برمجة المتحكمات المصغرة

التجارب العملية

الجلسة الثالثة

Programming

Embedded Systems Microcontroller

You Can Practice Microcontroller Programming Easily Now!

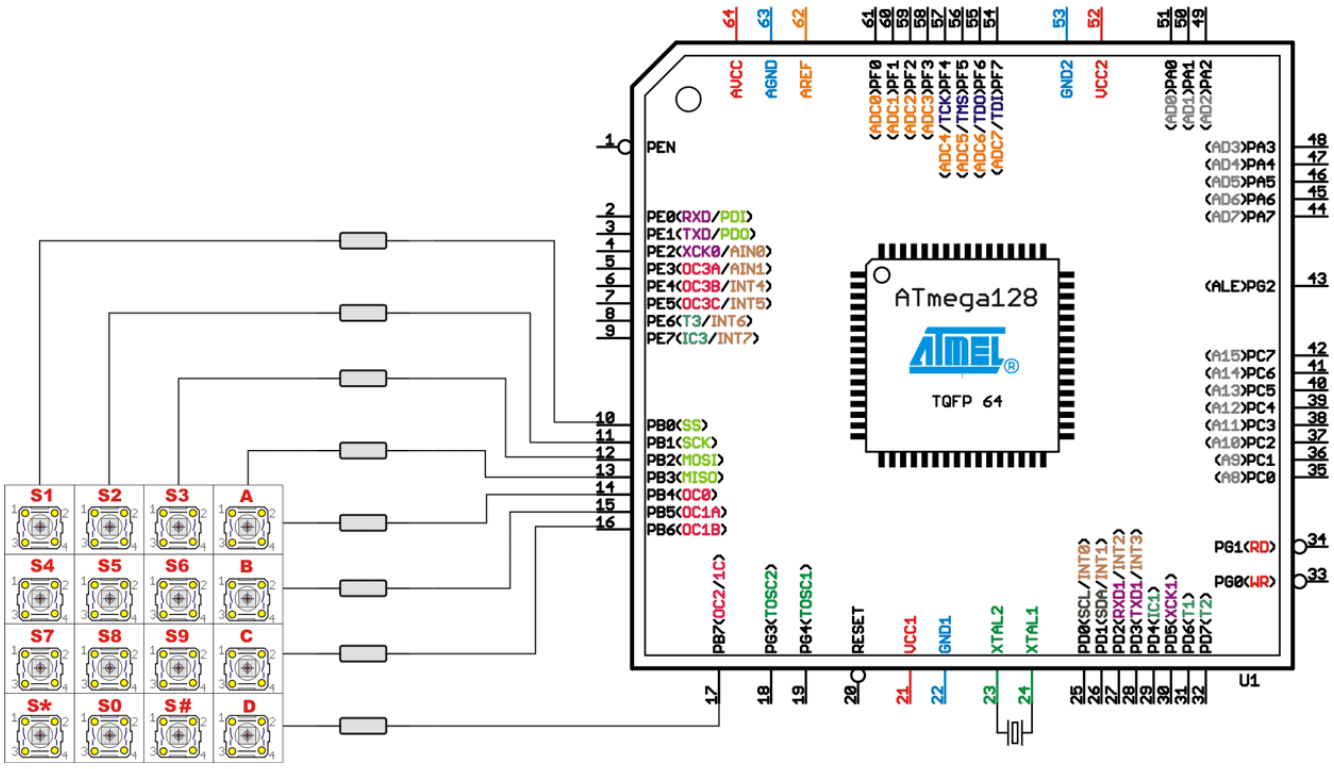
WALID BALID, Tuesday, December 15, 2009

التجربة الخامسة: لوحة مفاتيح ست عشرية مصفوفية
Exp.5: Hexadecimal Array Keypad

الغاية من التجربة:

توصيل وبرمجة لوحة مفاتيح ست عشرية.

مخطط التوصيل:



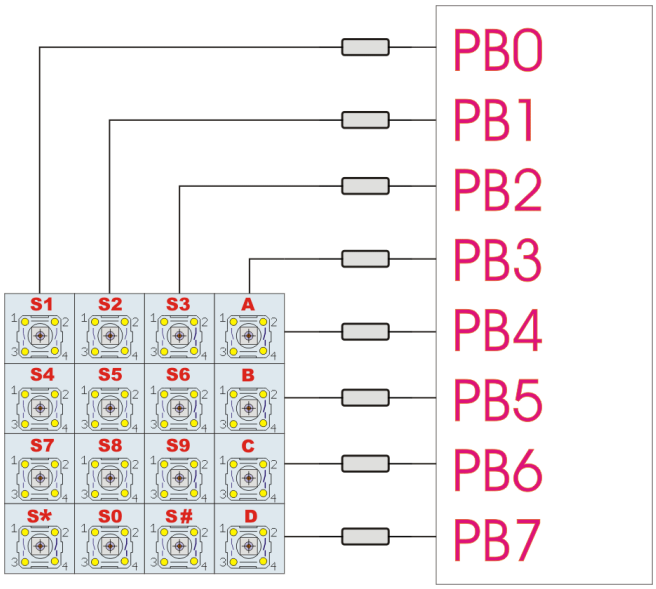
شرح عمل الدارة:

سوف نقوم بكتابة برنامج لقراءة حالة المفاتيح ومعرفة المفتاح المضغوط وطباعة اسم المفتاح المضغوط على النافذة التسلسلية.

