

# كيف تبني منظومة سكاذا

بواسطة المايكروكنترولر و الـ I/O

Gauge



Gauge 2



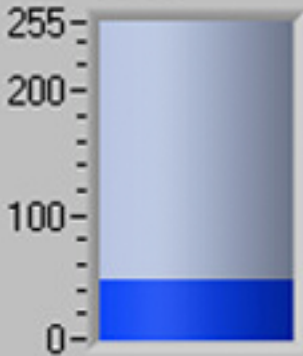
Gauge 3



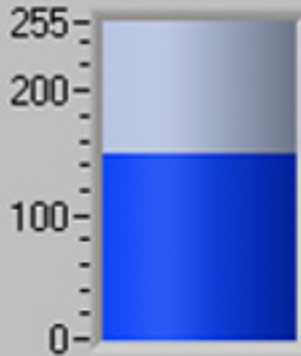
Gauge 4



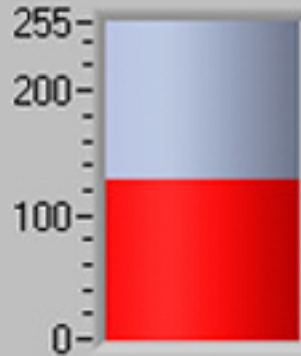
Tank



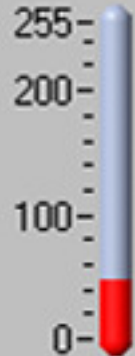
Tank 2



Tank 3



Thermometer 1



# جدول المحتويات

2.....

4..... : المايكروكنترولر

4..... 1. هاردوير المايكروكنترولر

6..... 2. المايكروكنترولر

9..... : اللاب فيو

10..... 1

11..... 2

13..... 3

15.....

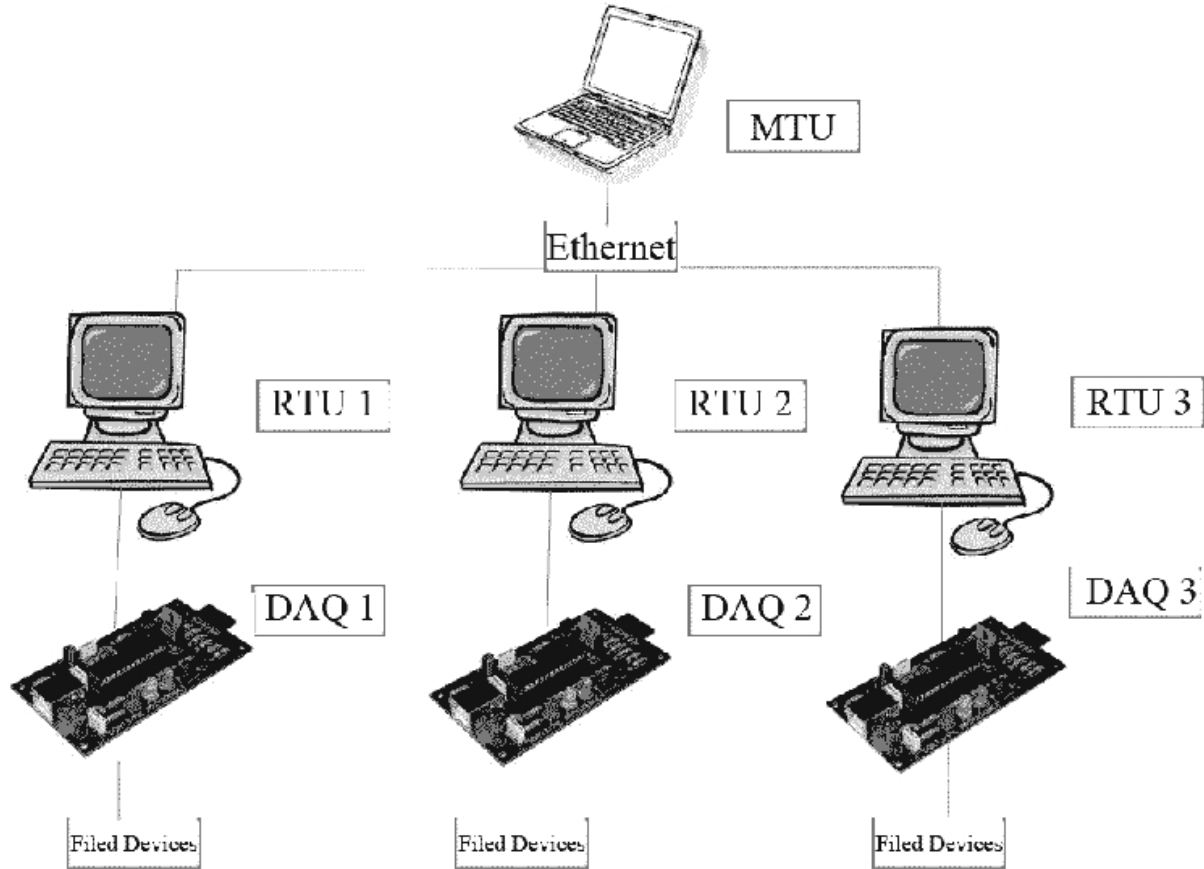


## المقدمة

هي اختصار (SCADA) Supervisory Control and Data Acquisition

هو نظام يقوم بوظيفتي المراقبة و التحكم في العمليات الصناعية المختلفة و ذلك عن طريق تجميع البيانات من الحساسات الموجودة في العملية الصناعية و إرسالها إلى كمبيوتر رئيسي يقوم بعرض قيم الحساسات و إرسال أشارات تحكم إلى المشغلات الموجودة في العملية الصناعية من خلال شاشة التشغيل و يكون الوسيط بين الحساسات و الكمبيوتر هو Data Acquisition Card حيث يقوم بمهمة قراءة الحساسات وتخزينها بداخله و من ثم الأتصال بالكمبيوتر عن طريق منفذ Serial كما سنشرح في هذا الكتاب أو عن طريق منفذ USB

