الجامعة السورية الخاصة
SYRIAN PRIVATE UNIVERSITY

شاشة نقطية لإظهار الرسائل النصية

Dot Matrix Text Message Display

تقرير أعد في إطار المشروع الفصلي - كلية هندسة الحاسوب والمعلوماتية

قسم هندسة الاتصالات والشبكات - قسم هندسة المتحكمات

إعداد:

هشام صالح مزهر

جلال أديب عقيل

عمرو لؤي نعيم

جولي بسام قروشان

إشراف:

د.علي سكاف

جميع حقوق النشر © محفوظة للجامعة السورية الخاصة

تشرين الثاني 2013

الشكر و التقدير: بكل محبة وإحترام ...

نتوجه بالشكر إلى الكادر التدريسي في الجامعة السورية الخاصة من دكاترة ومهندسين على جهودهم المبذولة وعطائهم الكبير ، الذي أوصلنا الى هذه المرحلة من المعرفة.

ونخص بالشكر الدكتور المهندس علي سكاف المشرف على المشروع على أتعابه معنا ومتابعتنا الدائمة ودعمه الكامل بكل المعلومات اللازمة .

الخلاصة

تعد النظم المضمنة من أكثر المواضيع التي تلقى اهتماماً في التطبيقات العملية المختلفة و خصوصاً بعد انتشار المتحكمات الصغرية على نطاق واسع ، تعتبر لوحات الإظهار الضوئية المتحركة أحد الحلول الإقتصادية والدعاية أو تطبيقات عرض المعلومات لإستخدامها في المحال التجارية، المطاعم الصالونات، المعارض، المطارات ومحطات السكك الحديدية، هي شاشات قليلة التكلفة تمتاز بدقتها العالية و ألوانها اللطيفة، و سهولة عرض الإعلانات، و الترويج للمنتجات من خلالها.

نعرض في هذا التقرير نتيجة عملنا في إطار مقرر المشروع الفصلي في كلية هندسة الحاسوب والمعلوماتية من الجامعة السورية الخاصة و الذي يتناول تصميم و تنفيذ شاشة عرض نقطية تقوم بإظهار الرسائل النصية، مبنية حول متحكم صغري .

و يكون تحميل الرسائل المعروضة على الشاشة المنفذة عن طريق تطبيق برمجي طورناه على الحاسب من خلال الإتصال عبر المنفذ التسلسلي RS232، يستقبل المتحكم الصغري البيانات و يقوم بعرضها على شاشة نقطية أحادية اللون ، بدقة (8 x24) كشريط نصي متحرك.

جدول المحتويات:

المقدمة.....6

الفصل الأول: دفتر الشروط.....7

الفصل الثاني: بنية أقسام المشروع 8
2.1. الكيان الصلب لشاشة العرض 8
2.2. برنامج المتحكم 9
2.3. برنامج الحاسب 10
2.3.1. خوارزمية العمل 11
الفصل الثالث: التنفيذ العملي 15
3.1. وحدة التغذية 15
3.2. المتحكم ATMEGA 16
3.2.1. لمحة عن المتحكم 16
3.2.2. شرح أقطاب المتحكم 16
3.3. الدارة المتكاملة IC SN74HC595N 18
3.4. شاشة العرض النقطية 19
الفصل الرابع: توصيل الدارة 21
الفصل الخامس: النتائج العملية 24
الخاتمة 25
المراجع 27
ملحقات 27
برنامج الحاسب 27
برنامج المتحكم 32

جدول الأشكال

الشكل 1 عمل سير المشروع 7
الشكل 2 الكيان الصلب لشاشة العرض 8
الشكل 3 إدخال النص 12

الشكل 4 إظهار النص..... 12
الشكل 5 إيقاف العرض..... 12
الشكل 6 مسح النص 12
الشكل 7 حركة الدارة عبر واجهة البرنامج 13
الشكل 8 عكس عملية الاظهار..... 13
الشكل 9 رسالة خطأ 14
الشكل 10 وحدة التغذية..... 15
الشكل 11 المتحكم ATMEGA8L 16
الشكل 12 PDIP..... 18
الشكل 13 الدارة المتكاملة SN74HC595N..... 19
الشكل 14 شاشة العرض النقطية..... 20
الشكل 15 واجهة البرنامج PROTOS..... 21

المقدمة

الأنظمة المضمنة أو الأنظمة المدفونة أو الأنظمة المخبأة ..... كلها تعبيرات تشير إلى نفس المعنى وهي

الأنظمة الالكترونية , التي تحتوي ليس فقط على دوائر الكترونية ، تحتوي أيضاً على برمجيات تكون

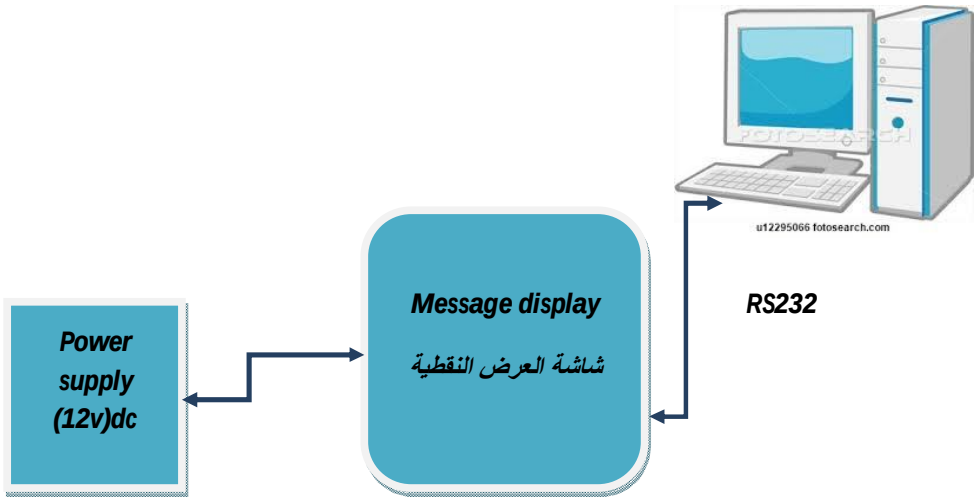
مخزنة على الدوائر الالكترونية و تعمل من خلالها على تنفيذ التطبيق المطلوب كمعالجة البيانات وإتخاذ القرارالمناسب، توجد الأنظمة المضمنة اليوم في العديد من الأجهزة حَولَنا و ذلك لسرعتها و دقة أدائها، و سهولة تنفيذها نسبياً بالمقارنة بالأجهزة الالكترونية التقليدية، و مثل هذه الأجهزة الهواتف المحمولة و أجهزة الملاحة و التحكم بالطائرات و السفن و السيارات , و الأجهزة المنزلية كالتلفاز و الفيديو وحتى الغسالات و الثلاجات . و وجود البرمجيات بهذه الأجهزة جميعاً يجعلها أكثر مرونة و ملائمة للتطبيق و بالتالي

أكثر فاعلية و توفيراً للطاقة و أسهل في تحديد الأعطال والصيانة .

يهدف هذا العمل إلى تصميم شاشة عرض نقطية و قيادتها عبر المتحكم، و هذا التطبيق الذي قمنا ببنائه هو إحدى الوسائل التي تقوم بتسهيل الإعلان للمنتجات و ترويجها و دعمناه من خلال إعطاءه القدرة على الإدخال عبرَ الحاسب و يظهر الخرج على الشاشة.

نتناول في الفصل الأول مواصفات المشروع ، لننتقل للفصل الثاني لنعرض بنية أقسامه: البرنامج المستخدم في المتحكم و برنامج الحاسب و الكيان الصلب لشاشة العرض، من ثم نعرض في الفصل الثالث التنفيذ العملي والقطع المستخدمة في الدارة ، وأخيراً نغطي النتائج العملية في الفصل الرابع .

الفصل الأول : دفتر الشروط (مواصفات المشروع)



الشكل (1) عمل سير المشروع

نقوم بإدخال النص المراد عرضه عبر الحاسب , ومن ثم تجرى عليه عملية المعالجة , ليظهر النص على شاشة العرض النقطية من النوع (24x8) نقطة , المؤلفة من 8 أسطر و 24 عاموداً .