التعليمات الجديدة:

من أجل قراءة لوحة مفاتيح ست عشرية فإننا نحتاج إلى تعليميتين أساسيتين:

شرح التعليمة	التعليمة البرمجية
تعريف البوابة الموصول معها لوحة المفاتيح وتعريف زمن التأخير لتفادي أثر العطالة الميكانيكية للمفاتيح.	`Config Kbd = Portb , Delay = 150`
قراءة حالة المفاتيح حيث أنه: 1- ستعود هذه التعليمة بالقيمة 16 = Var إذا لم يكن هناك أي مفتاح مضغوط. 2- ستعود هذه التعليمة بقيمة المفتاح المضغوط على الشكل التالي:	`Var = Getkbd()`



S1 = 0
S2 = 1
S3 = 2
S4 = 4
S5 = 5
S6 = 6
S7 = 8
S8 = 9
S9 = 10
S* = 12
S0 = 13
S# = 14
A = 3
B = 7
C = 11
D = 15

برنامج تشغيل الدارة:

```
$regfile = "m128def.dat"
$crystal = 1000000
$baud = 4800
'-----------------------
Config Kbd = Portb , Delay = 150
'-----------------------
Dim Var As Byte
'-----------------------
Do
   Var = Getkbd()
      If Var < 16 Then
         Gosub Check_number
      End If
Loop

End
'-----------------------
Check_number:
Select Case Var
   Case 0 : Print "Key Pressed is (1)"
   Case 1 : Print "Key Pressed is (2)"
   Case 2 : Print "Key Pressed is (3)"
   Case 3 : Print "Key Pressed is (A)"
   Case 4 : Print "Key Pressed is (4)"
   Case 5 : Print "Key Pressed is (5)"
   Case 6 : Print "Key Pressed is (6)"
   Case 7 : Print "Key Pressed is (B)"
   Case 8 : Print "Key Pressed is (7)"
   Case 9 : Print "Key Pressed is (8)"
   Case 10 : Print "Key Pressed is (9)"
   Case 11 : Print "Key Pressed is (C)"
   Case 12 : Print "Key Pressed is (*)"
   Case 13 : Print "Key Pressed is (0)"
   Case 14 : Print "Key Pressed is (#)"
   Case 15 : Print "Key Pressed is (D)"
End Select
Return
'-----------------------
```

التوجيهات.

تعريف البوابة الموصلة معها لوحة المفاتيح.

تعريف المتحولات

حلقة البرنامج الرئيسي يتم فيها قراءة حالة المفاتيح.

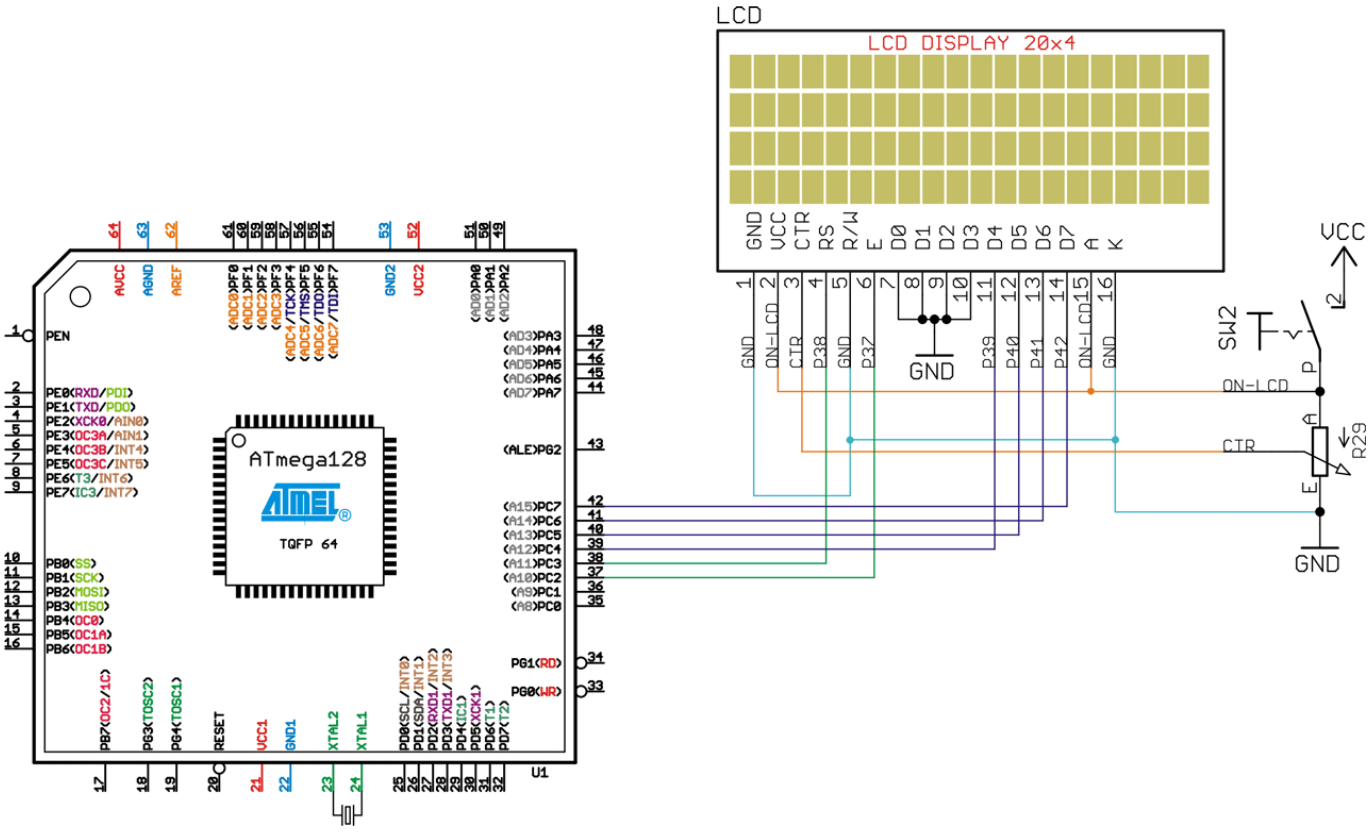
البرنامج لفرعي طباعة تعريف بالمفتاح المضغوط ثم العودة إلى البرنامج الرئيسي.

