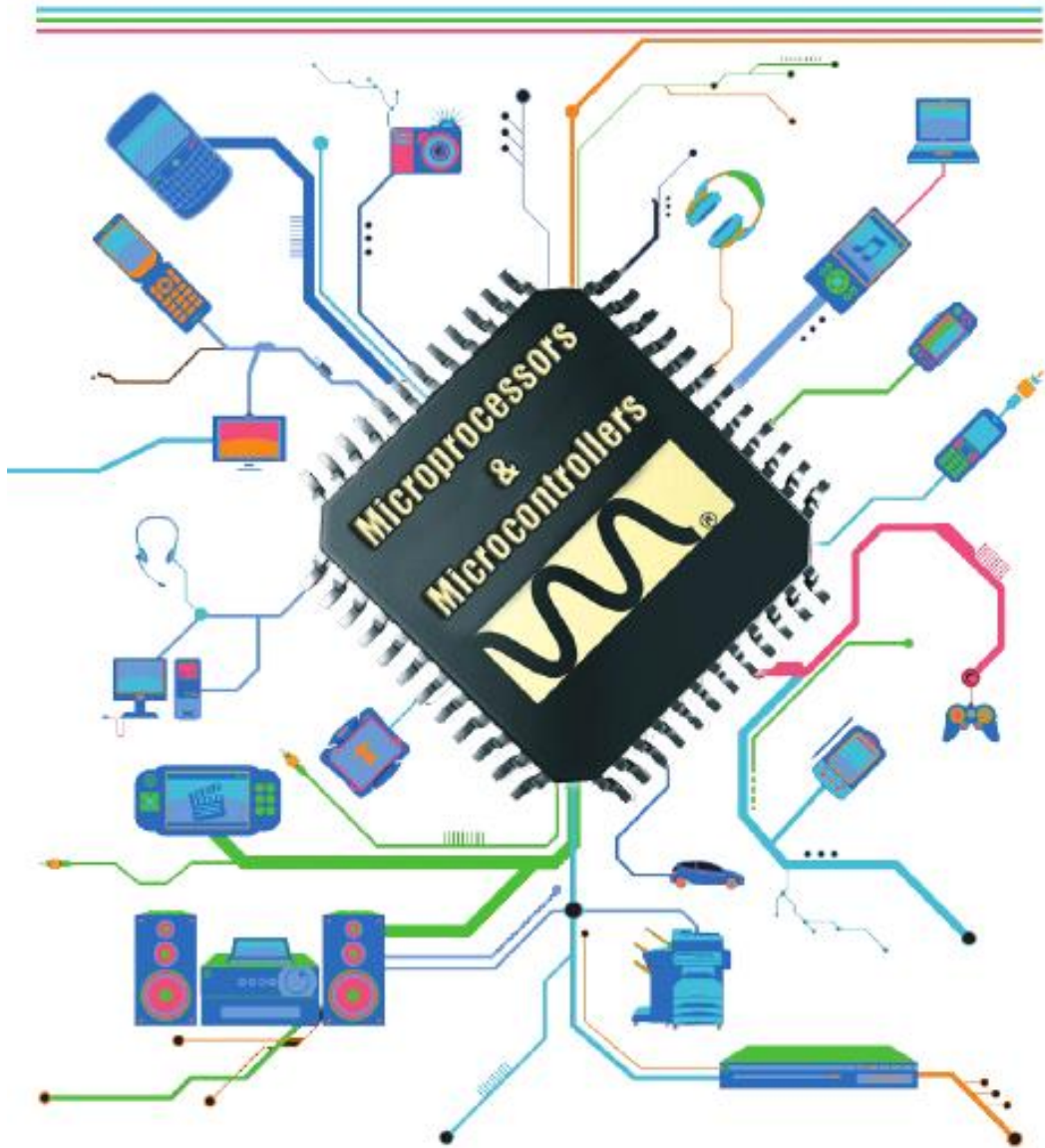


المتحكمات الصغيرة

Microcontroller



إعداد:

م. روزان المسالخي

يمكن اعتبار الأتمتة بشكل أو بآخر هي استبدال بعض أو كل التدخل البشري المطلوب لتنظيم وتوجيه عملية التحكم.

أهمية الأتمتة:

- ١ - إعطاء النظم الدقة الكبرى في العمل.
- ٢ - السهولة في التعامل.
- ٣ - الوثوقية عند التنفيذ.
- ٤ - الاقتصادية في تكلفة البناء والتشغيل وتوفير الوقت اللازم لذلك.

عناصر نظام التحكم:

يمكن أن نعتبر أن أي نظام تحكم يتألف من ثلاثة أقسام رئيسية:

١ - الدخل Input:

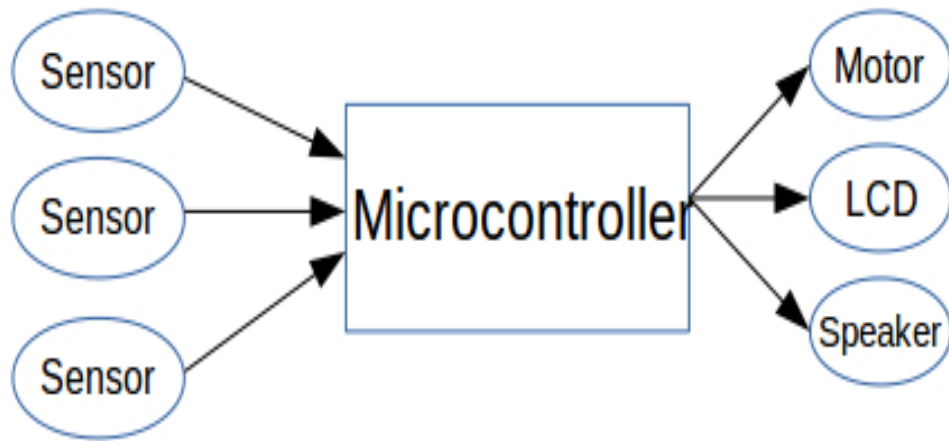
التي تمثل الحساسات المختلفة وأزرار الضغط أو أي وسيلة إدخال معلومات للمتحكم.

٢ - المعالجة Processing:

وتمثل الميكروكنترولر الذي يعتبر العقل المتحكم في النظام والذي يعتمد على استراتيجية معينة /قاعدة معرفية/ بحيث يوجه المخارج وفق الاستراتيجية المعدة مسبقاً.

٣ - الخرج Output:

التي تمثل المضخات والمكابس والمحركات وشاشات الإظهار والمضخات



مقارنة بين النظم التقليدية والنظم المؤتمتة:

المعيار	النظم التقليدية	النظم المؤتمتة
استراتيجية التحكم	عناصر مع أسلاك في الدارة	برنامج في الدارة
المعالجة	صلابة تعتمد على الريليهات وغيرها	برمجية بواسطة معالج كمبيوتر
تعديل استراتيجية التحكم	صعبة معقدة خصوصاً مع الأنظمة الضخمة، وحالة توقف	سهلة وسريعة بتغيير جزء من البرنامج، حالة عمل
المساحة المشغولة	كبيرة نسبياً	صغيرة نسبياً
التكلفة	ضخمة	بسيطة
تكرار نظام التحكم	يحتاج إلى وقت لإنجاز نفس التوصيلات	فقط نسخ البرنامج
الوثوقية	منخفضة بسبب وجود الحركة الميكانيكية	عالية بسبب استخدام العناصر الالكترونية
العمل في شبكة	صعب جداً	بسيط
الصيانة والفحوصات	صعبة وتحتاج إلى وقت وتكلفة	مباشرة وبلا تكلفة

الميكروكونترولر هو عبارة عن حاسب آلي متكامل محصور ضمن رقاقة صغيرة من أنصاف النواقل يقوم بنفس المهام التي يقوم بها جهاز الحاسب الآلي.

يمكننا المقارنة بين الحاسب الآلي والميكروكونترولر بالشكل التالي :

- الحاسب الآلي مهمته التفاعل مع المستخدم البشري من أجل قراءة البيانات والمعلومات من وحدات الإدخال المتنوعة كلوحة المفاتيح والفأرة ومن ثم معالجتها وإظهار النتائج على وحدات الإخراج مثل الشاشات والطابعات.

- الميكروكونترولر هو أصغر بكثير من الحاسب الآلي مهمته أيضاً التفاعل مع مستخدمه وهو الجهاز الذي يتم تركيب الميكروكونترولر داخله أي أن المستخدم الحقيقي له هو آلة وليس إنسان حيث يقوم الميكروكونترولر بتجميع البيانات والمعلومات من الجهاز عبر وحدات الإدخال الخاصة به من وأزرار التحكم وحساسات الحرارة أو الرطوبة أو الضغط أو الضوء ثم يقوم بمعالجة هذه البيانات والمعلومات في وحدة المعالجة الخاصة به ومن ثم يقوم بإظهار النتائج على وحدات الإخراج الخاصة به وهي البوابات التحكمية التي تقود الترانزستورات والحوالك وباقي الدوائر الالكترونية في الجهاز.

لماذا سمي الميكروكونترولر بهذا الاسم؟؟ وذلك لأن وظيفته هي التحكم فسمي بـ controller حيث يتحكم بالدائرة الالكترونية وبماتحويه من عناصر الكترونية وسمي بمايكرو لأن حجمه صغير جداً بالنسبة لإمكاناته الكبيرة .

بنية الميكروكنترولر:

تتألف معظم أنواع الميكروكنترولر المتوفرة في الأسواق من الأقسام الرئيسية التالية:

وحدة المعالجة المركزية CENTRAL PROCESSING UNIT CPU
ذاكرة البرنامج الويضية Flash Program Memory ROM
ذاكرة الوصول العشوائية Random Access Memory RAM
الناقل العام BUS
وحدة التحكم بالناقل Bus Unit
الساعة والهاز Clock & Oscillator
بوابات الدخل والخرج I/O Ports

١- وحدة المعالجة المركزية CPU: وهي الوحدة المسؤولة عن معالجة برنامج الميكروكنترولر وإجراء العمليات الحسابية والمنطقية على البيانات المتوفرة لها والقادمة من بوابات الدخل أو من الذاكرة وتجهيز النتائج للإخراج على بوابات الخرج.

٢- ذاكرة البرنامج الوميضية: هي عبارة عن نوع من أنواع الذاكرة ROM والتي نقوم بوضع البرنامج التشغيلي للميكروكنترولر فيها وبالتالي فإن وحدة المعالجة المركزية تقرأ التعليمات من هذه الذاكرة وتقوم بتنفيذها واحدة تلو الأخرى بالتزامن مع نبضات الساعة وطبعاً بما أن هذه الذاكرة وميضية فإن المعلومات لاتزول منها بعد قطع الطاقة عن المايكروكنترولر.

٣- ذاكرة الوصول العشوائي RAM: ويمكن أن نطلق عليها ذاكرة البيانات المؤقتة ومنه نستنتج وظيفتها وهي تخزين البيانات الناتجة عن قراءة القيم من العالم الخارجي أو من عمليات المعالجة والاحتفاظ بها بشكل مؤقت لاستعمالها في عمليات المعالجة أو التحكم الأخرى بالبرنامج وهذه البيانات المخزنة في الذاكرة تزول بمجرد قطع الطاقة عن الميكروكنترولر.

٤- الناقل العام BUS: ويمثل الطريق الذي تسلكه الإشارات الكهربائية داخل الميكروكنترولر بين مختلف الوحدات.

٥- وحدة التحكم بالناقل BUS Unit : ويمثل شرطي السير الذي يقوم بالتحكم بدور الناقل العام ويسيطر على عمليات انتقال البيانات والأوامر بين مختلف أجزاء الميكروكنترولر لضمان عدم حدوث تضارب بين هذه البيانات بين مختلف الوحدات، وهي الوحدة التي تخبر باقي الوحدات أي منها يكتب المعلومات على الناقل وأي منها يقرأ هذه المعلومات وارتباط هذه الوحدة بوحدة المعالجة المركزية وثيق جداً.

٦- الساعة أو الهزاز Oscillator : وهي القسم المسؤول عن ضبط الفواصل الزمنية بين مختلف عمليات الميكروكنترولر حيث أنها توجد ترتيباً دقيقاً ومتزامناً من الإشارات الكهربائية التي تزامن عمل وحدة المعالجة المركزية وباقي الوحدات لتنفيذ التعليمات البرمجية للميكروكنترولر الواحدة تلو الأخرى.

٧- بوابات الدخل والخرج: وهي المنافذ الكهربائية التي يقوم من خلالها الميكروكنترولر بالاتصال مع العالم الخارجي فيأخذ البيانات والمعلومات من عناصر القراءة الخارجية كالحساسات والمفاتيح كما أنه يعطي الأوامر التحكمية إلى ترانزستورات ولمبات الإشارة وغيرها.

👉 أهم ما يميز الميكروكنترولر هو سهولة استخدامه ورخص ثمنه ونسبة الخطأ فيه قليلة.

أنواع الميكروكنترولر:

يوضح الجدول التالي أشهر الشركات التي قامت بتطوير الميكروكنترولر:

م	اسم الشركة	شعار الشركة
1	شركة انتل intel	
2	شركة أتميل ATMEL	
3	شركة سوني SONY	
4	شركة مايكرو شيب MICROSHIP	
5	شركة توشيبا TOSHIBA	
6	شركة إن إي سي NEC	

تعد شركتي Microchip & Atmel هما الأبرز و الأكثر انتشاراً حيث تنتج شركة Microchip متحكمات PIC بينما تنتج شركة Atmel متحكمات AVR .

مقارنة بين PIC & AVR من عدة نواحي لنعرف من هو الاختيار الأنسب والأفضل في تطبيق معين:



في البداية يجب أن نعرف أنه في مجال النظم المؤتمتة لا يوجد ما يسمى "ما هو افضل متحكم دقيق" بصورة مطلقة ولكن هناك من هو "الأنسب" للاستخدام في تطبيق معين.

