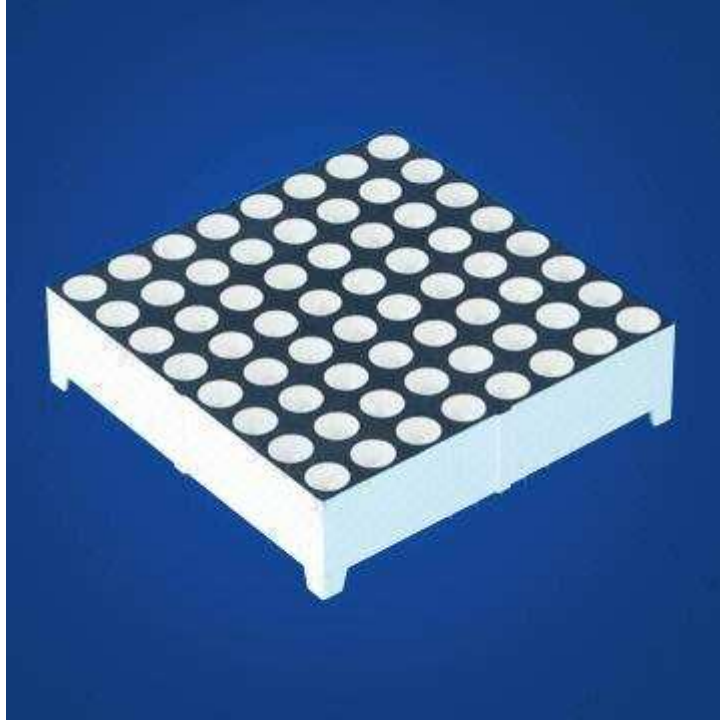


بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

طريقة عمل الماتريكس ليد

*Matrix Dot Led*



شرح وتصميم /

م/ ناصر أحمد

تجميع و تنسيق /  
كريم

[رابط الموضوع الأصلي : اضغط هنا](#)



اولا اخترت هذا العنوان للموضوع

لاننى كنت فى بداياتى لتعلم الماتركس

كنت اجد هذا العنوان

واختلغت المواقع والمنتديات التى بها نفس الموضوع

ولكن كان الكل مجرد نسخة واحدة  
**وللاسف ماكان به اى معلومة مفيدة**

وبناء على ذلك

وبعد ان تعلمنا الكثير والكثير من استاذنا ومهندسنا م/ احمد سمير  
فكرت ان اضع هنا هذا الموضوع وليس نسخة من اى منتدى اخر طبعاً

ولو باعتبار 1% من رد الجميل لاستاذنا احمد

ومعهدنا الكريم (منتدى القرية الالكترونية))

**وهو شرح مبادئ اللىد ماتركس**

على ان نتعلم سوياً طريقة صنع ال led matrix من الالف

ولن اقول الى اليا فمّن يقول هذه الكلمة لا يعرفها  
لان المجال واسع جدا ومفتوح

وانما نتعلم منه ما نستطيع ان شاء الله

والله الموفق لما يحبه ويرضاه

فعلى من يبحث عن معلومات عن ال led matrix

او بمعنى ادق **مبادئ اللىد ماتركس**

فانا مازلت تلميذ ولا ادعى اكثر من هذا

فعلى من يبحث عن هذه المعلومة

او يشجع لوضع هذا الموضوع

عليه ان يضع رداً فى الموضوع

وسوف تكون الاكواد بلغة السى micro c

وعلى من لا يعرفها الدخول هنا اولاً

<http://www.qariya.com/vb/showthread.php?t=40760&page=3>

حيث ان بدون معرفة لغة البرمجة لن يفيدك هذا الموضوع بشيء

واحب اقول اننا قد تعينا بين المواقع العربية والاجنبية حتى الصينية واليابانية وايضا اليهودية كل هذا للبحث عن معلومات عن هذا الموضوع

ومتابعة مشروعاتهم وتجاريمهم

ولن انكر اننى استفدت والحمد لله الكثير والكثير

ولكن لم يكن لى ان استطيع الاستفادة او حتى فهم كلمة واحدة من ماكنت اقرا

السبب

السبب كان جهلى بلغات البرمجة

وبفضل الله

قد سخر لنا استاذنا احمد سمير لاعطائنا دروس البرمجة بلغة السي والحمد لله من سلاسة شروحاته وسيرة فى خطوات ثابتة واذادة الله من فضله (فهو يمتلك ميزة قد لا يمتلكها الاغلبية )

وهى ميزة الاسلوب الجميل فى الشرح بحيث انه يخاطب المبتدا وغير المبتدا فى ان واحد

دون ان يشعر اى منهم باى فرق

وبحمد لله من الدرس الاول والثانى

قد بدأت افهم الاوامر (طبعا فى حدود ماكان فى الدروس)

---

بسم الله وباذن الله نبدا اولاً



بتعريفات عامة عن الماتركس  
وانمنى من كل زميل يدخل هذا الموضوع ان يفهم ان فى هذا الموضوع

نتحدث عن الماتركس من الالف

بمعنى انه سوف نبدا من الصفر والله المستعان

77777777777777

وقبل البداية اتمنى ان تعزرونى فقد لا امتلك الاسلوب الجميل مثل استاذنا احمد ولكنى احاول والله المستعان

77777777777777

---

سوف نبدا ان شاء الله يعمل مشاريع بسيطة ونستخدم متحكم واحد فقط وهذا لكى نستطيع متابعة الامور من اول درجة



- \_\_\_\_\_





## تابع الدرس الاول

بسم الله الرحمن الرحيم

نكمل هنا مابدانة فى المشاركة السابقة



والتي فيها فقط تعرفنا على مخارج ومدخل مكعب الماتركس بالبريتوس

وبناء عليه انه يتم وضع الطرف الموجب بالاعلى والطرف السالب باسفل مكعب الليد

ولو عكسنا المداخل والمخارج فانها لن تعمل طالما ان الطرف الثانى سالب فولت

ولو قمنا بتبديل السالب فولت بموجب فوت

سوف تلاحظ ان الليدات المقصودة سوف تجدها منطفئة وباقى الليدات سوف تضىء

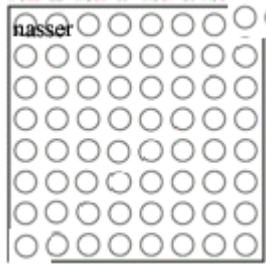
مع هذا او السبب فى ذلك هو انه معروف انه موجب+موجب= لا شىء (هذا بالنسبة للكونترول وال5فولت طبعاً)

اما بالنسبة للتوصيلات الكهربائية فهذا يعنى بووووووووووووم

قفلة

ار يو اوك ؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟

حتى الان نحن بصدد تنفيذ هذا المشروع البسيط



وقمنا باستخدام مخارج بورت B من الميكرو

وان الاوان حتى نستخدم البورت الاخر وهو بورت A

هذا هو الكود السابق كما ذكرنا

كود:

```
void main()
{
TRISB=0;
PORTB=0;
loop:
PORTB=0B00000001; delay_ms(500);
PORTB=0B00000010; delay_ms(500);
PORTB=0B00000100; delay_ms(500);
PORTB=0B00001000; delay_ms(500);
PORTB=0B00010000; delay_ms(500);
PORTB=0B00100000; delay_ms(500);
PORTB=0B01000000; delay_ms(500);
PORTB=0B10000000; delay_ms(500);

goto loop;
}
```

وسوف نقوم الان بتفعيل واستخدام بورت A

بهذا الكود

كود:

```
void main()
{
TRISB=0;
PORTB=0;

TRISA=0;
PORTA=0;

loop:

PORTA=0B111110;
PORTB=0B00000001; delay_ms(500);
//////////

PORTA=0B111101;
PORTB=0B00000010; delay_ms(500);
//////////

PORTA=0B111011;
PORTB=0B00000100; delay_ms(500);
//////////

PORTA=0B110111;
PORTB=0B00001000; delay_ms(500);
//////////

PORTA=0B011111;
PORTB=0B00010000; delay_ms(500);

goto loop;
}
```

وتلاحظون هنا هذه العلامة /////////////// فانا اضعتها كفاصل بين كل امر واذا يلية  
بامكانكم فقط المقارنة بين الكود الاول والكود الثانى لكى نتعرف على ماتم تغييره

ونلاحظ نقطة فى منتهى الاهمية

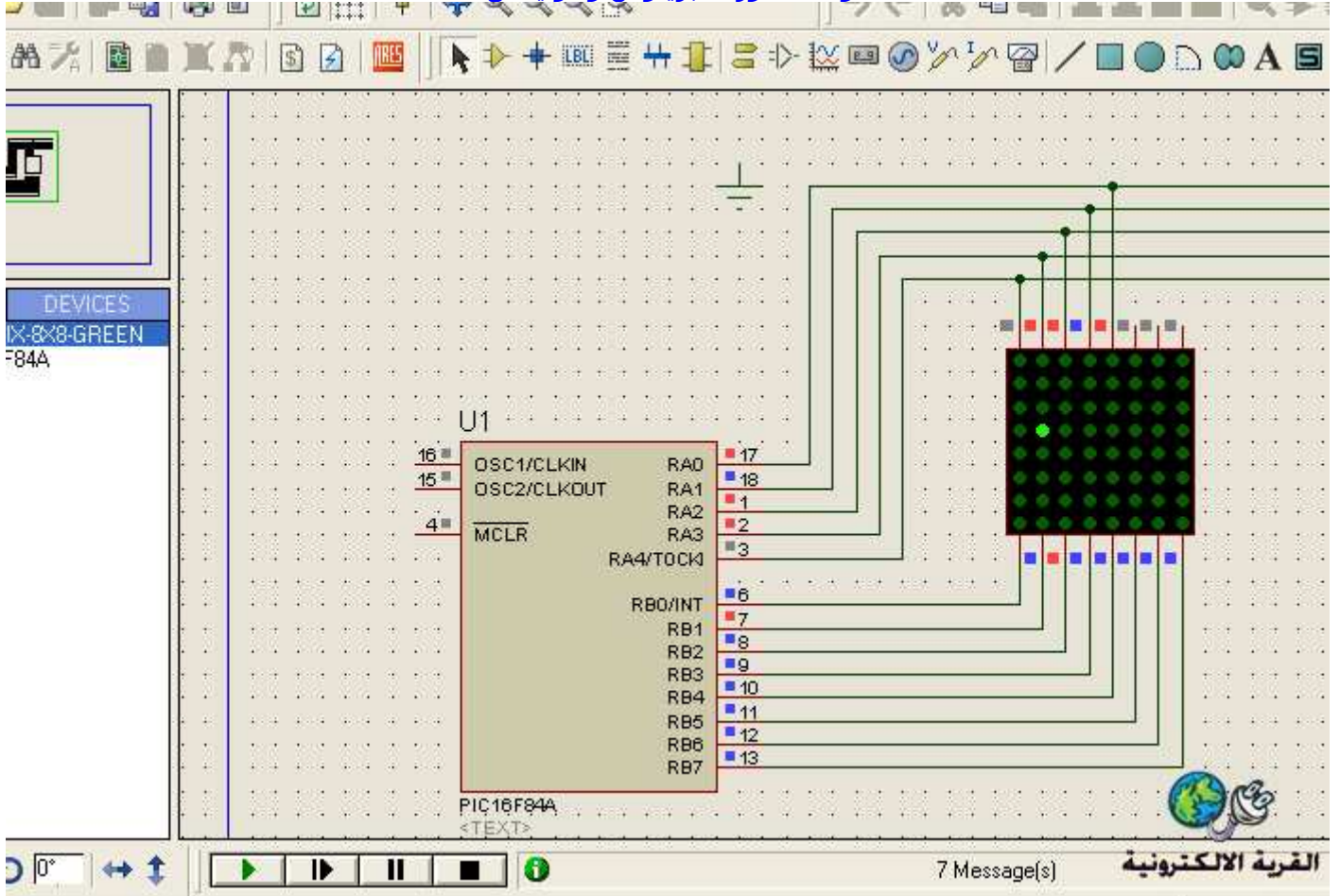
الا وهى اننا نعطى الديلاى او التوقيت بعد اعطاء الامر لبورت a وبورت B معا

هذه نقطة هامة جدا جدا  
والا فسوف تعمل بورت اى لمدة التوقيت ثم تنطفئ، وبعدها بورت بى الوقت المحدد ثم تنطفئ،

وطبعا لن يلتقى الاثنان معا .....