التجربة السادسة: برمجة شاشة الإظهار الكريستالية
Exp.6: Programming LCD

الغاية من التجربة:

توصيل وبرمجة شاشة إظهار محرفية 20x4.

مخطط التوصيل:



متطلبات توصيل:

يجب إغلاق المفتاح SW2.

شرح عمل الدارة:

سوف نقوم بكتابة برنامج لإظهار محارف نصية على شاشة إظهار كريستالية بالإضافة إلى استعراض الحركات الخاصة بالشاشة الكريستالية.

التعليمات الجديدة:

شرح التعليمة	التعليمة البرمجية
تعريف أقطاب شاشة الإظهار الموصلة مع المعالج	`Config Lcdpin = Pin , Db4 = Portc.4 , Db5 = Portc.5 , Db6 = Portc.6 Db7 = Portc.7 , E = Portc.2 , Rs = Portc.3`
تعريف أبعاد شاشة الإظهار (أعمدة ❖ أسطر)	`Config Lcd = 20 * 4`
تعريف محرف إضافي باستخدام الأداة LCD Designer	`Deflcdchar 0 , 14 , 17 , 14 , 17 , 17 , 14 , 17 , 14`
مسح شاشة الإظهار	`Cls`

Lcd var	طباعة قيمة/محرف على شاشة الإظهار
Lowerline	تحريك مؤشر الكتابة إلى السطر التالي
Shiftlcd Right	إزاحة جميع المعلومات المظهرة على الشاشة عموداً إلى اليمين
Shiftlcd Left	إزاحة جميع المعلومات المظهرة على الشاشة عموداً إلى اليسار
Locate X , Y	توضع مؤشر الكتابة في موقع (سطر/عمود)
Shiftcursor Right	إزاحة مؤشر الكتابة عموداً واحداً إلى اليمين
Shiftcursor Left	إزاحة مؤشر الكتابة عموداً واحداً إلى اليسار
Home Upper	الانتقال إلى السطر/العمود الأول (نقطة البداية)
Cursor Off Noblink	إلغاء مؤشر الكتابة.
Cursor On Blink	تشغيل مؤشر الكتابة مع خفقان
Display Off	إطفاء شاشة الإظهار
Display On	تشغيل شاشة الإظهار
Thirdline	الانتقال إلى السطر الثالث
Fourthline	الانتقال إلى السطر الرابع
Home Third	وضع مؤشر الكتابة في بداية السطر الثالث
Home Fourth	وضع مؤشر الكتابة في بداية السطر الرابع
Lcd Chr(x)	طباعة المحرف الإضافي

برنامج تشغيل الدارة:

```
$regfile = "m128def.dat"
$crystal = 8000000
'-----------------------
Config Lcdpin = Pin , Db4 = Portc.4 , Db5 = Portc.5 ,
Db6 = Portc.6 , Db7 = Portc.7 , E = Portc.2 , Rs = Portc.3

Config Lcd = 20 * 4

Deflcdchar 0 , 14 , 17 , 14 , 17 , 17 , 14 , 17 , 14
Deflcdchar 1 , 17 , 14 , 17 , 14 , 14 , 17 , 14 , 17
'-----------------------

Dim A As Byte
'-----------------------
Do
   Cls
   Lcd "hello world!"
   Wait 1

   Lowerline
   Wait 1

   Lcd "LCD Test"
   Wait 1

   For A = 1 To 8
      Shiftlcd Right
```

التوجيهات.

تعريف أقطاب شاشة الإظهار الموصلة مع المتحكم.