يمكن أن نعقد بعض المقارنات التقنية والمالية بين كلا النوعين من نفس المستوى تقريباً بالشكل التالي:

أولاً: مقارنة السرعة:

إن متحكمات AVR - 8 bit تتفوق بفارق كبير جداً ويعتبر أدائها أسرع بحوالي أربعة أضعاف عن مثيلتها PIC- 8 bit وذلك لأن متحكمات AVR تستطيع أن تنفذ عدد أوامر في الثانية الواحدة يساوي التردد الذي تعمل به أما متحكمات PIC تستطيع أن تنفذ ربع التردد فقط.

مثلاً لو معنا متحكم PIC و AVR وكلاهما يعمل بنفس التردد 16 ميغا هرتز (16 مليون هرتز) سنجد أن AVR يمكنه تنفيذ 16 مليون أمر برمجي في الثانية الواحدة بينما PIC يمكنه تنفيذ 4 مليون أمر برمجي في الثانية الواحدة. يرجع هذا الأمر إلى تقنية Pipeline التي تتميز بها جميع متحكمات AVR ولا تتواجد إلا في بعض فئات PIC المتطورة نسبياً.

ثانياً: استهلاك الطاقة:

هنا يتفوق PIC على AVR بفارق واضح، حيث تتميز متحكمات PIC باستهلاك منخفض للطاقة.

ثالثاً: السعر مقارنة بالميزات المدمجة:

في الأسواق المحلية تعتبر متحكمات AVR و PIC متقاربة في الأسعار لنفس العائلات التي لها خصائص وإمكانيات مشتركة (مثل سعة الذاكرة أو الحجم أو الطاقة المستهلكة أو...).

بالمقابل كما ذكرنا توفر متحكمات AVR قدرات مضاعفة (مثل السرعة والذاكرة و...) مقارنة بسعر PIC المماثل لها.

رابعاً: التوافر الكمي في الأسواق:

هنا نجد أن شركة Microchip (المصنعة لـ PIC) تتفوق على Atmel حيث أننا نجد أنه بكلا السوقين المحلي والعالمي منتجات Microchip متوفرة ويسهل الوصول إليها بسرعة مقارنة مع AVR.

كيف تختار بين عائلات الميكروكنترولر المختلفة ؟

إن الشركات المنتجة للمتحكمات الدقيقة عادة ماتصنع المئات من المتحكمات الدقيقة وتقسمها إلى عائلات تختلف فيما بينها حسب السعر والإمكانيات لكل متحكم ، لذا يتوجب عليك أن تتقن اختيار المتحكم المناسب لأداء أفضل تصميم بأقل سعر ممكن.

تعتبر أهم العوامل المؤثرة في تصنيف المتحكمات الدقيقة هي:

- ١- سرعة معالجة البيانات والاستجابة المطلوبة.
- ٢- عدد أطراف التحكم العامة أي عدد المداخل والمخارج.
- ٣- المجال أو البيئة التي سيعمل بها المتحكم مثل درجة الحرارة ومعدل استهلاك الطاقة.

٤- مساحة الذاكرة المطلوبة والتي ستحدد حجم البرنامج الذي سيوضع على المتحكم.

❖ تقنية الذاكرة memory technology :

على الرغم من الأهمية العظمى التي تحظى بها نوعية الذاكرة في المتحكم PIC فإنها لا تؤثر على منطق عمل الشريحة إطلاقاً.

تتوزع متحكمات PIC من حيث تقنية بناء الذاكرة على ثلاثة أصناف ويشار إلى صنف ذاكرة المتحكم بواسطة حرف رمزي يشير إلى نوع الذاكرة كما يلي :

- الرمز C - كما في المتحكمات PIC16CXXX – يشير إلى أن ذاكرة البرنامج من النوع EPROM.
- الرمز CR – كما في المتحكمات PIC16CRXXX – يشير إلى أن ذاكرة البرنامج من النوع ROM.
- الرمز F – كما في المتحكمات PIC16FXXX – يشير إلى أن ذاكرة البرنامج من النوع FLASH.

أ-شرائح الذاكرة EPROM

وهي اختصار (Erasable Programmable Read Only Memory) والتي تعني ذاكرة القراءة فقط القابلة للبرمجة والمحي أي أن هذه الشرائح يمكن محو محتوياتها عن طريق تسليط أشعة فوق بنفسجية من خلال نافذة في الجزء العلوي من الشريحة بعد ذلك فإن البيانات الجديدة يمكن حرقها في الذاكرة من

جديد، تتميز هذه الذاكرة بأنها تعطي مرونة كبيرة للمصممين أثناء كامل دورة التصميم وتتوفر بعدة خيارات تغليف مناسبة لجميع الاستخدامات.

ملاحظة: يجب تغطية نافذة المحي بطبقة كتيمة لمنع اختراق الأشعة فوق البنفسجية المنبعثة مع أشعة الشمس أو من مصابيح الفلوريسانت والتي تؤدي مع مرور الزمن إلى ضياع المعلومات إذا لم يتخذ الاحتياطات اللازمة.

ب- شرائح الذاكرة ROM

تتوفر شرائح متحكمات PIC ذات الذاكرة ROM المقنعة (masked ROM) وفق العديد من السعات المختلفة والكبيرة وذلك بغية إعطاء المستثمر التصميم الأقوى بالسعر الأقل.

ج- شرائح الذاكرة Flash

إن الذاكرة الوميضية Flash memory هي من الذاكر غير المتطايرة وهي شكل من أشكال الذاكرة EEPROM والتي تعني ذاكرة القراءة فقط القابلة للبرمجة والمحي كهربائياً، أي يمكن للمستخدم محو محتوياتها عن طريق إرسال نوع خاص من الإشارات الكهربائية المتسلسلة إلى أطراف معينة فيها وإعادة برمجتها حتى بدون أن تنزع من الدارة.

وتتميز الذاكر Flash أنه يتم محوها على شكل كتل في حين أن الذاكر EEPROM يتم محوها بمعدل بايت واحد كل مرة.

معلومة إضافية: الشرائح الإلكترونية الـ DIP (Dual in-line Package) هي التي تمتلك صفين من الأرجل المعدنية والتي يمكن توصيلها بثقوب على لوحة التجارب أو الـ PCB أما SMD وهي اختصار لكلمة Surface Mount Device هي الشرائح صغيرة الحجم و تمتلك أرجل معدنية صغيرة جداً ويتم لحامها على سطح الـ PCB فقط



ATmega32 (SMD)



ATmega32 (DIP)

قراءة دليل البيانات Datasheet :

تساعدك الـ Datasheet على فهم المتحكم الدقيق بصورة مفصلة فهي تحتوي على طريقة تشغيله وبرمجته، وتحتوي أيضاً على جميع البيانات التقنية المتعلقة بالمتحكم مثل: التصميم الداخلي، وظائف الأطراف، الطاقة، تقنيات البرمجة، كيفية تفعيل القدرات التي يملكها المتحكم أو إلغائها...

يتوفر من دليل البيانات نسختين summery مختصر سريع أو complete الدليل الكامل..

فمن خلال ملفات Datasheet يمكننا التعرف على المتحكم الدقيق الذي يمتلك الإمكانات المناسبة للمشروع الذي نريده.

المتحكم PIC16F877A:**منافذ الإدخال والإخراج للمتحكم PIC16F877A:****ماهي المنافذ Ports:**

تعتبر المنافذ واحدة من أهم السمات التي تميز ميكروكنترولر معين عن آخر، فهي عدد من أطراف الإدخال والإخراج المستخدمة للاتصال مع الأجهزة والوسط الخارجي.

هذه الأطراف تم توزيعها داخلياً بنظام 8 بت على خمسة منافذ A, B, C, D, E وهذه المنافذ لديها العديد من المميزات المشتركة.

معظم أطراف الإدخال والإخراج يمكنها القيام بوظيفتين إلى ثلاثة وظائف، بحيث إذا تم استخدام طرف ما في وظيفة معينة .. فإنه لا يمكن استخدام هذا الطرف في الأغراض العامة للإدخال والإخراج.

المنفذ A:

فقط 6 أطراف من هذا المنفذ يمكن استخدامها على الشكل التالي:

الأطراف من RA0 إلى RA5 يمكن استخدامها كمداخل رقمية.

الأطراف من RA0 إلى RA5 يمكن استخدامها كمخارج رقمية.

الأطراف من AN0 إلى AN4 يمكن استخدامها كمدخل تماثلية (أي أن عدد المداخل التماثلية لمنفذ A هي خمسة فقط..).

المنفذ B:

في هذا المنفذ فإن الأطراف الثمانية جميعهم يمكن استخدامها على الشكل التالي:
الأطراف من RB0 إلى RB7 يمكن استخدامها كمدخل رقمية.
الأطراف من RB0 إلى RB7 يمكن استخدامها كمخارج رقمية.
يمكن استخدام الطرف RB0 كمقاطعة خارجية.

المنفذ C:

في هذا المنفذ فإن الأطراف الثمانية جميعهم يمكن استخدامها على الشكل التالي:
الأطراف من RC0 إلى RC7 يمكن استخدامها كمدخل رقمية.
الأطراف من RC0 إلى RC7 يمكن استخدامها كمخارج رقمية.
الطرفين RC1, RC2 يمكن أن تستخدم في الحصول على PWM (تعديل عرض الموجة).

المنفذ D:

في هذا المنفذ فإن الأطراف الثمانية جميعهم يمكن استخدامها على الشكل التالي:
الأطراف من RD0 إلى RD7 يمكن استخدامها كمدخل رقمية.
الأطراف من RD0 إلى RD7 يمكن استخدامها كمخارج رقمية.

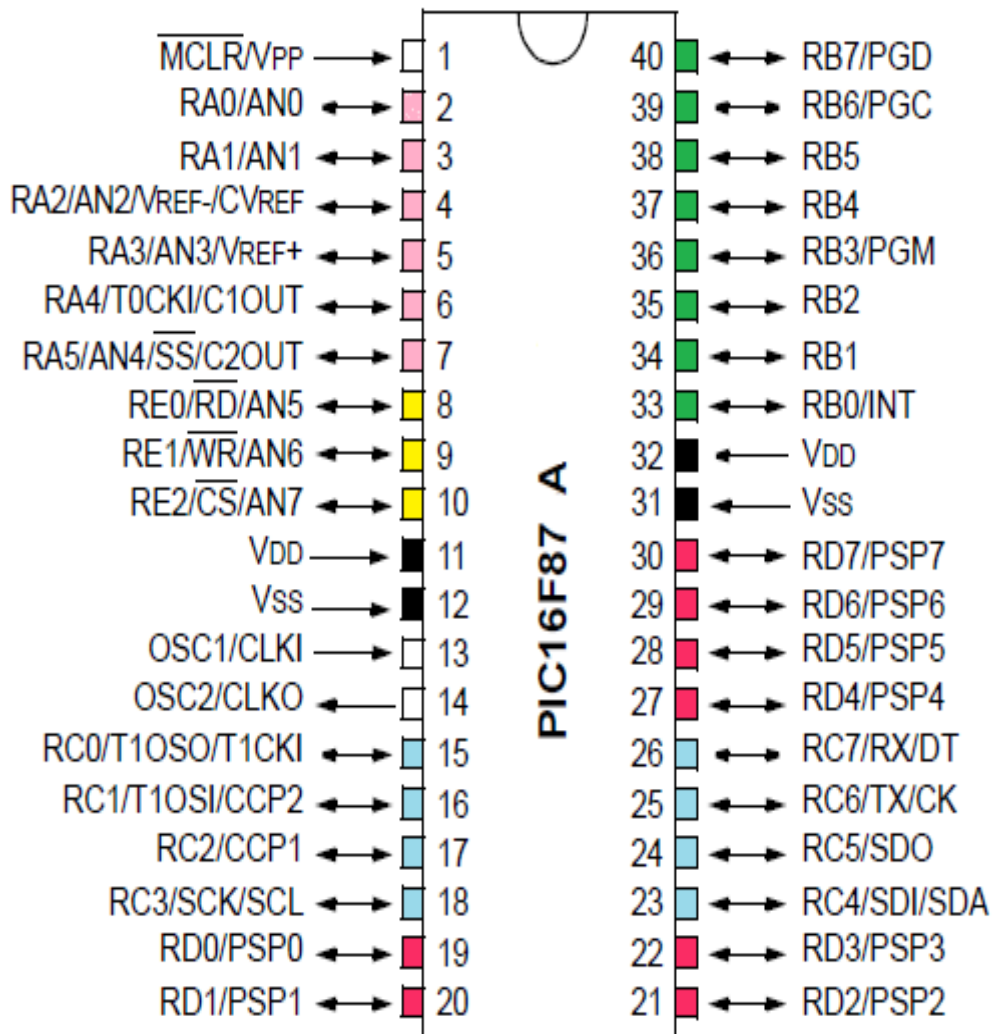
المنفذ E:

في هذا المنفذ فإن ثلاثة أطراف فقط يمكن استخدامها على الشكل التالي:

الأطراف من RE0 إلى RE2 يمكن استخدامها كمداخل رقمية.

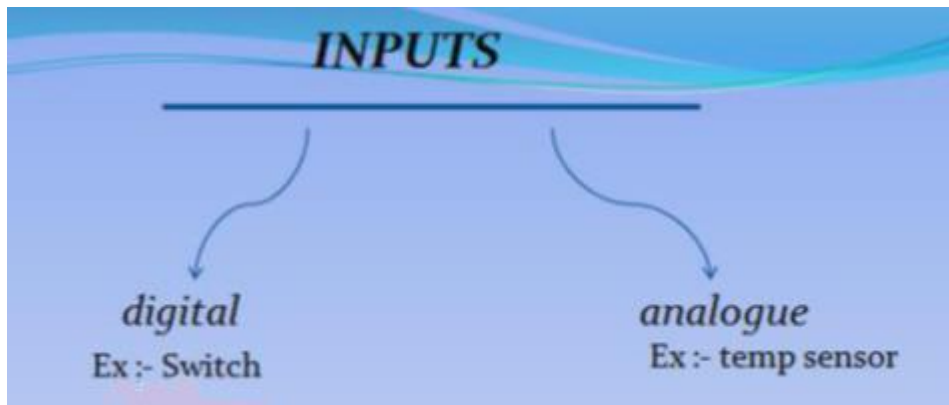
الأطراف من RE0 إلى RE2 يمكن استخدامها كمخارج رقمية.