يتم شراء كارت DAQ من شركة لاب فيو بأسعار تبدأ من 200 دولار وفي هذا الكتاب سنتناول كيفية بناء كارت DAQ عن طريق المايكروكنترولر بتكلفة بسيطة جدا باستخدام ATMEGA32 وهو ما يمكن تطبيقه على PIC16F877A بكل سهولة حيث أن الكود مكتوب بلغة MIKROC



يتكون نظام سكاذا SCADA من ثلاثة أجزاء :

**الجزء الأول** Hardware يتمثل في كارت DAQ الذى يقوم بتجميع المعلومات من الحساسات الموجودة فى العملية الصناعية بالإضافة إلى جهاز الكمبيوتر.

**الجزء الثانى** Software هو البرنامج المثبت على جهاز الكمبيوتر الذى يقوم بالاتصال بالهاردوير و أستقبال البيانات و تمثيلها للمستخدم فى شكل منحنيات و قيم و إصدار تقارير Reports توضح سير العملية الصناعية و أذارات Alarms عند حدوث خطأ معين .

**الجزء الثالث** Communication و يمثل وسيلة الأتصال التى تقوم بنقل المعلومات من المحطات الفرعية ألى المحطة الرئيسية و التى تصل غالبا لمئات الأميال و قد يستخدم الأترنت أو الأتصال عن طريق القمر الصناعى و يجب أستخدام وسيلة أتصال مؤمنة ومشفرة حتى لا يتاح لأى شخص من خارج النظام بالتحكم فى المعدات و قراءة التقارير الخاصة بالعملية الصناعية

يقوم المايكروكنترولر بجمع المعلومات من 8 حساسات أنالوج و 8 حساسات ديجتال و يقوم بأستقبال أوامر من شاشة سكاذا للتحكم فى 8 أجهزة و أيضا هناك شاشة LCD صغيرة موصلة بالمايكروكنترولر يمكن أستخدامها لعرض أى قيمة و يتم التوصيل بين جهاز الكمبيوتر و المايكروكنترولر من خلال USART عن طريق كابل السريال .

و سيتم تقسيم الشرح إلى جزئين:

**الجزء الأول** : المايكروكنترولر و يشمل بناء الدائرة الخاصة بالمايكروكنترولر على برنامج المحاكاة و كتابة الكود الخاص بلغة الميكروسى و كيفية إرسال و أستقبال البيانات من و إلى اللاب فيو.

**الجزء الثانى** : اللاب فيو و يشمل شرح كيفية أستقبال البيانات من المايكروكنترولر و عرضها على المستخدم حيث يتم تمثيل Analog Sensor عن طريق عدادات و يتم تمثيل Digital Sensor عن طريق ليدات مع توافر 8 مفاتيح فى شاشة اللاب فيو يمكنهم التحكم فى 8 أجهزة على خرج المايكروكنترولر.

