاب0و 1 و2 و3 و4 كمداخل.

```
bcf    STATUS,RP0    ;Bank0
clrf   PORTA         ;PORTA=0
bsf    STATUS,RP0    ;Bank1
movlw  0x1F          ;Defining input and output pins
movwf  TRISA         ;Writing to TRISA register
```


EEDATA
8
EEADR
00h
01h
02h
3Fh
ذاكرة المعطيات
64x8إيبيروم
عداد البرنامج
PC<12:0>
13
CALL, RETURN
RETFIE, RETLW
Stack Level 1
Stack Level 8
خط العنونة
Reset Vector
0000h
Peripheral Interrupt Vector
0004h
Program Memory
1024x14
3FFh
1FFFh
User Memory Space
منطقة المستخدم
Data Bus
File Address
File Address
سجلات الوظائف الخاصة
Special Function Registers
00h Indirect addr. Indirect addr. 80h
01h TMR0 OPTION 81h
02h PCL PCL 82h
03h STATUS STATUS 83h
04h FSR FSR 84h
05h PORTA TRISA 85h
06h PORTB TRISB 86h
07h 87h
08h EEDATA EECON1 88h
09h EEADR EECON2 89h
0Ah PCLATH PCLATH 8Ah
0Bh INTCON INTCON 8Bh
سجلات عامة
General Function Registers
0Ch 8Ch
68 General Purpose registers (SRAM)
Mapped (accesses) in Bank 0
4Fh CFh
50h D0h
للاستخدام المستقبلي
For Future Use
7Fh FFh
Bank 0
Bank 1
تحصيل هذه المواقع
له نفس النتيجة
بغض النظر من أي
منها نقرأ.
مواقع غير مستخدمة
قراءتها تعطي الصفر
دائماً.
0 0
0 1
RP1 RP0
Status Register

TRISA(Tri-State Buffer)

المُسجل

وهو سجل مكون من 8-bit (المستخدم منها فقط 5-bit) لانة يتعامل مع port A المكون من 5-bit ، وضيفتة تحديد أتجاة نقل البيانات Data ، أي تحديد طرف الأدخال Input أو الأخراج Output، يوجد هذا المسجل في الموقع 0x85 من bank1 ، القيمة صفر تعني الطرف المحدد من port a هو طرف أدخال input ، أما القيمة واحد فتعني ان الطرف هو طرف أخراج Output ، لكي نجعل porta كلة منفذ أخراج ما علينا سوى تحميل المسجل trisa بالقيمة 0x00

```
Trisa = 0x00  // the binary is 0000 0000
```

ولكي نجعل porta كلة منفذ أدخال ما علينا سوى تحميل المسجل trisa بالقيمة 0xFF

```
Trisa = 0xff  // the binary is 1111 1111
```

ولكي نجعل اول طرف RA0 من porta طرف ادخال Input وباقي الاطراف هي اطراف خروج Output ما علينا سوى تحميل المسجل trisa بالقيمة 0xFE

```
Trisa = 0xFE  //the binary is 1111 1110
```

PORTA المُسجل

وهو سجل مكون من 8-bit وضيفتة استقبال البيانات في حالة الكون المنفذ A أدخال ، أو أخراج البيانات الى العالم الخارجي في حالة كون المنفذ A أخراج ، ويوجد هذا المسجل في الموقع 0x05 من bank0

PORTB المُسجل

وهو سجل مكون من 8-bit وضيفتة استقبال البيانات في حالة الكون المنفذ B أدخال ، أو أخراج البيانات الى العالم الخارجي في حالة كون المنفذ B أخراج ، ويوجد هذا المسجل في الموقع 0x06 من bank0

Stephen M. Charlwood, The University of Birmingham © 2004

التعليمات

Bit Orientation Operations

التعَليمات على مُستوى البت

Mnemonic, Operands		Description	Cycles
BIT-ORIENTED FILE REGISTER OPERATIONS			
BCF	f, b	Bit Clear f	1
BSF	f, b	Bit Set f	1
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)