تعريف موديل شاشة الظهار

تعريف محارف إضافية باستخدام الأداة LCD Designer

تعريف المتحولات

حلقة البرنامج الرئيسي يتم فيها الإظهار على الشاشة وعرض الحركات لأساسية وإظهار المحارف الإضافية.

```
      Waitms 500
   Next A

   For A = 1 To 8
      Shiftlcd Left
      Waitms 500
   Next A

   Locate 2 , 9
   Lcd "-L2,C9"
   Wait 1

   Shiftcursor Right
   Lcd "-*-"
   Wait 1

   Shiftcursor Left
   Lcd "*-*"
   Wait 1

   Home Upper
   Lcd "HELLO WORLD"
   Wait 1

   Cursor Off Noblink
   Wait 1

   Cursor On Blink
   Wait 1

   Display Off
   Wait 1

   Display On
   Wait 1

   Thirdline
   Lcd "   /Line 3"
   Wait 1

   Fourthline
   Lcd "   /Line 4"
   Wait 1

   Home Third
   Lcd ":)"
   Wait 1

   Home Fourth
   Lcd ":("
   Wait 1

   Cls
   Locate 1 , 9 : Lcd Chr(0)
   Wait 1
   Locate 1 , 11 : Lcd Chr(1)
Loop

End
```

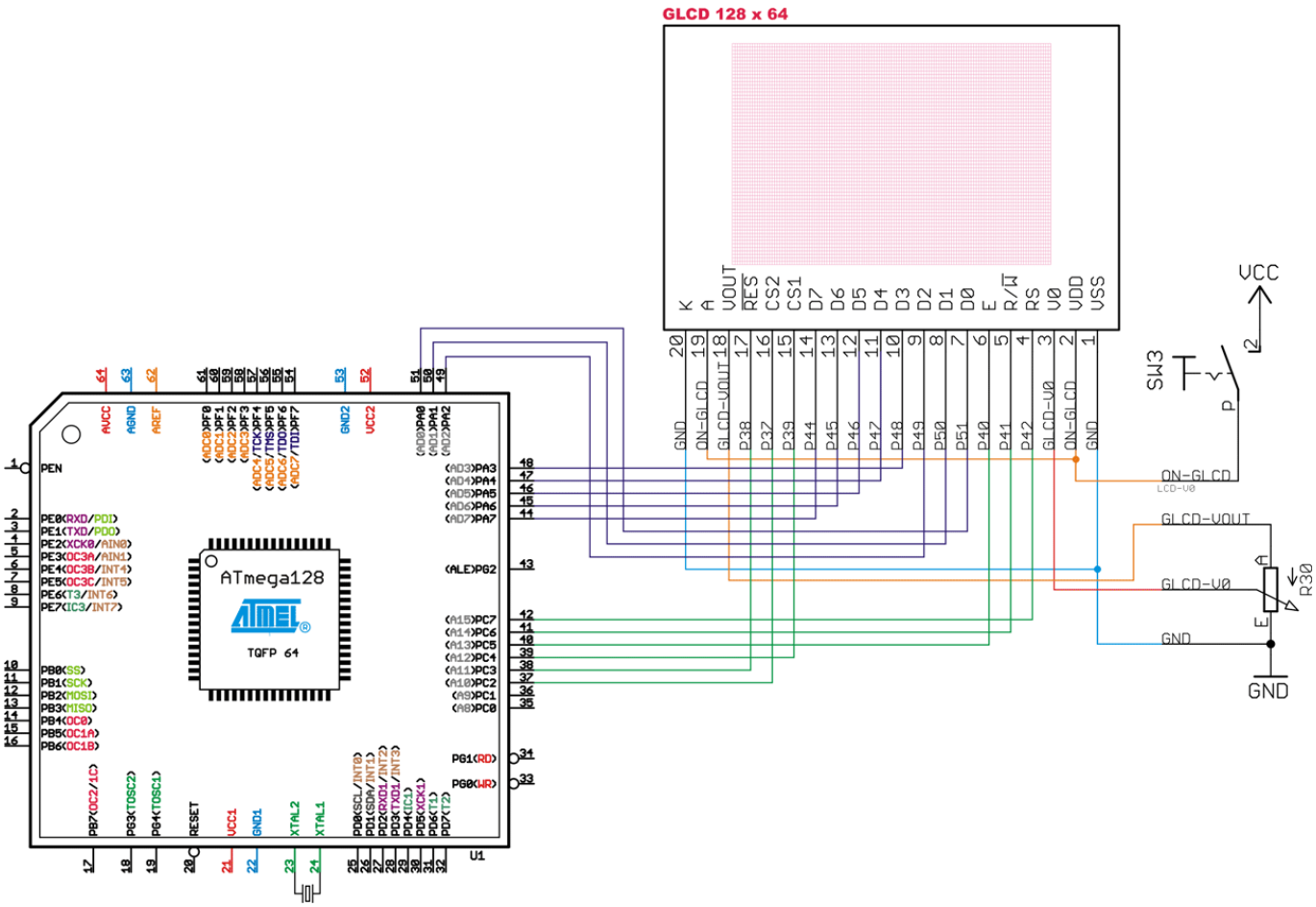
Exp.7: Programming Graphic LCD

التجربة السابعة: برمجة شاشة الإظهار الرسومية

الغاية من التجربة:

توصيل وبرمجة شاشة إظهار رسومية 128x64.

مخطط التوصيل:



متطلبات توصيل:

يجب إغلاق المفتاح SW3.