الفصل الثاني: بنية أقسام المشروع

نميز فيها ثلاثة أجزاء :

- الكيان الصلب لشاشة العرض .
- برنامج المتحكم .
- برنامج الحاسب.

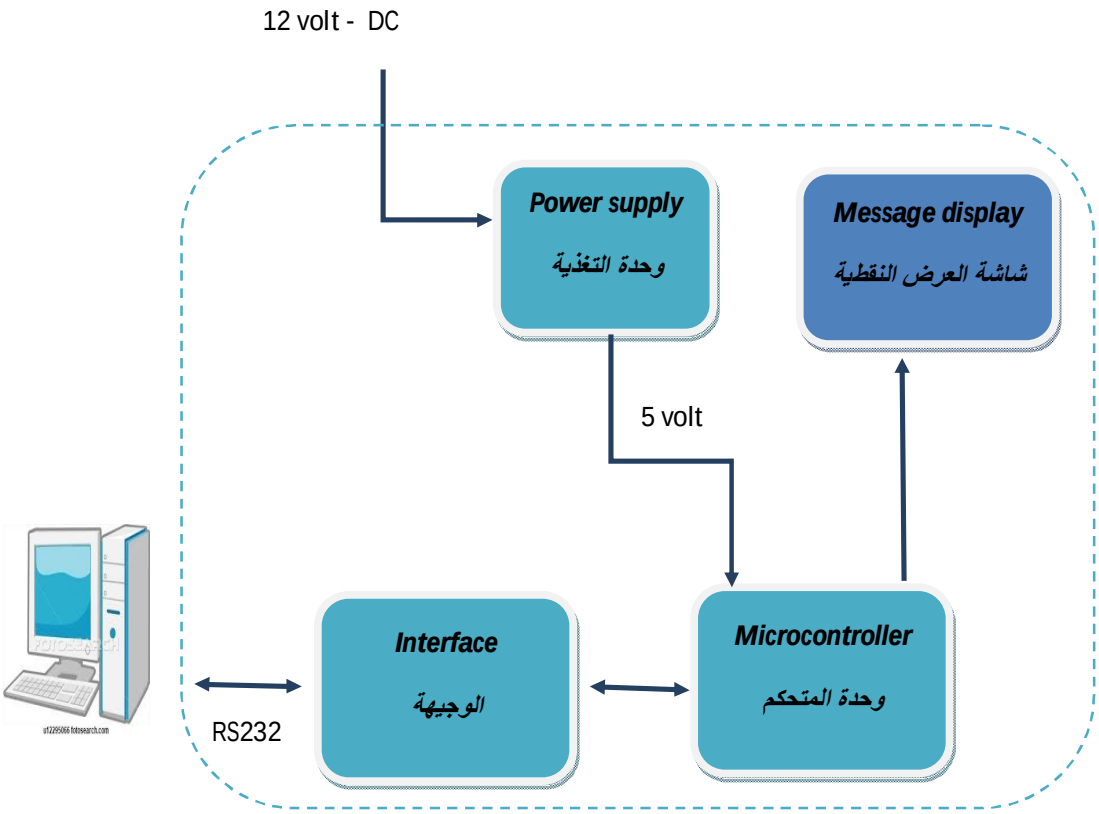
2.1. الكيان الصلب لشاشة العرض :

يتكون الكيان الصلب من أربع كتل هي:

وحدة التغذية التي تؤمن 5 V للمتحكم و لشاشة الإظهار انطلاقا من 12 V , و بالمواصفات التي ذكرناها سابقاً

و وحدة التحكم التي تقود شاشة الإظهار وتتخاطب مع الحاسب عبر كتلة الوجيهة.

يمكننا أن نعطي المخطط الصندوقي للكيان الصلب اللازم كما موضح في الشكل (2)

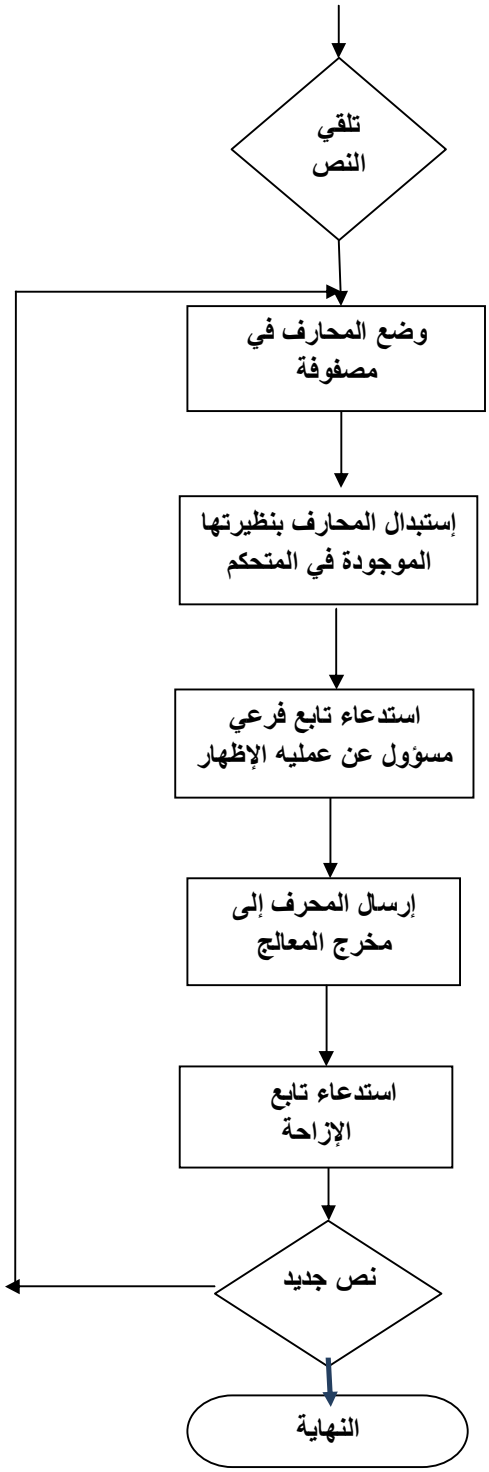


الشكل (2) الكيان الصلب لشاشة العرض

2.2. برنامج المتحكم :

بدايةً تقوم الدارة بتلقي النص المراد إظهاره على الشاشة من الحاسب , حيث تقوم بمعالجة البيانات المرسلة من الحاسب و تحويلها للشكل الذي تفهمه الدارة و ترسله إلى الشاشة , و تتم عمليات المعالجة حسب المخطط الإنسيابي التالي:

نقطة البدء



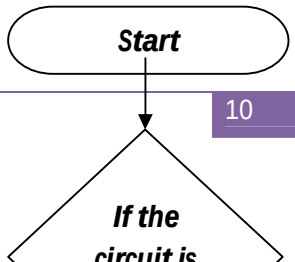
2.3. برنامج الحاسب :