# الجزء الأول : المايكروكنترولر

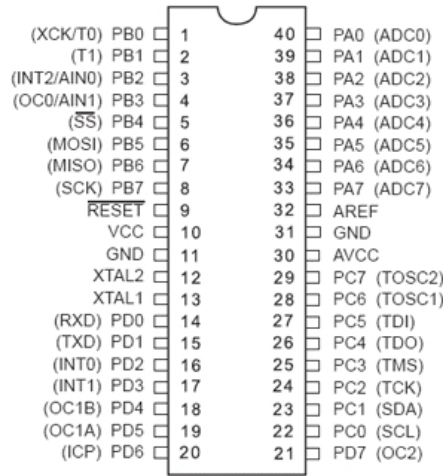
## 1. هاردوير المايكروكنترولر

### مكونات DAQ CARD

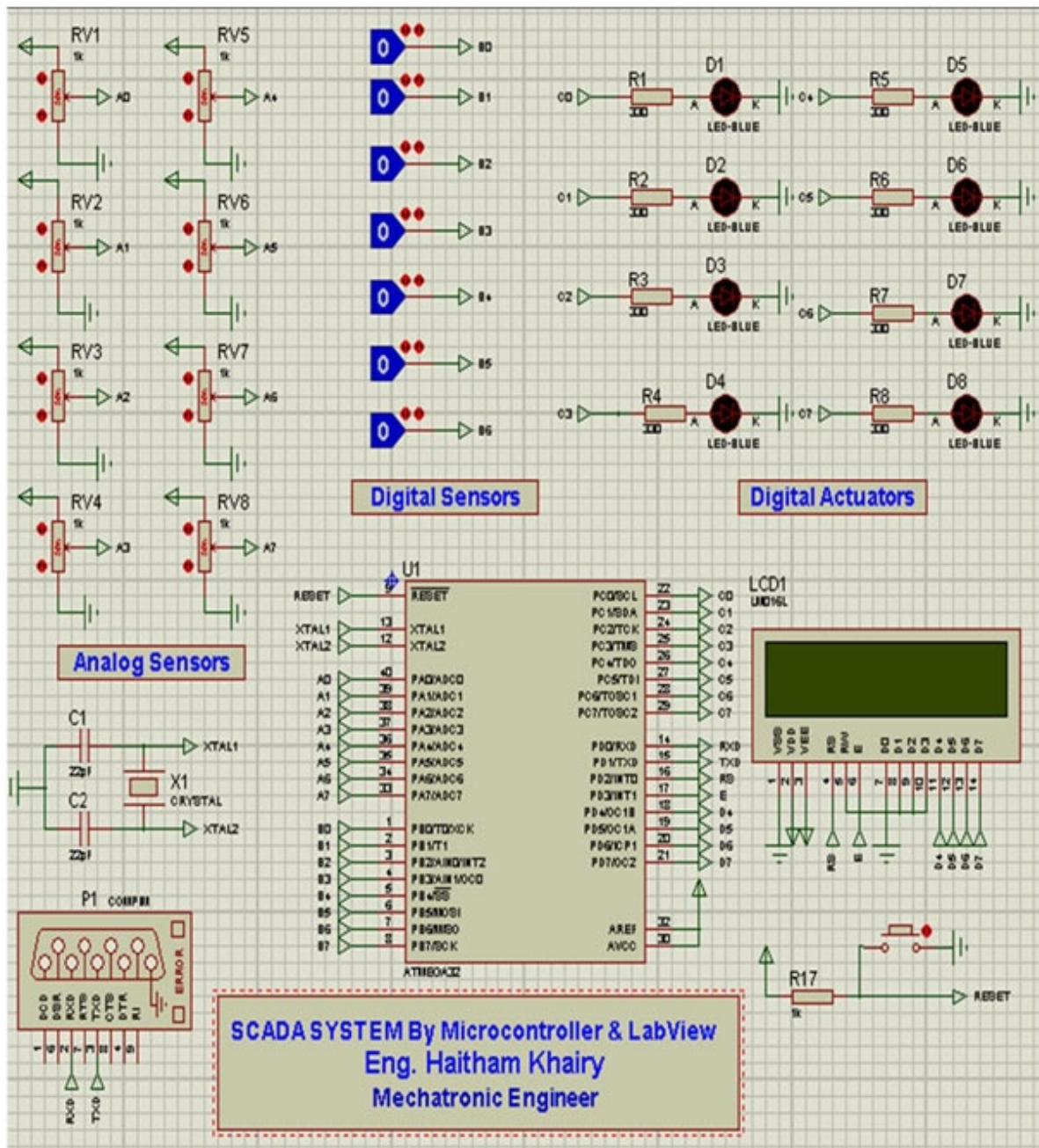
ATMEGA 32	المايكروكنترولر
MikroC Pro for AVR 5.4	برنامج كتابة كود المايكروكنترولر
Protues 7.10	برنامج المحاكاة
LabView 2011	برنامج اللاب فيو
Virtual Serial Port Kit	برنامج سرىال وهمى

يحتوى المايكروكنترولر ATMEG32 على أربعة منافذ

PORTA	8 Analog sensor
PORTB	7 Digital sensor
PORTC	8 Digital Output
PORTD	USART + LCD



يتم توصيل الميكروكنترولر بمكونات تحاكي الحساسات و المشغلات الموجودة في العملية الصناعية



بعد تنصيب برنامج Virtual Serial Port Kit و عمل منفذين سريرال و هميين مثلا COM3>COM4 يتم اختيار أحدهما في بروتوس و اختيار الأخر في اللاب فيو

## 2. برنامج المايكروكنترولر

SCADA System With Microcontroller & LabView.c

10/10/2012 11:06:02 AM

```
1: /*----- SCADA SYSTEM By Microcontroller & LabView-----  
2: -----Eng.Haitham Khairy-----  
3: -----Mechatronic Engineer-----  
4: -----ArabMicro-----*/  
5:  
6: #include <built_in.h>  
7: char DIG Input,Read[1];  
8: unsigned int adc rd0,adc rd1,adc rd2,adc rd3,adc rd4,adc rd5,adc rd6,adc rd7 ;  
9: char adc rd0c[6],adc_rdc[6],adc_rd2c[6],adc_rd3c[6],adc_rd4c[6],adc_rd5c[6],adc_rd6c  
c[6],adc_rd7c[6] ;  
10: CHECK PORT B();  
11: CHECK PORT_A();  
12: send();  
13: void main() {  
14: //////////////////////////////////////////////////INITIALIZE////////////////////////////////////  
15:   DDRB=0x00 ;  
16:   DDRC=0xFF ;  
17:   PORTC=0x00;  
18:   UART1 Init(9600);  
19:   Delay ms(100);  
20:   UART1 Write(10);  
21:   while(1){  
22:     //////////////////////////////////Acquire_Data////////////////////////////////////  
23:     CHECK PORT B();  
24:     CHECK PORT A();  
25:     //////////////////////////////////Recvie_Data////////////////////////////////////  
26:     if(UART1 Data Ready() == 1 ){  
27:       Uart1 Read text(Read,"EN",10);  
28:       PORTC=Read[0];  
29:     //////////////////////////////////Send_Data////////////////////////////////////  
30:     send();  
31:   }  
32: }  
33: }  
34:  
35: CHECK PORT B(){  
36:   DIG_Input=PINB;  
37: }  
38:  
39: CHECK PORT A(){  
40:   adc rd0 = ADC Read(0);  
41:   Wordtostr(adc rd0,adc rd0c);  
42:   adc rd1 = ADC Read(1);  
43:   Wordtostr(adc rd1,adc rdc);  
44:   adc rd2 = ADC Read(2);  
45:   Wordtostr(adc rd2,adc rd2c);  
46:   adc rd3 = ADC Read(3);  
47:   Wordtostr(adc rd3,adc rd3c);  
48:   adc rd4 = ADC Read(4);  
49:   Wordtostr(adc rd4,adc rd4c);  
50:   adc rd5 = ADC Read(5);  
51:   Wordtostr(adc rd5,adc rd5c);  
52:   adc rd6 = ADC Read(6);  
53:   Wordtostr(adc rd6,adc rd6c);  
54:   adc rd7 = ADC Read(7);  
55:   Wordtostr(adc rd7,adc rd7c);  
56: }  
57:
```



```

58: send() {
59:   if (UART1 Tx Idle() == 1) {
60:     Uart1 write text(adc rd0c);
61:     Uart1 write text(adc rd1c);
62:     Uart1 write text(adc rd2c);
63:     Uart1 write text(adc rd3c);
64:     Uart1 write text(adc rd4c);
65:     Uart1 write text(adc rd5c);
66:     Uart1 write text(adc rd6c);
67:     Uart1 write text(adc rd7c);
68:     Uart1 write(DIG Input);
69:     Uart1_write(10);
70:   }
71: }
72:

```

## أساسيات

- يمكن تخزين البيانات بعدة صور منها bit,char,int,float بالنسبة لل char يخزن في صورة Byte و يمكن تمثيله بعدة أشكال مثال

ASCII	DECIMAL	BINARY	HEXADECIMAL
1	49	00110001	0X31

ويمكن التحويل بكل سهولة من خلال MIKROC

DECIMAL	HEXADECIMAL	BINARY	CHARACTER
49	31	00110001	1

و يمكن أيضا الاطلاع على جدول ASCII Chart من قائمة Tools

## مثال عملي

```

PORTC='1';
PORTC=49;
PORTC=0b00110001;
PORTC=0X31;

```

- لدينا يتراوح 5 0 عن طريق محول ADC يمكن الحصول على قيمة الحساس التحويل على resolution 8 bit 10 bit حيث يتم تقسيم 5 فولت إلى 1024 جزء و كل جزء قيمته 4.8 ميلي فولت لذلك أقل قيمة يمكن الأحساس بها 4.8 ميلي فولت و مضاعفاتها و لو الدقة 8 bit سيتم تقسيم 5 فولت إلى 256 جزء و كل جزء قيمته 19.6 ميلي فولت.

$$Q = \frac{(V_{\max} - V_{\min})}{2^n - 1} \cdot Q = \frac{5 - 0}{1024 - 1} = 4.8mV$$