الأطراف من AN5 إلى AN7 يمكن استخدامها كمداخل تماثلية (أي أن عدد المداخل التماثلية للمنفذ E هي ثلاثة فقط..)



إدخال البيانات إلى الميكروكنترولر:

بعد أن تعرفنا على جميع منافذ الميكروكنترولر لاحظنا أنه يمكنه استقبال البيانات في كل من الأشكال التناظرية والرقمية، فلذلك يكون للمبرمج المرونة في اختيار ما إذا كانت بعض أطراف البيك ينبغي أن تستخدم كمدخلات تماثلية أو مدخلات رقمية أو حتى مخرجات رقمية.



فعلى سبيل المثال في المتحكم السابق شرحه نلاحظ أن Pin2 تأخذ الرمز RA0/AN0 وهذا يعني أن Pin2 يمكن أن تستخدم 0 bit من المنفذ A كمدخل رقمي إلى استندنا إلى الرمز RA0 أو أن تستخدم كمدخل تناظري استناداً إلى الرمز AN0.

إخراج البيانات من الميكروكنترولر:

الميكروكنترولر هو عبارة عن جهاز رقمي كما ذكرنا سابقاً فلذلك نواتج الخرج بالنسبة له هي إشارات رقمية حيث أنه:

الخرج "1" يعني حالة التشغيل ON.

الخرج "0" يعني حالة الإيقاف OFF.

المقاومة الرافعة للجهد والمقاومة الخافضة للجهد:

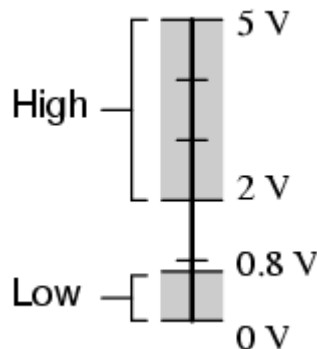
من أشهر الكلمات التي قد تسمعها في عالم الالكترونيات الرقمية هي المقاومة الرافعة للجهد والمقاومة الخافضة للجهد، هذه المقاومات التي تتراوح قيمتها بين 2.2 كيلو أوم إلى 10 كيلو أوم تستخدم بصورة أساسية في دخل الانظمة الرقمية حيث توصل مع المفاتيح العادية بدلاً من وصلها مع التغذية بشكل مباشر وذلك لثلاثة أسباب:

الاستخدام الأول (إلغاء الدخل العائم):

جميع المكونات الرقمية التي تتعامل بالصففر والواحد تعاني من مشكلة خطيرة جداً تسمى المنطقة العائمة أو نطاق الشوشرة هذه المنطقة هي فارق الجهد بين 0 و 1 فتستخدم هذه المقاومات لحل هذه المشكلة.

والسبب في ذلك أن معظم المكونات الالكترونية تعتبر أي جهد دخل بين الصففر و 0.8 فولت هو LOW=0 بينما أي جهد بين 2 حتى 5 فولت يعتبر HIGH=1

فإذا كان هناك دخل بقيمة 0.9 فولت أو 1.25 أو 1.9 تحدث مشكلة المنطقة العائمة حيث لاتستطيع الالكترونيات الرقمية أن تتعرف على هذا الجهد وبالتالي لايمكنها ان تحدد بدقة هل هو HIGH=1 أو LOW=0 وتبدأ بالتصرف العشوائي.



الاستخدام الثاني (عكس الجهد الداخل):

في بعض الحالات يكون مطلوب عكس الجهد الداخل من المفتاح..

عندما يتم توصيل المقاومة مع المفتاح بأسلوب Pull Down مما يجعل المفتاح يدخل جهد 5 فولت للطرف PB0 مثلاً عند الضغط عليه وعند ترك المفتاح يكون الجهد صفر فولت (أرضي).

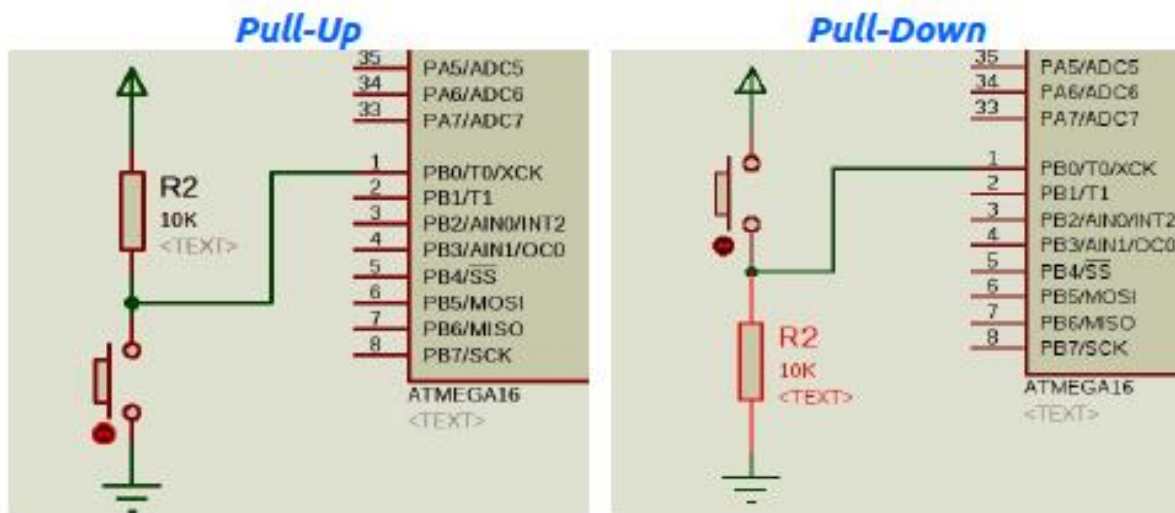
Press Button = HIGH

Releas Button = LOW

عندما يتم توصيل المقاومة مع المفتاح بأسلوب Pull Up فبذلك يصبح الجهد الأساسي للطرف RB0 هو 5 فولت وعند الضغط على المفتاح يتحول الجهد إلى صفر (أرضي).

Press Button = LOW Releas Button = HIGH

توضح الصورة التالية الفرق بين توصيلة كلاً من المقاومتين:



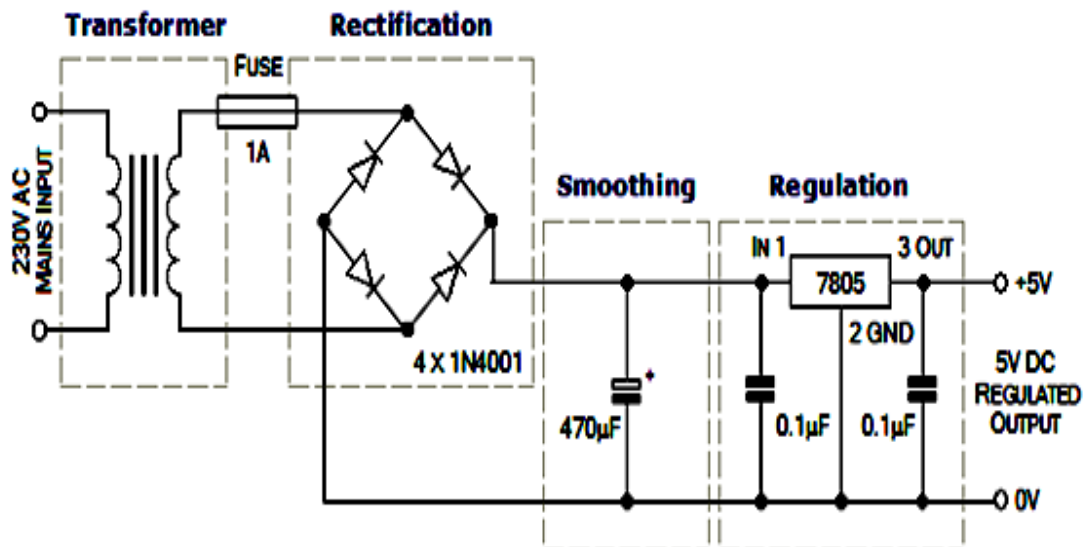
الاستخدام الثالث (محول تيار إلى جهد):

بعض الحساسات الشهيرة مثل الميكروفون أو المقاومة الضوئية أو الحرارية تقوم بتحويل الحرارة أو الضوء إلى تغير في التيار الكهربائي وليس تغير في الجهد مما يمثل مشكلة في فهم هذه الحساسات، حيث نجد أن جميع المتحكمات الدقيقة التي تحتوي على ADC يمكنها قراءة تغير الجهد فقط ولا تستطيع التعرف على التيار الكهربائي لذا يتم استخدام المقاومات الرافعة والخافضة للجهد لتحويل التيار المتغير إلى جهد.

المتطلبات الأساسية لعمل الميكروكونترولر:



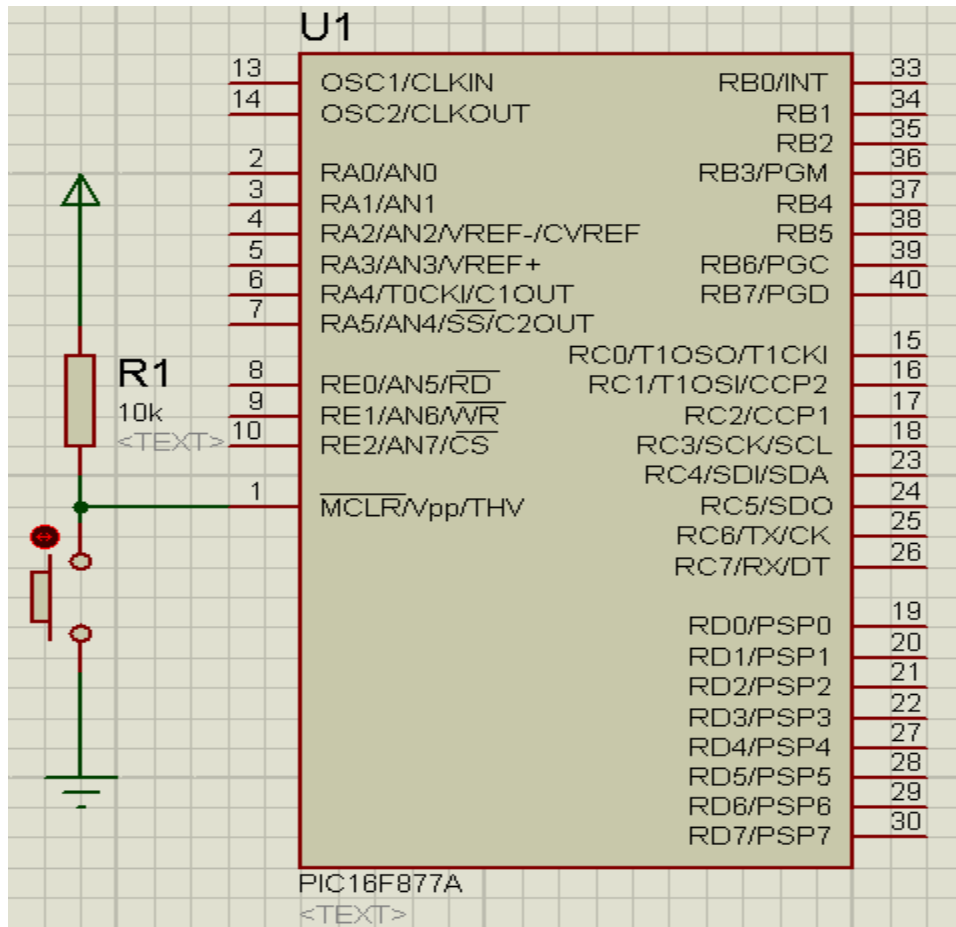
١- مصدر التغذية :



٢- إعادة المؤقت لحالة الانضباط (حالته الطبيعية) Reset :

في أي ميكروكونترولر نجد طرف يسمى MCLR والتي تعني Master Clear بمعنى أنه يقوم بعمل Reset فمثلا لو كان الميكروكونترولر يقوم بتنفيذ برنامج معين و عملت Reset فسيقوم بتنفيذ البرنامج من البداية.

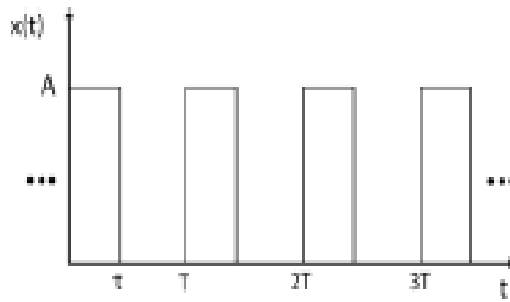
يمكن اعتبار طرف MCLR بأنه Active Low Pin أي أنه لكي يعمل نوصله بجهد منخفض أي 0V.



٣-المذبذب Clock Source:

إن أي ميكروكنترولر يحتاج لإشارة ساعة حتى يقوم بالعمل المطلوب منه على أكمل وجه والتي يقوم المذبذب بتوليدها.





شكل نبضات الساعة عبر الزمن

عندما نشتري أجهزة الحاسوب الخاصة نحاول

دائماً شراء الجهاز الأسرع وهنا نجد الشركات

المصنعة دائماً تتباهي بأن منتجاتها هي الأفضل ...

لأنها تعمل بأعلى سرعة معالجة (تقاس

بالجيجاهرتز - مليار هرتز). ومثل أجهزتنا

الشخصية نجد أن سرعة المُتحكّمت الدقيقة هي

أيضاً تحدد عبر التردد الذي تعمل عليه أو كما تسمى (مصدر الساعة clock source).