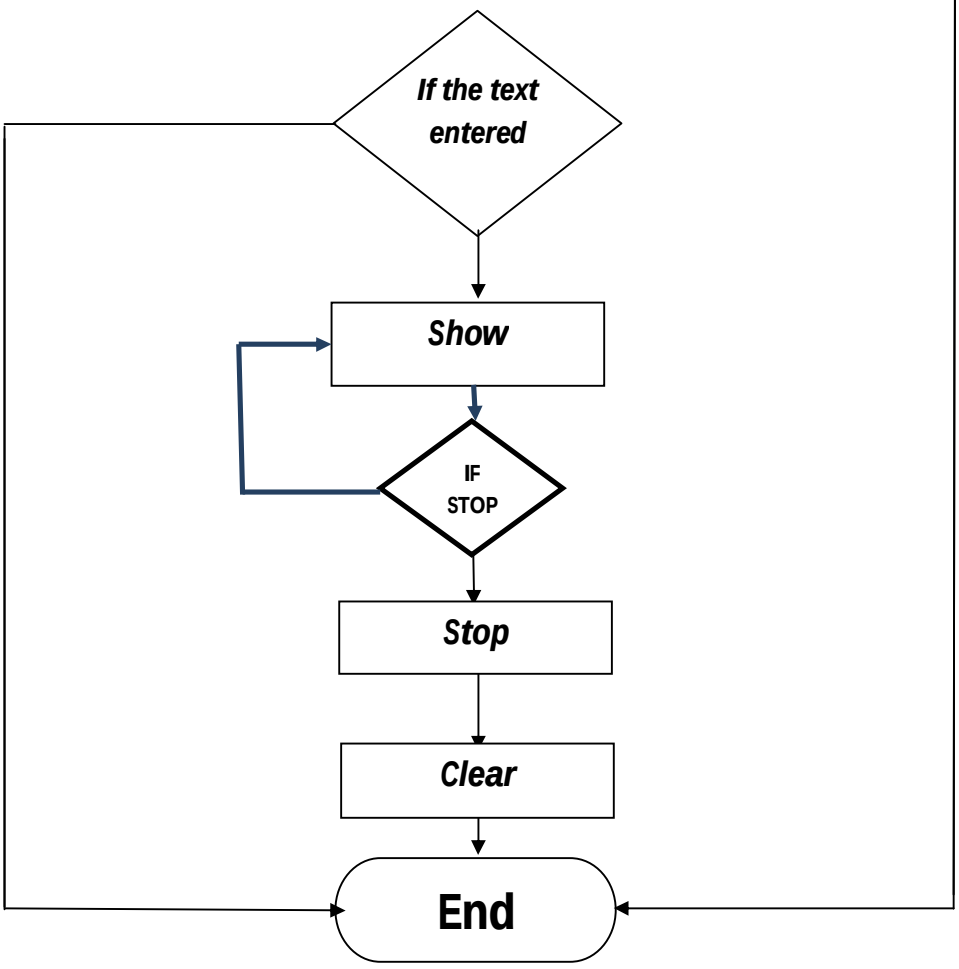
البرنامج الذي قمنا بتصميمه للاتصال بالدارة عبر الحاسب وتطبيق العمليات المطلوبة، حيث نقوم بإدخال النص المطلوب إظهاره و ارساله عبر مدخل RS232 إلى الدارة ، و اللغة البرمجية المتبعة في هذا البرنامج هي C# .

العمليات الموجودة في البرنامج هي :

ندخل النص عبر برنامج الحاسب إذا كانت الدارة موصولة يتم الإرسال عبر المنفذ USB. تتلقى الدارة الرسالة القادمة من الحاسب و إذا لم تتلقاها يظهر على الحاسب رسالة تنبيه بعدم اتصال الحاسب بالدارة عبر المنفذ.
هنا لدينا حالتين : الرسالة وصلت للدارة ودخلت الحلقة , أو لم تصل .
إن لم يدخل النص في الحلقة فوراً يتحول على إيقاف العرض . أما إذا دخلت الرسالة على الحاسب تظهر شاشة صغيرة في واجهة البرنامج توضح لنا حركة الرسالة . و بعد الإنتهاء من عرض الرسالة نقوم بإيقاف أو بمسح الرسالة وهنا يتوقف العرض لندخل رسالة أخرى .

2.3.1. خوارزمية العمل :





و هنا نعرض شرح بسيط ل واجهات البرنامج :

1- إدخال النص:

الشكل (3) إدخال النص

نقوم هنا بإدخال النص متضمناً شرطاً , إذا كان المربع النصي فارغ لن تتفعل خانة الإظهار.

2- إظهار النص على الدارة:

الشكل (4) إظهار النص

نقوم عبر هذه الإجرائية بإرسال النص إلى الدارة , وإظهاره عليها وعلى شاشة العرض الداخلية في البرنامج أيضاً ، متضمناً شرط عدم تفعيل خانة الإيقاف والمسح إذا لم يتم الضغط عليه.

3- إيقاف العرض على الدارة:

الشكل (5) إيقاف العرض

تقوم هذه الإجرائية بإيقاف العرض على شاشة الدارة والشاشة الداخلية في البرنامج.

4- مسح النص من الدارة:

الشكل (6) مسح النص

تقوم هذه الإجرائية بمسح النص من الدارة ومن شاشة العرض الداخلية .

5- شاشة صغيرة تظهر حركة الدارة عبر واجهة البرنامج:

الشكل (7) حركة الدارة عبر
الواجهة

استخدمنا في هذه الإجرائية مكتبة ال Graph كي تظهر الجملة متحركة على شاشة البرنامج الداخلية كما هي على شاشة الدارة، إضافة إلى خانة Inverse Color كي تنعكس عملية الإظهار كما هو موضحاً في الرسم التوضيحي اللاحق .

الشكل (8) عكس عملية لإظهار

6- رسالة خطأ في حال لم تكن الدارة موصولة على الجهاز:

الشكل (9) رسالة خطأ

يظهر هذا المربع في حال لم تكن الدارة موصولة على الحاسب , وقمنا بإدخال هذه الإجرائية ضمن إجرائية

الإظهار (show) حيث يكون شرط لتطبيق عملية الإظهار .

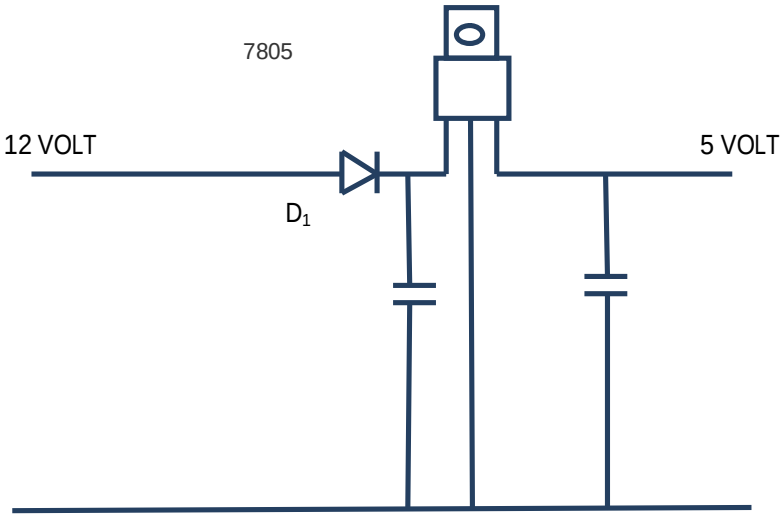
7- إيجاد المدخل الموصولة عليه الدارة في الحاسب وإرسال البيانات عبره:

وتقوم هذه الإجرائية بالدوران على جميع مداخل الحاسب ودراسة إذا كان موصول عليها الدارة أم لا و في حال إيجاد مدخل موصولة الدارة عليه تقوم بإرسال البيانات عبره ، وهذه الإجرائية ضمن إجرائية الإظهار (show) .

الفصل الثالث: التنفيذ العملي

سنقوم هنا بشرح تفاصيل الكتل المكونة للكيان الصلب لشاشة العرض :

3.1. وحدة التغذية : هي الوحدة الخاصة بإمداد التيار الكهربائي اللازم لأجزاء الحاسب بالجهد والأمبير المطلوبين كل حسب حاجته.



الشكل (10) وحدة التغذية

يظهر الشكل تفاصيل وحدة التغذية المستخدمة ,و قمنا بوضع الديود D_1 لتجنب أي إشكال قلب قطبية المدخل لحماية الدارة من أي عطل محتمل .

3.2. المتحكم : ATMEGA8L (ATMEL AVR) [1]....