شرح عمل الدارة:

سوف نقوم بكتابة برنامج لإظهار محارف نصية وصور وإنشاءات رسومية على شاشة إظهار رسومية بالإضافة إلى استعراض الحركات الخاصة بالشاشة الرسومية.

التعليمات الجديدة:

شرح التعليمة	التعليمة البرمجية
تعريف الأقطاب الموصلة مع شاشة الإظهار الرسومية.	`Config Graphlcd = 128 * 64sed , Dataport= Porta, Controlport= Portc , Ce = 2 , Ce2 = 4 , Cd = 7 , Rd = 6 , Reset = 3 , Enable = 5`
تعريف مكتبة شاشة الإظهار الرسومية	`$lib "glcdKS108.lib"`

تعيين الخط المراد استخدامه	`Setfont Font`
مسح النصوص والرسوم	`Cls`
مسح النصوص فقط	`Cls Text`
مسح الرسوم فقط	`Cls Graph`
طباعة متحول أو نص على الشاشة في موقع محدد.	`Lcdat y , x , var [, inv]`
رسم خط استناداً إلى نقطة بداية ونقطة نهاية	`Line(X0 , Y0) - (X1 , Y1) , Color`
التحكم بإظهار أو إخفاء نقطة (SET/RST Pixel)	`Pset X , Y , Color`
رسم دائرة في نقطة مرجعية وبقطر r	`Circle(X , Y) , r , Color`
إظهار صورة في موقع محدد موجودة عند لافتة محددة	`Showpic X , Y , label`
تضمين ملف الخطوط من ملف خطوط خارجي	`$include "Font5x8.font"`
تضمين الصورة من ملف bgf خارجي	`$bgf "Smiley1.bgf"`

برنامج تشغيل الدارة:

التوجيهات.

```
$regfile = "m128def.dat"
$crystal = 8000000
$lib "glcdKS108.lib"
```

تعريف البوابة الموصلة معها شاشة الإظهار الرسومية.

```
'-------------------------
Config Graphlcd = 128 * 64sed , Dataport
= Porta , Controlport = Portc , Ce = 2 ,
Ce2 = 4 , Cd = 7 , Rd = 6 , Reset = 3 ,
Enable = 5
```

تعريف المتحولات

```
'-------------------------
Dim X As Byte , Y As Byte
```

مسح شاشة الإظهار

```
'-------------------------
Cls : Cls Text : Cls Graph
```

طباعة نصوص باستخدام خط من الحجم 8x8

```
'-------------------------
Setfont Font8x8
   Lcdat 3 , 11 , "Font 5x8 Test"
   Lcdat 4 , 11 , "Font 6x8 Test"
   Lcdat 5 , 11 , "Font 8x8 Test"
   Wait 2
```

طباعة نصوص باستخدام خط من الحجم 5x8

```
   Cls
   Setfont Font5x8
      Lcdat 1 , 1 , "LINE1" : Wait 1
      Lcdat 2 , 1 , "LINE2" : Wait 1
      Lcdat 3 , 1 , "LINE3" : Wait 1
```

طباعة نصوص باستخدام خط من الحجم 6x8

```
   Setfont Font6x8
      Lcdat 4 , 1 , "LINE4" : Wait 1
      Lcdat 5 , 1 , "LINE5" : Wait 1
      Lcdat 6 , 1 , "LINE6" : Wait 1
```

طباعة نصوص باستخدام خط من الحجم 8x8

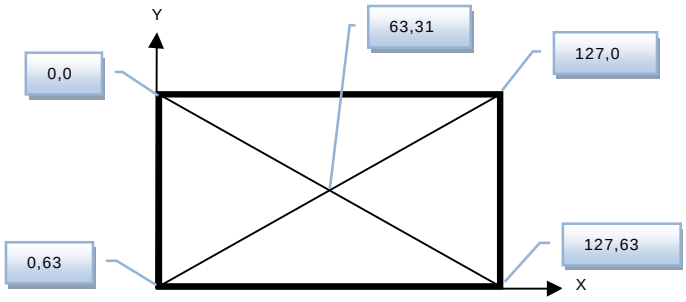
```
   Setfont Font8x8
      Lcdat 7 , 1 , "LINE7" : Wait 1
      Lcdat 8 , 1 , "LINE8" : Wait 1
'/------------/
```