حيث يتم تنفيذ الأوامر بسرعة = $1 \div (\text{التردد})$ الذي يعمل به المُتحكّم، على سبيل المثال.

جميع شرائح ال AVR تقوم بتنفيذ أمر واحد كل 1 نبضة من ال clock source.

• إذا كان التردد = 1 ميگاهرتز (السرعة الافتراضية لمعظم مُتحكّمت (هذا

يعني أن المُتحكّم يمكن تنفيذ 1 مليون أمر بلغة الأسمبيلي في الثانية الواحدة

ويستغرق الأمر 1 ميكروثانية

• إذا كان التردد = 1 ميگاهرتز هذا يعني أن المُتحكّم يمكنه تنفيذ 1 مليون أمر بلغة

الأسمبيلي في الثانية الواحدة ويستغرق الأمر 1 ميكروثانية

• إذا كان التردد = 2 ميگاهرتز هذا يعني أن المُتحكّم يمكنه تنفيذ 2 مليون أمر بلغة

الأسمبيلي في الثانية الواحدة ويستغرق الأمر نصف ميكروثانية أو (500 نانوثانية).

• إذا كان التردد = 4 ميگاهرتز هذا يعني أن المُتحكّم يمكنه تنفيذ 4 مليون أمر بلغة

الأسمبيلي في الثانية الواحدة ويستغرق الأمر 250 نانوثانية nano second

• إذا كان التردد = 8 ميگاهرتز هذا يعني أن المُتحكّم يمكنه تنفيذ 8 مليون أمر بلغة

الأسمبيلي في الثانية الواحدة ويستغرق الأمر 125 نانو ثانية

النانو ثانية (nS) Nano Second = جزء من مليار جزء من الثانية

الميكرو ثانية (uS) Micro Second = جزء من مليون جزء من الثانية.

أنواع المذبذبات

المذبذب الداخلي المصنوع من مقاومة ومكثف

Internal RC Oscillator

وهي أرخص أسلوب للحصول على Clock Source ولا تتطلب هذه الطريقة توصيل أية مكونات خارجية وبالتالي يتم تصغير حجم المشروع وتقليل الكلفة. **عيوبها:** نسبة الخطأ في التردد والتوقيت تبلغ 3% - هذه النسبة لا تؤثر في أغلب التطبيقات لكنها خطيرة جداً في التطبيقات التي تحتاج توقيت دقيق.

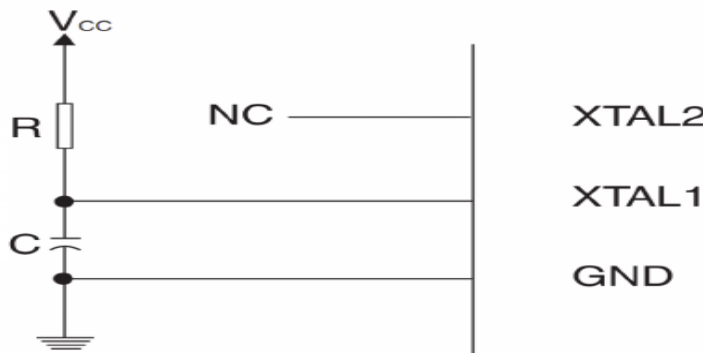
المذبذب الخارجي المصنوع من مقاومة ومكثف

External RC Oscillator

هذه الطريقة مماثلة للطريقة السابقة لكن الاختلاف الوحيد هو دائرة RC تكون موجودة خارج شريحة المتحكم وتوفر ترددات أكبر من 8 ميغا هرتز، حيث يمكنك ضبط التردد عبر التحكم في قيمة المقاومة وسعة المكثف عبر القانون التالي:

$$F(\text{frequency}) = \frac{1}{(R * C)}$$

حيث يتم توصيل XTAL1 بالطريقة التالية (NC أي غير متصل بشيء)



المميزات: يمكنك الحصول على ترددات أكبر من المذبذب الداخلي بتكلفة قليلة، كما يمكنك تغيير التردد في أي وقت عبر وضع مقاومة متغيرة بدلاً من المقاومة الثابتة.

العيوب: مثل النوع السابق، غالباً لا يمكن تحديد قيمة المكثف ولا المقاومة بدقة كبيرة حيث تحتوي هذه العناصر على نسبة خطأ 5% في حال أنك تريد استخدام هذه الطريقة يجب الانتباه إلى قيمة المكثف حيث يجب أن تكون 22 بيكوفاراد أو أكثر ويجب أن يكون من النوع السيراميكي (منعدم القطبية).

مذبذب الكريستالة الخارجي

External Crystal Oscillator

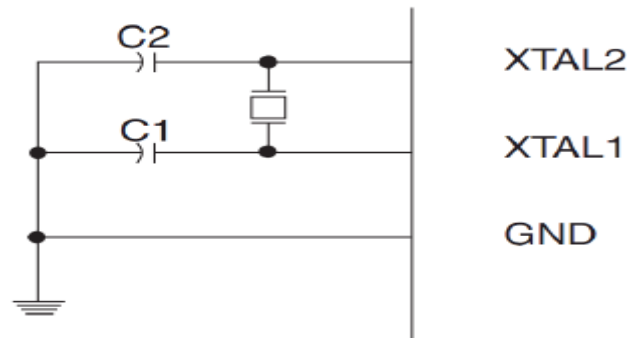


تعتبر هذه الطريقة هي الأكثر شيوعاً والمفضلة لدى جميع الشركات ومصممي الأنظمة المدمجة، حيث يتم استخدام الكريستالة ذات الطرفين لتوليد التردد المطلوب بدقة عالية جداً وبأقل نسبة خطأ ممكنة وهي أكثر دقة بنحو 1000 مرة من مذبذبات RC.

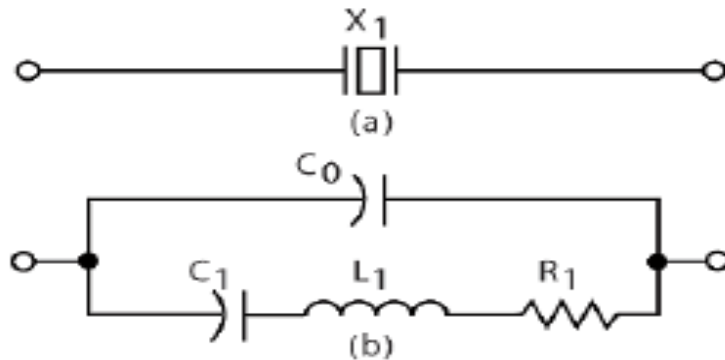
المميزات: الدقة العالية جداً وثبات التردد مهما تغيرت درجة الحرارة وبالتالي فهي توفر أداء ممتاز طوال فترة التشغيل.

العيوب: التكلفة حيث تحتاج هذه الكريستالات إلى مكثفات إضافية عدد 2 بسعة 22 بيكوفاراد.

يتم توصيل الكريستالة والمكثفات بالصورة التالية:



الكريستالة هي عنصر الكتروني مصنوع من مادة بلورية اسمها الكوارتز وعندما يتم توصيلها بالكهرباء فإنها تعمل كأنها دائرة رنين مكونة من مكثف وملف ومقاومة تنتج موجة جيبيه، وغالباً ما نجد أن التردد الخاص بالكريستالة مكتوب على جسم القطعة.



التركيب الداخلي الذي تمثله الكريستالة عندما
تتصل بدائرة كهربائية

مستلزمات برمجة المتحكم الصغير:

لكي نتمكن من برمجة المتحكم الصغير Microcontroller يتطلب توفر مايلي:

١- حاسب آلي.

٢- جهاز برمجة.

يبين الشكل التالي جهاز برمجة الميكروكنترولر وهو يصلح لأنواع متعددة من Pic ويتطلب هذا الجهاز كابلاً للربط مع الحاسب الآلي.



يوضح الشكل التالي الكابل المستخدم للربط مع الحاسب الآلي:



٣- البرامج المستخدمة في عملية البرمجة وتوليد ملف hex.

٤- البرنامج الذي يكتب على الميكروكنترولر.

٥- ميكروكنترولر من النوع Pic مثلاً.

٦- بعض المكونات الكهربائية والالكترونية من مقاومات ومكثفات وغيرها.

برنامج فلوكد Flowcode:

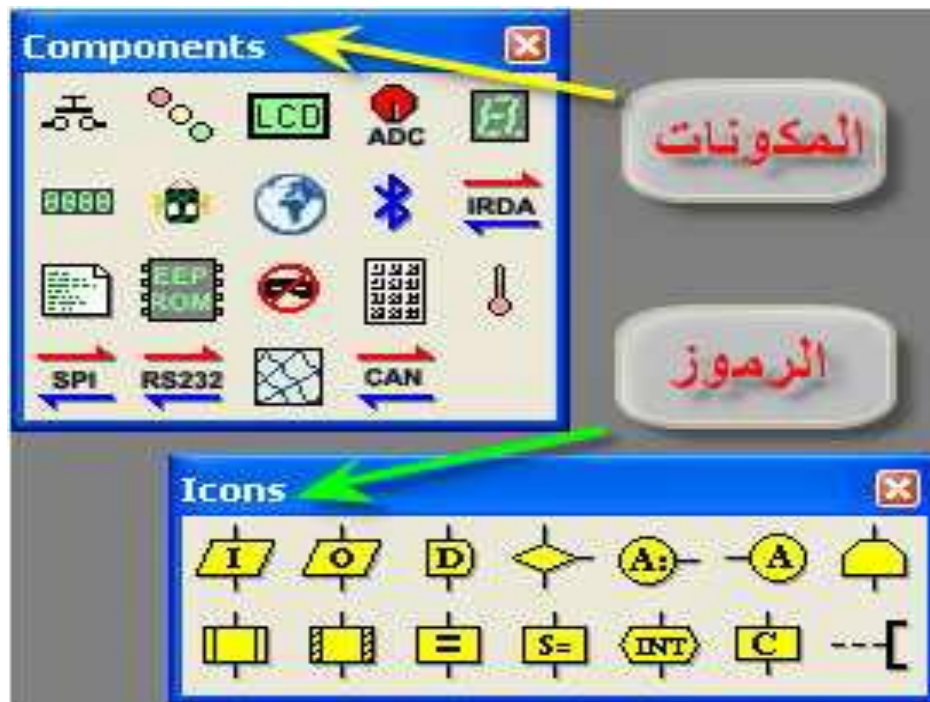
فلوكود هو إنتاج شركة ماتركس يستخدم لبرمجة الميكروكنترولر باستخدام مخطط التدفق، حيث أنه بمجرد الانتهاء من تصميم مخطط التدفق للبرنامج على شاشة الكمبيوتر وبضغط زر واحد يتم ترجمته إلى الترميز الست عشري (كود hex).

يحتوي فلوكد على:

١- رموز أساسية main icon لرسم مخطط التدفق.

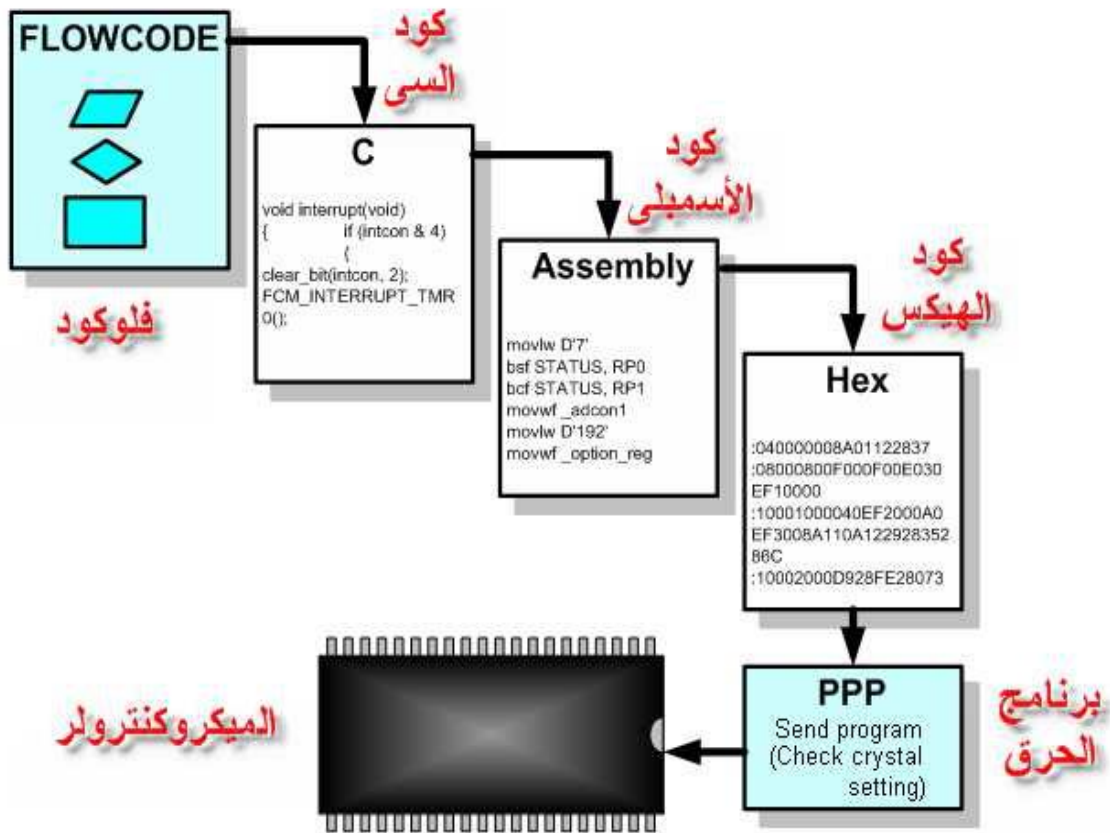
٢- مكونات الكترونية electronic components تسمح لنا بإنشاء نظام الكتروني افتراضي على الشاشة وهي واسعة حيث أنها تتضمن مفاتيح تشغيل بسيطة Simple Switches وليدات Leds وشاشة عرض LCD وعارضات سباعية 7Segment وغيرها..