الشكل (11) المتحكم
ATMega8L

3.2.1. لمحة: ....[1]

استخدمت عائلة AVR مزايا بنية ال RISC المحسنة , حيث تتمتع بالأداء العالي و بطاقة بنية ال RISC المنخفضة , تحتوي قائمة التعليمات على (130) تعليمة , ينفذ معظمها خلال دورة ساعة واحدة . تحتوي على (32) مسجل عمل , تستخدم للأغراض العامة , و تتصرف فيهم كما تشاء , و سرعة التنفيذ تصل إلى 8 مليون تعليمة في الثانية عند تردد(8MH) . لديها ذاكرة معطيات و ذاكرة برنامج غير قابلة للزوال . لديه ذاكرة برنامج من نوع فلاش مبنية داخل الشريحة حجمها (4Kbyte) قابلة لإعادة البرمجة 1000 دورة كتابة و مسح . ذاكرة معطيات داخلية RAM بطول 128 byte , ذاكرة معطيات داخلية RAM بطول 128 byte ذاكرة معطيات EEPROM بطول (256 byte) , كما تحوي على أقفال برمجية لحماية كل من ذاكرة البرنامج الفلاش و ذاكرة المعطيات , و مؤقت (عداد) بطول 8 بت عدد 2 بمقسم تردد Prescaler منفصل بنمط مقارنة واحد , و مؤقت (عداد) بطول 16 byte بمقسم تردد Prescaler منفصل , ونمط مقارنة و نمط المسك , يحتوي على ثلاث قنوات تعديل عرض Pwm , ومقارن تشابهي مبني على شريحة المتحكم , نافذة تسلسلية ثنائية الاتجاه UART , و نافذة تسلسلية ثنائية الأسلاك I2C 2WIRE , القيادة بالوصلة التسلسلية SPI (قائد / مقاد) , و ستة قنوات للتحويل التشابهي الرقمي ADC بدقة 10bit .

[1].... المتحكم ATMEL ATMEGA8L : http://www.atmel.com/devices/atmega8.aspx

3.2.2.شرح أقطاب المتحكم :

- قطب جهد التغذية : Vcc

- قطب التغذية الصفري : GND

- النافذة (pb0 ....pb5) : وهي عبارة عن نافذة دخل وخرج ذات ستة أقطاب ثنائية الإتجاه و زودت هذه النافذة بمقاومة رفع داخلية , و بإمكان دائرة قيادة الخرج النافذة b تقديم تيار قدره 20MA ,

وعند استخدامها كدخل فإنها ستصبح منبعا للتيار إذا كانت مقاومات الرفع الداخلية ممكنة , و ذلك عند ربط أقطاب هذه النافذة خارجياً مع المنطق المنخفض , و تتمتع أقطاب النافذة b بمزايا خاصة أخرى بعمل المتحكم 8mega , وعند حدوث حالات التصفير فإن أقطاب هذه النافذة تصبح في حالة الممانعة العالية حتى لو كانت دورة الساعة متوقفة عن العمل.

- النافذة (port c): و هي عبارة عن دخل / خرج ذات ستة اقطاب ثنائية الاتجاه و زودت بمقاومة رفع داخلية و تعطى تيار 20MA , و مميزاتها كالنافذة (portb) أيضاً .

- النافذة (port d) : وهي عبارة عن i/o ذات ثمانية أقطاب مزودة بمقاومة رفع , و تعطى تيار 20MA .

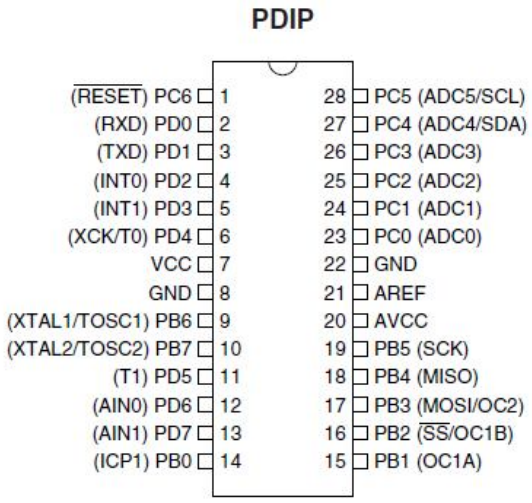
- قطب التصفير (reset) : يولد التصفير الخارجي عند تطبيق نبضات كهربائية أطول من 50ns ذات منطق منخفض على القطب reset , و ذلك حتى إن لم تكن الساعة فى حالة عمل , و النبضات الاقصر من ذلك قد لا تؤدي بالضرورة إلى توليد التصفير .

- مدخل الهزاز (xtal1) : هو عبارة عن مدخل مضخم الهزاز العاكس , وهو أيضاً مدخل لدارة تشغيل الساعة الداخلية.

- مدخل الهزاز: (xtal2) : هو عبارة عن مخرج مضخم الهزاز العاكس .

- aVcc : وهو قطب جهد التغذية للمبدل التشابهي الرقمي ADC و يجب وصله خارجياً مع القطب Vcc من خلال مرشح منخفض .

- Aref :وهو قطب دخل الجهد المرجعي التشابهي للمبدل ADC ويجب أن تتراوح قيمة هذا الجهد عند عمل المبدل (aVcc ADC) , يتم التحكم بالمتحكم AVR من قبل البرامج . و بعدم وجود البرامج لا يستطيع العمل , تتم كتابة برامج AVR على جهاز الكمبيوتر باستخدام AVR-BASCOM ,هذا البرنامج هو نوع من أنواع برامج الكمبيوتر يسمى بالمترجم ، لأنه يأتي من www.mcselec.com مجاني , فيمكن لجميع الطلاب تحميل البرنامج واستخدامه ب حرية في المنزل . و يتم توصيل AVR إلى جهاز الكمبيوتر بإستخدام مبرمجة خاصة بمتحكمات AVR .



الشكل (12) PDIP

من القطع التي تم استخدامها في بنية الدارة

3.3. الدارة المتكاملة : IC SN74HC595N

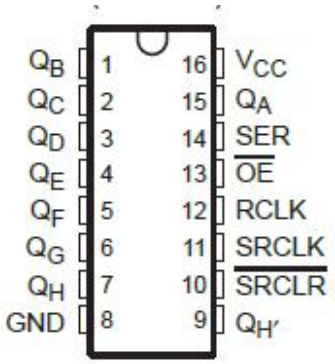
- الدخل 8 بت على التسلسل , الخرج على التفرع .
- مدى جهد التشغيل يتراوح بين (2 إلى 6) فولط .
- انخفاض استهلاك الطاقة الى -80 واط .
- 6 ميلي أمبير خرج , يقاد ب 5 فولط .
- انخفاض تيار وحدلت الدخل إلى 1 أمبير .

إن الدارات من نوع "595HC" تحتوي على 8bit متسلسلة , يوازيها 8bit خرج من نوع d-type , و تحتوي على 3 مخارج من سجلات التخزين و يتم توفير ساعات منفصلة تزود سجلات التخزين و الإزاحة.

عندما يكون الخرج (oe) مفعل الدخل عالي , ويكون الخرج مقاومته عالية . **[3]...**

يكون مؤقت سجلات الإزاحة (srclk) و مؤقت سجلات التخزين (rclk) موجب النبضة إذا كانت كل الساعات معاً.

[3].... الدارة المتكاملة IC SN74HC595N datasheet:https://www.sparkfun.com/datasheets/IC/SN74HC595.pdf



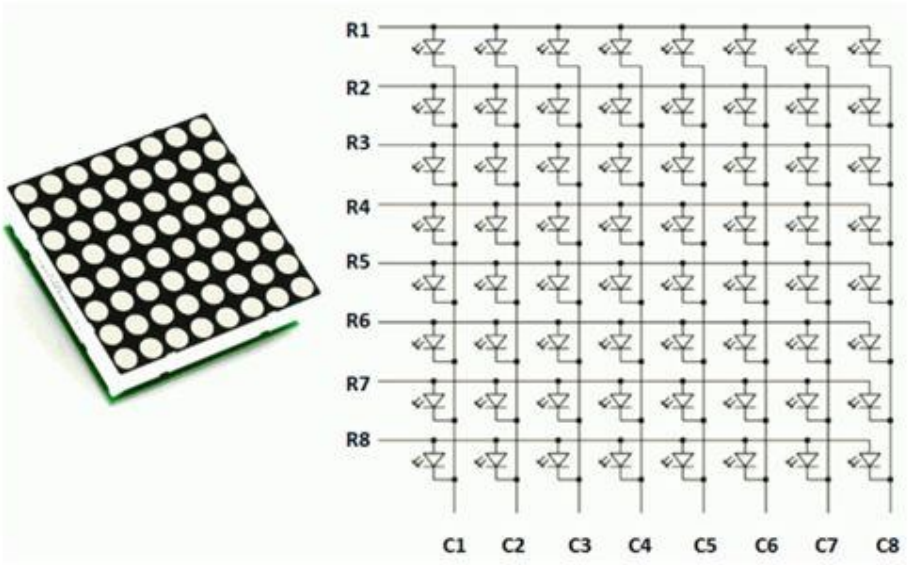
الشكل (13) الدارة المتكاملة
SN74HC595N

3.4. شاشة العرض النقطية (DOT MATIX):

في هذه الشاشة يتم توصيل الليدات مع بعضها في شكل صفوف rows , و أعمدة columns بغرض تقليل عدد الأطراف المطلوبة لتشغيلها . فعلى سبيل المثال وحدة عرض 8X8 سوف تحتاج إلى 64 طرف مدخل / مخرج (طرف لكل نقطة ليد) . بتوصيل جميع الأنودات مع بعضها في شكل صفوف (من الصف الأول R1 إلى الصف الثامن R8) , و توصيل الكاثودات مع بعضها فى شكل أعمدة (من العامود الأول C1 إلى العامود الثامن C8) فإن أطراف المداخل / المخارج المطلوبة للتشغيل تقل إلى 16 طرف.