Bit Clear Flag

التَعليمة **BCF**

تَقوم هَذه التَعليمة بتَصفير أحد بتات مُسجل Register ، وتأخذ التَعليمة الصيغة التالية ،

```
BCF  f , b
```

حيث :

f: يمُثل المُسجل المُراد تَصفير أحد بتاتة،

b: يمثل رقَم البت المراد تَصفيرة من المُسجل f ،

Stephen M. Charlwood, The University of Birmingham © 2004

التَعليمة BSF — Bit Set Flag

تَقوم هذه التَعليمة برفع أحد بتات مسجل Register الى القيمة واحد ، وتأخذ التَعليمة الصيغة التالية ،

```
BSF  f , b
```

التعليمة BTFSS — Bit Test Flag Skip If Set

لكي نَفهم وضيفة هَذه التَعليمة ، دَعنا نأخذ الشفرة Code التالية،

```
1 BTFSS  0x0C, 2
2 BSF    0x0C , 2
3 BCF    0x0C , 3
```

تقوم التعليمة الأولى والتي هي BTFSS 0x0C,2 بفحص البت Bit الثاني من المسجل 0x0C فأذا كان واحد ، يُحمل عداد البرنامج PC بعنوان التَعليمة الثالثة التي هي BCF 0x0C,3 لينفذها ، بصيغة أخرى عندما تُنفذ التَعليمة BTFSS وكان البت المراد فحصة يساوي واحد يتم قَفز التَعليمة التي تَلي تَعليمة BTFSS ليتم تنفيذها أي سيتم قفز التعليمة الثانية الى التعليمة الثالثة والتي هي BCF 0x0C,3 لقوم المعالج بتنفيذها ، أما أذا كان البت Bit المُراد فحصة يَحمل القيمة صفر فيتم تنفيذ التعليمة التي تلي التعليمة BTFSS مباشرتاً أي سيتم تنفيذ التعليمة الثانية BSF 0x0C,3 ويستمر تنفيذ البرنامج ' أي سيتم تنفيذ التعليمة الثالثة ايضاً والرابعة وهكذا ، اذن التعليمة BTFSS تعتبر من تعليمات القفز Jump وتأخذ التعَليمة الصيغة التالية ،

Stephen M. Charlwood, The University of Birmingham © 2004

Bit Test Flag Skip If Clear

التَعليمة BTFSC

تَعمل هذه التَعليمة عَكس عَمل تَعليمة BTFSS حيثُ أن تعليمة BTFSC تقفز Jumping أذا كان البت Bit المُراد فَحصة يَحمل القيمة صفر ،أما اذا كان البت المراد فحصة يحمل القيمة واحد فسيتم تنفيذ التعليمة التي تلي تَعليمة BTFSC مباشرتاً ، تأخذ التعليمة الصيغة التالية ،

```
BTFSC   f , b
```

حيث:

f: يُمثل المُسجل المُراد فحَص أحد بتاتة ،

b: يُمثل رَقم البت المُراد فَحصة من المُسجل f ،

Stephen M. Charlwood, The University of Birmingham © 2004

Lateral and Control Instruction

تعليمات السيطرة والثوابت

وهي التعليمات التي تسيطر على سير عمل البرنامج مثل تعليمات القفز Jump ، وكذلك تتضمن التعليمات الخاصة بتحميل قيم ثابتة K الى مسجل معين ، الجدول Table 5.2 يوضح مجموعة تعليمات السيطرة والثوابت ،

Mnemonic, Operands		Description	Cycles
LITERAL AND CONTROL OPERATIONS			
ADDLW	k	Add literal and W	1
ANDLW	k	AND literal with W	1
CALL	k	Call subroutine	2
CLRWDT	-	Clear Watchdog Timer	1
GOTO	k	Go to address	2
IORLW	k	Inclusive OR literal with W	1
MOVLW	k	Move literal to W	1
RETFIE	-	Return from interrupt	2
RETLW	k	Return with literal in W	2
RETURN	-	Return from Subroutine	2
SLEEP	-	Go into standby mode	1
SUBLW	k	Subtract W from literal	1
XORLW	k	Exclusive OR literal with W	1

MOV Lateral to W Reg.