والشكل التالي يوضح لنا شكل كل من الرموز والمكونات:



وبالاستفادة من سحب وإسقاط هذه الرموز والمكونات على الشاشة نكون قد بدأنا ببناء مشروعنا خطوة تلو الأخرى.

ففي البداية يقوم البرنامج بمعالجة البرنامج للحصول على كود السي C Code ثم بعد ذلك يتم تمريره عبر المجمع Assembler للحصول على كود الاسمبلي وفي النهاية يتم تحويله إلى كود الهيكس وهي اللغة التي يفهمها الميكروكنترولر والذي نقوم بإرسال هذا الكود إليه ليقيم بتنفيذه.



لإرسال كود الهيكس إلى الميكروكنترولر فإن فلوكود يمتلك برنامج فرعي للقيام بذلك يسمى PPP الذي يقوم بحرق الكود في ذاكرة البرنامج للميكروكنترولر .

مميزات البرمجة بالفلوكود:

تعتبر الميزة الكبرى لبرنامج Flowcode هو أنه يتيح لمن لديهم خبرة قليلة في البرمجة باللغات الأخرى القدرة على إنشاء برامج لأنظمة الكترونية معقدة .

يمكن تصنيف أهم العوامل التي تميز البرمجة بالفلوكود كالتالي:

١- له واجهة سهلة الاستخدام حيث أنه يمكن ببساطة سحب واسقاط الرموز والمكونات المرفقة مع البرنامج على الشاشة لإنشاء نظام الكتروني دون الدخول في كتابة سطور التعليمات المرهقة كمافي اللغات الأخرى.

٢- يوفر الوقت والجهد عن طريق سهولة وسرعة تصميم النظام الالكترونية.

٣- يعطي نتائج بدون أخطاء حيث أنه نتيجة ما يتم رؤيته من تصميم ومحاكاة على شاشة الحاسوب هو بالفعل ما يحدث على أرض الواقع بعد تحميل البرنامج داخل الميكروكنترولر.

٤- يخدم اللغات الأخرى فهو يسمح لك بعرض كود C لجميع البرامج التي يتم إنشاؤها.

خطوات البرمجة بالفلوكود:

تتلخص خطوات تنفيذ المشروع باستخدام فلوكود بثلاث خطوات رئيسية فقط:

الخطوة الأولى: تصميم البرنامج Program Design

يتم البدء في إنشاء مخطط التدفق لبرنامج المشروع عن طريق استخدام خاصية السحب والإسقاط drag and drop للرموز والمكونات المطلوبة من واجهة برنامج فلوكوند لتصميم المشروع المطلوب. ثم يتم النقر على كل رمز أو عنصر لضبط الإجراءات والخصائص التي نحتاج إليها .

الخطوة الثانية: محاكاة المشروع Project Simulation

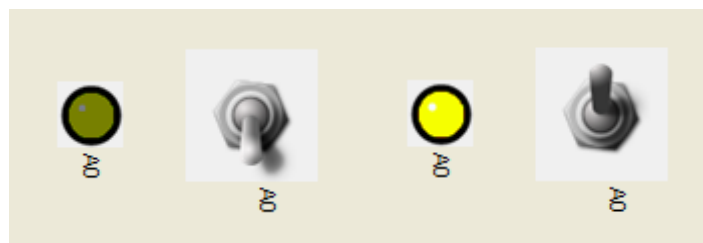
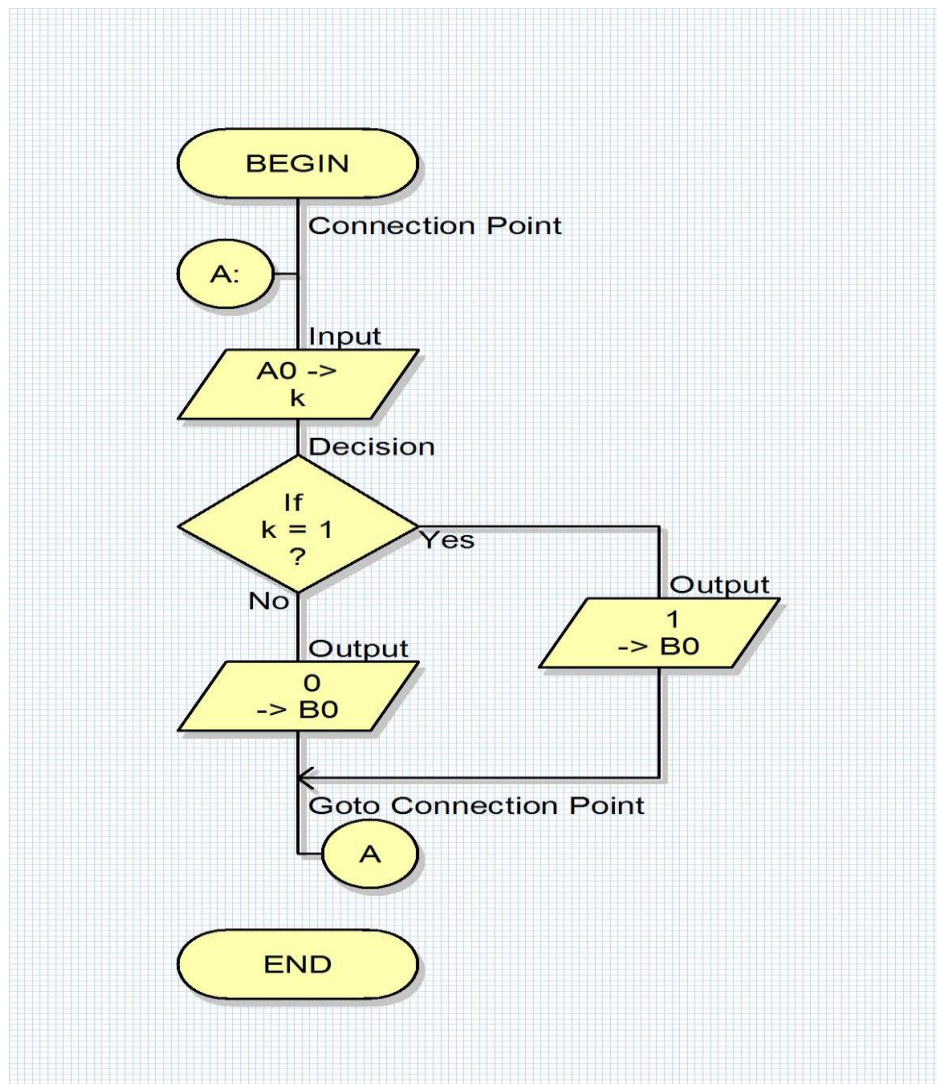
بمجرد الانتهاء من تصميم البرنامج المطلوب فإن فلوكوند يتيح لنا إمكانية محاكاة النظام وذلك عن طريق الضغط على زر في قائمة التحكم للبرنامج مختص بذلك. يتيح لنا فلوكوند أثار كل خطوة من البرنامج على المكونات والمتحكم وذلك على الشاشة مباشرة. يمكن التبديل والتغيير في القيم ورؤية تأثير هذه التغيرات على سير البرنامج وبالتالي على نتائج المشروع .

الخطوة الثالثة: توليد كود الهيكس Hex code Generation

عندما ننهي من المحاكاة والحصول على الشكل النهائي للمشروع وبعد الرضا الكامل عن النتائج. نقوم بالنقر على زر واحد خاص بتوليد ملف الهيكس في صورته النهائية تمهيدا لارساله الى الميكروكنترولر باستخدام البرنامج المساعد PPP والمبرمجة. ويجب ملاحظة ان كود الهيكس المتولد من فلوكوند (مرورا بكود السي والاسمبلى) يكون متوافق مع معظم انواع المبرمجات .

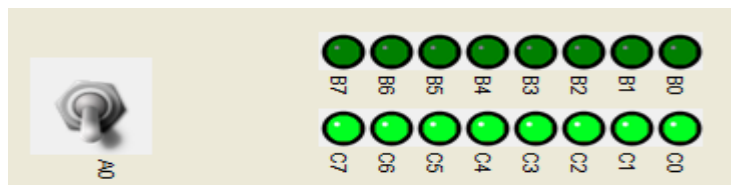
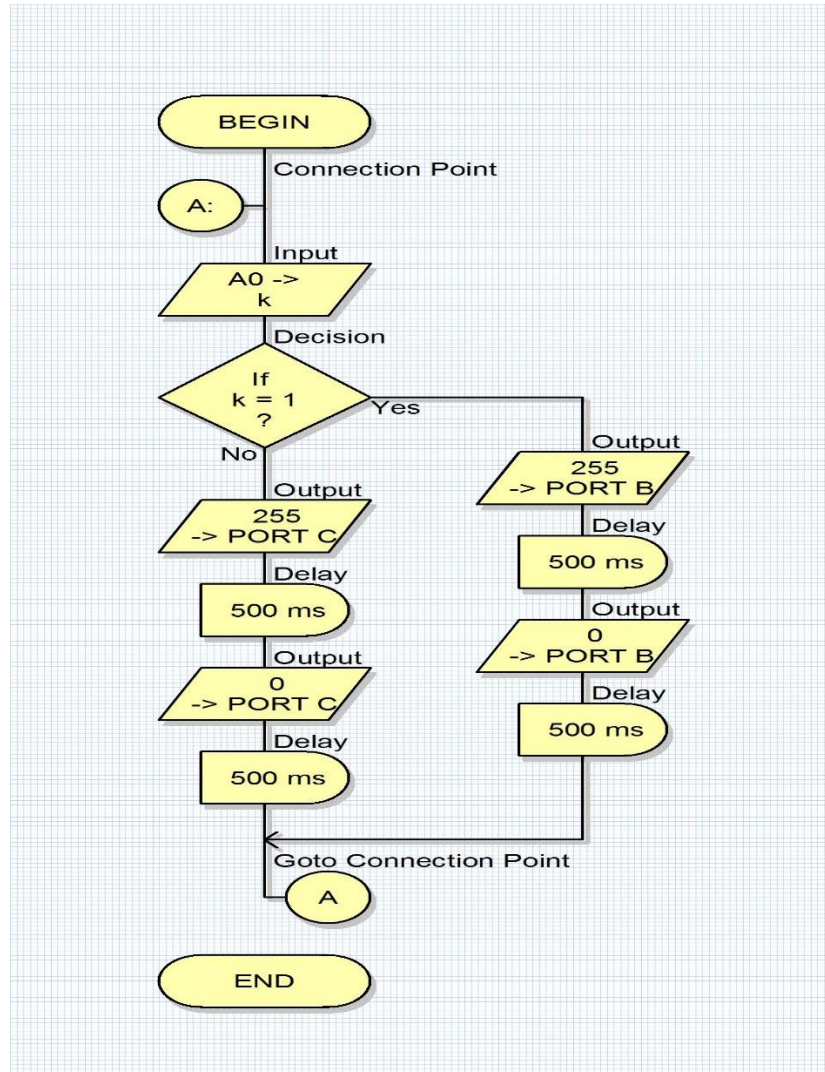
مثال:

برنامج يقوم بإضاءة ليد في حالة الضغط على السويتش وإطفائها في حال تحرير السويتش.



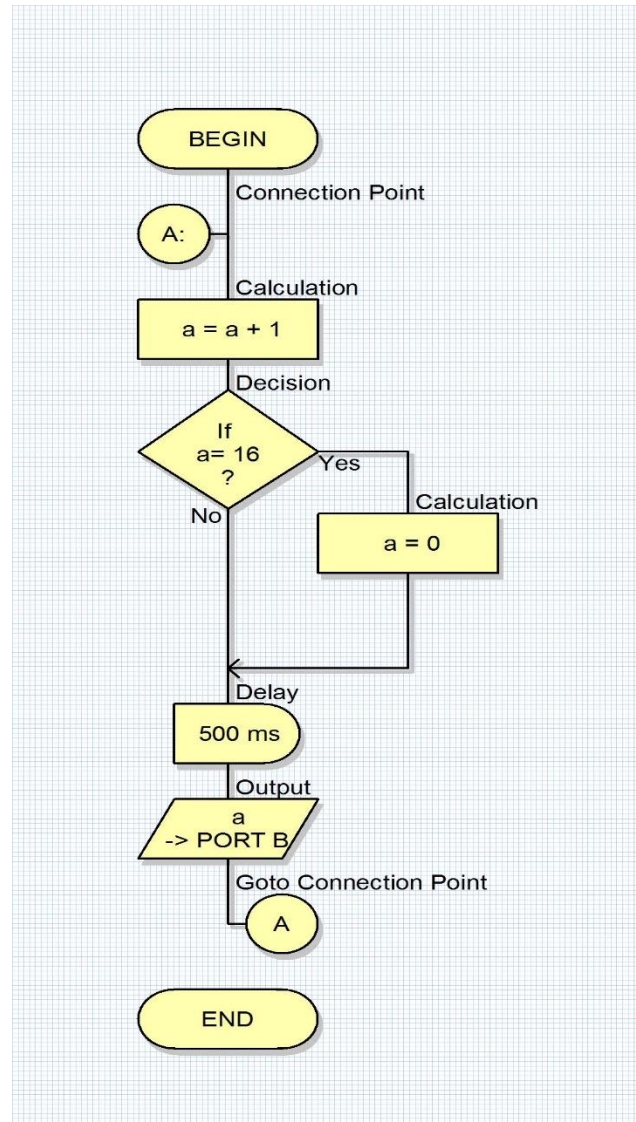
مثال:

برنامج يقوم في حالة الضغط على السويتش بإضاءة شريط من الليدات لمدة معينة ومن ثم إطفائها لمدة معينة وفي حالة عدم الضغط على السويتش يقوم بإضاءة شريط ثاني من الليدات لمدة معينة ومن ثم إطفائها لمدة معينة.