كل ليد يعنون برقم صفه وعاموده . ففى الشكل التالي إذا جعلنا الصف الرابع R4 في الحالة المرتفعة (الأنودات) وجعلنا العامود الثالث C3 في الحالة المنخفضة (الكاثودات – لاحظ حرف C مشترك بين معنى العامود والكاثود) فإن الليد الموجود في الصف الرابع و العامود الثالث سوف يضيء . يمكن عرض الحروف بالمسح السريع لأي من الصفوف أو الأعمدة .

فى هذه التجربة سوف نناقش طريقة مسح الأعمدة . column scanning



الشكل (14) شاشة العرض النقطية

فمثلاً لعرض الحرف : A

1. نختار أولاً العامود C1 هذا يعني جعله في الحالة المنخفضة , وفي نفس الوقت عدم اختيار باقي الأعمدة بمنع توصيلها بالأرضي (أحد الطرق لعمل ذلك هو جعل C2 إلى C5 في الحالة المرتفعة) .

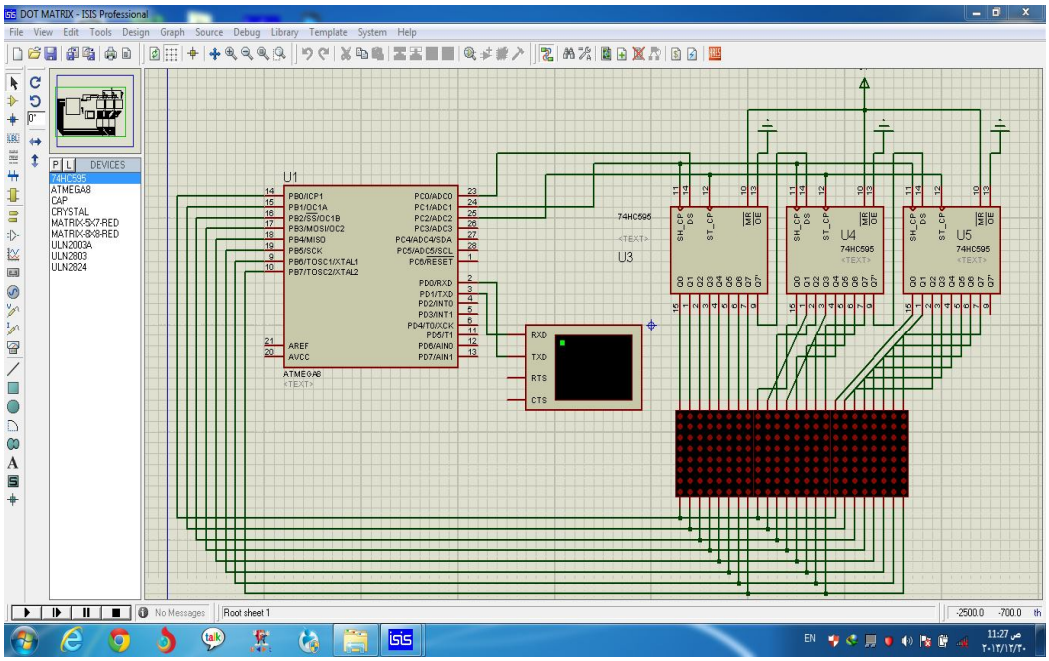
2. الآن أصبح العامود الأول فعال , و يحتاج إلى توصيل الليدات بالصفوف من R2 إلى R7 الموجودة فى العامود الأول C1 و التي تمثل الجزء الأول فقط من الحرف , و يتم ذلك بتوصيل الجهد الموجب (الحالة المرتفعة) لهذه الصفوف.

3. الخطوة التالية اختيار العامود الثاني C2 (وعدم اختيار باقي الأعمدة) , و توصيل الجهد الموجب إلى الصفوف R1 , R5 و هكذا . هذه العملية تسمى "عملية مسح الأعمدة و تغذية قيم البيانات إلى الصفوف" .

4. نتيجة لذلك بالمسح السريع خلال الأعمدة (سرعة المسح يجب أن تكون أكبر من 100 مرة بكل ثانية) و توصيل الليدات المناظرة في كل صف لهذا العامود فإن صورة العرض تبدو ساكنة ومستقرة للعين (خداع بصري) .

الفصل الرابع:توصيل الدارة

لقد قمنا بمحاكاة الدارة عل برنامج protos للتحقق من توافق القطع مع بعضها البعض و الحصول على النتائج المطلوبة , و بعد ذالك تم وصل المنافذ (B) من المتحكم إلى مجموعة من الترانزيستورات لقيادة أسطر الشاشة حيث أن كل رجل من أرجل المنفذ (B) موصلة بقاعدة الترانزيستور . عند تفعيل الترانزيستور يقوم بتمرير تيار قدره 5v إلى السطر المراد إضاءته . و من خلال المنفذ (C) من المتحكم تم توصيل ثلاثة أرجل إلى ثلاثة دائرات متكاملة من النوع 74hc595 و كل دارة متكاملة موصولة مع أعمدة كل بلوك , وصلنا الأرجل الثاني و الثالث (TX,RX) من المنفذ (D) مع منفذ com لتبادل البيانات مع الحاسب .



الشكل (15) واجهة البرنامج PROTOS

يتم توصيل الليدات مع بعضها في شكل صفوف و أعمدة بغرض تقليل عدد الأطراف المطلوبة لتشغيلها . فعلى سبيل المثال وحدة عرض 8x8 سوف تحتاج الى 64 طرف مدخل / مخرج , طرف لكل نقطة ليد .

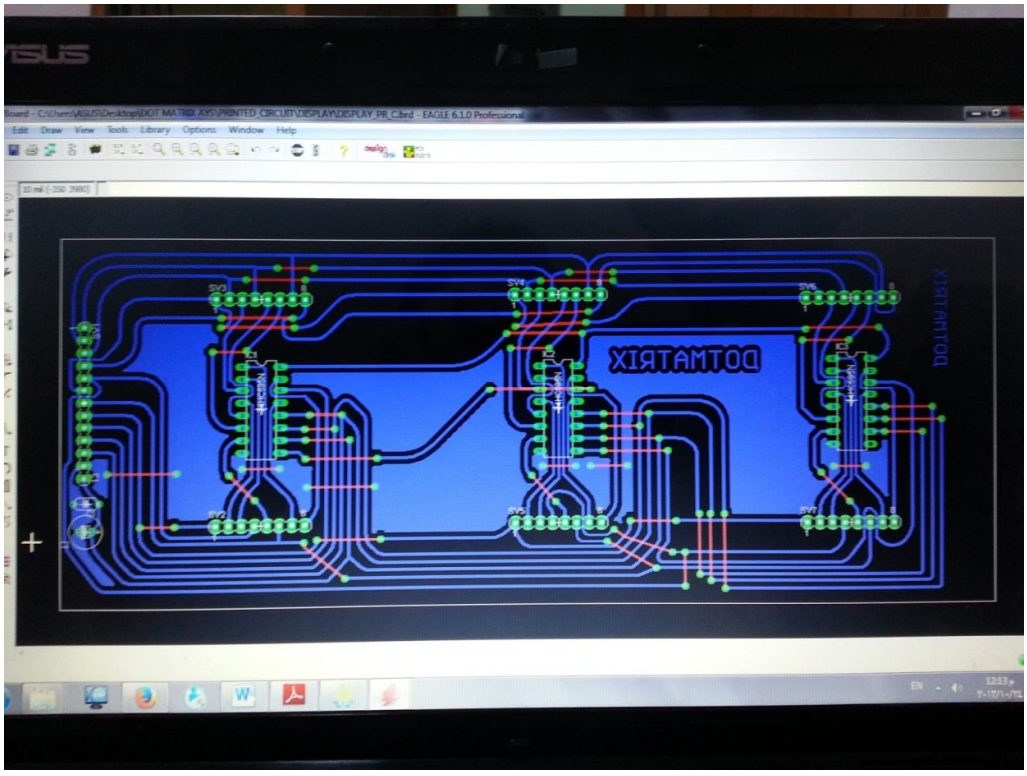
بتوصيل الصف الأول R1 إلى الصف الثامن , و توصيل الكاثودات مع بعضها في شكل أعمدة من العامود الأول C1 إلى العامود الثامن . كل ليد يعنون برقم صفه و عامود. سوف تحتاج الدارة الكهربائية إلى مقاومات