اليكم الملف بعد التعديل  
وتطبيق النقطة المتحركة اعلى

وهذه صورة البريتوس وهو يعمل



وهذا الملف بالكامل

[lesson 1.rar - 0.06MB](#)

ملاحظة هامة :

فى المثال للتجربة سوف تجد ان الكرة تتحرك 8مرات

من اسفل اليسار وحتى اعلى اليمين

اما فى تجربتنا ع البريتوس سوف تجد انها تتحرك 5 مرات فقط

فما هو السبب؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟

طبعا السبب هو عدد مخارج البورت A داخل ميكرو f84a16 الذى نستخدمه فى تجاربنا الاولى

ولهذا السبب فى اختلاف العدد بين a and B



قمنا بالاكتماء بـ5 حركات فقط

واحب انى اقول فى اخر هذه المشاركة

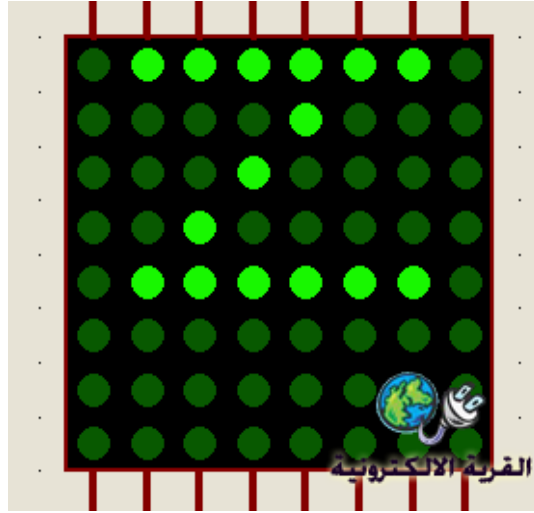
مبروك لقد تمت اول خطوة من خطوات اللىد ماتركس



انتظرونا فى الدرس رقم 2

وسوف يكون باذن الله عبارة عن  
اظهار حرف N على شاشة الماتركس

مثل هذه الصورة





وهو كما تشاهدون عبارة عن خط مائل من اقصى يسار الشاشة  
حتى اعلى اليمين للشاشة  
ولتنفيذ هذا الشكل سوف نستخدم نفس الكود لكرة البينج بونج  
ولكننا سوف نقوم بالغير فقط داخل الكود  
هذا هو الكود السابق

كود:

كود:

```
void main()
{
  TRISB=0;
  PORTB=0;

  TRISA=0;
  PORTA=0;

loop:

PORTA=0B111110;
PORTB=0B00000001; delay_ms(500);
//////////

PORTA=0B111101;
PORTB=0B00000010; delay_ms(500);
//////////

PORTA=0B111011;
PORTB=0B00000100; delay_ms(500);
//////////

PORTA=0B101111;
PORTB=0B00001000; delay_ms(500);
//////////

PORTA=0B011111;
PORTB=0B00010000; delay_ms(500);

goto loop;
}
```

داخل هذا الكود قمنا باستخدام ال porta للازاحة (بمعنى الحركة)  
او التبديل بين الاعمدة وبة نقوم بتفعيل العمود الذي نريد

وهو بوضع القيمة 0 لهذا العمود ووضع القيمة 1 لباقي الاعمدة

وقمنا باستخدام portb لاجراج ما نريد ان نشاهده على الشاشة وهذا سوف نسمة  
ال (data)





سبحان الله والحمد لله ولا حول ولا قوة الا بالله

بعد استيعابنا لما سبق ذكره

اقول نعم صحيح ماتوقعتومة وهو التغيير سوف يتم فقط فى التوقيت

واليكم الكود الجديد

كود:

```
void main()
{
  TRISB=0;
  PORTB=0;

  TRISA=0;
  PORTA=0;

loop:

  PORTA=0B111110;
  PORTB=0B00000001; delay_us(500);
  ///////////////

  PORTA=0B111101;
  PORTB=0B00000010; delay_us(500);
  ///////////////
  PORTA=0B11011;
  PORTB=0B00000100; delay_us(500);
  ///////////////
  PORTA=0B10111;
  PORTB=0B00001000; delay_us(500);
  ///////////////
  PORTA=0B01111;
  PORTB=0B00010000; delay_us(500);

goto loop;
}
```

بعد استخدام هذا الكود

سوف تشاهد عند استخدام البريتوس

ان النقط المتحركة من الاسفل الى الاعلى قد تحول الى شكل ثابت

وهو خط كامل من الاسفل الى الاعلى





طبعا كما ذكرنا وتعلمنا ان مهمة بورت a هي فقط تفعيل الاعمدة

وبورت b هي التي تخرج لنا ماتشاهدة العين مmmmmmm

سوف نقوم هنا بوضع جزء واحد فقط من الكود وهو المختص بالوامر التي وضعناها لبورت B

كود:

```
PORTB=0B00000001; delay_us(500);  
PORTB=0B00000010; delay_us(500);  
PORTB=0B00000100; delay_us(500);  
PORTB=0B00001000; delay_us(500);  
PORTB=0B00010000; delay_us(500);
```

بالنظر والتدقيق داخل هذا الكود سوف تشاهد ماترأة عينك على شاشة الماتركس

وهو الخط المائل من الاعلى الى الاسفل

وهذا للتوضيح اكثر ركزو في هذا الشكل جيدا  
ولاحظو ماذا رسم الرقم 1 بين الارقام 0

```
00000001  
00000010  
00000100  
00001000  
00010000
```

نعم هذا هو شكل الخط المائل

وبناء على ذلك

نقوم بالتغييرات التي نريدها فقط داخل هذه الارقام

ونذهب الى انشاء الحرف

هنا اخترنا حرف N

ومن الممكن ان تختار اي حرف اخر



## اليكم الان حرف N بالتشكيل بالارقام

01111110  
00000100  
00001000  
00010000  
01111110

ركزو جيدا للشكل السابق سوف تجد حرف ال N

نعود الان الى استخدام نفس الكود السابق

ونضع فقط التغييرات التي تم شرحها الان

### واليكم الكود :

كود:

```
void main()
{
  TRISB=0;
  PORTB=0;

  TRISA=0;
  PORTA=0;

loop:

  PORTA=0B111110;
  PORTB=0B01111110; delay_us(500);
  ////////////

  PORTA=0B111101;
  PORTB=0B00000100; delay_us(500);
  ////////////

  PORTA=0B111011;
  PORTB=0B00001000; delay_us(500);
  ////////////

  PORTA=0B101111;
  PORTB=0B00010000; delay_us(500);
  ////////////

  PORTA=0B011111;
  PORTB=0B01111110; delay_us(500);

goto loop;
}
```

### للتاكيد هنا

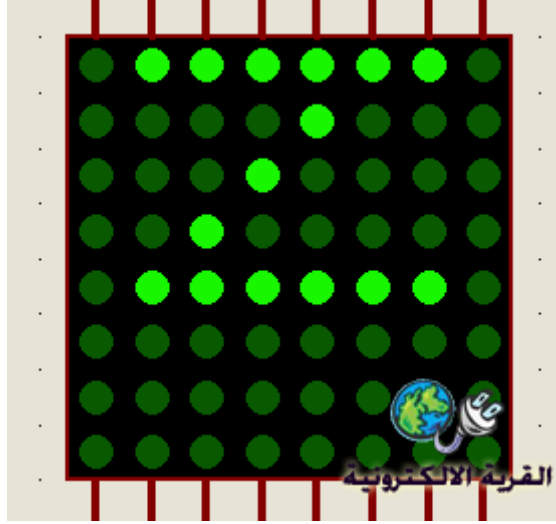
تم هنا كما سبق وشرحنا قد تم تغيير التوقيت من ms الى التوقيت الاسرع وهو us

وقمنا بتغيير ال data ( ما تشاهدة العين على الشاشة )

بعد التغييرات التي اجريناها

نذهب الى الا خ بریتوس ونشاهد بعد التشغيل

سوف تشاهدون هذا الشكل باذن الله



مبروك عليكم جميعا

لقد قمنا الان بكتابة اول حرف بالليد ماتريكس

واتمنى لكل من تابع هذا الدرس

ان يقوم بالتعديلات كما يحب ويخرج احرف اخرى

ماسوف تحتاجة داخل البريتوس

هو ان تعدل شكل مكعب الليد ماتريكس

( نقوم بزر الماوس الايسر بتحديد المكعب ثم نضغط على زر الماوس الايمن ونختار السهم الذي يدور حول نفسه )

وهكذا سوف تشاهد شاشة الليد بشكل معتدل للمشاهدة



وتتعرفو على الاوامر التالية

char

for

(while(1

واليكم هذا الكود:

هو ليس له صلة بالماتركس

وانما يشمل على مجموعة اوامر لابد من استخدامها لتحريك النصوص في الماتركس  
كود:

```
void main(){
char x;
TRISB=0; PORTB=0;
delay_ms(1000);
while(1){
for(x=0;x<9;x++){ PORTB++; delay_ms(1000);}
portb=0;delay_ms(1000);
}
}
```

كما تلاحظون في هذا الكود:

فقد احتوى هذا الكود كل الاوامر المطلوب منا ان نستخدمها في برامجنا



اتركم الان مع دروسكم



بسم الله ولا حول ولا قوة الا بالله



وتراجعوا دروس البرمجة لاستاذنا احمد

واتمنى ان تكونو قد تعرفتو على :

(char) وهى طبعا تستخدم لحجز مساحة فى الذاكرة للميكرو ونعطيها اى اسم وهذه المساحة كما قال لنا استاذنا احمد اسمها متغير بمعنى اننا من الممكن ان نقوم بتغييرها كما نشاء فى اى وقت .

مثال:

```
char n ; char x ; char m;
```

(for) وهذا الامر نستخدمه اذا اردنا ان يقوم البك بامر معين عدد معين من المرات

مثال :

كود:

```
for(x=0;x<9;x++)
```

العدد هنا اعتقد واضح 9

(while(1

كود:

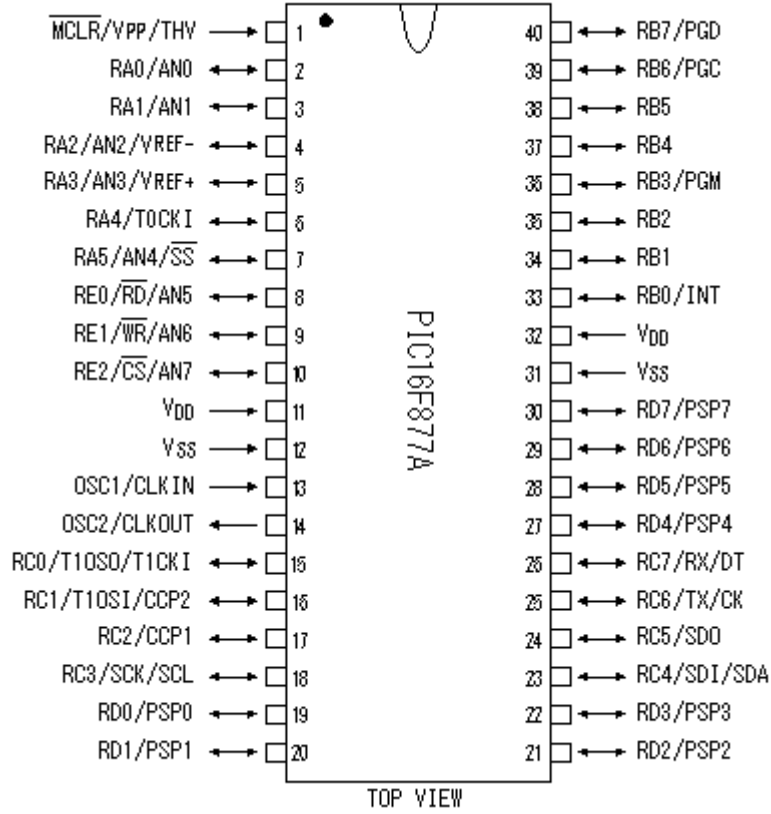
```
while(1)
```

وهذا الامر ان يعيد البك الاوامر كلها وباستمرار

وهى بالظبط مثل loop: وهذه يتم وضعها فى اول الكود و goto loop؛ وهذه يتم وضعها اخر الكود