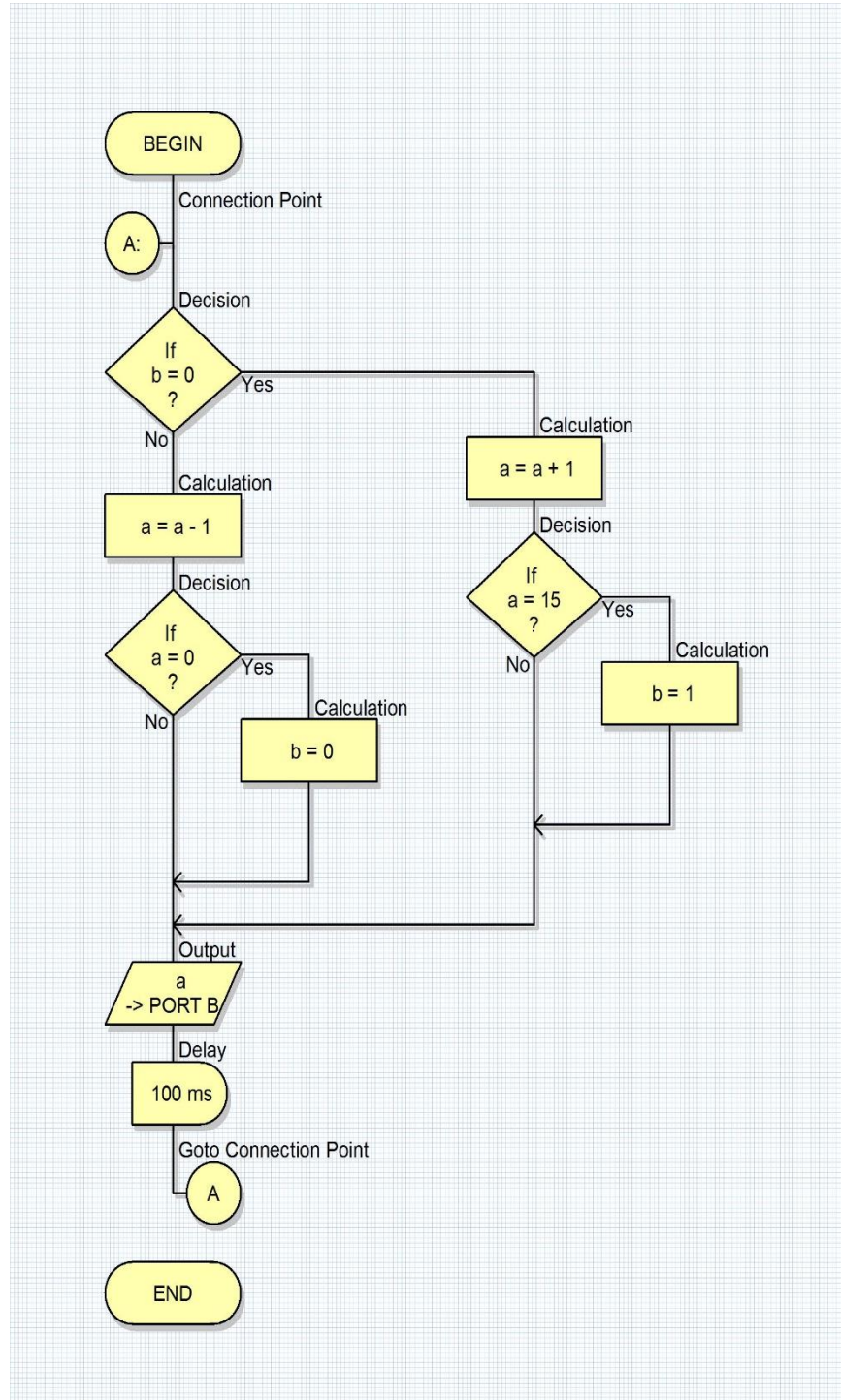
مثال:

دائرة عداد تصاعدي من 0 حتى 15

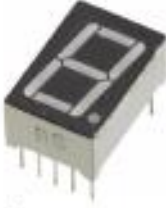


مثال:

دمج دارتي عداد تصاعدي و تنازلي بين 0 و 15



المقاطعة السباعية 7 Segment:

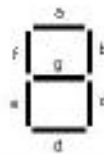


"المقاطعة السباعية - تنطق سيفين سيجمنت " 7segment وهي عبارة عن مستطيل صغير يحتوي على 7 مقاطع مضيئة باستخدام دايودات ضوئية (متوفرة باللون الأحمر والأخضر والأزرق). وتستخدم في عرض الأرقام وبعض حروف اللغة الإنجليزية.

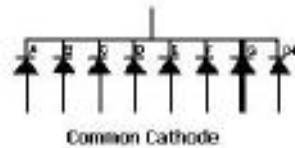
تتوفر هذه القطعة الإلكترونية في الأسواق بمختلف الأحجام فمنها ما هو صغير جداً مثل المستخدمة في الساعات الرقمية الرخيصة ومنها ما هو كبير الحجم مثل المستخدمة في إشارات المرور (اللوحة المضيئة التي تعرض الوقت المتبقي لتفتح إشارة المرور).



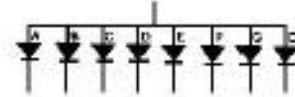
تتكون السيفين سيجمنت من 7 دايودات ضوئية متصلة ببعضها البعض إما عن طريق توصيل الطرف الموجب وتسمى common anode أو عن طريق توصيل الطرف السالب وتسمى common cathode (سنستخدم في التجارب التالية النوع common cathode). ويسمى كل دايود ضوئي بأحد حروف الأبجدية الإنجليزية A,B,C,D,E,F,G كما هو موضح في الصورة التالية:



7-segment display with segment identification.



Common Cathode



Common Anode

Equivalent circuit, with decimal point.

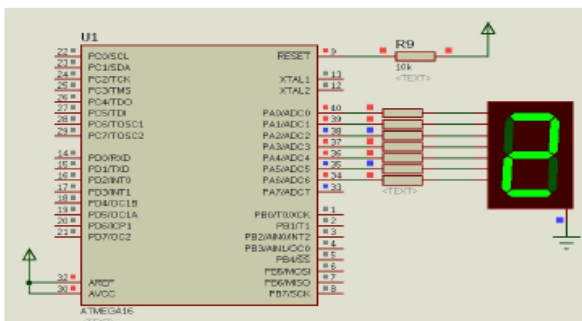
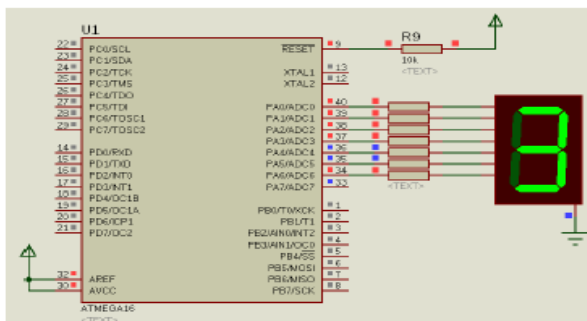
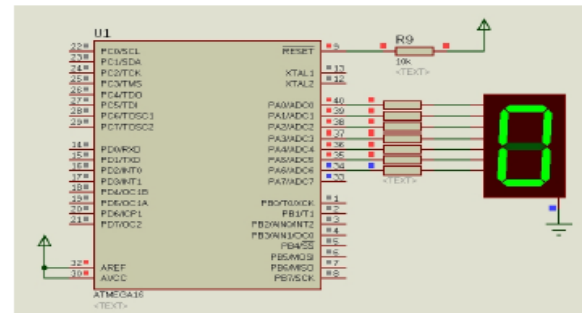
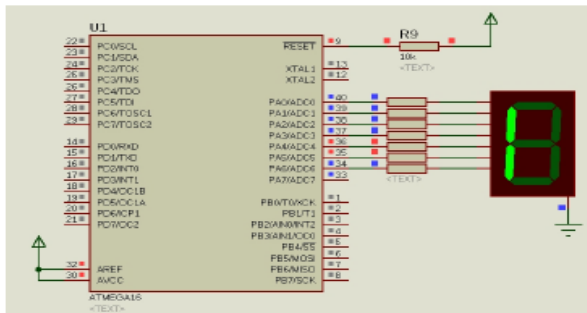
0123456789

لتشغيل هذا العنصر نقوم بتوصيل الأطراف السبعة بأحد المنافذ في المتحكم الدقيق.

تحتوي بعض 7 Segment على طرف إضافي وهو دايود ضوئي صغير موجود على الجانب الأيمن السفلي ويستخدم لعرض الفاصلة العشرية (.).
لعرض أي رقم من الأرقام العشرية سنستخدم الجدول التالي والذي يوضح الحالة التي يجب أن يكون عليها كل دايود ضوئي حتى يتم عرض رقم معين.

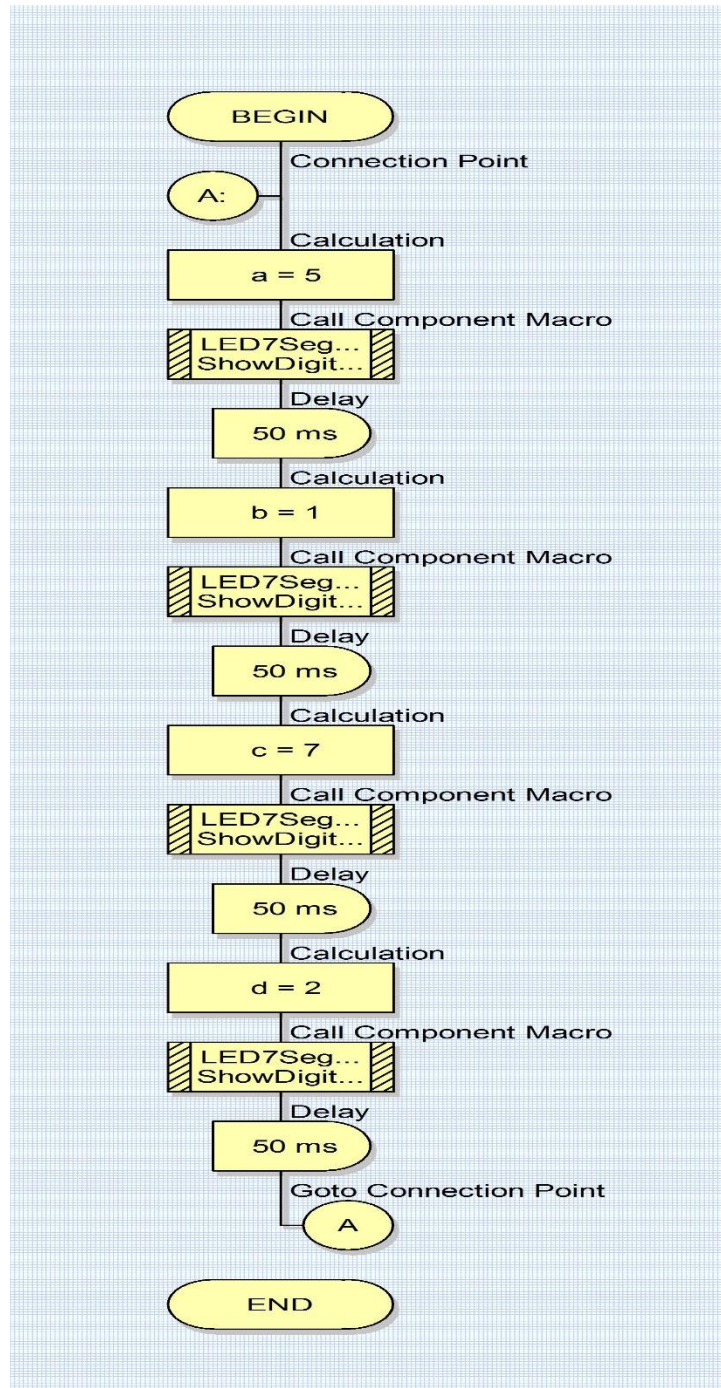
DIGIT	LEDs TO GLOW						
	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

الشكل التالي يوضح طريقة توصيل المتحكم بلسيفن سيغمنت.