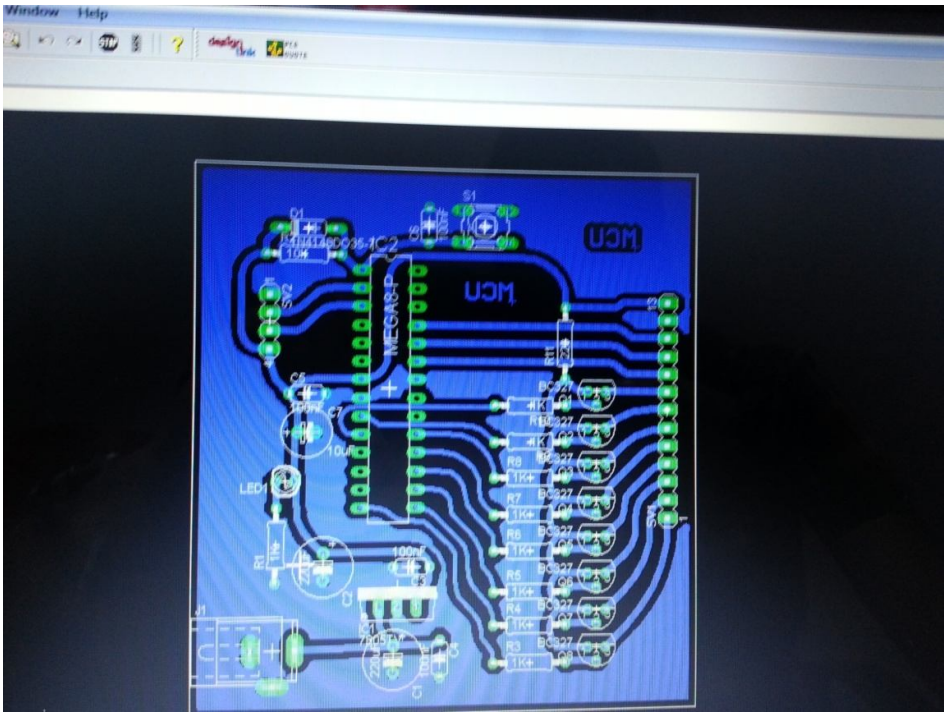
330 أوم على التوالي مع الصفوف من R1 إلى R7 لتحديد التيار المار خلال الليدات . بعد ذلك يتم توصيل الصفوف بأطراف المنفذ PORTB(RB0-RB6) للميكروكونترولر . الأعمدة تتصل بخمسة مخارج من الدارة المتكاملة يتم التحكم بها من خلال أطراف المنفذ PORTD(RD0-RD4) . بهذه الطريقة يقوم الميكروكونترولر بمسح الأعمدة بارسال الخانات المناظرة للمنفذ PORTD على سبيل المثال تحديد الخانة RD0 بواحدة مسح (تصفير) الخانات RD1 إلى RD4 , يؤدي إلى اختيار العامود الأول . سوف ينتظر الميكروكونترولر حوالي واحد مل ثانية قبل التحول إلى العامود التالي .

عند كل عامود سوف يخرج الميكروكونترولر قيم الصفوف المناظرة على المنفذ PORTB لتوصيل الليدات المناظرة في العامود لعرض الحرف المحدد.

التحويل بين الأعمدة يكون سريعا بما يكفي لخداع البصر و ليبدو العرض مستقرا.و يمكن عرض الحروف بالمسح السريع لأي من الصفوف .

و تم استخدام برنامج Eagle لطباعة الدارة من خلاله , و توضح الأشكال التالية طريقة رسم المخططات الدارة




Window Help
UCM
MEGA8-P
LED1
100nF
10uF
1K
BC327

TRI

الخاتمة:

استطعنا أن نبني دارة متضمنة الأجزاء التي تم ذكرها سابقاً, و إيجاد خوارزمياتها البرمجية و تضمينها وتطوير قدراتنا البرمجية و قدرتنا على مقاربة مسائل الحياة اليومية بخوارزميات قابلة للتطبيق برمجياً, كما حاولنا جاهدين أن نطلع على معظم الأعمال التي سبقتنا في هذا المجال . ووضعنا خوارزميات وتطبيقاتها التي أعطت نتائج دقيقة وممتازة .إضافة الى ذلك إن تقريرنا قابل لإضافة تحسينات كبيرة منها أنه بالإمكان أن نصل الدارة على جهاز خليوي وجعلها تقرأ الرسائل النصية وتحويلها إلى اللوحة وطباعة الرسائل عليها .

أيضاً بالإمكان القيام بتعديل على المتحكم حيث يصبح قادر على قراءة كل ما يدخل إليه كمجموعة من البلوكات الصورية اي يستقبلها كصورة بغض النظر عن ما هو الموجود داخلها إن كانت أحرف او أشكال هندسية أو أي شكل آخر وإظهاره كما هو على الشاشة ويتم ذلك بتعديل على برنامج الحاسوب أيضا حيث يقوم بإرسال البيانات المدخلة كمجموعة صور مقسمة إلى بلوكات بطول معين، نضيف إلى ذلك أنه يمكن تكبير الشاشة إلى عدد كبير جداً من المصابيح وذلك يعتمد على طلب المستخدم .

المراجع:

[1].... المتحكم ATMEL ATMEGA8L :
http://www.atmel.com/devices/atmega8.aspx

[2].... الترانزستور BC327 Datasheet:

http://www.alldatasheet.com/view.jsp?Searchword=Bc327&gclid=COy73Ibt17sCFUnjwgodJl0AFg

[3].... الدارة المتكاملة IC SN74HC595N datasheet :

https://www.sparkfun.com/datasheets/IC/SN74HC595.pdf

[4]....LED DOT MATRIX PDF FILE:

http://www.mat.ucsb.edu/200C/2008_Students/MAT-200C_2008_Files/matt_stabile/MAT200CStabile.pdf

ملحقات :