00000000000000000000000000000000

واليكم الان صور الاخ f877a16 مممممممم يعتبر غول بالمقارنة بالاخير f84a16



وللزيادة فى المعلومات اليكم الرابط

[http://hobby\\_elec.piclist.com/e\\_pic6\\_h04.htm](http://hobby_elec.piclist.com/e_pic6_h04.htm)

وكما تشاهدون عدد المخارج كثيرة جدا

كنا نستخدم المخارج A B

وسوف نستخدم اليوم البورت B ونستبدل البورت A بالبورت D فى PIC16F877A

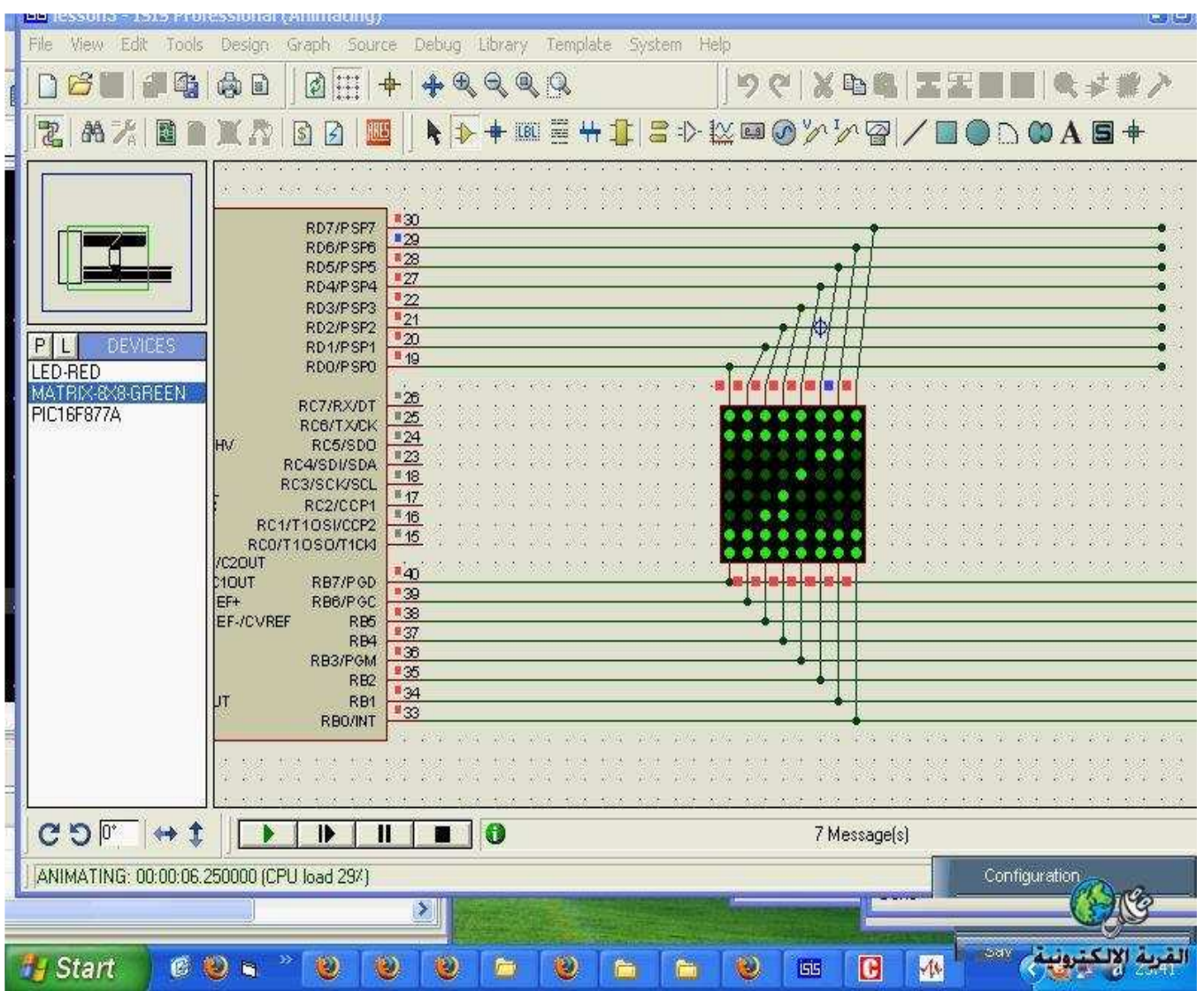
وهذا يعطينا 8 مخارج + 8 مخارج

وبامكاننا الان ان نقوم بكتابة الحرف N بشكل اكبر ونقط اكثر

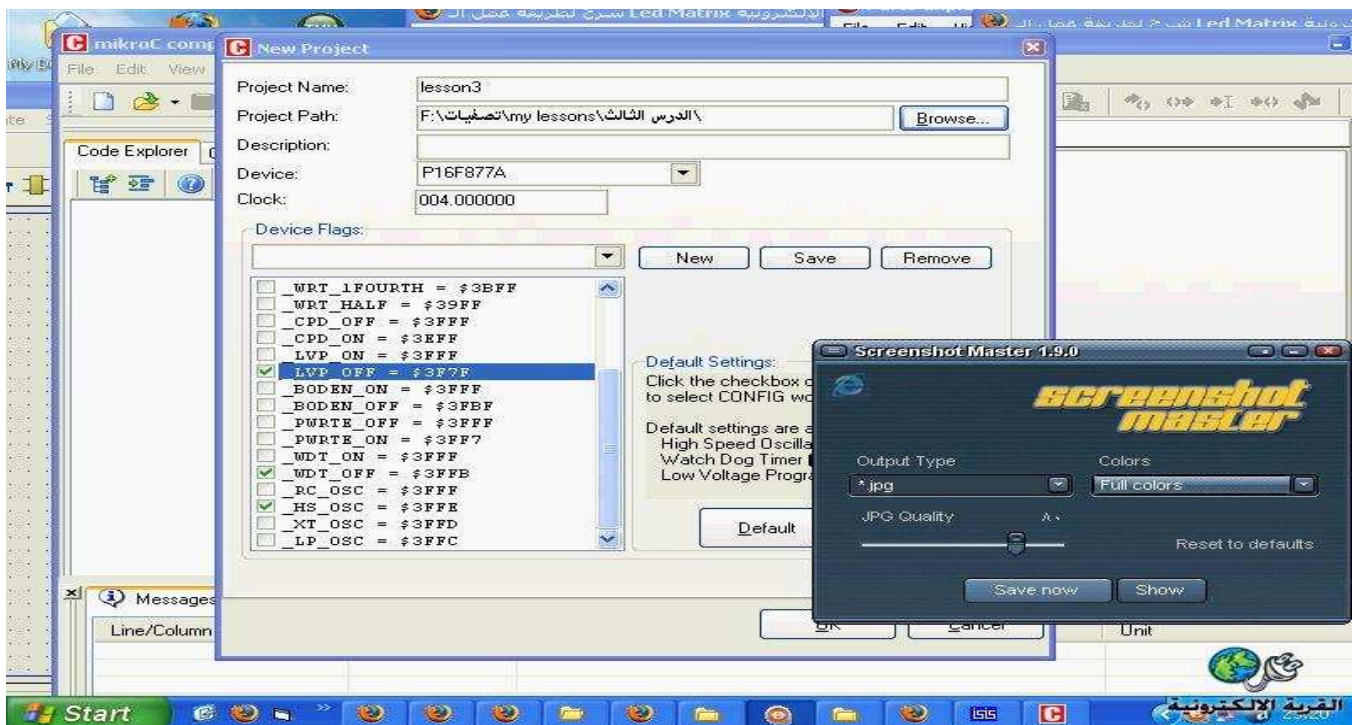








واليكم ايضا صورة لاعدادات الميكروسى هذه العلامة باللون الازرق مهم فيها هذا  
(LVE\_OFF الخيار)



ياترى مارايكم فى المتحكم الجديد؟؟

HH

والان فلنذهب جميعا الى الاكشن

واللعب

والاشياء الجميلة

كلنا نقوم بتحميل المعدات وننصب الخيم

led matrix for animation ونلعب مع ال



ولكن قبل ان نتحرك

نحتاج الى عودة سريعة لما سبق فى هذا الدرس

لتحرك الصفوف D كما شاهدتم فقد استخدمنا البورت

او ماتشاهدة العين على الشاشة صح ام لا ؟ DATA لاجراج ال B واستخدمنا بورت

اوك

وهذا ماتم عرضة

```
0B11111111
0B11111111
0B00000110
0B00001000
0B00010000
0B00110000
0B11111111
0B11111111
```

كما تلاحظون فقد قمت بتظليل ماتم اظهارة على الشاشة

والا ن سوف نحتاج الاوامر التى سبق التنوية عنها

واولهم امر ال char وكما قلنا من قبل ان هذا لحجز مكان فى الذاكرة ويكون هذا دائما فى اول الكود وقبل كلمة void main

هنا لدينا 8 اعمدة

اذن سوف نحتاج الى حجز 8 اماكن

وسوف نعطيهم اسماء طبعاً مختلفة

هكذا سوف تجدوهم فى اعلى البرنامج

كود:

```
char x1=0B11111111;  
char x2=0B11111111;  
char x3=0B00000110;  
char x4=0B00001000;  
char x5=0B00010000;  
char x6=0B00110000;  
char x7=0B11111111;  
char x8=0B11111111;
```

وكما تشاهد ياخى ويازميلى



انه تم وضع الاداتا بنفس الترتيب ونفس الشكل مmmmmmmmm

سوف نعيد التجربة السابقة الان

ولكن مع استخدام المتغيرات

واليكم الكود :

كود:

```
char x1=0B11111111;  
char x2=0B11111111;  
char x3=0B00000110;  
char x4=0B00001000;  
char x5=0B00010000;  
char x6=0B00110000;  
char x7=0B11111111;  
char x8=0B11111111;
```

```
char n;
```

```

void main()
{
TRISB=0;
PORTB=0;

TRISd=0;
PORTd=0;

loop:

PORTd=0B11111110;
PORTB=x1; delay_us(300);
//////////

PORTd=0B11111101;
PORTB=x2; delay_us(300);
//////////

PORTd=0B11111011;
PORTB=x3; delay_us(300);
//////////

PORTd=0B11110111;
PORTB=x4; delay_us(300);
//////////

PORTd=0B11101111;
PORTB=x5; delay_us(300);

//////////

PORTd=0B11011111;
PORTB=x6; delay_us(300);
//////////

PORTd=0B10111111;
PORTB=x7; delay_us(300);
//////////

PORTd=0B01111111;
PORTB=x8; delay_us(300);

goto loop;
}

```

**قم بوضع هذا الكود وسوف تشاهد نفس النتيجة السابقة**

**ولكن استخدمنا هنا التعريفات وهي**

**%%char**

**سوف نتقل الان الى الخطوة التالية فى طريقنا لتحريك الماتركس**

**وهو الامر فور for**

**سوف نقوم باستخدام الامر for لكى يقوم البرنامج باعادة الاوامر الى عدد ممكن  
نختار 100 مثلا اوك 100 مرة ان شاء الله**

## واليكم الكود :

كود:

```
char x1=0B11111111;
char x2=0B11111111;
char x3=0B00000110;
char x4=0B00001000;
char x5=0B00010000;
char x6=0B00110000;
char x7=0B11111111;
char x8=0B11111111;

char n;

void main()
{
TRISB=0;
PORTB=0;

TRISd=0;
PORTd=0;

loop:

for(n=0;n<100;n++){

PORTd=0B11111110;
PORTB=x1; delay_us(300);
//////////

PORTd=0B11111101;
PORTB=x2; delay_us(300);
//////////
PORTd=0B11111011;
PORTB=x3; delay_us(300);
//////////
PORTd=0B11110111;
PORTB=x4; delay_us(300);
//////////
PORTd=0B11101111;
PORTB=x5; delay_us(300);

//////////

PORTd=0B11011111;
PORTB=x6; delay_us(300);
//////////
PORTd=0B10111111;
PORTB=x7; delay_us(300);
//////////
PORTd=0B01111111;
PORTB=x8; delay_us(300);

}

goto loop;
}
```

استخدم هذا الكود طبعا على نفس ملف الهاردوير



ومرة بعدها

والتي بعدها هى نهاية البرنامج بالكامل

اذهب الى ما قبل جو تو لوب

وسوف نضع هذا السلم

او الدوران

سمية كما تحب

$x1=x2$

$x2=x3$

$x3=x4$

$x4=x5$

$x5=x6$

$x6=x7$

$x7=x8$

$x8=x1$

????????????????????????????