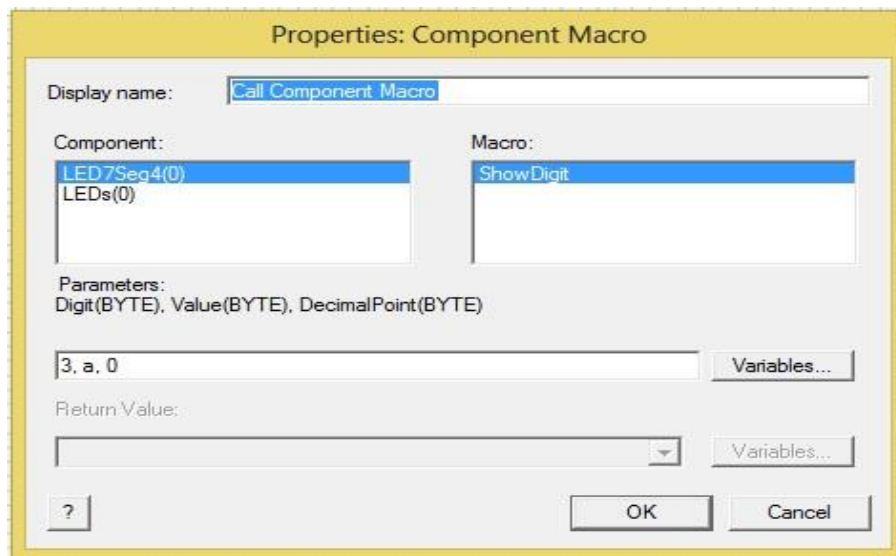


مثال:

برمجة عرض العدد ٥١٧٢ على ٤ عارضات موصولة بطريقة متوازية:

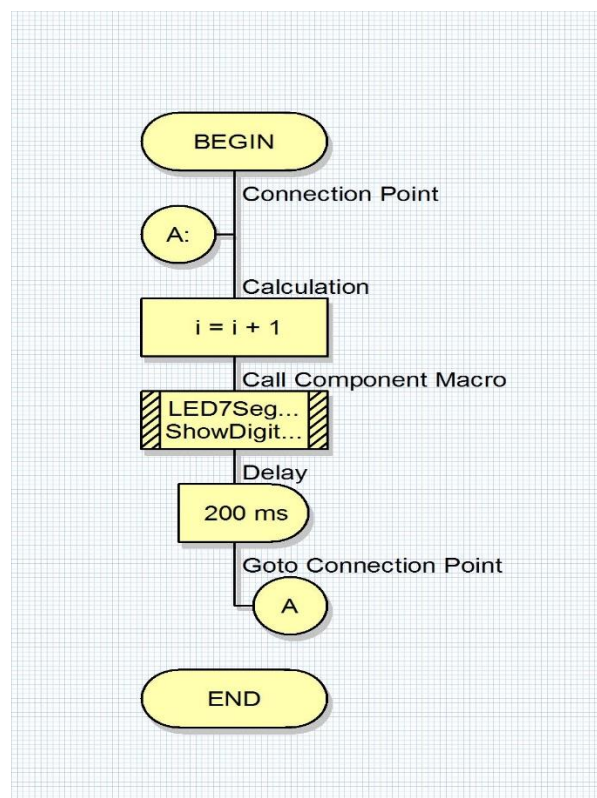


ونقوم بضبط مانريد إظهاره بكل خانة بالشكل التالي:

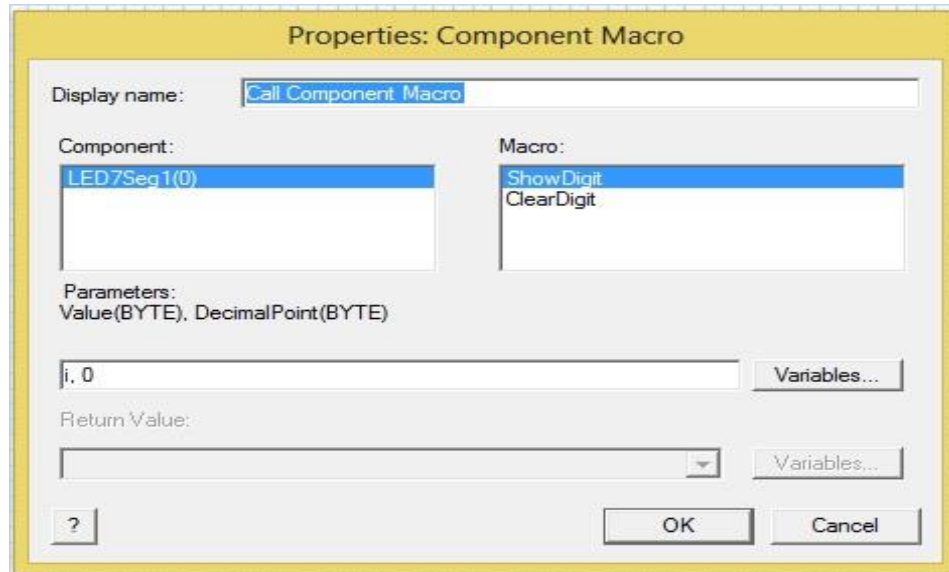


مثال:

دائرة عداد تصاعدي باستخدام العارضة السباعية 7Segment:



ونضبط إعدادات الخانة بالشكل التالي:



بعد أن تعرفنا على 7 Segment نلاحظ أن لها محدودية في عملية الإظهار، حيث أنه لا يمكننا إظهار سوى الأرقام وبعض الأحرف المحدودة فماذا لو احتجنا إلى إظهار رسالة للمستخدم تطالبه بإدخال رقم معين أو إظهار رسالة توضيحية له أو غير ذلك..؟

عندها فإننا سنستخدم شاشة الإظهار الكريستالية السائلة LCD ..

شاشة الإظهار LCD:

إن شاشة LCD هي عبارة عن شاشة مؤلفة من سطر أو أكثر يحتوي كل سطر على عدد من الخانات والخانة هي عبارة عن مربع صغير يتم إظهار المحرف عليه، أي أن كل خانة تستطيع إظهار محرف واحد فقط، وأكثر الشاشات شيوعاً هي الشاشات ذات القياسات التالية:

1*16 , 1*20 , 1*24 , 2*16 , 2*20 , 4*16 , 4*20

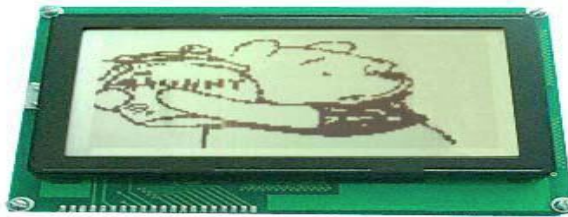
أنواع شاشات LCD:

١- شاشة المحارف CHARACTER LCD:



في هذه الشاشة كل مربع يكتب فيه رمز أو حرف أو رقم واحد فقط .

٢- شاشة الرسومات GRAPHICAL LCD:



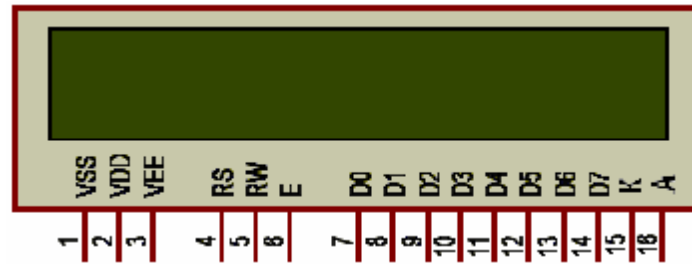
يمكن عرض أي شي عليها لكن من دون ألوان ويمكن الرسم عليها أيضاً..

٣- الشاشات الملونة COLOR LCD:



وهي مثل الموجودة في التلفزيونات والموبايلات الملونة وتقاس بالبيسكل وجودة الألوان لها.

يبين الشكل التالي أقطاب التحكم بشاشة LCD ذات السطرين والـ 16 عمود:



أقطاب شاشة الـ LCD :

القطب V_{SS} :

هو قطب التغذية لشاشة الـ LCD , وهو جهد الأرضي (0) منطقي (GND) .

القطب V_{DD} :

هو أيضاً قطب التغذية لشاشة الـ LCD , ولكن ذو القيمة (+5V) .

القطب V_{EE} :

ويدعى أيضاً بـ V_0 وهو قطب جهد التباين , ويقصد بالتباين هو حدة ظهور الرمز على الشاشة وأقل تباين أن لا نرى شيئاً على الشاشة يكون عند تطبيق (+5V) على هذا القطب , وأعلى تباين للشاشة يكون عند تطبيق (0V) على هذا القطب . ويمكن التحكم بتباين الشاشة عن طريق وصل قطب التباين (V_0) إلى مقاومة متغيرة .

القطب RS :

وهو مسجل اختيار الدخول لشاشة الـ LCD وذلك في حال طبق عليه (0) منطقي عندها نريد إرسال كلمة تحكم , بينما في حال طبق (1) منطقي فعندها نريد إرسال معطيات إلى الشاشة , وفي الحالتين عن طريق أقطاب المعطيات .

القطب R/W :

وهو للقراءة أو الكتابة إلى الشاشة , نطبق (0) منطقي على هذا القطب عندما نريد كتابة (إرسال) المعلومات إلى شاشة الـ LCD , ونطبق (1) منطقي عندما نريد قراءة (استقبال) المعلومات من شاشة الـ LCD .

القطب E :

وهو قطب تمكين شاشة الـ LCD , فكل معلومة يتم كتابتها أو قراءتها من شاشة الـ LCD يجب إرفاقها بنبضة تمكين على هذا القطب , ونبضة التمكين هذه تحدث عند الجبهة الهابطة , أي تتم هذه النبضة برفع القطب إلى الـ (1) منطقي وإنزاله إلى الـ (0) منطقي بعد تأخير مناسب .

الأقطاب DB7 ... DB0 :

وهي أقطاب المعطيات (DATA) , حيث يتم كتابة المعطيات أو كلمات التحكم عبر هذه الخطوط إلى شاشة الـ LCD وكذلك قراءة المعطيات منها ...

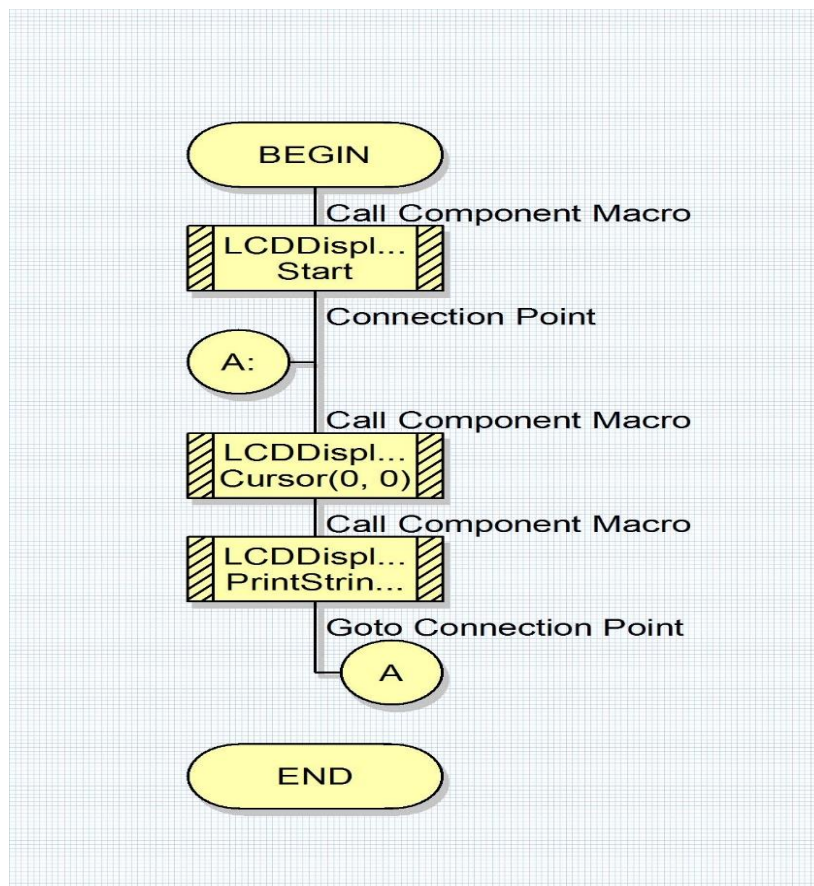
القطبين K و A :

تزود بعض الشاشات بهذين القطبين , وهما على الترتيب قطبي المهبط و المصعد لليد المسطح المستطيل الموجود خلف شاشة الـ LCD , والذي يضاء عند تطبيق الـ (1) منطقي على المصعد (A) و الـ (0) منطقي على المهبط (K) .

وظيفة هذا الليد هو تأمين الإضاءة الكافية لشاشة الـ LCD ليتمكن المستخدم من رؤية العبارات المكتوبة عليها في ظلمة الليل ...

مثال:

برنامج يقوم بعرض "I love Damascus" على شاشة LCD.

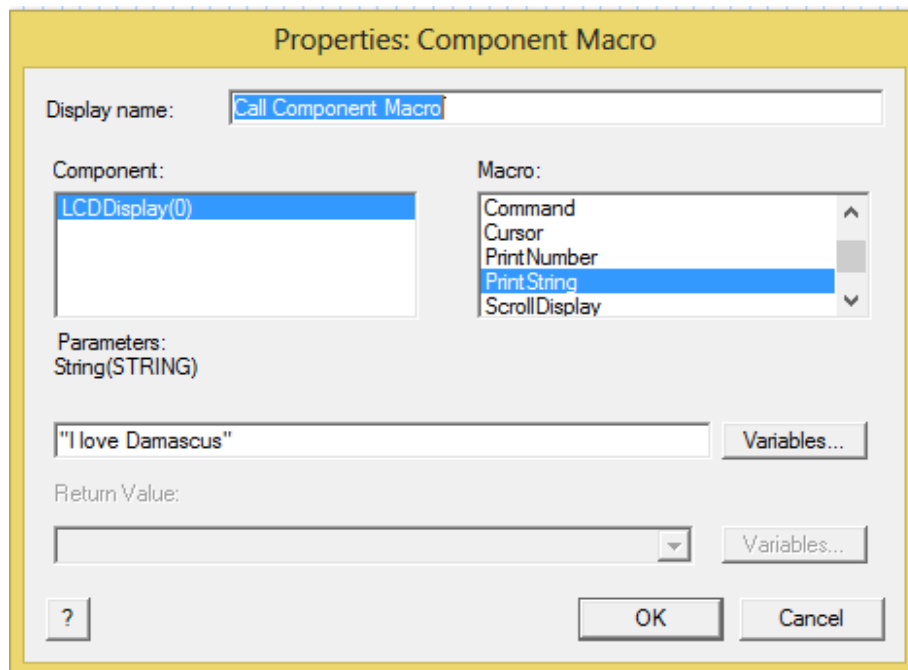


I	l	o	v	e	D	a	m	a	s	c	u	s

❖ أهم دوال شاشة الإظهار LCD التي نحتاجها خلال عملية البرمجة هي:

- ١- دالة البدء Start: لنهيء البرنامج لاستخدام شاشة الإظهار.
- ٢- مؤشر الكتابة Cusror: لتحديد مكان الكتابة على الشاشة.
- ٣- طباعة رقم PrintNumber: لنوضح للبرنامج أنه سيكتب حروف وأرقام.
- ٤- طباعة سلسلة PrintString: لنوضح للبرنامج أنه سيكتب جمل.
- ٥- دالة المسح Clear: لمسح المكتوب على الشاشة.