1) برنامج الحاسب:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;

namespace dotmatrix
{
    public partial class Form1 : Form
    {
        string MyText = "";
        int x = 0;
        int Font_size = 50;

        public Form1()
        {
            InitializeComponent();
            x = pictureBox1.Width;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                MyText = textBox1.Text + "~";
                serialPort1.PortName = textBox2.Text;
                serialPort1.Open();
                serialPort1.WriteLine(MyText);
                serialPort1.Close();
                button2.Enabled = button3.Enabled = true;
                timer1.Enabled = true;
            }
            catch (Exception)
            {
                MessageBox.Show("there is no
circuit","error",MessageBoxButtons.OK,MessageBoxIcon.Error);
            }
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
            if (textBox1.Text != "")
```

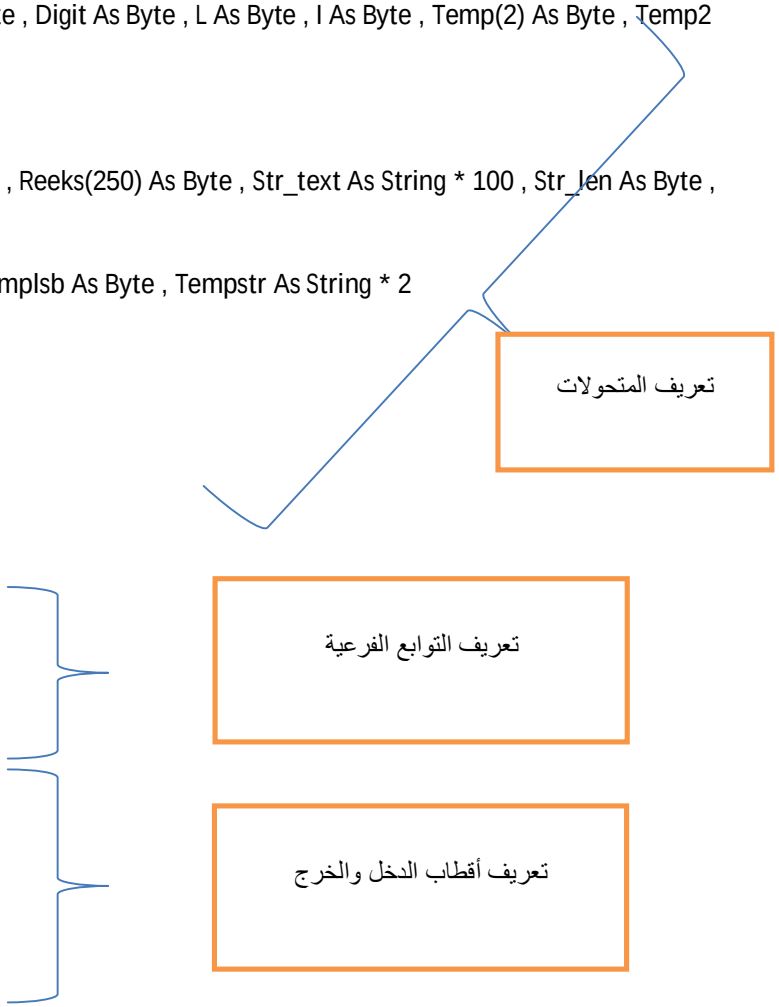
```
        {
            button1.Enabled = true;
            button4.Enabled = true;
        }
        else
        {
            button1.Enabled = false;
            button4.Enabled = false;
        }

        MyText = textBox1.Text;
    }


    private void button2_Click(object sender, EventArgs e)
    {
        serialPort1.Open();
        serialPort1.WriteLine(".");
        serialPort1.Close();
        timer1.Enabled = false;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        serialPort1.Open();
        serialPort1.WriteLine(".");
        textBox1.Text = "";
        serialPort1.Close();
        ClearImage();
    }


    private void button4_Click(object sender, EventArgs e)
    {
        try
        {
            MyText = textBox1.Text + "~";
            foreach (string port in SerialPort.GetPortNames())
                serialPort1.PortName = port;
            serialPort1.Open();
            serialPort1.WriteLine(MyText);
            serialPort1.Close();
            button2.Enabled = button3.Enabled = true;
            timer1.Enabled = true;
        }
        catch (Exception )
        {
            MessageBox.Show("there is no circuit", "error",
MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
```

```
        }

        void DrawText(int x)
        {
            if (checkBox1.Checked != true)
            {
                Bitmap image = new Bitmap(pictureBox2.Width,
pictureBox2.Height);
                for (int i = 0; i < image.Width; i++)
                    for (int j = 0; j < image.Height; j++)
                        image.SetPixel(i, j, Color.Black);
                Graphics g = Graphics.FromImage(image);


                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Red, x, 0);
                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Red, pictureBox2.Width + x, 0);
                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Red, x - pictureBox2.Width, 0);


                pictureBox2.Image = image;
            }

            else
            {
                Bitmap image = new Bitmap(pictureBox2.Width,
pictureBox2.Height);
                for (int i = 0; i < image.Width; i++)
                    for (int j = 0; j < image.Height; j++)
                        image.SetPixel(i, j, Color.Red);
                Graphics g = Graphics.FromImage(image);


                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Black, x, 0);
                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Black, pictureBox2.Width + x, 0);
                g.DrawString(MyText, new Font("Candara", Font_size),
Brushes.Black, x - pictureBox2.Width, 0);


                pictureBox2.Image = image;
            }
        }

        void ClearImage()
        {
            if (checkBox1.Checked != true)
```

```
        {
            Bitmap image = new Bitmap(pictureBox2.Width,
pictureBox2.Height);
            for (int i = 0; i < image.Width; i++)
                for (int j = 0; j < image.Height; j++)
                    image.SetPixel(i, j, Color.Black);

            pictureBox2.Image = image;
        }
        else
        {
            Bitmap image = new Bitmap(pictureBox2.Width,
pictureBox2.Height);
            for (int i = 0; i < image.Width; i++)
                for (int j = 0; j < image.Height; j++)
                    image.SetPixel(i, j, Color.Red);
               pictureBox2.Image = image;
        }
    }
    private void timer1_Tick(object sender, EventArgs e)
    {
      if (x > -(MyText.Length * Font_size +20))
        {
            x -= 5;
            DrawText(x);
        }
        else
        {
            x = pictureBox2.Width +100 ;
        }
    }
     }}
```

1) برنامج المتحكم:

```
$regfile = "M8def.dat"

$crystal = 8000000

$baud = 2400

Dim Char As Byte , C As Byte , Digit As Byte , L As Byte , I As Byte , Temp(2) As Byte , Temp2
As Byte

Dim X As Word

Dim Str_temp As String * 1 , Reeks(250) As Byte , Str_text As String * 100 , Str_len As Byte ,
Col_len As Byte , R As Byte

Dim Tempmsb As Byte , Templsb As Byte , Tempstr As String * 2

Dim Scroll_speed As Byte

Dim Arr As Byte

Dim A As Byte

Dim S As Byte

Declare Sub Show_text

Declare Sub Set_text

Declare Sub Test_display

Config Portb = Output

Config Portc = Output

Config Pind.6 = Input

Portd.6 = 1

Conv Alias Pind.6
```



```
For S = 1 To 250 Step 1
Reeks(s) = 255
Waitms 1
Next S
```

جعل الليدات مطفأة خارج نطاق العبارة المعروضة

وضع (1)منطقي في كافة خانات المصفوفة

```
A = 1
Arr = "~"
Str_text = "DOT MATRIX"
Goto Main
'****************
```

```
Sub Set_text
C = 1
For Digit = 1 To Str_len
Str_temp = Mid(str_text , Digit , 1)
Char = Asc(str_temp)
X = Char * 5
For I = 1 To 5
Reeks(c) = Lookup(x , Text_data)
If Conv = 0 Then
Reeks(c) = 255 - Reeks(c)
End If                                  '***** ADD
X = X + 1
C = C + 1
Next I
If Conv = 0 Then
Reeks(c) = 255                          'SET THE SPACE BETWEEN CHARECTERS (LIGHT OR
DARK)(0,255) CONSECUENTLY
C = C + 1
```

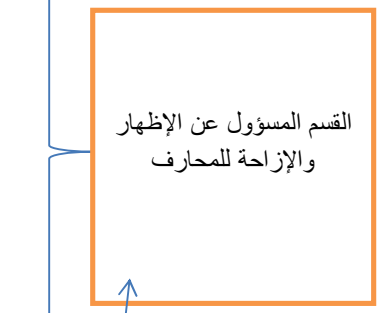
التابع الفرعي المسؤول عن وضع شيفرة المحارف الموجودة في العبارة المستقبلة ضمن مصفوفة لتجهزها لعملية الإظهار

```
Reeks(c) = 255                    'SET SPACE
C = C + 1
End If
If Conv = 1 Then
Reeks(c) = 0                      'SET THE SPACE BETWEEN CHARECTERS (LIGHT OR
DARK)(0,255) CONSECUENTLY
C = C + 1
Reeks(c) = 0                      'SET SPACE
C = C + 1
End If
Next Digit
End Sub Set_text
```