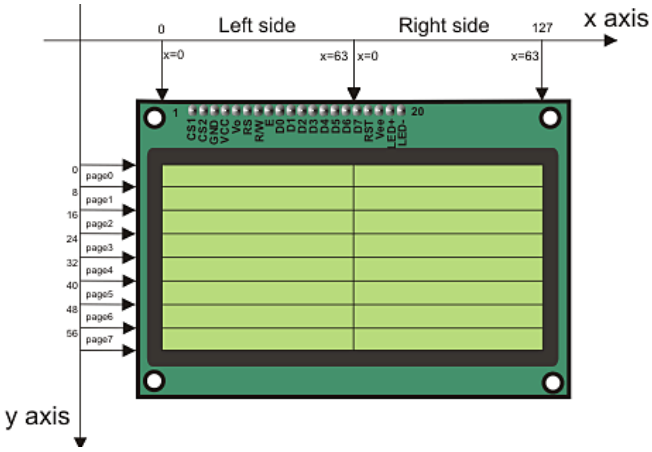
طباعة نصوص باستخدام خط من الحجم 16x16

```
Cls
Setfont Font16x16
```

```
   Lcdat 4 , 23 , "16x16" : Wait 2
   Cls
      Lcdat 1 , 1 , "LINE1" : Wait 1
      Lcdat 3 , 1 , "LINE2" : Wait 1
      Lcdat 5 , 1 , "LINE3" : Wait 1
      Lcdat 7 , 1 , "LINE4" : Wait 1
'/------------/
Setfont Font8x8
   Lcdat 4 , 11 , "Drawing Lines" :
Wait 2
   Cls Text
      Line(0 , 0) -(127 , 0) , 255
      Wait 1
      Line(0 , 63) -(127 , 63) , 255
      Wait 1
      Line(0 , 0) -(0 , 63) , 255
      Wait 1
      Line(127 , 0) -(127 , 63) , 255
      Wait 1
      Line(0 , 0) -(127 , 63) , 255
      Wait 1
      Cls Graph
'/------------/
   Lcdat 4 , 11 , "SET/RST Pixel" :
Wait 2
   Cls Text
      For X = 0 To 127
         Pset X , 20 , 255
         Pset X , 43 , 255
         Waitms 200
      Next X

      For Y = 0 To 63
         Pset 42 , Y , 255
         Pset 86 , Y , 255
         Waitms 200
      Next Y

      For X = 127 To 0 Step -1
         Pset X , 20 , 0
         Pset X , 43 , 0
         Waitms 200
      Next X

      For Y = 63 To 0 Step -1
         Pset 42 , Y , 0
         Pset 86 , Y , 0
         Waitms 200
      Next X
      Cls Graph
'/------------/
   Lcdat 4 , 11 , "Drawing Circle" :
Wait 2
   Cls Graph
   For X = 1 To 31
      Circle(63 , 31) , X , 255
      Waitms 200
      Circle(63 , 31) , X , 0
   Next X
```

رسم خطوط وفق التقسيم التالي لشاشة الإظهار





رسم خطوط نقطة بنقطة

ومن ثم مسحها نقطة بنقطة

رسم دائرة

```
   Cls Graph
'/------------/
   Showpic 0 , 0 , Smiley1
   Wait 2
   Cls : Cls Text : Cls Graph

   Showpic 0 , 0 , Smiley2
   Wait 2
   Cls Graph

   Showpic 0 , 0 , Smiley3
   Wait 2
   Cls Graph

   Showpic 0 , 0 , Smiley4
   Wait 2
'/------------/
  Glcdcmd &H3E , 1 : Glcdcmd &H3E , 2
  Wait 1
  Glcdcmd &H3F , 1 : Glcdcmd &H3F , 2
'/------------/
End
'--------------------
$include "Font5x8.font"
$include "Font6x8.font"
$include "Font8x8.font"
$include "Font16x16.font"
'--------------------
Smiley1:
$bgf "Smiley1.bgf"

Smiley2:
$bgf "Smiley2.bgf"

Smiley3:
$bgf "Smiley3.bgf"

Smiley4:
$bgf "Smiley4.bgf"
```

إظهار الصور المضمنة من ملفات خارجية

- إطفاء قسمي شاشة الإظهار الرسومية اليميني واليساري
- إعادة تشغيل القسمين

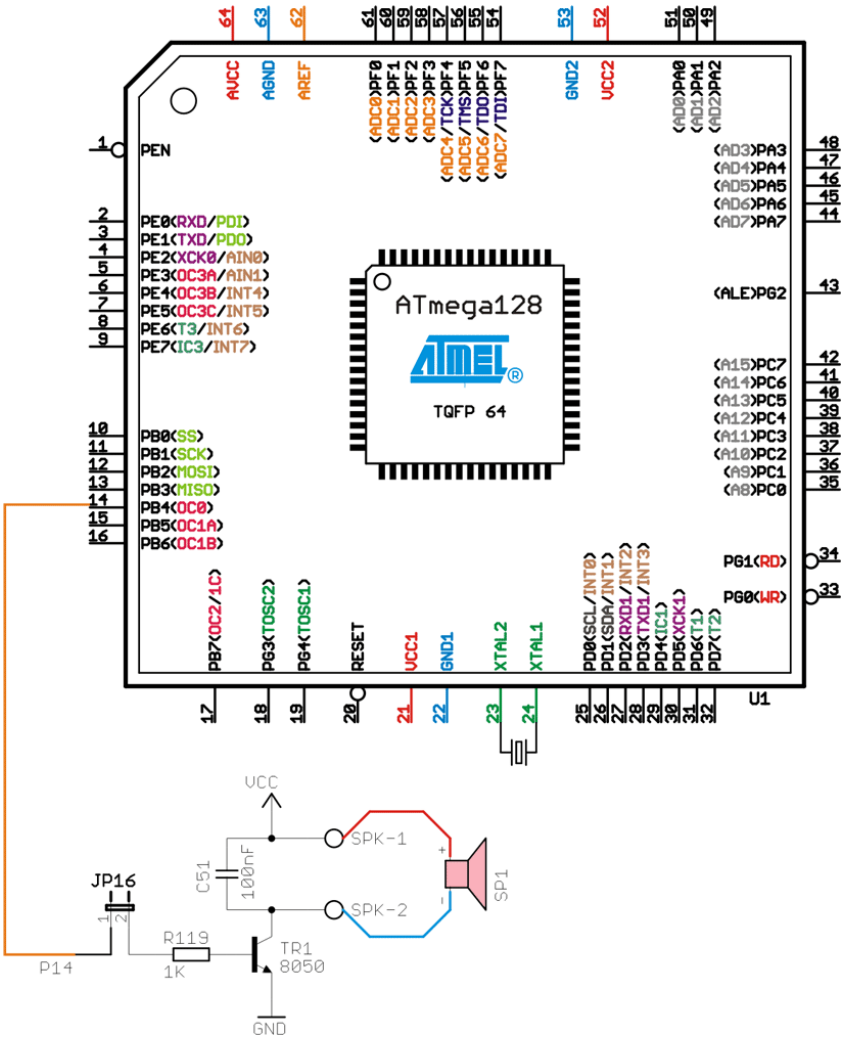
تعليمة تضمين ملفات النصوص من ملفات برمجية خارجية