التعليمة MOVLW

تقوم هذه التعليمة بنقل قيمة ثابتة Lateral الى مسجل العمل W Reg.، وتأخذ التعليمة الصيغة التالية ،

```
MOVLW  K
```

حيث:

التعليمة ADDLW **ADD Lateral to W Reg.**

تقوم هذه التعليمة بجمع قيمة ثابتة Lateral مع قيمة مسجل العمل .W Reg ، وتخزن النتيجة في مسجل العمل .W Reg ، تأخذ التعليمة الصيغة التالية ،

```
ADDLW  K
```

حيث:

K: يمثل الرقم الثابت المراد جمعة مع مسجل العمل .W Reg .

التعليمة ANDLW **AND Lateral and W Reg.**

تقوم هذه التعليمة بأجراء عملية AND المنطقية بين مسجل العمل .W Reg وقيمة ثابتة ، وتأخذ التعليمة الصيغة التالية ،

```
ANDLW  K
```

حيث:

K: يمثل الرقم الثابت المراد أجراء عملية AND المنطقية علية .

CALL Statement

التعليمة CALL

تقوم هذه التعليمة باستدعاء برنامج ثانوي Subroutine مخزون في موقع اخر من الذاكرة ، تأخذ التعليمة الصيغة التالية ،

```
CALL  K
```

حيث:

K: يمثل أسم البرنامج الثانوي Subroutine المراد استدعاءة .

Clear Watchdog Timer

التعليمة CLRWDT

تقوم هذه التعليمة بتعطيل وضيفة مؤقت الحراسة ، ، تأخذ التعليمة الصيغة التالية.

Goto Statement

التعليمة Goto

تقوم هذه التعليمة بالقفز الى موقع أخر من الذاكرة ، تأخذ التعليمة الصيغة التالية ،

```
Goto  K
```

حيث:

K: يمثل عنوان موقع الذاكرة المراد القفز الية .

Stephen M. Charlwood, The University of Birmingham © 2004

التعليمة XORLW

Exclusive OR Lateral With W Reg.

تقوم هذه التعليمة بأجراء عملية Ex-OR المنطقية بين مسجل العمل .W Reg وقيمة ثابتة ، وتأخذ التعليمة الصيغة التالية ،

```
XORLW  K
```

حيث:

K: يمثل الرقم الثابت المراد أجراء عملية Ex-OR المنطقية علية .

التعليمة IORLW

Inclusive OR Lateral With W Reg.

تقوم هذه التعليمة بأجراء عملية OR المنطقية بين مسجل العمل .W Reg وقيمة ثابتة ، وتأخذ التعليمة الصيغة التالية ،

```
IORLW  K
```

حيث:

K: يمثل الرقم الثابت المراد أجراء عملية OR المنطقية علية .

Stephen M. Charlwood, The University of Birmingham © 2004

التعليمة SUBLW — Subtract Lateral from W Reg.

تقوم هذه التعليمة بطرح قيمة ثابتة Lateral من قيمة مسجل العمل W Reg. ، وتخزن النتيجة في مسجل العمل W Reg. ، تأخذ التعليمة الصيغة التالية ،

```
SUBLW  K
```

حيث:

K: يمثل الرقم الثابت المراد طرحة من مسجل العمل W Reg. .

التعليمة SLEEP — SLEEP Instruction

تقوم هذه التعليمة بأدخال المسيطر الدقيق في وضع توفير الطاقة standby Mode حيث يتوقف المعالج عن العمل ، وعند ورود أي اشارة مقاطعة الى المسطر يعود المسيطر للنهوض من جديد ، وتاخذ التعليمة الصيغة التالية ،

```
SLEEP
```