حد فاهم حاجة ؟؟؟



اعتقد انه هناك بعض فهم والاخر مmmmmmmmm

او ك ماذا تم هنا

ماتم هنا انا قمنا بتغيير ما يتم عرضة على الشاشة

ولكن بدون مانغير مكانهم

وانما قمنا بامر الرموز وهى التى تحتوى على الداتا بالدوران حول نفسها  
واليكم الكود النهائى

.....



كود:

```
char x1=0B11111111;
char x2=0B11111111;
char x3=0B00000110;
char x4=0B00001000;
char x5=0B00010000;
char x6=0B00110000;
char x7=0B11111111;
char x8=0B11111111;

char n;

void main()
{
TRISB=0;
PORTB=0;

TRISd=0;
PORTd=0;

loop:

for(n=0;n<100;n++){

PORTd=0B11111110;
PORTB=x1; delay_us(300);
//////////
PORTd=0B11111101;
PORTB=x2; delay_us(300);
//////////
PORTd=0B11111011;
PORTB=x3; delay_us(300);
//////////
PORTd=0B11110111;
PORTB=x4; delay_us(300);
//////////
PORTd=0B11101111;
PORTB=x5; delay_us(300);
//////////
PORTd=0B11011111;
PORTB=x6; delay_us(300);
//////////
PORTd=0B10111111;
PORTB=x7; delay_us(300);
//////////
PORTd=0B01111111;
PORTB=x8; delay_us(300);
}
x1=x2;          /// بداية الدوران
x2=x3;
x3=x4;
x4=x5;
x5=x6;
x6=x7;
x7=x8;
x8=x1;          /// نهاية الدوران

goto loop;
}
```

مبروك على الجميع قد تم تحريك حرف ال N



السلام عليكم ورحمة الله وبركاته

بسم الله الرحمن الرحيم

اهلا وسهلا بكم جميعا  
اتمنى ان تكونوا استفدتم من ماسبق

NNNNNNNNNN

كما تعودنا فى هذه الدروس اننا نستخدم طريقة البناء السلس وليس الانتقال المفاجيء (بمعنى  
اننا نقدم الدرس بما سبق فى الدرس السابق من اكواد)  
وهذا حتى لا تحدث فجوة بين كل درس وماقبله او مابعدة

وحتى تكون المتابعة للمبتدىء ايسر

MMMMMMMMMMMMnasserMMMMMMMMMMMMMMMM

فى درسنا السابق تمكنا خطوة بخطوة من اظهار حرف ال N ثم قمنا بتحريكة

وهذه الخطوة فى حد ذاتها قد تسعد الكثيرين

ار يو اوك؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟؟

فى درسنا السابق ايضا تعلمنا ان مجموعة char او مجموعة المتغيرات هى التى نشاهدة على  
العين

وتعلمنا ايضا ان هذه المجموعة :

كود:

```
char x1=0B11111111;  
char x2=0B11111111;  
char x3=0B00000110;  
char x4=0B00001000;  
char x5=0B00010000;  
char x6=0B00110000;  
char x7=0B11111111;  
char x8=0B11111111;
```

نستطيع ان نعطيها اسم (مادة العرض ) او (الداتا) او (الشريط السينمائى)

او ك سوف نعطيها الاسم الاخير (الشريط السينمائى )

هذا هو الشريط السينمائي كما تشاهدون وعليه يظهر حرف ال N

```
؛char x1=0B11111111
؛char x2=0B11111111
؛char x3=0B00000110
؛char x4=0B00001000
؛char x5=0B00010000
؛char x6=0B00110000
؛char x7=0B11111111
؛char x8=0B11111111
```

وكما نلاحظ جميعا ان عدد الاسطر 8 وعدد مكعب الماتركس 8

وهكذا لا يوجد فرق في العدد

وبناء على ذلك تجد ان الحرف لا يكاد يختفى حتى يظن من جديد

(والحل)

الحل انه يجب علينا ترك مسافة بين عرض الحرف وبين تكرار عرضة  
وهذا سوف يكون باضافة سطر فارغ بمعنى انه سوف يكون بالكامل 0

```
؛char x9=0B00000000
```

هل تلاحظون ان عدد اسطر الماتركس 8 وعدد المتغيرات 9

نعم

فسوف تقوم شاشة العرض بعرض ال 8 المسجلين بالترتيب

وعند التقليب سوف ياخذ كل سطر موقع السطر الذي قبله

وهكذا سوف يتم العرض بالتتابع

في تجربتنا نستخدم ال X88 وبة 8 اعمدة ولكننا نريد اليوم

**اظهار كلمة كاملة او جملة**

## هذا بالظبط كما شرحنا

سوف تقوم الشاشة بعرض 8 من المجموعة char وعند التقليل سوف ياخذ كل متغير قيمة ماسبقه

سوف نقوم الان بوضع الكود الجديد ولاحظو التغييرات التى تمت بينهم

طبعاً لن نجد اشرف من((( لفظ الجلالة))) لاطهارة على تجاربنا الاولى

واليكم الكود :

كود:

```
char x1=0b00000000;  
char x2=0b00000000;  
char x3=0b11111110;  
char x4=0b11111110;  
char x5=0b00000000; ////  
char x6=0b11111110;  
char x7=0b11111110;  
char x8=0b00000110;  
char x9=0b00000110;  
char x10=0b11111110;  
char x11=0b11111110;  
char x12=0b00000110;  
char x13=0b00000110;  
char x14=0b00111110;  
char x15=0b00101110;  
char x16=0b00101000;  
char x17=0b00111000;  
char x18=0b00000000; ////  
char x19=0b00000000;  
char x20=0b00000000;  
char x21=0b00000000;
```

```
char n;
```

```
void main()
```

```
{
```

```
TRISB=0;
```

```
PORTB=0;
```

```
TRISd=0;
```

```
PORTd=0;
```

```
loop:
```

```
for(n=0;n<100;n++){
```

```
PORTd=0B11111110;
```

```
PORTB=x1; delay_us(300);
```

```
//////////
```

```
PORTd=0B11111101;
```

```
PORTB=x2; delay_us(300);
```

```
//////////
```

```
PORTd=0B11111011;
```

```
PORTB=x3; delay_us(300);
```

```
//////////
```

```
PORTd=0B11110111;
PORTB=x4; delay_us(300);
//////////
PORTd=0B11101111;
PORTB=x5; delay_us(300);

//////////

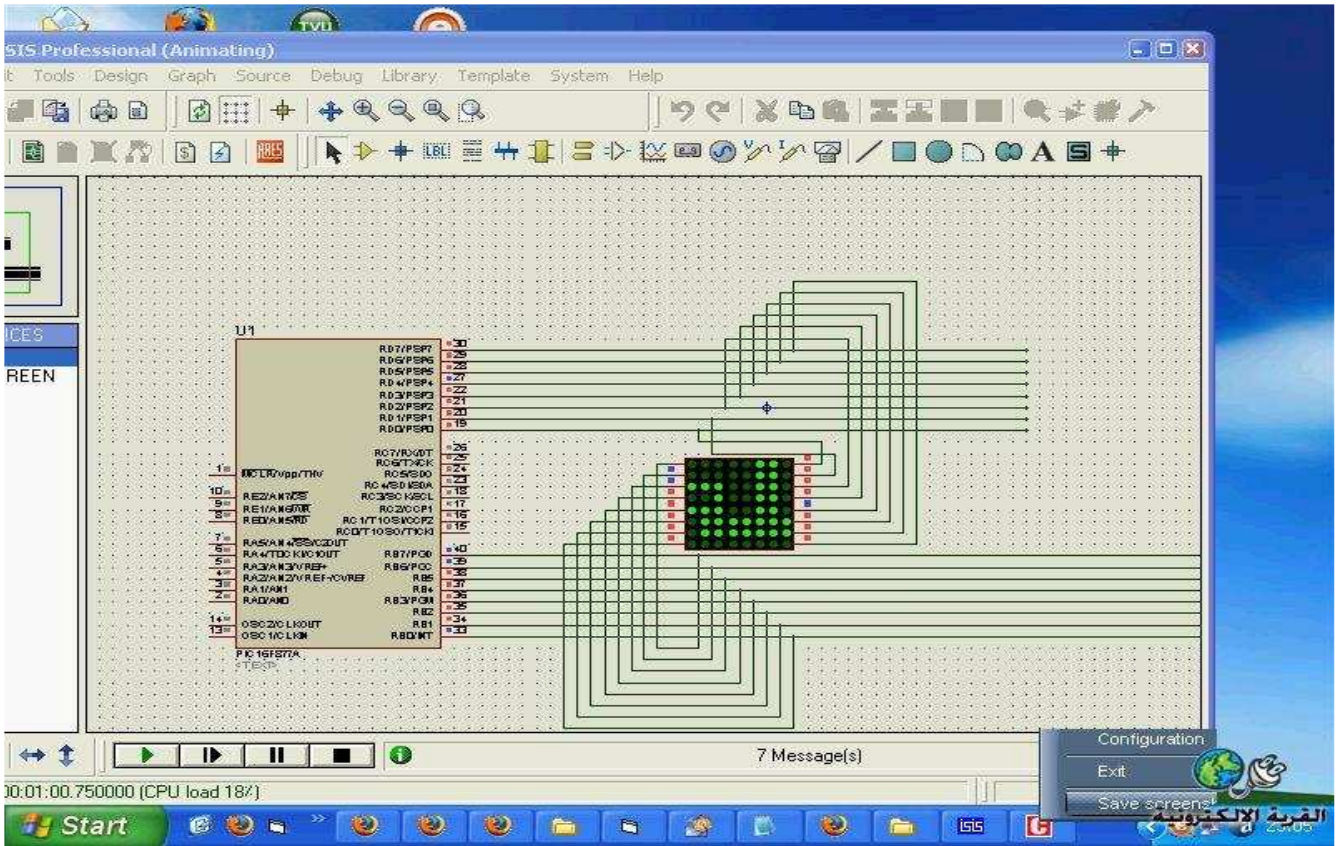
PORTd=0B11011111;
PORTB=x6; delay_us(300);
//////////
PORTd=0B10111111;
PORTB=x7; delay_us(300);
//////////
PORTd=0B01111111;
PORTB=x8; delay_us(300);

}

x1=x2;
x2=x3;
x3=x4;
x4=x5;
x5=x6;
x6=x7;
x7=x8;
x8=x9;
x9=x10;
x10=x11;
x11=x12;
x12=x13;
x13=x14;
x14=x15;
x15=x16;
x16=x17;
x17=x18;
x18=x19;
x19=x20;
x20=x21;
x21=x1;

goto loop;
}
```

سوف نشاهد جميعا لفظ الجلالة تتحرك من اليسار الى اليمين



😊 وهكذا نتعلم سويا 😊

((((( (((((((((((((ومعلومة اليوم))))))))))))))

اننا نستخدم (الشريط السينمائي) كما نريد حتى لو كان 100 سطر

لن يوقفنا غير ذاكرة الميكرو 😊

```

0000 0000000000000000000011000000000000110000000000000000
0000 0000000000000000000011110000000000110000000000000000
0000 0000000000000000000011001100000000110000000000000000
0000 0000000000000000000011000011000000110000000000000000
0000 0000000000000000000011000000110000110000000000000000
0000 0000000000000000000011000000001100110000000000000000
0000 0000000000000000000011000000000011110000000000000000
0000 0000000000000000000011000000000000110000000000000000