تضمين ملفات الصور البرمجية من ملفات خارجية

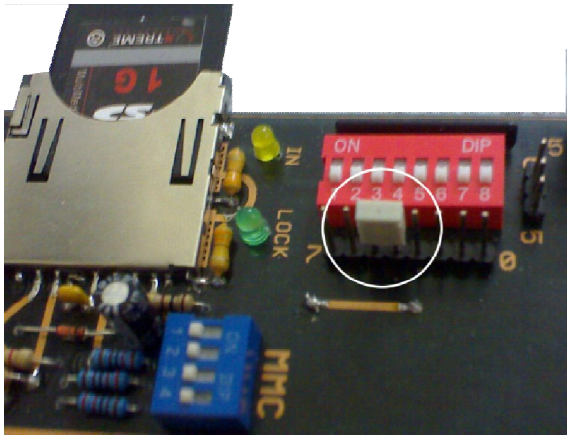
الغاية من التجربة:

توليد نغمات صوتية لأغراض التنبيه الصوتي.

مخطط التوصيل:



متطلبات توصيل:



يجب إغلاق نقطة الوصل JP16 باستخدام Jumper. ووضع نقطة وصل بين PB4,PB5 كما في الصورة.

شرح عمل الدارة:

سوف نقوم بكتابة برنامج لتوليد نغمات تنبيه صوتية تظهر على مجهار صوتي خارجي. يمكن تشكيل أي نغمة من خلال دمج عدة تعليمات sound التي تعتمد على توليد نبضة وتغيير عرض ودور النبضة وبالتالي يتغير التردد على قاعدة المفتاح الترانزستوري الذي يتحكم بالمجهار الخارجي (Speaker)

التعليمات الجديدة:

شرح التعليمة	التعليمة البرمجية
توليد صوت على القطب (Speaker) وبعرض نبضة محدد بالمتحول (Pulses) وزمن النبضة محدد بالمتحول (Periods).	`Sound Speaker , Pulses , Periods`

برنامج تشغيل الدارة:

التوجيهات.	`$regfile = "m128def.dat"` `$crystal = 8000000`

تعريف القطب الموصل في خرجه المكبر الصوتي	`'----------------------`
	`Config Portb.4 = Output`
	`Speaker Alias Portb.4`

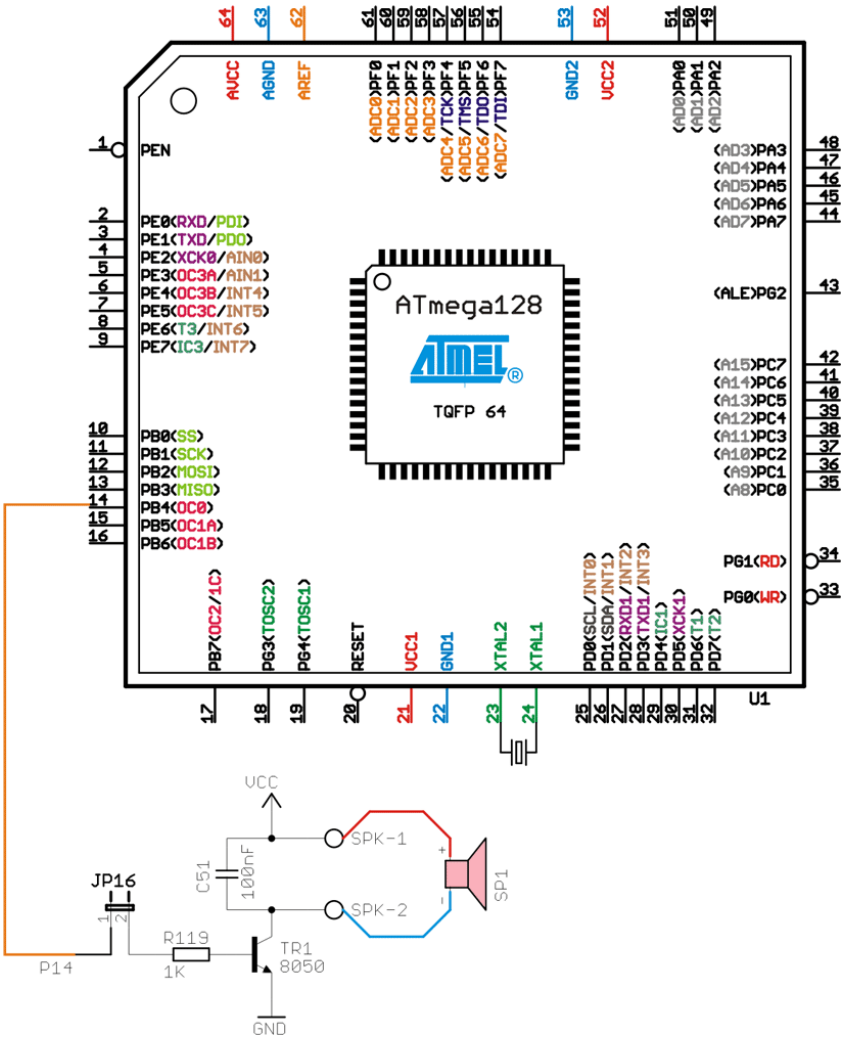
تعريف المتحولات وإسناد القيم الابتدائية لها	`'----------------------`
	`Dim Pulses As Word , Periods As Word`
	`Dim I As Byte`
	