Return flag interrupt enable instruction — التعليمة RETFIE

لكي نفهم عمل هذه التعليمة ، يجب ان نفهم ما هي المقاطعة ، مبدء المقاطعة هو مقاطعة او توقف المعالج عن تنفيذ البرنامج الحالي او الرئيس Main Program لتنفيذ برنامج أخر يسمى برنامج خدمة المقطاعات interrupt service routine(ISR) ، وبعد الانتهاء من برنامج خدمة المقاطعات ، يجب ان يعود التنفيذ الى البرنامج الرئيسي Main Program ، أذن تعليمة RETFIE تقوم بنقل التنفيذ من برنامج خدمة المقاطعات الى البرنامج الرئيسي ، او بصيغة اخرى هي تعليمة العودة الى البرنامج الرئيسي ، سيتم مناقشة هذا الموضوع بشكل مفصل لاحقاً، وتاخذ التعليمة الصيغة التالية .

```
RETFIE
```

Return Instruction — التعليمة RETURN

تستخدم هذه التعليمة للعودة من برنامج الفرعي Subroutine ، الى البرنامج الرئيس، التعليمة تأخذ الصيغة التالية .

```
RETURN
```

Return with lateral to W Reg. **التعليمة RETLW**

تستخدم هذه التعليمة للعودة من برنامج فرعي Subroutine ، الى البرنامج الرئيس ، ولكن عند العودة تحمل قيمة مسجل العمل W Reg. بقيمة ثابتة Lateral value ، التعليمة تأخذ الصيغة التالية .

```
RETLW  K
```

حيث:

K: يمثل الرقم الثابت المراد نقلة الى مسجل العمل W Reg. عند العودة من البرنامج الفرعي subroutine .

Stephen M. Charlwood, The University of Birmingham © 2004

Program Structure

هيكل البرنامج

يتكون هُيكل البرنامج من دالة رئيُسيُة main function تكتب ضمنها التعلُيمات بلغة التجمُيع , Assembly Languageكما تحتوي على تعلُيمة القفز go to لكي يدخل المتحكم بحلقة تكرار غير منتهية :

التعليُمتان الأولى والثانُية تسمى موجُهات Directiveتستخدم لتوجُة المترجم Assemblerلُترجمة البرنامج بماُتناسب مع المتحكم pic16f84a أما التعلُيمة الثالثة mainفهي عبارة عن عنوان label الذي ترجع له التعلُمية Goto main لتشكل حلقة لا متناهُية ومن خلال هذة الحلقة نكتب التعلُيمات التَي ستتكرر بشكل مستمر , أما التعلُمية الأخيرة End فهَي تدل على نهاُية البرنامج .
يتكون المتحكم pic16f84aمن 13منفذ ولكَي نستخدم هذة المنافذ ُيجب تهيئتها , أي تحدُيد أي طرف من أطراف المنافذ هَي طرف دخل أو طرف خرج .

```
list p=16f84a
include <p16f84a.inc>
main
        ;instructions
        ;instructions
goto main
end
```

Stephen M. Charlwood, The University of Birmingham © 2004

PORTA &PORTB Configuration

تهيئة المنافذ

```
 list p=16f84a
include <p16f84a.inc>
      BSF 0x03,5        ;  Go to Bank 1   "Bit Set Flag"  Register, Bit
      MOVLW 0xff        ; mov  xxx1 1111 to W Register
      MOVWF   0x85      ; Move W to Trisa
      MOVLW 0x00        ; mov 0000 0000 to W Register
      MOVWF   0x86      ; Mov W to Trisb
      BCF 0x83,5        ;  Go to Bank 0   "Bit Clear Flag"  Register, Bit

main
            ;instructions
            ;instructions
goto main
end
```

Stephen M. Charlwood, The University of Birmingham © 2004

Writing

عملية الكتابة
Operations

```
list p=16f84a
include <p16f84a.inc>
    BSF 0x03,5        ; Go to Bank 1   "Bit Set Flag"  Register, Bit
    MOVLW 0xfb        ; mov xx11 1011 to W Register
    MOVWF   0x85      ; Move W to Trisa
    BCF 0x83,5        ; Go to Bank 0   "Bit Clear Flag"  Register, Bit
    Main
            BSF 0x05,2
     goto main
end
```

قمنا اولاً بعملّة تهّيئة المنفذ portaلّكون البت الثالث هو طرف خرج outputعن طرّق تحمّل السجل trisa بالقّيمة , 0 xFB ثم قمنا بكتابة تعلّيماتنا داخل الدالة الرئّيسية main function وذلك برفع البت الثالث من porta بالقّيمة واحد وذلك عن طرّق التعلّيمة bsf 0x05,4 تذكر أن عنوان porta هو 0 x05 فّي الـ file register إذا وصلت ديود باعث للضوء LED ستجده في حالة عمل .