```

ويظل وسط الكود كما هو

ان يتم ادراج الداتا واحدة تلو الاخرى

## ملحوظة هامة

(( لو عكسنا الخرج )) فسوف يتم عكس العرض

عرضنا اليوم كان من اليسار الى اليمين

ولو تم عكس العرض سوف يصبح العرض من اليمين الى اليسار

كود:

```
PORTd=0B11111110;
PORTB=x1; delay_us(300);
//////////

PORTd=0B11111101;
PORTB=x2; delay_us(300);
//////////

PORTd=0B11111011;
PORTB=x3; delay_us(300);
//////////

PORTd=0B11110111;
PORTB=x4; delay_us(300);
//////////

PORTd=0B11101111;
PORTB=x5; delay_us(300);

//////////

PORTd=0B11011111;
PORTB=x6; delay_us(300);
//////////

PORTd=0B10111111;
PORTB=x7; delay_us(300);
//////////

PORTd=0B01111111;
PORTB=x8; delay_us(300);
```

وهذا هو العكس الذي اقصد

كود:

```
PORTd=0B11111110;
PORTB=x8; delay_us(300);
//////////

PORTd=0B11111101;
PORTB=x7; delay_us(300);
//////////

PORTd=0B11111011;
PORTB=x6; delay_us(300);
//////////

PORTd=0B11110111;
PORTB=x5; delay_us(300);
//////////

PORTd=0B11101111;
PORTB=x4; delay_us(300);

//////////

PORTd=0B11011111;
```



```
PORTB=x3; delay_us(300);  
/////////  
PORTd=0B10111111;  
PORTB=x2; delay_us(300);  
/////////  
PORTd=0B01111111;  
PORTB=x1; delay_us(300);
```

بعدد الاعمدة التى نستخدمها

ويتم اضافة اى شىء الى الشريط السينمائى

على ان يتم ادراج نفس العدد  
الى نظام التقلاب

واخيرا يجب ان يكون اخر رقم يتم انتقاله ياخذ موقع اول رقم



والا لن يتم العرض الا مرة واحدة

\*  
\*  
\*

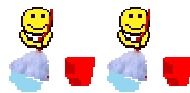
والان وكالمعتاد

اترككم مع اجهزكم ولا تنسوا ان تقوموا بالتجارب باستمرار



نسيت اليوم رش الماء امام الدكان

نروح نرش المية











---

---

سوف نقوم الان وكما تعودنا ان نستخدم الكود السابق

فى نفس التجربة واليكم الكود السابق

كود:

```
char x1=0b00000000;
char x2=0b00000000;
char x3=0b11111110;
char x4=0b11111110;
char x5=0b00000000; ////
char x6=0b11111110;
char x7=0b11111110;
char x8=0b00000110;
char x9=0b00000110;
char x10=0b11111110;
char x11=0b11111110;
char x12=0b00000110;
char x13=0b00000110;
char x14=0b00111110;
char x15=0b00101110;
char x16=0b00101000;
char x17=0b00111000;
char x18=0b00000000; ////
char x19=0b00000000;
char x20=0b00000000;
char x21=0b00000000;
```

```
char n;
```

```
void main()
```

```
{
```

```
TRISB=0;
```

```
PORTB=0;
```

```
TRISd=0;
```

```
PORTd=0;
```

```
loop:
```

```
for(n=0;n<100;n++){  
  
PORTd=0B11111110;  
PORTB=x1; delay_us(300);  
//////////  
  
PORTd=0B11111101;  
PORTB=x2; delay_us(300);  
//////////  
  
PORTd=0B11111011;  
PORTB=x3; delay_us(300);  
//////////  
  
PORTd=0B11110111;  
PORTB=x4; delay_us(300);  
//////////  
  
PORTd=0B11101111;  
PORTB=x5; delay_us(300);  
  
//////////  
  
PORTd=0B11011111;  
PORTB=x6; delay_us(300);  
//////////  
  
PORTd=0B10111111;  
PORTB=x7; delay_us(300);  
//////////  
  
PORTd=0B01111111;  
PORTB=x8; delay_us(300);  
  
}
```

```
x1=x2;  
x2=x3;  
x3=x4;  
x4=x5;  
x5=x6;  
x6=x7;  
x7=x8;  
x8=x9;  
x9=x10;  
x10=x11;  
x11=x12;  
x12=x13;  
x13=x14;  
x14=x15;  
x15=x16;
```







ولتبسي الامور

اقول لكم ببساطة شديدة

مدخلى الداتا وهم B&A والتي اوصلناهم ببورت 1 هم الداتا

وماتعنية كلمة داتا هنا  
اننا نسمح للمزيح بالعمل او لا نسمح

ولكى نسمح له بالعمل فسوف نعطي البورت المتصل بهم هذا الامر



؛porta.f1=1; porta.f1=0

على ان تكون الداتا متصلة بالبورت رقم  $F1 < 1$

هكذا نكون خالصنا من الخطوة الاولى

ونقوم باعداد هذا الامر

كود:

```
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

هذا الامر يعنى اننا قمنا بفتح الداتا وفتح النبضة  
ثم اغلقنا الداتا واغلقنا النبضة

نحتاج هنا الى تركيز اكثر

قمنا بفتح الداتا مع النبضة فى وقت واحد

ثم اغلقناهم.....



( لو فتحنا احدهم واغلقناها وفتحنا الاخرى واغلقناها )

لن يحدث اى شىء  
بمعنى ان المسجل لن يتحرك

ويجب ان يتم فتح الاثنين فى وقت واحد

ثم غلقهم

∇

∇

∇

لكى يتقدم الاى سى خطوة الى الامام

معنى خطوة للامام

انة فى بداية عمل مسجل الازاحة تكون مخارجة بالكامل تعطى موجب

وعند اعطائة هذا الامر

سوف تجد ان البورت الاول لمسجل الازاحة اصبح سالب (لونة ازرق) فى بريتوس

---

وهكذا نفهم ان الامر الاول هذا لمجموع الاوامر كلها

والامر الاخير معناه خطوة للامام

اوك؟؟؟؟؟؟؟؟؟؟؟؟؟؟

وان قمنا باعطاء الامر الاخير مرتين متتابعتين مثل هذا



ار يوووو اوك:

نعود الان الى الماتركسات

كما فهمنا من قبل

انة لاطهار صورة ما او اى شىء ع الماتركس

فاننا نقوم باظهار الداتا

مع طرف سالب  
من مخارج بورت A فى الميكروكونترول

وهذا هو مايحدث بالظبط مع الماتركس باستخدام مسجل الازاحة

نظهر الداتا ونذهب بالخطوات الى المكان الذى نريده

ثم نعطى التوقيت للامرين معا

كمثال

```
portb=0b10101010
```

```
؛porta.f1=0; porta.f0=0 ؛porta.f0=1;porta.f1=1  
؛porta.f0=0 ؛porta.f0=1;porta.f1=1; porta.f1=0
```

اعتقد ان الاغلبية قد فهمت

ان هذا الامر

سوف يضىء العمود رقم 2 من الماتركس

لماذا؟؟؟؟؟؟؟؟

لانا تقدمنا خطوتين الى الامام

ولا ننسى طبعا انه فى بداية الكود نقوم بتفعيل الداتا كما سبق

=====

الان اليكم الكود الجديد:



كود:

```
char x1=0b00000000;
char x2=0b00000000;
char x3=0b11111110;
char x4=0b11111110;
char x5=0b00000000; ////
char x6=0b11111110;
char x7=0b11111110;
char x8=0b00000110;
char x9=0b00000110;
char x10=0b11111110;
char x11=0b11111110;
char x12=0b00000110;
char x13=0b00000110;
char x14=0b00111110;
char x15=0b00101110;
char x16=0b00101000;
char x17=0b00111000;
char x18=0b00000000; ////
char x19=0b00000000;
char x20=0b00000000;
char x21=0b00000000;

char n;

void main()
{
TRISB=0;
PORTB=0;

TRISa=0;
PORTa=0;

loop:

for(n=0;n<100;n++){

porta.f1=1;   porta.f1=0;//////////*****//////////

//////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x1;
delay_us(300);
```

```
////////////////////////////////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x2;
delay_us(300);
```

```
////////////////////////////////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x3;
delay_us(300);
```

```
////////////////////////////////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x4;
delay_us(300);
```

```
////////////////////////////////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x5;
delay_us(300);
```

```
////////////////////////////////////
```

```
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x6;
delay_us(300);
```

```
////////////////////////////////////
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x7;
delay_us(300);
```

```
////////////////////////////////////
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portb=x8;
delay_us(300);
```

```
}
```

```
x1=x2;
x2=x3;
x3=x4;
x4=x5;
x5=x6;
x6=x7;
x7=x8;
x8=x9;
x9=x10;
x10=x11;
x11=x12;
x12=x13;
```

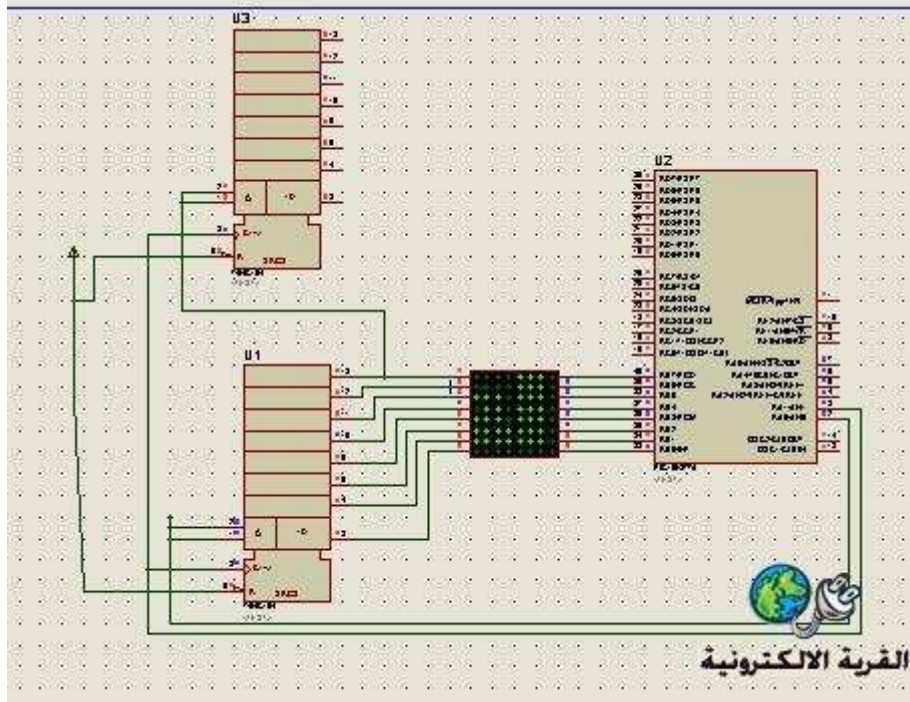
```
x13=x14;  
x14=x15;  
x15=x16;  
x16=x17;  
x17=x18;  
x18=x19;  
x19=x20;  
x20=x21;  
x21=x1;
```