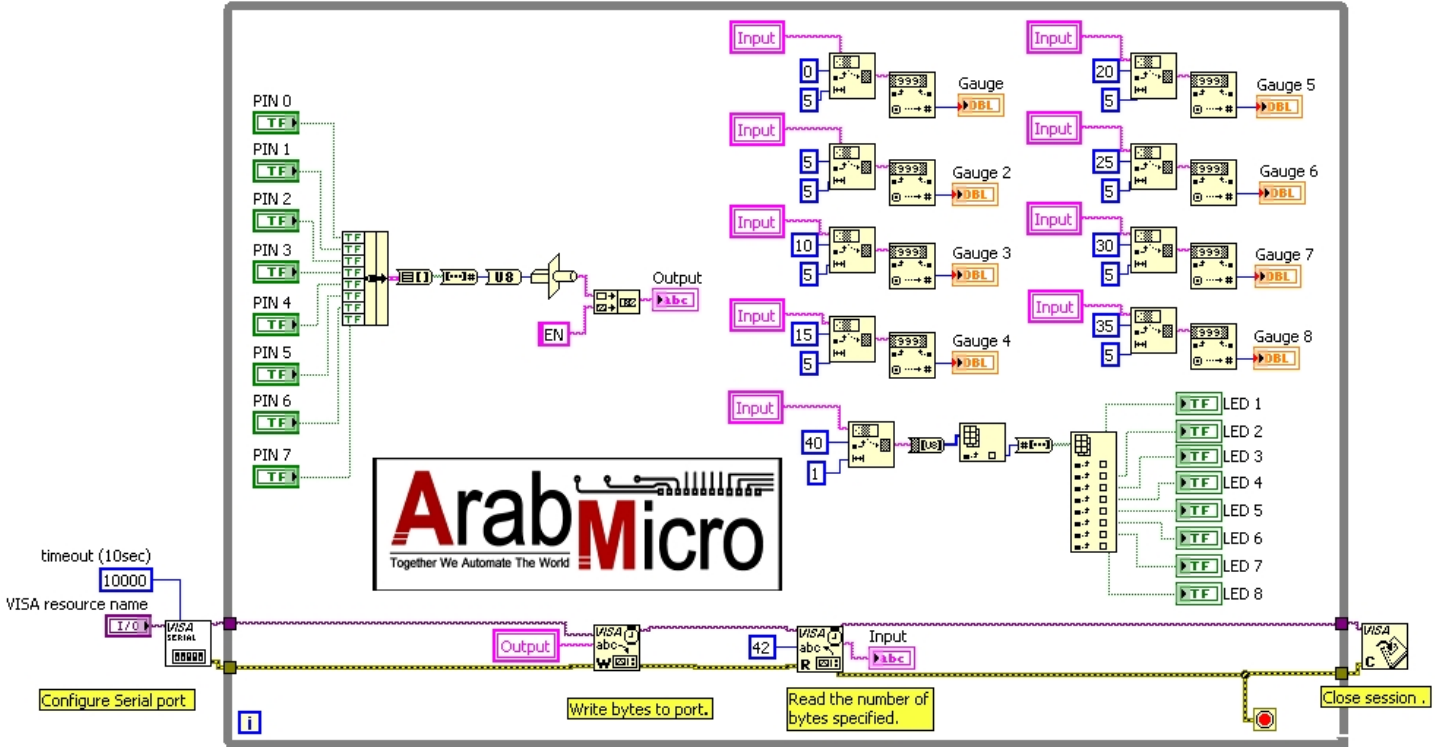


الوظيفة	السطر
أستدعاء المكتبة الخاصة بمحول ADC	6
أنشاء المتغيرات	7,8,9
دالة وظيفتها الحصول على قيم PORTB و الذى يحاكي Digital Sensor	10
دالة وظيفتها الحصول على قيم PORTA و الذى يحاكي Analog Sensor	11
دالة وظيفتها إرسال قيم الحساسات إلى برنامج اللاب فيو	12
تهيئة PORTB ليكون دخل و PORTC ليكون خرج	15,16
تهيئة الأتصال عن طريق UART	18,19
إرسال قيمة عشوائية فى البداية	20
بداية حلقة لا نهائية لتكرار البرنامج بصورة مستمرة	21
تحصيل البيانات المراد إرسالها إلى اللاب فيو عن طريق أستدعاء الدوال الخاصة بالحصول على قيم الحساسات بالسطر 10 و 11	23,24
دالة تفيد بوجود بيانات مستقبلية عبر السريال	26
وظيفة هذه الدالة هو أستقبال النص القادم من خلال السريال حتى يتم العثور على "EN" و من ثم تخزين القيمة فى المتغير Read (مرتبطة ببرنامج اللاب فيو)	27
تخزين أول قيمة من مصفوفة Read[0] بداخل PORTC (مرتبطة ببرنامج اللاب فيو)	28
أستدعاء الدالة Send المختصة بأرسال البيانات إلى اللاب فيو	30
الحصول على قيمة PORTB و ذلك عن طريق الأمر PINB و تخزينها بداخل المتغير DIG_Input و تكون هذه القيمة Byte و هى ناتج 8 Bit تبع البورت	36
تقوم الدالة ADC_Read(0) بالحصول على قيمة Analog للطرف ADC0 و تعطى قيمة من 0 إلى 1023 وذلك لأن دقة المحول 10 bit	40
وظيفة الدالة هو تحويل قيمة السنسور من إرقام إلى نصوص وذلك لتطويعها للأرسال عبر منفذ السريال مثال يتم تحويل 125 إلى "125" و يتم تخزينه فى 6 Byte ( للمزيد من الشرح قف على الدالة بالماوس ثم أضغط F1 )	41
دالة للتأكد من أن البيانات السابقة تم إرسالها و التحقق أن سجل الأرسال فارغ أستعداداً لأرسال البيانات الجديدة	59
إرسال قيم 8 حساس أنالوج	60
إرسال قيم 8 حساس ديجتال	68
إرسال الرقم 10 و هو عبارة عن Termination Character كرمز لنهاية الأرسال	69

## الجزء الثاني : اللاب فيو

هو عبارة عن لغة برمجة من أنتاج شركة National Instruments و تختلف عن لغات البرمجة الأخرى في :

1. تستخدم الأيقونات في البرمجة بدلا من كتابة الكود
2. مصممة للتخاطب مع الأجهزة الختلفة و تدعم بروتوكولات الأتصال المختلفة Rs232,USB
3. مصممة للتعامل مع الشبكات و تدعم TCP,UDP



ينقسم البرنامج بالأعلى إلى ثلاثة أجزاء

الجزء الأول : أنشاء الأتصال و تهيئته و أستقبال-أرسال البيانات و هو الجزء الموجود بأسفل الصورة و يتم تخزين البيانات المستقبلية في متغير يسمى Input و يتم تخزين البيانات المراد إرسالها في متغير يسمى Output

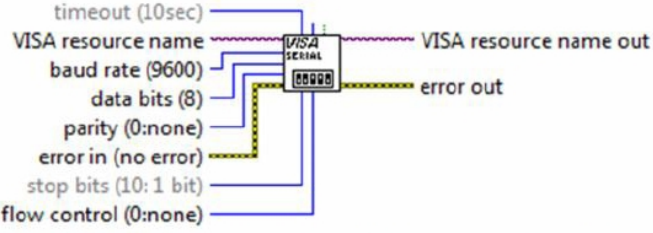
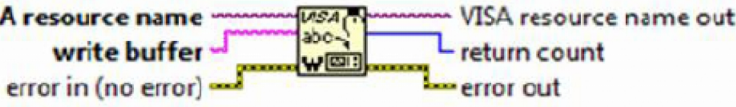
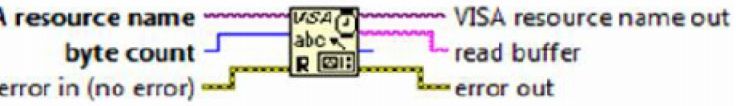
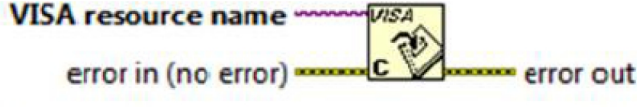
الجزء الثاني : و هو الموجود بأعلى اليمين و يتم فيه عمل فك لتشفير الداتا المستقبلية من المايكروكنترولر و المخزنة في المتغير Input بداخل اللاب فيو و ذلك بتقسيمها إلى 8 عدادت و 8 ليدات