Stephen M. Charlwood, The University of Birmingham © 2004

Reading Operations

عملية القراءة

عملُية القراءة Readingمن المنافذ مهمة لقراءة البُيانات من المحُيط الخارجَي مثلاً لتحسس المفاتُح Switches أو الحسسات , Sensorsالبرنامج التالَي ُيبُين عملُية القراءة Readingمن المنفذ port bعن طرُق البت السادس , إذا كان البت السادس فَي حالة توصُيل نشغل البت الرابع من portb .

```
 list p=16f84a
include <p16f84a.inc>
            BSF 0x03,5        ;  Go to Bank 1
            MOVLW 0x40        ; move 0100 0000 to W Register
            MOVWF   0x86      ; Mov W to Trisb
            BCF 0x83,5        ;  Go to Bank 0
   Main
             BTFSS 0x06,6     ; "Bit Test Flag Skip if Set " Register,bit
             GOTO   bit_is_reset
             GOTO Main
bit_is_reset  BSF 0x06,4
            GOTO Main
   end
```

قمنا اولا بعملُة تهُئة المنفذ portbلُكون البت السادس هو طرف دخول inputعن طرُق تحمُّل السجل trisbبالقُمة , x400ثم قمنا بكتابة تعلُماتنا داخل الدالة الرئُسة main functionوذلك عن طرُق فحص البت السادس من portbوذلك عن طرُق التعلُمة btfss وبعد تحقق الشرط قمنا برفع البت الرابع من , portbراجع الوحدة الخامسة لمزُّد من المعلومات حول التعلُمات .

Stephen M. Charlwood, The University of Birmingham © 2004

Variable المتغيرات

يمكن تعرّف المتغّيرات على غرار باقّي لغات البرمجة , وتكون الصّيغة كالآتّي :

Examble:
Variable level= 50
Variable Set_point = 20

Constant الثواب

هًي قٌيم مسندة لها أسماء معرفة لتسهٌيل التعامل مع القٌيمٌ Valuesيمكن استخدام تعلٌيمة EQUلإسناد أسماء للقٌيم , Valuesوتكون الصّيغة كالآتي :

NameOf_ Constant EQU Value

Examble:
Trisa EQU 0x85
Porta EQU 0x05

Examble:
Constatnt trisa= 0x85
Constatnt trisb= 0x05

Stephen M. Charlwood, The University of Birmingham © 2004

```
list p=16f84a
include <p16f84a.inc>
    Trisa   equ 0x85
    Porta  equ 0x05
    Bank0 equ 0x03
    Bank1 equ 0x83
    BSF Bank0,5         ;  Go to Bank 1    "Bit Set Flag"  Register, Bit
    MOVLW 0xfb          ; mov xx11 1011 to W Register
    MOVWF   Trisa       ; Move W to Trisa
    BCF Bank1,5         ;  Go to Bank 0    "Bit Clear Flag"  Register, Bit
Main
     BSF Porta,2
     goto Main
end
```

التعليمة ORG

تقوم هذه التعلُيمة بتحدُّد بدأية موقع تخزُّن البرنامج فًي الذاكرة , وبشكل افتراضًّي ُتم تخزُّين البرنامج عند الموقع 0 x00 إن لم تكتب التعلُيمة **ORG**

```
Start org 0×00
 movlw 0xFF
 movwf PORTB
```