```
goto loop;  
}
```

كالمعتاد قم بمقارنة ماتم تغييرة بين الكودين

نتبته فقط من اننا ابدلنا بورت D ببورت A

واليكم الان صورة كاملة للهاردوير



نلاحظ فى الصورة اننا قمنا بتوصيل

2 مسجل ازاحة

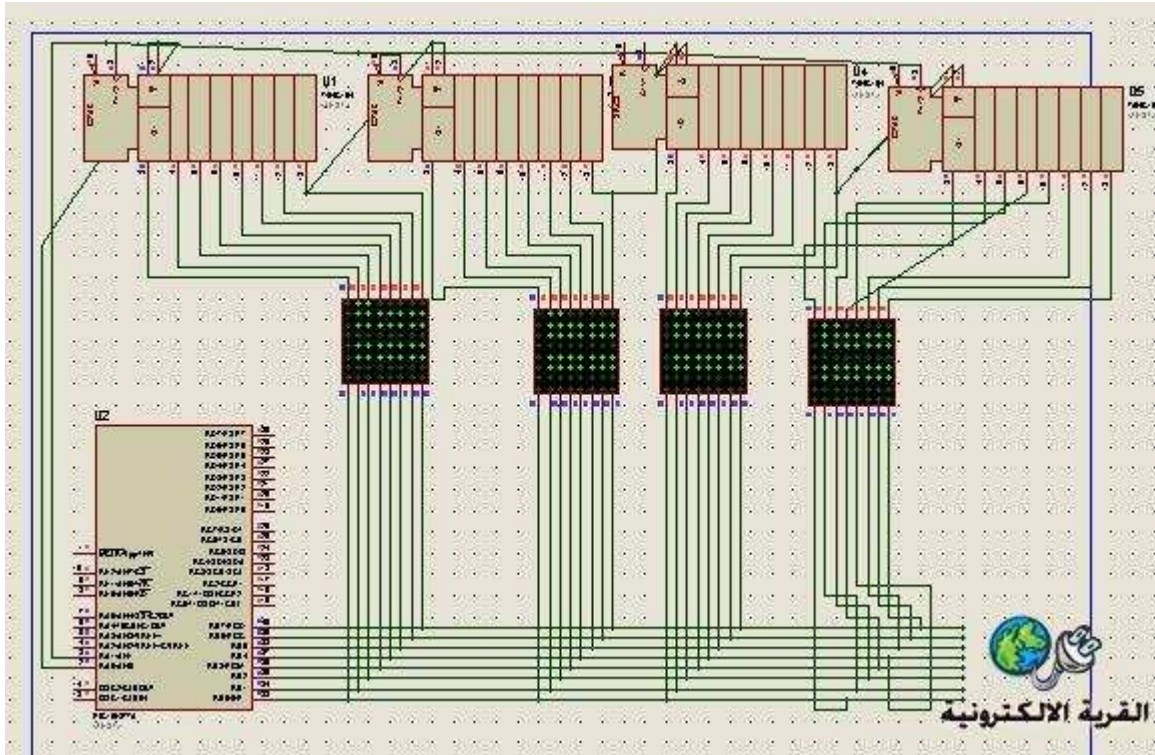
وهذا لكى نتعلم سويا كيفية توصيل الامسجلات بعضها البعض







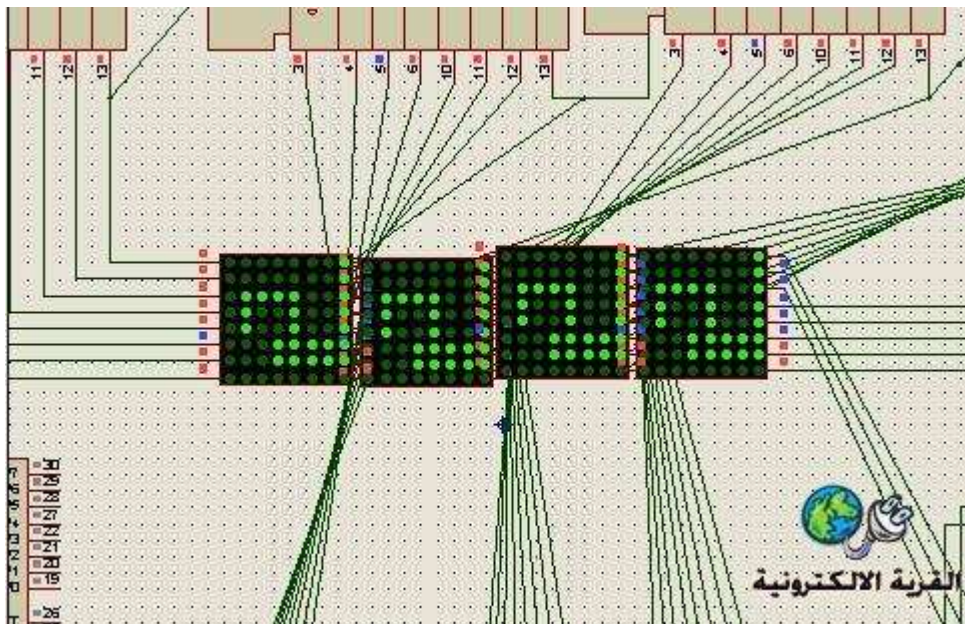
اعتقد ان جميعا يعرف ذلك ؟



((( )))

عموما هذه الصورة قمت بترتيب الماتركسات نسبيا

(معلش الماوس معاكسنى كثير اليوم خخخخخخخخخخخخخخ)



عموما بقدر المستطاع

كما نلاحظ فى الصور هنا ان الماتركسات كلها تعطى نفس الشكل

وهو اخر جزء من لفظ الجلالة المتحرك

هذا طبعا لانى استخدمت الكود السابق وهو لو تذكرنا جميعا  
الجزء الاوسط كان عبارة عن 8 خطوات لمسجل الازاحة والخطوة هنا تعنى  
خطوة للامام .....ياترى فاكرين ولا

---

وهكذا دخلنا الى درسنا الجديد دون ان نشعر باى تغيير

نعم

وللتوضيح اكثر

ننتبه جميعا الى الكلمات القادمة

اولا سوف نقوم بتجزئة الكود السابق الى 3 اجزاء

وهى:

الجزء الاول

اعلى الكود

وهو عبارة عن الداتا التى سوف نرسلها (تشاهدها العين على الشاشة)

الجزء الاوسط  
وهو مجموعة الاوامر الخاصة بالكود

وفى درسنا رقم (5) قمنا بالتحرك 8 خطوات للامام .....صح ام لا؟





```

char d28=0b00000000;
char d29=0b00000000;
char d30=0b00010000;
char d31=0b00111000;
char d32=0b01111100;
char d33=0b11111110;
char d34=0b00111000;
char d35=0b00111000; ////
char d36=0b00111000;
char d37=0b00111000;
char d38=0b00111000;
char d39=0b11111110;
char d40=0b11101110;
char d41=0b11000110;
char d42=0b10000010;
char d43=0b10000010;
char d44=0b00000000;
char d45=0b00000000;

char x;
void main() {
TRISA=0;
TRISB=0;
portb=0;
porta=0;
loop:

for (x=0;x<32;x++) {
porta.f2=1;
////////////////////////////////////
porta.f1=1;  porta.f1=0;

portb=d45;  delay_us(250);
////
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d44;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d43;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d42;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d41;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d40;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d39;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d38;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d37;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portb=d36;  delay_us(250);

porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

```

```
portb=d35; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d34; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d33; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d32; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d31; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d30; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d29; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d28; delay_us(250);
////
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d27; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d26; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d25; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d24; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d23; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d22; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d21; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d20; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d19; delay_us(250);

porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;

portb=d18; delay_us(250);
porta.f0=1;porta.fl=1; porta.fl=0; porta.f0=0;
```

```
portb=d17; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portb=d16; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portb=d15; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portb=d14; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portb=d13; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portb=d12; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
}
```

```
d0=d1;  
d1=d2;  
d2=d3;  
d3=d4;  
d4=d5;  
d5=d6;  
d6=d7;  
d7=d8;  
d8=d9;  
d9=d10;  
d10=d11;  
d11=d12;  
d12=d13;  
d13=d14;  
d14=d15;  
d15=d16;  
d16=d17;  
d17=d18;  
d18=d19;  
d19=d20;  
d20=d21;  
d21=d22;  
d22=d23;  
d23=d24;  
d24=d25;  
d25=d26;  
d26=d27;  
d27=d28;  
d28=d29;  
d29=d30;  
d30=d31;  
d31=d32;  
d32=d33;  
d33=d34;  
d34=d35;  
d35=d36;  
d36=d37;  
d37=d38;  
d38=d39;  
d39=d40;  
d40=d41;
```



```
d41=d42;  
d42=d43;  
d43=d44;  
d44=d45;  
d45=d0;  
  
goto loop;  
  
}
```

معنا فى هذا الكود

تم ادراج داتا تحتوى على 45 عمود

يشمل كلمة ((لفظ الجلالة )) ثم بعدها حرف N ثم سهم

وكل هذا يتحرك من اليسار الى اليمين



ملاحظة



فلنتذكر جميعا

عملية التبديل فى الاتجاهات

ماهى الا مجرد ترتيب اخراج الداتا

فلو كانت اول خطوة = اول سطر من الداتا

فالحركة سوف تكون من اليمين الى اليسار

واذا كانت اول خطوة = اخر سطر من الداتا

فالعكس



اتمنى ان تكون هذه النقطة واضحة

لانىا تحتاج الى فهم

وسوف نستخدم الاتنين معا فى دروسنا القادمة ان شاء الله

+++++

فلنعود الان الى درس اليوم



لو قارنت بين هذا الكود وماسبق من الاكواد

فلن تجد تغييرا الى **فى وسط الكود فقط**

ففى الاكواد السابقة كنا نتحرك خطوة للامام ثم نرسل داتا

فكنا نتحرك 8 خطوات

لاننا كنا من قبل نتعامل فى حدود ال8 اعمدة

اما اليوم فلدينا 32 عمود

ولذلك لو قمت بالعد للخطوات سوف تجدهم 32 خطوة

وهذه هى الخطوة الاخيرة على سبيل المثال

كود:

```
portb=d12; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

وسوف نجد ايضا ان الديلاى ( delay\_us ) او فترة الانتظار اقل من قبل

وهذه فقط لنزيد من سرعة الحركة وليس الا



فكلما ذات المدة كانت الحركة ابطا وكلما قلت المدة يكون البرنامج اسرع

هذا بالاضافة الى عدد مرات الاعدادة الموجودة فى اسفل كلمة لوبوب loop  
والموجودة فى امر فور for





دتم فى رعاىة الله

سبحان الله وبحمدة سبحان الله العظيم

((الدرس السابع))

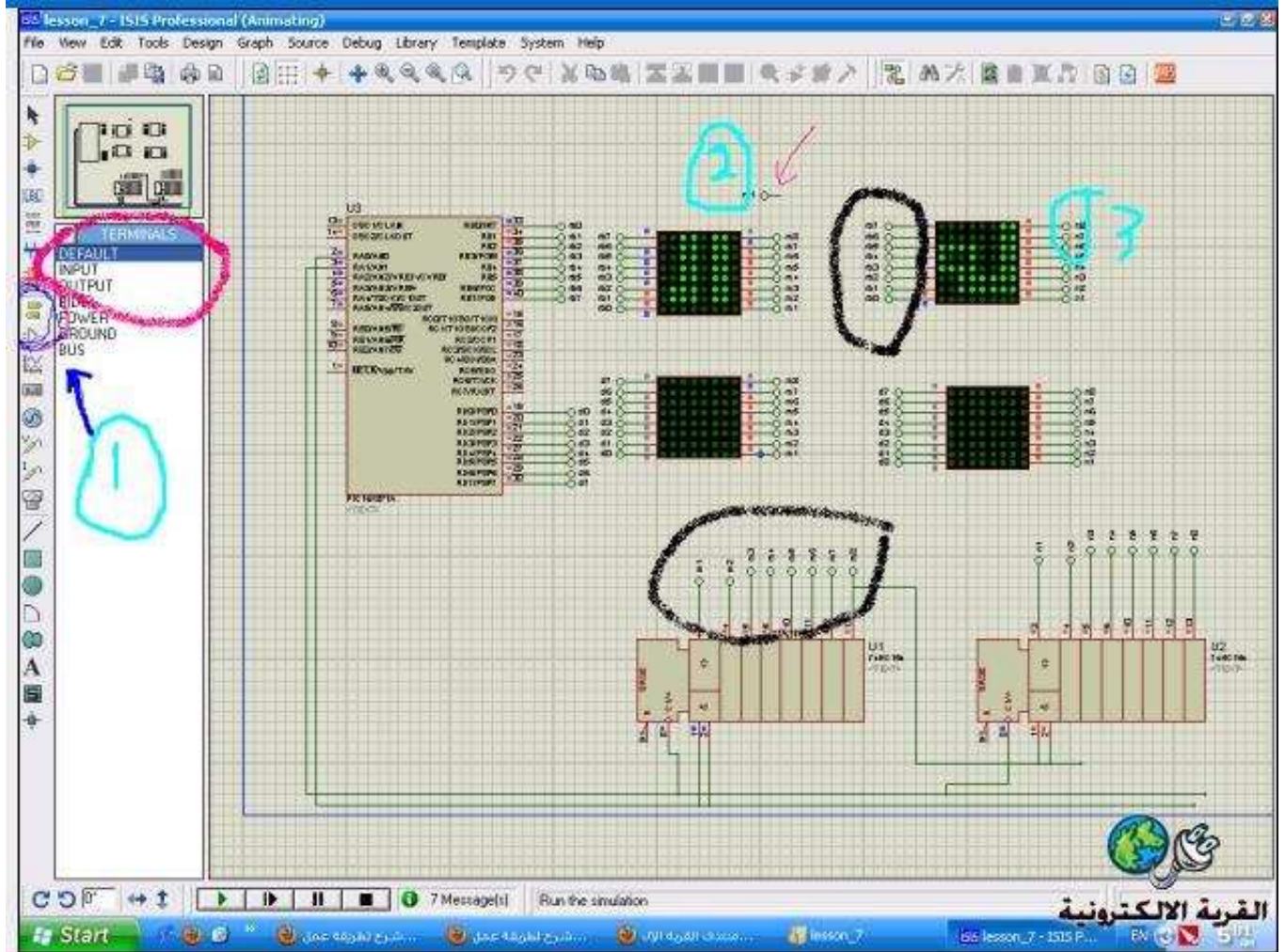




واردت ان اعرض لكم صورة اخرى للتوصيلات بدون خطوط

اعتقد ان الكثيرين يعرفونها

ولكن هذه الصورة لمن لا يعرفونها



وهكذا تكون الشروحات اوضح

والتوصيلات مفهومة اكثر

وسوف نقوم باستخدام نفس الكود السابق لهذا الدرس  
كما تعودنا

اليكم ملفات الدرس السابع

[http://www.herosh.com/download/20952...son\\_7.rar.html](http://www.herosh.com/download/20952...son_7.rar.html)