الجزء الثالث : و هو الموجود بأعلى اليسار و يتم فيه تشفير الداتا المراد إرسالها إلى المايكروكنترولر حيث أننا نريد إرسال قيمة 8 مفاتيح فيتم تشفيرهم لما يقابلهم من ASCII Code و يتم تخزين قيمته في المتغير Output و من ثم توصيلها ب VISA Write

## الأول :

يتم إنشاء الأتصال مع أى جهاز خارجى عن طريق NI-VISA و هى عبارة عن لغة برمجة مخصصة للتخاطب مع الاجهزة الخارجية أياً كان طريقة أتصالها بالكمبيوتر و يتم **تنصيبها** بشكل منفصل و لإنشاء الأتصال مع السريال بورت يلزم أربعة مكونات رئيسية ويتم الحصول عليهم من

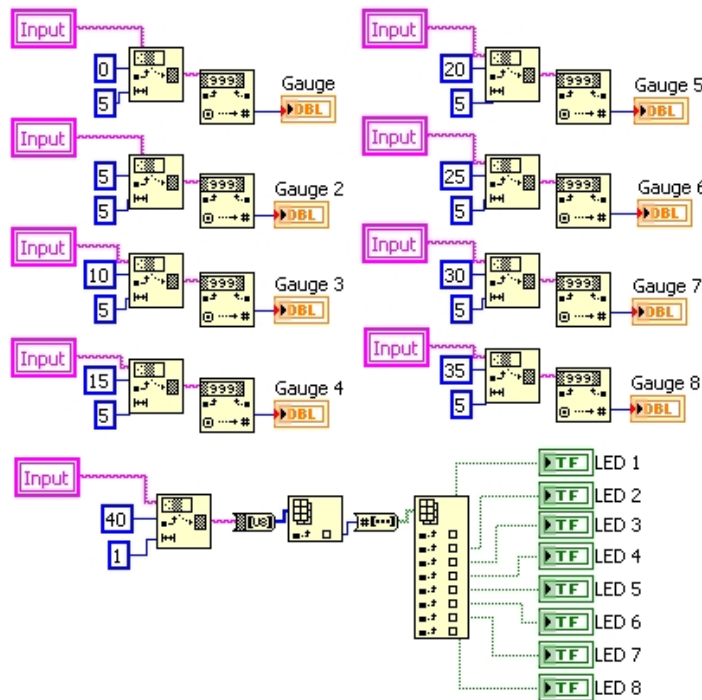
Instrument I/O > Serial > ....

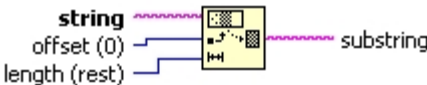
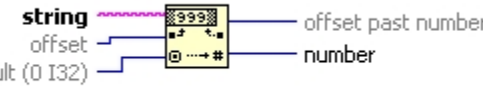
<p><b>VISA Configure Serial Port</b></p>  <p>timeout (10sec) VISA resource name baud rate (9600) data bits (8) parity (0:none) error in (no error) stop bits (10:1 bit) flow control (0:none)</p> <p>VISA resource name out error out</p> <p>Initializes the serial port specified by <b>VISA resource name</b> to the specified settings. The VISA class you wire to the <b>VISA resource name</b> input determines the polymorphic instance to use.</p>	<p>تقوم هذه الدالة بفتح جلسة أتصال مع منفذ السريال و يتم وضعها قبل <b>While</b></p>
<p><b>VISA Write</b></p>  <p>VISA resource name write buffer error in (no error)</p> <p>VISA resource name out return count error out</p> <p>Writes the data from <b>write buffer</b> to the device or interface specified by <b>VISA resource name</b>.</p>	<p>تقوم هذه الدالة بأرسال النص الموصل ب <b>Write buffer</b> إلى المايكروكنترولر</p>
<p><b>VISA Read</b></p>  <p>VISA resource name byte count error in (no error)</p> <p>VISA resource name out read buffer error out</p> <p>Reads the specified number of bytes from the device or interface specified by <b>VISA resource name</b> and returns the data in <b>read buffer</b>.</p>	<p>تقوم هذه الدالة بأستقبال النص القادم من المايكروكنترولر على <b>read buffer</b></p>
<p><b>VISA Close</b></p>  <p>VISA resource name error in (no error)</p> <p>error out</p> <p>Closes a device session or event object specified by <b>VISA resource name</b>.</p>	<p>تقوم هذه الدالة بغلاق جلسة الأتصال مع منفذ السريال و يتم وضعها بعد <b>while</b> و تنفذ مرة واحدة فقط فى حالة تحقق شرط <b>While</b> و ذلك فى حالة <b>error</b></p>