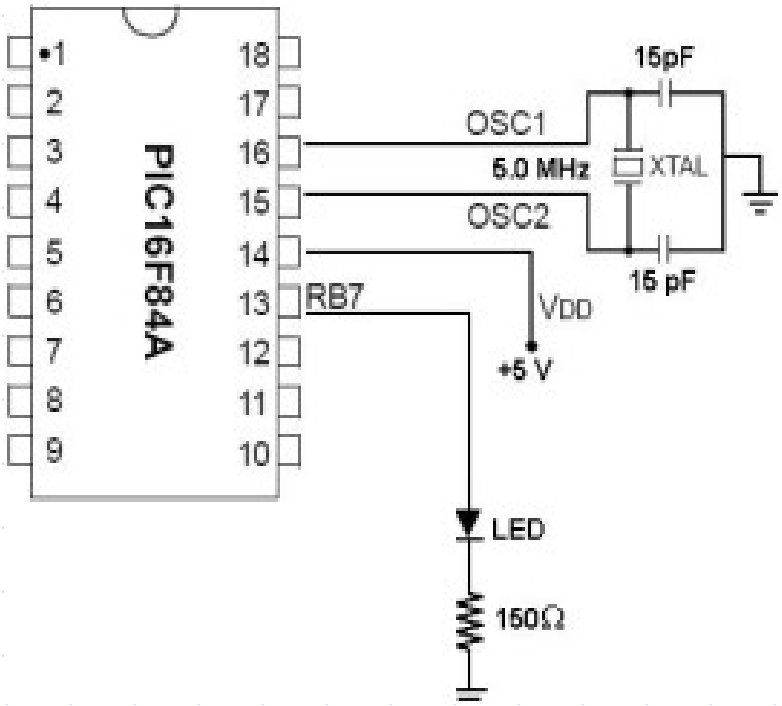
Stephen M. Charlwood, The University of Birmingham © 2004

مثال :

اكتب برنامج لجعلٌ LEDيومض (Flashing) بيِّن فترة وأخرى باستخدام الإجرائية Subroutine .

الحل:

المطلوب هو جعل LED يومض بُن فترة وأخرى كما هو مبُين فَي الشكل وذلك عن طرُق الطرف RB7