(((((((( nasser56 فك الضغط)))))))))

واليكم فيديو للمحاكاة للدرس

<http://www.youtube.com/watch?v=7sGMXRSTVzU>  
ملاحظة الفيديو بطيء بسبب برنامج تصوير الشاشة

لكي نتسوعب هذا الدرس

علينا فهم الاساس

كما لاحظتم فى الدروس السابقة

كنت داما اقول :

ان الكود مبنى من  
3 اجزاء

الداتا ( ماتشاهدة العين ويتم عرضة ع الشاشة )

والجزء الاوسط

وهذا هو المتحكم فى الكود

ولو جعلنا الداتا تبدا من اعلى الكود  
فان النص سوف يتحرك من اليمين الى اليسار

واذا بدا من اعلى رقم للداتا فسوف يتحرك من اليسار الى اليمين

؟؟؟؟؟؟؟؟؟؟؟؟؟؟

ممممممممممممم

اعتقد انه هناك البعض قد فهم درس اليوم قبل ان تكمل الشرح

فعلا

اليوم قمنا باستخدام الاتجاهين معا  
من اليمين لليساار ومن اليسار الى اليمين

وقمنا باستخدام جملة واحدة لكى نسهل الامور وتكون اوضح

+++++

بالمناسبة كان من الممكن ان اجعل الاداتا نفسها ونستخدمها فى بورت B  
وهى نفسها نستخدمها فى بورت D

ولكن ذلك لا يعطينا فرصة للتغيير

ولذلك:

قمنا بانشاء داتا جديدة باسم M

واستخدمناها مع البورت D

=====  
=====  
=====

اهم نقطة فى درسنا اليوم هو ملاحظة

الخروج للنص

فى اوسط الكود

كود:

```
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m1;
```

```
portb=d44; delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

سوف تلاحظ

ان بورت B=d44  
وان بورت D=m1

????????????????

هل فهمنا ام لا ؟؟؟؟؟؟؟؟؟

=====

ولم ننسى اضافة الجزء الاسفل من الكود

وهو كما نعرف جميعا للتبديل ((( اوالتحريك للكود )))

واليكم الكود النهائى

كود:

```
char d0=0b00000000;
char d1=0b00000000;
char d2=0b11111110;
char d3=0b11111110;
char d4=0b00000000; ////
char d5=0b11111110;
char d6=0b11111110;
char d7=0b00000110;
char d8=0b00000110;
char d9=0b11111110;
char d10=0b11111110;
char d11=0b00000110;
char d12=0b00000110;
char d13=0b00111110;
char d14=0b00101110;
char d15=0b00101000;
char d16=0b00011000;
char d17=0b00000000; ////
char d18=0b00000000;
char d19=0b00000000;
char d20=0b00000000;
char d21=0b11111111;
char d22=0b11111111;
char d23=0b00000110;
char d24=0b00111000;
char d25=0b01100000;
char d26=0b11111111;
char d27=0b11111111;
```

```
char d28=0b00000000;
char d29=0b00000000;
char d30=0b00010000;
char d31=0b00111000;
char d32=0b01111100;
char d33=0b11111110;
char d34=0b00111000;
char d35=0b00111000; ////
char d36=0b00111000;
char d37=0b00111000;
char d38=0b00111000;
char d39=0b11111110;
char d40=0b11101110;
char d41=0b11000110;
char d42=0b10000010;
char d43=0b10000010;
char d44=0b00000000;
char d45=0b00000000;
char m0=0b00000000;
char m1=0b00000000;
char m2=0b11111110;
char m3=0b11111110;
char m4=0b00000000; ////
char m5=0b11111110;
char m6=0b11111110;
char m7=0b00000110;
char m8=0b00000110;
char m9=0b11111110;
char m10=0b11111110;
char m11=0b00000110;
char m12=0b00000110;
char m13=0b00111110;
char m14=0b00101110;
char m15=0b00101000;
char m16=0b00011000;
char m17=0b00000000; ////
char m18=0b00000000;
char m19=0b00000000;
char m20=0b00000000;
char m21=0b11111111;
char m22=0b11111111;
char m23=0b00000110;
char m24=0b00111000;
char m25=0b01100000;
char m26=0b11111111;
char m27=0b11111111;
char m28=0b00000000;
char m29=0b00000000;
char m30=0b00010000;
char m31=0b00111000;
char m32=0b01111100;
char m33=0b11111110;
char m34=0b00111000;
char m35=0b00111000; ////
char m36=0b00111000;
char m37=0b00111000;
char m38=0b00111000;
char m39=0b11111110;
char m40=0b11101110;
char m41=0b11000110;
char m42=0b10000010;
char m43=0b10000010;
char m44=0b00000000;
char m45=0b00000000;
```

```
////////////////////////////////////
```

```

char x;
void main() {
TRISA=0;
TRISB=0;
portb=0;
porta=0;
TRISd=0;
portd=0;

loop:

for (x=0;x<32;x++) {
porta.f2=1;
////////////////////////////////////
porta.f1=1;  porta.f1=0;

portd=m0;
portb=d45;

delay_us(250);
////
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m1;
portb=d44;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m2;
portb=d43;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m3;
portb=d42;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m4;
portb=d41;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m5;
portb=d40;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m6;
portb=d39;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portd=m7;
portb=d38;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m8;
portb=d37;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portd=m9;
portb=d36;  delay_us(250);

porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portd=m10;
portb=d35;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portd=m11;
portb=d34;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;

portd=m12;
portb=d33;  delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0;  porta.f0=0;
portd=m13;
portb=d32;  delay_us(250);

```

```
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m14;  
portb=d31; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m15;  
portb=d30; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m16;  
portb=d29; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m17;  
portb=d28; delay_us(250);  
////  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m18;  
portb=d27; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m19;  
portb=d26; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m20;  
portb=d25; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m21;
```

```
portb=d24; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m22;  
portb=d23; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m23;  
portb=d22; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m24;  
portb=d21; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m25;  
portb=d20; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m26;  
portb=d19; delay_us(250);
```

```
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m27;  
portb=d18; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m28;  
portb=d17; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m29;  
portb=d16; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;  
portd=m30;  
portb=d15; delay_us(250);  
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m31;
portb=d14; delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m32;
portb=d13; delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
portd=m33;
portb=d12; delay_us(250);
porta.f0=1;porta.f1=1; porta.f1=0; porta.f0=0;
```

```
}
```

```
m0=m1;
m1=m2;
m2=m3;
m3=m4;
m4=m5;
m5=m6;
m6=m7;
m7=m8;
m8=m9;
m9=m10;
m10=m11;
m11=m12;
m12=m13;
m13=m14;
m14=m15;
m15=m16;
m16=m17;
m17=m18;
m18=m19;
m19=m20;
m20=m21;
m21=m22;
m22=m23;
m23=m24;
m24=m25;
m25=m26;
m26=m27;
m27=m28;
m28=m29;
m29=m30;
m30=m31;
m31=m32;
m32=m33;
m33=m34;
m34=m35;
m35=m36;
m36=m37;
m37=m38;
m38=m39;
m39=m40;
m40=m41;
m41=m42;
m42=m43;
m43=m44;
m44=m45;
m45=m0;
```

////////////////////////////////////

```
d0=d1;  
d1=d2;  
d2=d3;  
d3=d4;  
d4=d5;  
d5=d6;  
d6=d7;  
d7=d8;  
d8=d9;  
d9=d10;  
d10=d11;  
d11=d12;  
d12=d13;  
d13=d14;  
d14=d15;  
d15=d16;  
d16=d17;  
d17=d18;  
d18=d19;  
d19=d20;  
d20=d21;  
d21=d22;  
d22=d23;  
d23=d24;  
d24=d25;  
d25=d26;  
d26=d27;  
d27=d28;  
d28=d29;  
d29=d30;  
d30=d31;  
d31=d32;  
d32=d33;  
d33=d34;  
d34=d35;  
d35=d36;  
d36=d37;  
d37=d38;  
d38=d39;  
d39=d40;  
d40=d41;  
d41=d42;  
d42=d43;  
d43=d44;  
d44=d45;  
d45=d0;
```

////////////////////////////////////