والتي يمكن ضبطها بالشكل التالي:



لوحة المفاتيح Keyboard :

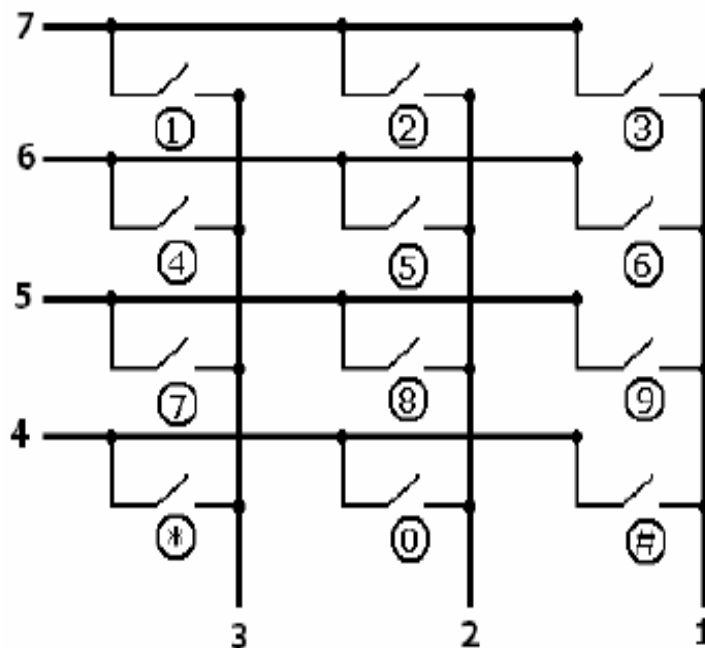
وصل لوحة المفاتيح إلى المتحكم الصغري :

كيف يمكن وصل لوحة المفاتيح إلى المتحكم الصغري ... ؟ لابد أن هذا السؤال قد تبادر إلى ذهنك أثناء دراستك للـ Microcontroller , فكيف يتم ذلك وأين تجد الإجابة عليه ... ؟ بالطبع ستجده في جلستنا هذه فتعال معنا يا صديقي لنكتشف سوية لغز لوحة المفاتيح ...



ليكن لدينا لوحة من المفاتيح اللحظية والمؤلفة من 12 مفتاحاً (لوحة مفاتيح الهاتف) , وبفرض أننا نريد وصله مع المتحكم الصغري , فإن أول ما يفكر به المرء هو أن يصل كل مفتاح إلى قطب دخل في المتحكم وبالتالي سنحتاج إلى 12 قطب دخل وهذا استهلاك كبير لأقطاب دخل , وبالتالي ما هو الحل الأمثل برأيك لربط هذا اللوح مع هذا المتحكم ... ؟

إن الحل الأمثل ... لا بل المبدأ العام لربط أية لوحة مفاتيح مهما كانت عدد أزرارها مع المتحكم الصغري هي كالتالي :
نقوم بوصل المفاتيح بالشكل التالي :



نلاحظ أنه تم اختصار الـ 12 خط إلى 7 خطوط فقط والتي هي عبارة عن 3 أعمدة و 4 أسطر , وبالتالي فإن أي كيبورد يمكن وصله مع المتحكم بخطوط عددها هو (عدد الأعمدة + عدد الأسطر) , فمثلاً : ليكن لدينا كيبورد مؤلف من 7 أعمدة و 5 أسطر , فيمكن أن يوصل مع المتحكم عن طريق ($7 + 5 = 12$) خط أي لوحة مفاتيح مؤلفة من 35 مفتاح .. لاحظ الاختصار الكبير في عدد الخطوط .

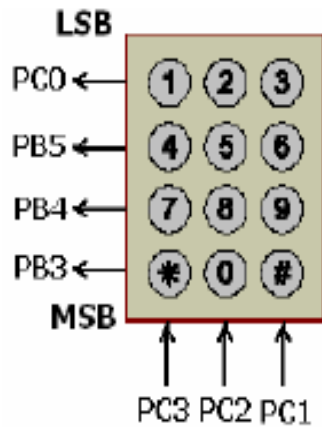
لكن كيف يمكن المتحكم أن يعرف أيّاً من الأزرار تم الضغط عليه ... ؟

للجواب على ذلك سنقوم بعملية مسح لأعمدة الكيبورد عموداً .. عموداً , وبالتالي يقوم المتحكم بفحص الأزرار في العمود الأول بعدها سيقوم بفحص الأزرار في العمود الثاني , ثم يفحص الأزرار في العمود الثالث , وبذلك في كل عملية مسح يقوم بها المتحكم يفحص عموداً واحداً فقط , وتكرر هذه العملية باستمرار من قبل المتحكم الصغري , تجري عملية المسح هذه بأن نجعل المتحكم يقوم بإرسال القيمة (0) منطقي إلى العمود الأول (ذو الرقم 1) فقط أما باقي الأعمدة فتُرسل لها القيمة (1) منطقي في عملية المسح التالية سنقوم بإرسال القيمة (0) منطقي إلى العمود الثاني فقط أما باقي الأعمدة سترسل لها القيمة (1) منطقي , ثم بعد ذلك في المرة الثالثة سيتم إرسال القيمة (0) منطقي إلى العمود الثالث فقط والقيمة (1) منطقي للأعمدة الباقية , وبالمقابل سيتم تفعيل خطوط الأسطر جميعها بالقيمة (1) منطقي من قبل المتحكم . في حال الضغط على أحد المفاتيح فإن القيمة (0) منطقي والتي تم تفعيل العمود بها ستنتقل إلى السطر الموافق , وتقلب القيمة المتوقعة على السطر الموافق للزر المضغوط من (1) إلى (0) .

وعندها نلاحظ بتشكيل شيفرة مختلفة عند ضغط كل مفتاح و مستقلة عن شيفرات المفاتيح الأخرى , هذه الشيفرة مكونة من أرقام و وحدات الأسطر والأعمدة مجتمعة مع بعضها البعض .

ولفهم هذه التقنية جيداً سنقوم بطرح المثال النموذجي التالي :

ليكن لدينا لوحة المفاتيح الموضحة جانباً :



تحتوي هذه اللوحة على 12 مفتاح (لوحة مفاتيح الهاتف النموذجية) , وهي مؤلفة من أربعة أسطر وثلاثة أعمدة , و سيتم وصل هذه اللوحة مع المتحكم الصغري AT90S4433 على الشكل التالي :

- PC1 , PC2 , PC3 هي مخارج للمتحكم السابق ومداخل للوحة المفاتيح (الأعمدة) والتي يتم من خلالها إجراء عملية المسح .
- PC0 , PB5 , PB4 , PB3 هي مداخل للمتحكم ومخارج للوحة المفاتيح (الأسطر) , وبما أنها مداخل للمتحكم فيتوجب على أن أقوم برفع مقاومات الرفع الداخلية للأقطاب السابقة , وبالتالي يتوضع عليها في الحالة الطبيعية (1) منطقي وهذا هو المطلوب يا سيدي ...

الآن نقوم بتطبيق القيمة (0) منطقي على العمود الأول عن طريق القطب (PC1) , ونطبق على العمودين الآخرين القيمة (1) منطقي عن طريق القطبين (PC2) و (PC3) , والآن عندما يتم الضغط على أحد مفاتيح العمود الأول من لوحة المفاتيح (3 أو 6 أو 9 أو #) سوف يتم وصل الصفر منطقي عبر المفتاح المضغوط إلى أحد الأقطاب (PC0 أو PB5 أو PB4 أو PB3) على الترتيب , وبالتالي سيتغير ال (1) منطقي المتوضع على القطب ليصبح (0) منطقي , ونكون بذلك قد حددنا الزر المضغوط عن طريق تغير أحد أقطاب الأسطر من (1) إلى (0) منطقي .

فبعد أن نقوم بمسح العمود الأول نطبق (0) منطقي على العمود الثاني عن طريق القطب (PC2) مع تطبيق (1) منطقي على العمودين الآخرين عن طريق القطبين (PC1) و (PC3) ... وهكذا بنفس الطريقة نقوم بمسح العمود الثالث عن طريق القطب (PC3) , وبالتالي نكون قد مسحنا كامل لوحة المفاتيح .

أي بكلام آخر :

عند تطبيق القيمة $B(110)$ على الأعمدة (تفعيل العمود الأول) :

- إذا نتج على الأسطر القيمة $B(1110) = (E)_H$; فهذا يعني أن المفتاح (3) مضغوط .
- إذا نتج على الأسطر القيمة $B(1101) = (D)_H$; فهذا يعني أن المفتاح (6) مضغوط .
- إذا نتج على الأسطر القيمة $B(1011) = (B)_H$; فهذا يعني أن المفتاح (9) مضغوط .

- إذا نتج على الأسطر القيمة $(0111)_B$; فهذا يعني أن المفتاح (#) مضغوط .
- إذا نتج على الأسطر القيمة $(1111)_B$; فهذا يعني أنه لم يتم الضغط على أي مفتاح .

عند تطبيق القيمة $(101)_B$ على الأعمدة (تفعيل العمود الثاني) :

- إذا نتج على الأسطر القيمة $(1110)_B$; فهذا يعني أن المفتاح (2) مضغوط .
- إذا نتج على الأسطر القيمة $(1101)_B$; فهذا يعني أن المفتاح (5) مضغوط .
- إذا نتج على الأسطر القيمة $(1011)_B$; فهذا يعني أن المفتاح (8) مضغوط .
- إذا نتج على الأسطر القيمة $(0111)_B$; فهذا يعني أن المفتاح (0) مضغوط .
- إذا نتج على الأسطر القيمة $(1111)_B$; فهذا يعني أنه لم يتم الضغط على أي مفتاح .

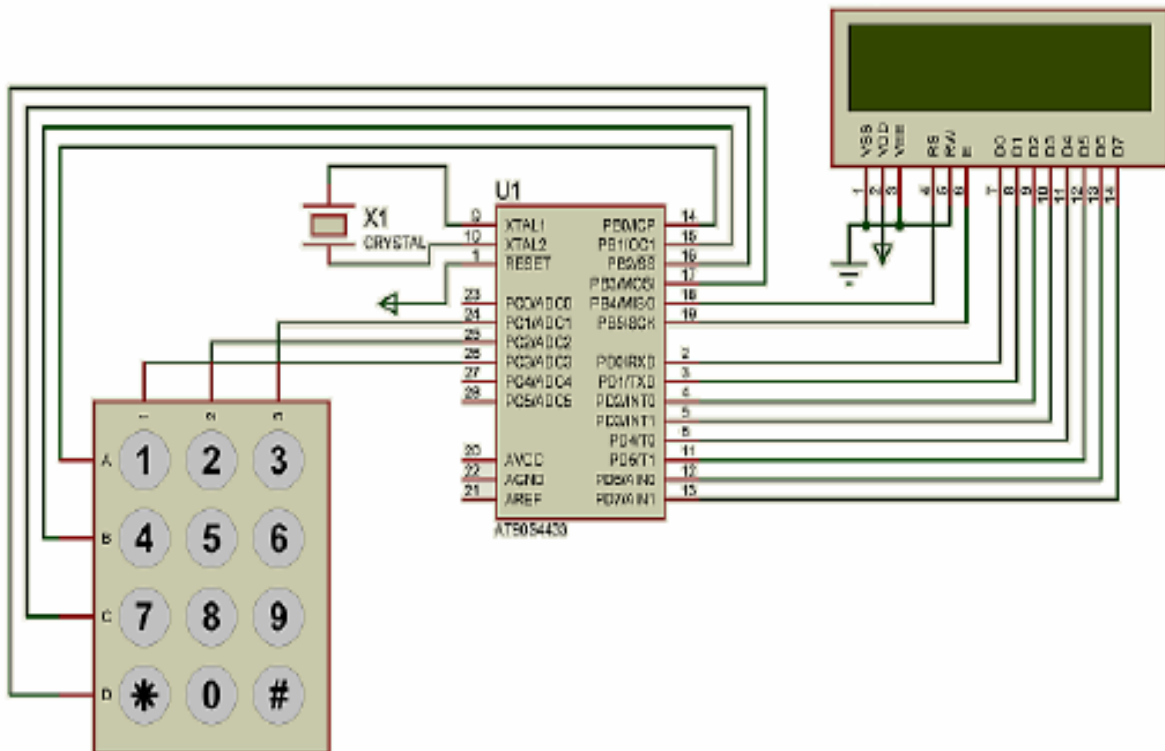
عند تطبيق القيمة $(011)_B$ على الأعمدة (تفعيل العمود الثالث) :

- إذا نتج على الأسطر القيمة $(1110)_B$; فهذا يعني أن المفتاح (1) مضغوط .
 - إذا نتج على الأسطر القيمة $(1101)_B$; فهذا يعني أن المفتاح (4) مضغوط .
 - إذا نتج على الأسطر القيمة $(1011)_B$; فهذا يعني أن المفتاح (7) مضغوط .
 - إذا نتج على الأسطر القيمة $(0111)_B$; فهذا يعني أن المفتاح (*) مضغوط .
 - إذا نتج على الأسطر القيمة $(1111)_B$; فهذا يعني أنه لم يتم الضغط على أي مفتاح .
- وبهذه الطريقة يتم مسح كامل المفاتيح لفحص أي منها يكون مضغوط ، ونستطيع باستخدام مفاتيح لحظية أن نشكل لوحة مفاتيح بالعدد الذي نريده من الأزرار لكن على أن يتم وصلهم بالشكل السابق الذكر .

مثال تطبيقي :

المطلوب تصميم دائرة مؤلفة من لوحة مفاتيح وشاشة LCD و يتم ربطهما مع المتحكم AT90S4433 . بحيث في حال الضغط على أحد المفاتيح (ولتكن لوحة مفاتيح هاتف) يظهر الرمز على شاشة الـ LCD مع اتخاذ كامل الإجراءات اللازمة لتهيئة شاشة الـ LCD بالشكل الذي تراه مناسباً .

الدائرة :



مثال:

برنامج يقوم بكتابة الرقم أو الرمز الذي تضغطه بالكيورد على شاشة LCD.