	`Pulses = 50 : Periods = 100`

حلقة البرنامج الرئيسي يتم فيها توليد أصوات نغمات مختلفة.	`'----------------------`
	`Sound Speaker , 100 , 200 : Wait 1`
	
	`Do`
	`  Sound Speaker , Pulses , Periods`
	`  Waitms 1`
	`  Periods = Periods + 5`
	`Loop Until Periods = 1000`
	
	`For I = 0 To 6`
	`  Sound Speaker , 150 , 800`
	`  Waitms 1000`
	`Next I`
	
	`Sound Speaker , 100 , 200`
	`End`

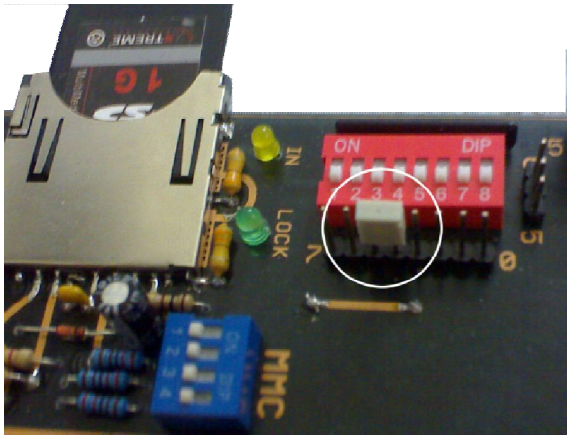
الغاية من التجربة:

توليد نغمات صوتية DTMF (Dual Tone Multi Frequency) لأغراض التنبيه الصوتي.

مخطط التوصيل:



متطلبات توصيل:



يجب إغلاق نقطة الوصل JP16 باستخدام Jumper. ووضع نقطة وصل بين PB4,PB5 كما في الصورة.

شرح عمل الدارة:

سوف نقوم بكتابة برنامج لتوليد نغمات تنبيه DTMF تظهر على مجهار صوتي خارجي موصل مع القطب OC1A.

التعليمات الجديدة:

شرح التعليمة	التعليمة البرمجية
توليد صوت على القطب (OC1A). **Number:** أحد أرقام لوحة مفاتيح الهاتف الستة عشر. **Duration**: زمن توليد النغمة بالميلي ثانية. **String**: مجموعة الأرقام التي تشكل رقم الاتصال كاملاً.	`Dtmfout Number , Duration` `Dtmfout String , Duration`

برنامج تشغيل الدارة:

التوجيهات.	`$regfile = "m128def.dat"` `$crystal = 8000000`
تفعيل المقاطعات لأن التعليمة DTMF تعتمد على المؤقت1 في عملها	`'-------------------` `Enable Interrupts`

تعريف المتحولات وإسناد القيم الابتدائية لها	`'-------------------`
	`Dim Var As Byte`
	`Dim Btmp As Byte , Sdtmf As String * 10`
	`Sdtmf = "1234567890"`

حلقة البرنامج الرئيسي يتم فيها توليد أصوات نغمات مختلفة.	`'-------------------`
	`Do`
	`  Dtmfout Sdtmf , 50`
	`  Waitms 1000`
	`  For Btmp = 0 To 15`
	`  Dtmfout Btmp , 100`
	`  Waitms 500`
	`  Next Btmp`
	`Loop`

ملاحظة1: إن التعليمة **Dtmfout** تعتمد على المؤقت1 في توليد النغمات لذا يجب تفعيل شعاع المقاطعة كما أنها ستقوم بإصدار النغمات على القطب OC1A فقط.

ملاحظة2: إن تردد الهزاز الكريستالي يجب أن يكون بين 4MHZ ~ 8MHZ.

ملاحظة3: الرجاء مراجعة الوثيقة الفنية Atmel application note (AN314) لفهم العلاقات الرياضية لهذه التعليمة.