```
show text
Sub Show_text
If Str_len > 1 Then
For L = 0 To Col_len
Do
If Iscarwaiting() = 1 Then
Arr = Inkey()
Waitms 1
If Arr = "." Then
Waitms 1
A = 1
Print "SHOW STOPED"
For S = 1 To 250 Step 1
Reeks(s) = 0
Waitms 1
```

التابع الفرعي المسؤول عن إظهار النص

```
Next
'Str_len = 0
Str_text = ""
Goto Main
End If
End If
C = 1
  For R = 1 To Scroll_speed                    'scrolling speed
     Reset Portc.0
     Toggle Portc.2
     Toggle Portc.2
     Toggle Portc.1
     Toggle Portc.1
     Char = C + L
     Portb = Reeks(char)
     Waitms 20
     C = C + 1
     Set Portc.0
     For I = 1 To 24
     Toggle Portc.2
     Toggle Portc.2
     Toggle Portc.1
     Toggle Portc.1

     Char = C + L
     Portb = Reeks(char)
     Waitms 1
     C = C + 1
    ' If C = 25 Then C = 1
     Next I
```



```
    C = 1
  Portb = 255
  Next R
Incr L
If L > Col_len Then L = 0
Next L
Str_text = ""
Loop
End If
End Sub Show_text

Sub Test_display
Portb = 255
For X = 1 To 20
      Reset Portc.0
      Toggle Portc.2
    Toggle Portc.2
      Toggle Portc.1
     Toggle Portc.1
Next X
End Sub Test_display

'--------------------
```

```
Main:
Call Test_display
Waitms 100

Do
If Ischarwaiting() = 1 Then
Arr = Inkey()
Insertchar Str_text , A , Arr
A = A + 1
Waitms 1
End If                                  '

Str_len = Len(str_text)
Col_len = Str_len * 8                   '*************AYS********
Scroll_speed = 5
                                'adjust the scroll speed

If Arr = "~" Then
Print "DATA RECEIVED"
Print "STARTING SHOW"
Call Set_text
Call Show_text
End If
Loop
```

```
End                                    'end program


Text_data:
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
Data 0 , 0 , 0 , 0 , 0 , 0             '
```

جدول شيفرة المحارف

```
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  '
Data 0 , 0 , 0 , 0 , 0                  'space
Data 0 , 0 , 95 , 0 , 0                  '!
Data 0 , 7 , 0 , 7 , 0                  '"'
Data 20 , 127 , 20 , 127 , 20                '#
Data 36 , 42 , 127 , 42 , 18                 '$
Data 39 , 21 , 107 , 84 , 114                '%
Data 54 , 73 , 86 , 32 , 80                 '&
Data 0 , 0 , 11 , 7 , 0                  ''
Data 0 , 28 , 34 , 65 , 0                  ' (
Data 0 , 65 , 34 , 28 , 0                  ')
Data 42 , 28 , 127 , 28 , 42                 '*
Data 8 , 8 , 62 , 8 , 8                  '+
Data 0 , 88 , 56 , 0 , 0                  ',
Data 8 , 8 , 8 , 8 , 8                  '-
Data 0 , 96 , 96 , 0 , 0                  '.
Data 32 , 16 , 8 , 4 , 2                  '/
Data 62 , 81 , 73 , 69 , 62                 '0
Data 0 , 66 , 127 , 64 , 0                  '1
Data 114 , 73 , 73 , 73 , 70                 '2
```

```
Data 34 , 65 , 73 , 73 , 54                    '3
Data 24 , 20 , 18 , 127 , 16                   '4
Data 39 , 69 , 69 , 69 , 57                    '5
Data 60 , 74 , 73 , 73 , 48                    '6
Data 1 , 113 , 9 , 5 , 3                     '7
Data 54 , 73 , 73 , 73 , 54                    '8
Data 6 , 73 , 73 , 41 , 30                     '9
Data 0 , 54 , 54 , 0 , 0                     ':
Data 0 , 91 , 59 , 0 , 0                     ';
Data 8 , 20 , 34 , 65 , 0                      '<
Data 20 , 20 , 20 , 20 , 20                    '=
Data 0 , 65 , 34 , 20 , 8                      '>
Data 2 , 1 , 81 , 9 , 6                      '?
Data 50 , 73 , 121 , 65 , 62                   '@
Data 124 , 18 , 17 , 18 , 124                    'A
Data 65 , 127 , 73 , 73 , 62                   'B
Data 62 , 65 , 65 , 65 , 34                    'C
Data 65 , 127 , 65 , 65 , 62                   'D
Data 127 , 73 , 73 , 65 , 65                   'E
Data 127 , 9 , 9 , 1 , 1                     'F
Data 62 , 65 , 73 , 73 , 58                    'G
Data 127 , 8 , 8 , 8 , 127                     'H
Data 0 , 65 , 127 , 65 , 0                     'I
Data 32 , 64 , 65 , 63 , 1                     'J
Data 127 , 8 , 20 , 34 , 65                    'K
Data 127 , 64 , 64 , 64 , 64                   'L
Data 127 , 2 , 12 , 2 , 127                    'M
```

```
Data 127 , 2 , 4 , 8 , 127                    'N
Data 62 , 65 , 65 , 65 , 62                   'O
Data 127 , 9 , 9 , 9 , 6                    'P
Data 62 , 65 , 65 , 33 , 94                   'Q
Data 127 , 9 , 25 , 41 , 70                   'R
Data 38 , 73 , 73 , 73 , 50                   'S
Data 1 , 1 , 127 , 1 , 1                    'T
Data 63 , 64 , 64 , 64 , 63                   'U
Data 7 , 24 , 96 , 24 , 7                    'V
Data 127 , 32 , 24 , 32 , 127                     'W
Data 99 , 20 , 8 , 20 , 99                    'X
Data 3 , 4 , 120 , 4 , 3                    'Y
Data 97 , 81 , 73 , 69 , 67                   'Z
Data 0 , 127 , 65 , 65 , 0                    '[
Data 2 , 4 , 8 , 16 , 32                    '\
Data 0 , 65 , 65 , 127 , 0                    ']
Data 4 , 2 , 1 , 2 , 4                    '^
Data 64 , 64 , 64 , 64 , 64                     '_
Data 0 , 0 , 7 , 11 , 0                    '`
Data 32 , 84 , 84 , 84 , 56                   'a
Data 127 , 40 , 68 , 68 , 56                    'b
Data 0 , 56 , 68 , 68 , 68                    'c
Data 56 , 68 , 68 , 40 , 127                    'd
Data 56 , 84 , 84 , 84 , 24                   'e
Data 8 , 126 , 9 , 9 , 2                    'f
Data 8 , 84 , 84 , 84 , 60                    'g
Data 127 , 16 , 8 , 8 , 112                   'h
```

```
Data 0 , 68 , 125 , 0 , 0                    'i
Data 32 , 64 , 68 , 61 , 0                   'j
Data 0 , 127 , 16 , 40 , 68                  'k
Data 0 , 65 , 127 , 64 , 0                   'l
Data 124 , 4 , 120 , 4 , 120                 'm
Data 124 , 8 , 4 , 4 , 120                   'n
Data 56 , 68 , 68 , 68 , 56                  'o
Data 124 , 20 , 20 , 20 , 8                  'p
Data 8 , 20 , 20 , 20 , 124                  'q
Data 124 , 8 , 4 , 4 , 8                     'r
Data &B00000000 , &B10001100 , &B10010010 , &B10010010 , &B10100010 , &B01000100
, &B00000000 , &B00000000 , &B00000000
Data 72 , 84 , 84 , 84 , 36                  's
Data 0 , 4 , 63 , 68 , 68                    't
Data 60 , 64 , 64 , 32 , 124                 'u
Data 28 , 32 , 64 , 32 , 28                  'v
Data 60 , 64 , 60 , 64 , 60                  'w
Data 68 , 40 , 16 , 40 , 68                  'x
Data 12 , 80 , 80 , 80 , 60                  'y
Data 68 , 100 , 84 , 76 , 68                 'z
Data 0 , 8 , 54 , 65 , 0                     '{
Data 0 , 0 , 119 , 0 , 0                     '|
Data 0 , 65 , 54 , 8 , 0                     '}
Data 2 , 1 , 2 , 4 , 2                       '~
Data 42 , 85 , 42 , 85 , 42                  '
```