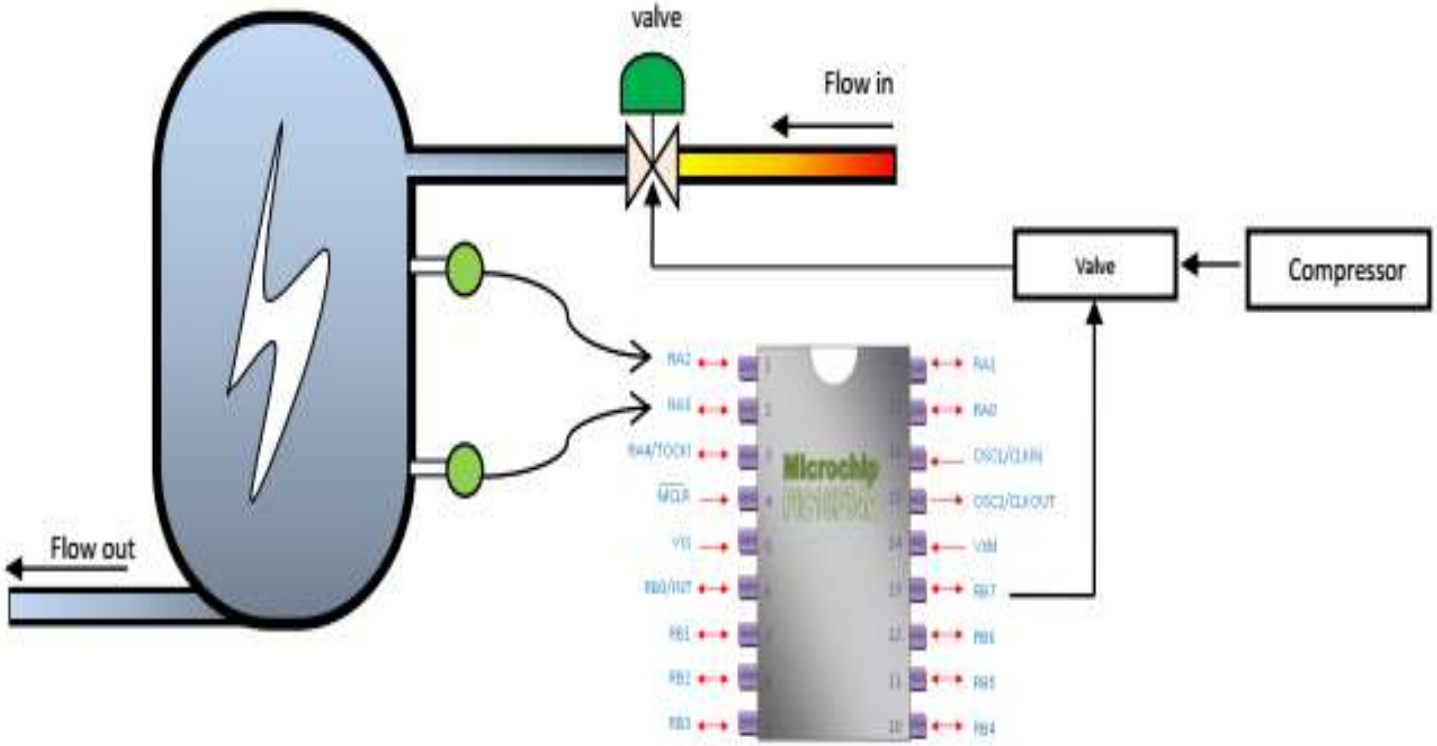
Stephen M. Charlwood, The University of Birmingham © 2004

```
list p=16f84a
include <p16f84a.inc>
        STATUS_BANK0  EQU 03h
        STATUS_BANK1  EQU 83h
        trisb   EQU 86
        portb  EQU 06
               BSF  STATUS_BANK0  ,5
               MOVLW  0x00                ; b"0000 0000"
               MOVWF  trisb
               BCF  STATUS_BANK1  ,5
 START
               BSF portb,7
               CALL  Delay
               BCF portb,7
               CALL Delay
               GOTO START
 Delay
                MOVLW 0xff
                MOVWF 0x0E
 Repeat         Decfz 0E,1
                GOTO Repeat
                RETURN
```

Stephen M. Charlwood, The University of Birmingham © 2004

مثال :

البرنامج التالِيٌ يستخدم للتحكم بمستوى خزان ، وذلك عن طرِّيق الفحص المستمر للحساس السفلِي ، عندما تصبح قُمة الحساس السفلِي صفر فهذاٌ يدل على أن الخزان فارغ ، نقوم بفتح الصمام ، وننتظر إلى أن تصبح قُمة الحساس العلوي تساوي واحد والتِّي تدل على أن الخزان قد ملئ وتعاد العملِّية :



Stephen M. Charlwood, The University of Birmingham © 2004

```
 list p=16f84a
include <p16f84a.inc>
    ;########## Control tanl level ############
       STATUS_BANK0  EQU 03h
       STATUS_BANK1  EQU 83h
       trisa   EQU 0x85
       trisb   EQU 0x86
       porta  EQU 0x05
       portb  EQU 0x06
        BSF             STATUS_BANK0 ,5
        MOVLW           0x0Ch   ;  xxxx 11xx bit3 =LL bit2=HL
        MOVWF           trisa
        MOVLW           0x00h   ; 0xxx xxxx
        MOVWF           trisb
        BCF             STATUS_BANK1  ,5

MAIN      BTFSC           porta  ,3   ;  test the tank if empty
          GOTO            MAIN
          BSF  portb  ,7  ;1xxx xxxx turn on bump
REPEAT    BTFSS           porta  ,2 ;  test the tank if full
          GOTO            REPEAT
          BCF  portb  ,7   ;1xxx xxxx turn on bump
GOTO MAIN

end
```

Stephen M. Charlwood, The University of Birmingham © 2004