```
goto loop;  
}
```

كالمعتاد دائما ننسى رش الماء ~~XXXXXXXXXXXX~~





انتهى بحمد الله

الدرس السابع

الدرس القادمة ان شاء الله سوف نعود للخلف قليلا

لكى نحاول ان نتعلم سويا اختصارات الاكواد

((لانى اجد الحاج كبير فى ذلك من ناحية المشاركين))

تنوية هام قد نتاخر قليلا فى الدروس القادمة  
لانى اعانى كثير من المشاكل بالجهاز

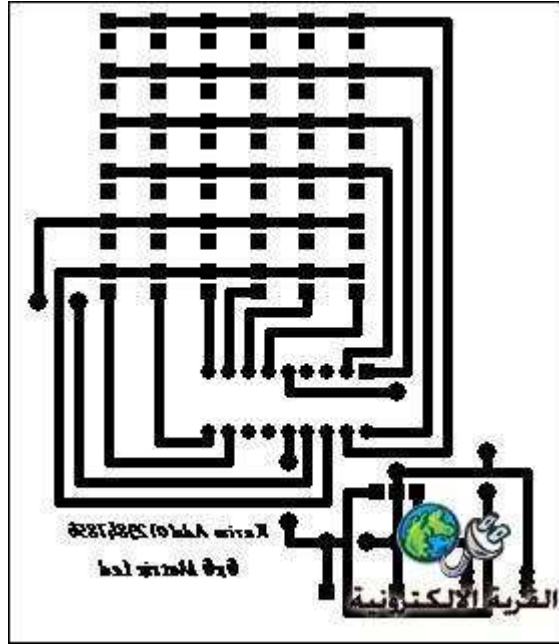


السلام عليكم

طبعاً شكر خالص جداً لاستاذى ناصر على المجهود الذى بذله معنا فى دروسه وشرحته و اقول بكل فخر ان هذا هو  
ثمره تعبك معنا ويارب دائماً تستمر فى طريقك الى الامام.

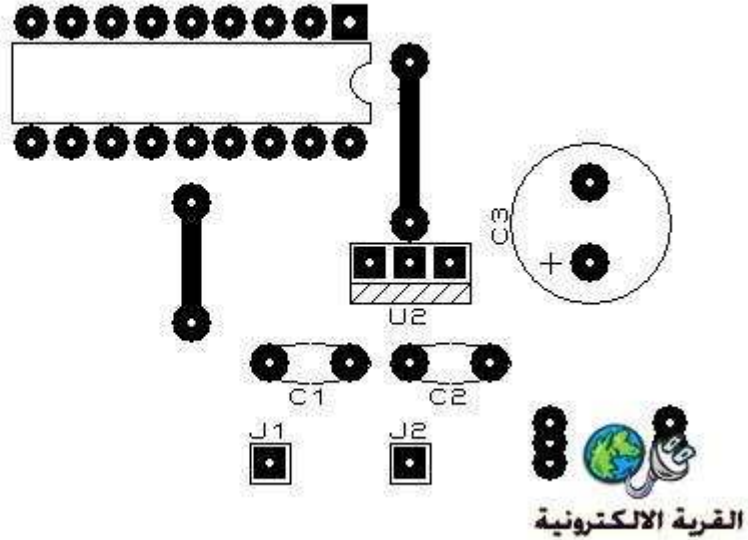
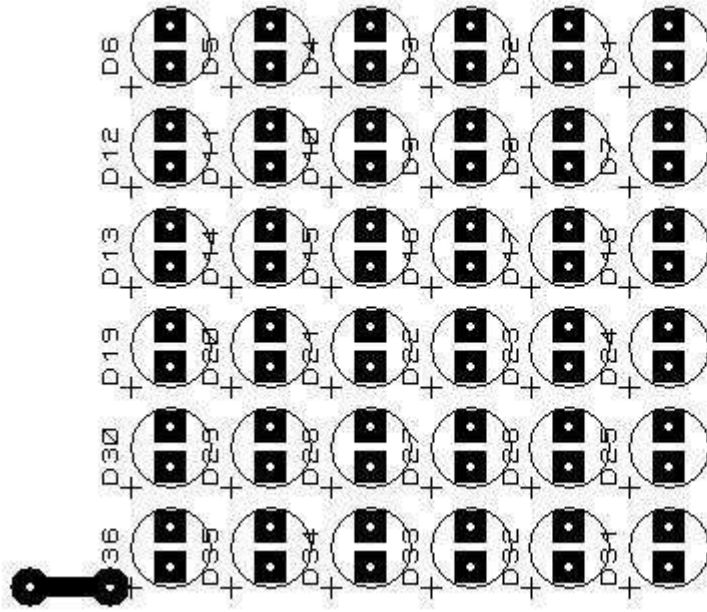
اولاً :

هذه الدائرة المطبوعة لشاشة **Matrix led 6x6** وبأستخدام ( **Pic 16F628A** ) واسباب استخدامى لهذه المتكمله  
سوف اشرحها فى اخر الرد .



وهذا رابط تحميل ملف PDF للدائرة [أضغط هنا](#).

وهذا توضيح للعناصر وطريقة التوصيل .



### العناصر هي :

- عدد 1 IC **PIC16F628A** السعر 12 جم .
- عدد 1 IC **7805** السعر 1 جم .
- عدد **36** ليد أى لون السعر الواحد 0.25 الكل 9 جم .
- عدد 1 مكثف **uf740** السعر 0.50 جم .
- عدد 2 مكثف ثابت **n100** السعر 0.50 جم .
- عدد 1 جاك Adaptor السعر 0.50 جم . أختياري

### ملاحظة :-

يتم توصيل اطراف الانود للموحدات الضوئية عن طريق سيلك او بتنى الاطراف وتشبيكها مع بعض .

وهذا الفيديو يوضح الدائرة وهى تعمل وتعرض كلمة ( الله أكبر )

<http://www.youtube.com/watch?v=Nf5XP8BudVs>

واعذرونى اذا كانت التصوير ردى

وهذا هم البرنامج

```
char x1=0b00000000;
char x2=0b00000000;
char x3=0b00000000;
char x4=0b00000000;
char x5=0b00000000;
char x6=0b00000000;
char x7=0b00000000;
char x8=0b00111111;
char x9=0b00000000;
char x10=0b00111111;
char x11=0b00000001;
char x12=0b00111111;
char x13=0b00000001;
char x14=0b00011111;
char x15=0b00010100;
char x16=0b00001000;
char x17=0b00000000;
char x18=0b00111111;
char x19=0b00000000;
char x20=0b00100000;
char x21=0b00100000;
char x22=0b00111100;
char x23=0b00000100;
char x24=0b00001101;
char x25=0b00000100;
char x26=0b00001100;
char x27=0b00000010;
char x28=0b00000001;
char x29=0b00000001;
char x30=0b00000000;
char x31=0b00000000;
char x32=0b00000000;
char x33=0b00000000;
char x34=0b00000000;
char x35=0b00000000;

char n;

void main()
{
TRISB=0;
PORTB=0;

TRISa=0;
PORTa=0;

loop:

for(n=0;n<100;n++){

PORTa=0B11001110;
PORTB=x1; delay_us(300);

PORTa=0B11001101;
PORTB=x2; delay_us(300);

PORTa=0B11001011;
PORTB=x3; delay_us(300);

PORTa=0B11000111;
PORTB=x4; delay_us(300);

PORTa=0B10001111;
PORTB=x5; delay_us(300);
```

```
PORTa=0B01001111;  
PORTB=x6; delay_us(300);
```

```
}
```

```
x1=x2;  
x2=x3;  
x3=x4;  
x4=x5;  
x5=x6;  
x6=x7;  
x7=x8;  
x8=x9;  
x9=x10;  
x10=x11;  
x11=x12;  
x12=x13;  
x13=x14;  
x14=x15;  
x15=x16;  
x16=x17;  
x17=x18;  
x18=x19;  
x19=x20;  
x20=x21;  
x21=x22;  
x22=x23;  
x23=x24;  
x24=x25;  
x25=x26;  
x26=x27;  
x27=x28;  
x28=x29;  
x29=x30;  
x30=x31;  
x31=x32;  
x32=x33;  
x33=x34;  
x34=x35;  
x35=x1;
```

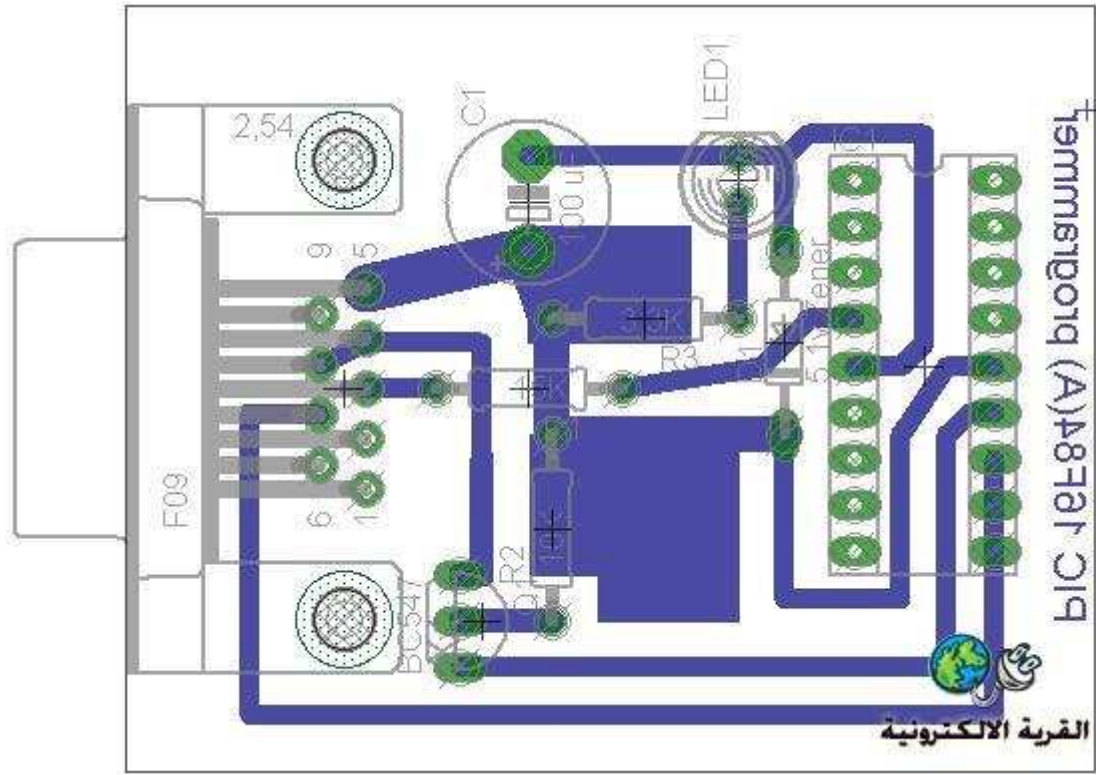
```
goto loop;  
}
```

**أما لاستخدامى هذا الـ ( Pic ) فقل اسباب الاتية :-**

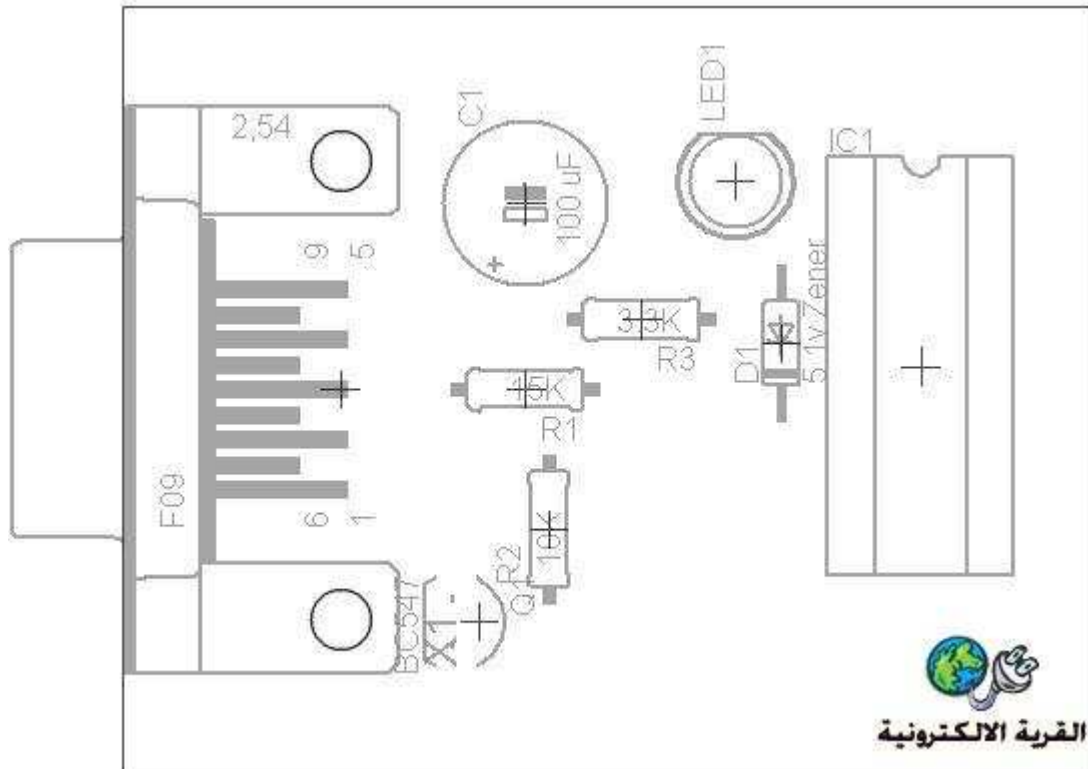
- 1- يوجد به مذبذب داخلى بقوة 4MHz.
- 2- يوجد به 16 طرف خرج Port A = 8 ، Port B = 8 على عكس مثيلة الثانى ( PIC 16F84 ) .
- 3- حجم الذاكرة 3 أضعاف مثيله الآخر .
- 4- حجم الرام 3 أضعاف مثيلة الآخر .
- 5- يوجد به مسجلات وقربان اكثر من الآخر .
- 6- لتفعيل المذبذب الداخلى وتشغيل الأطراف يجب اختيار هذا المذبذب قبل بدأ كتابة البرنامج ( INTRC\_OSC\_NOCLKOUT ) وأيقاف عمل MCLR.

**أما بالنسبة للمبرمجة فقد استخدمت هذه المبرمجة وهى JDM Programmer وتعمل معى بكفاءة عالية جداً جداً وان تكلفتها بسيطة لا تتعدى الـ 5 جنيهات**

**وهذه صورة للمبرمجة**



ولتحميل الدائرة المطبوعة [أضغط هنا](#) .  
وهذا توضيح للمكونات بأعلى الدائرة :



أما بالنسبة للبرنامج فهو برنامج PICPgm Programmer الذى وضعه الاستاذ الكبير أحمد سمير  
لتحميل البرنامج [أضغط هنا](#) .

وإذا واجهتكم أى مشكلة ضع سؤالك فى رد وانشاء الله سوف اجيب على جميع الاسئلة .