## الجزء الثاني :

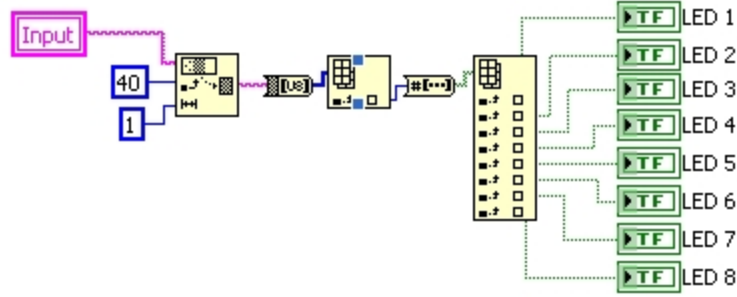
يتم استقبال البيانات في اللاب فيو على شكل text يتكون من 42 حرف مرتبة حسب كود المايكروكنترولر بحيث يرسل قيم حساسات الأنالوج ثم حساسات الديجتال و يتم تفكيك text بنفس الصورة بداخل اللاب فيو للحصول على قيم الحساسات ثم نهاية الأرسال

Analog Sensor 0	1	2	3	4	5	6	7	Digital sensor	End Character
5 Byte	5 Byte	5 Byte	5 Byte	5 Byte	5 Byte	5 Byte	5 Byte	1 Byte	1 Byte



<p><b>String Subset</b></p> <p><b>string</b> </p> <p>Returns the <b>substring</b> of the input <b>string</b> beginning at <b>offset</b> and containing <b>length</b> number of characters.</p>	<p>تقوم الدالة بقطع جزء من النص الأساسي ونحدد مكان بدء القطع حسب قيمة <b>offset</b> و نحدد طول النص المقطوع حسب قيمة <b>length</b></p> <p>Programming &gt; String</p>
<p><b>Decimal String To Number</b></p> <p><b>string</b> </p> <p>Converts the numeric characters in <b>string</b>, starting at <b>offset</b>, to a decimal integer and returns it in <b>number</b>.</p>	<p>تقوم بتحويل الأرقام من صيغة <b>character</b> إلى أرقام في صيغة <b>Int</b> بنفس الشكل يمكن استخدامها في عمليات الطرح و الجمع</p> <p>Programming &gt; String &gt; str/num conversion</p>

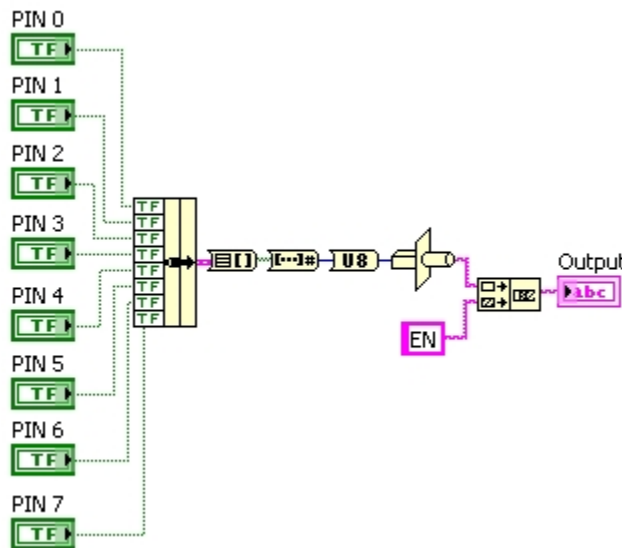
ثم نقوم بتكرار التقطيع مع تغيير offset في كل مرة للحصول على قيمة حساس مختلف







<p><b>String Subset</b></p> <p>Returns the <b>substring</b> of the input <b>string</b> beginning at <b>offset</b> and containing <b>length</b> number of characters.</p>	<p>دالة تقطيع text Programming &gt; String</p>
<p><b>String To Byte Array</b></p> <p>Converts a string into an array of unsigned bytes.</p>	<p>تقوم بتحويل text إلى Array و يتم تخزين الحرف كعنصر في المصفوفة مثال لو كان النص يتكون من ثلاثة حروف ستحتوى المصفوفة على ثلاث عناصر مع العلم بأن القيم تخزن داخل المصفوفة في صيغة Decimal كمثال حرف A سيسجل داخل المصفوفة 65 Programming &gt; String &gt; conversion</p>
<p><b>Index Array</b></p> <p>Returns the <b>element or subarray</b> of <b>n-dimension array</b> at <b>index</b>.</p>	<p>تستخدم هذه الدالة للحصول على قيمة عنصر معين داخل المصفوفة و سيكون الخرج منها Decimal Programming &gt; Array</p>
<p><b>Number To Boolean Array</b></p> <p>Converts an integer or fixed-point number to a Boolean array.</p>	<p>تقوم بتحويل الداتا من صورة Decimal إلى صورة Binary كمثال سيتم تحويل عنصر 255 إلى مصفوفة لها 8 عناصر و قيمة كل عنصر True وهو ما يمكن استخدامه في أضاءة ليد و يتم استخدام الدالة Index Array مرة أخرى لاستخراج عناصر المصفوفة Programming &gt; Numeric &gt; Conversion</p>

:

سنقوم الآن بتجميع قيمة 8 مفاتيح في Byte حيث تمثل قيمة كل مفتاح 1Bit و أيضا ألحاق رمز يمثل نهاية الأرسال و الذى تم تحديده في كود المايكروكنترولر



<p><b>Bundle</b></p> <p>cluster element 0 element 1 ... element n-1</p> <p>output cluster</p> <p>Assembles a cluster from individual elements.</p>	<p>تقوم بتجميع قيم 8 مفاتيح في Cluster Programming &gt; Cluster</p>
<p><b>Cluster To Array</b></p> <p>cluster → array</p> <p>Converts a cluster of elements of the same data type to a 1D array of elements of the same data type.</p>	<p>نقوم بالتحويل من Cluster إلى Array مع ملاحظة أن المصفوفة من نوع Boolean Programming &gt; Cluster</p>
<p><b>Boolean Array To Number</b></p> <p>Boolean array → number</p> <p>Converts a <b>Boolean array</b> to a 32-bit unsigned integer by interpreting the array</p>	<p>يتم تحويل المصفوفة و التى تحوى بداخلها 8 Bit إلى ما يقابلها من رقم Decimal مثال لو كانت كل المفاتيح on سيكون الخرج 255 Programming &gt; Numeric &gt; Conversion</p>

<p><b>To Unsigned Byte Integer</b></p> <p>number  unsigned 8bit integer</p> <p>Converts a number to an 8-bit unsigned integer in the range 0 to 255.</p>	<p>تقوم بتحويل نوع خرج الدالة السابقة من Bit 8 إلى Bit 32</p> <p>Programming &gt; Numeric &gt; Conversion</p>
<p><b>Type Cast</b></p> <p>type  x  *(type *) &amp;x</p> <p>Casts x to the data type,</p>	<p>تقوم بتحويل من صيغة Decimal إلى ASCII كمثال رقم 65 سيصبح حرف A</p> <p>Programming &gt; Numeric &gt; Data Manipulation</p>
<p><b>Concatenate Strings</b></p> <p>string 0  concatenated string</p> <p>string 1</p> <p>...</p> <p>string n-1</p> <p>Concatenates input strings and 1D arrays of strings into a single output string. For array inputs, this function concatenates each element of the array.</p>	<p>تقوم هذه الدالة بوصل أكثر من string و ستستخدم لألحاق رمز نهاية الكود و يمكن أستخدامها في حالة أردنا الكتابة على شاشة LCD</p> <p>Programming &gt; String</p>

## المتغيرات Variables في اللاب فيو

تمكننا المتغيرات من توفير الأسلاك المستخدمة في التوصيل بين الدوال و بعضها و تساهم في تنظيم الكود .

يمكن أدراجها بطريقتين:

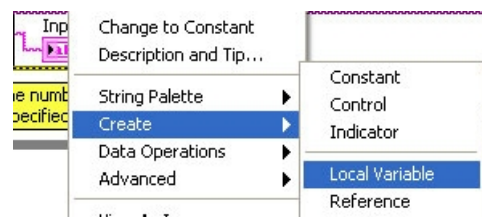
الطريقة الأولى

Programming > Structures > Local Variable

ستظهر لك المتغير في صورة علامة أستفهام ويمكن جعله للقراءة أو للكتابة وذلك بالضغط يمين ثم Read و لأختيار العنصر المراد ربط المتغير به يتم الضغط عليه بالزر الأيسر و أختيار أسم العنصر

الطريقة الثانية

يتم الضغط بالزر الأيمن على العنصر المراد أنشاء متغير له ثم أختيار Create



باللاب فيو

DAQ جاهزة من أنتاج ArabMicro

يمكنك

NI

بأسعار تنافسية مقارنة بأسعار

ArabMicro .

مجهز

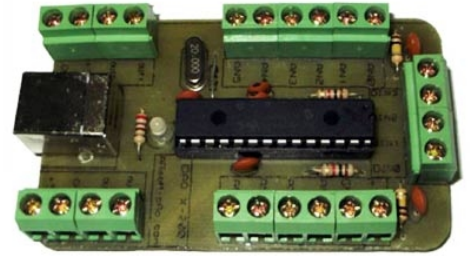
و هو ما يتيح لكم استخدام برنامج اللاب فيو في مشاريع التخرج أو التطبيقات الصناعية بسهولة و يسر و بسرعات تصل إلى أضعاف كروت NI .

6 Analog Input 10-bit , 6 Digital Input-6 Output  
100 Sample / Second for each channel  
USB 1.1 Low Speed up to 1.5Mbit/sec  
LabVIEW Compatible  
Atmega8 Main Controller



USB DAQ X-100

10 Analog Input 10-bit , 4 Digital Input-4 Output  
500 Sample / Second for each channel  
USB 2.0 Full Speed up to 12Mbit/sec  
LabVIEW Compatible  
PIC 18F2550 Main Controller



USB DAQ X-200

13 Analog Input 10-bit , 8 Digital Input- 8 Output  
500 Sample / Second for each channel  
USB 2.0 Full Speed up to 12 Mbit/sec  
LabVIEW Compatible  
PIC 18F4550 Main Controller



USB DAQ X-300

للأسعار و تنزيل الداتاشيت الخاصة

Website: [www.ArabMicro.Com](http://www.ArabMicro.Com)

Page: [www.facebook.com/pages/ArabMicro/250909681693209](https://www.facebook.com/pages/ArabMicro/250909681693209)

Email: Sales@arabmicro.com

Phone: 01112970435 OR +201112970435

