بنام خداوند بخشنده مهربان

عنوان مقاله :

ارتباط سریال کامپیوتر و میکروکنترلر با استفاده از VB و VC++

ارائه دهنده : بهزاد خزاما

دانشجوی برق دانشگاه آزاد سبزوار

مقدمه :

در این مقاله سعی در تشریح چگونگی کار با پورت سریال و ایجاد ارتباط بین دو **کامپیوتر** و یا **کامپیوتر** و **میکروکنترلر** را داریم.

همچنین چگونگی ارتباط با مودم و انتقال اطلاعات از طریق آن بررسی شده است . در اینجا ما از **VB** برای ارتباط با روالهای **API** و یا کنترل های **ACTIVEX** استفاده کرده ایم که البته توضیحات ذکر شده در زبانهای دیگری مانند **VC++** نیز صادق میباشد .که به تشریح آن خواهیم پرداخت

در این مقاله سعی در تشریح چگونگی کار با ***پورت سریال کامپیوتر*** و ارتباط آن با ***کامپیوترهای دیگر*** و یا با ***میکروکنترلر*** را داریم.

همچنین چگونگی ارتباط با مودم و انتقال اطلاعات از طریق آن بررسی شده است . در اینجا ما از VB برای ارتباط با روالهای API و یا کنترل های ACTIVEX استفاده کرده ایم که البته اکثر توضیحات ذکر شده در زبانهای دیگری مانند VC++ نیز صادق میباشد .که به تشریح آن خواهیم پرداخت .

هر کامپیوتر دارای تعدادی پورت سریال میباشد که به ترتیب به نامهای **Com۱,Com۲,Com۳,,** نامیده میشوند. و هر کدام ممکن است که برای ارتباط با وسیله خاصی در نظر گرفته شود . مثلا' ممکن است موش به COM۱ و اسکنر به COM۲ و هر دستگاه سریال دیگری نیز به همین صورت بروی یکی از پورتهای سریال نصب شده در PC متصل شود . پورت سریال کانالی برای انتقال اطلاعات و دریافت آن به صورت بیت به بیت است.I/O به روش سریال نمونه ای از نحوه' ارتباط حمایت شده توسط ویندوز است این روش هنگام ارتباط (ONE TO ONE) بین دو وسیله استفاده می شود این روش میتواند با استفاده از کابل یا مودم یا استفاده از فرستنده گیرنده های مادون قرمز ایجاد گردد .ارتباط سریال با استفاده از روالهای API .بسیار شبیه کار با فایلها میباشد . همچنین ویندوز دارای توابعی برای باز یا بسته کردن و دستکاری مشخصات پورت میباشد. برای کار با پورت سریال توابعی موجود است تا بتوان اطلاعات CPUرا به زبان پورت ترجمه نمود و آنرا به فرم بیت به بیت درآورد تا برای ارسال آماده شوند همچنین در حالت دریافت اطلاعات نیز با کنار هم قرار گرفتن این بیتها یک بایت ایجاد شده و به CPU برای پردازش فرستاده میشود. البته این فرایند توسط لایه های مختلفی صورت میگیرد ویندوز برای کار با پورت سریال از یک درایور انتقال سریال بنام COMM.DRV استفاده میکند. البته با وجود روالهای API دیگر نیازی به چنین ارتباطهای سطح پائینی برای برنامه نویس احساس نمیشود. به عنوان یک برنامه نویس بسته به نیاز خود میتوانید از روالهای API و یا از یکسری ACTIVEX استفاده نمائید. در این مقاله ما به ذکر برخی از روالهای API مهم و توضیح یک ACTIVEX ارئه شده همراه با VISUAL STADIO PRO بنام COMM۳۲ خواهیم پرداخت درانتها نیز به طریقه' ارتباط با میکروکنترلر خواهیم پرداخت .

قدم اول : ایجاد ارتباط با پورت سریال توسط COMM۳۲ .

برای شروع به کار و دسترسی به پورت توسط COMM۳۲ یکسری property گنجانده شده است که با setکردن آنها میتوان پورت را آماده دسترسی نمود . در زیر به توضیح این property ها میپردازیم.

۱ – Set کردن شماره'پورت `mscomm۱.Comport=[value]`

این property شماره پورت مورد نظر را SET مینماید. این مقدار میتواند بین **۰- ۱۶** باشد. اگر پورتی با شماره'مورد نظر شما موجود نباشد یک خطا رخ میدهد که میتوان از آن برای اطمینان از انجام کار سود برد.

۲ – set کردن مشخصات پورت `mscomm۱.setting = [string value]`

این property امکان تعریف مشخصات پورت را میدهد .این مشخصات بایستی به صورت یک رشته تعریف شود.این رشته شامل تعریف " **boud rate , parity bit , data bit ,stop bit** " میباشد

مقادیر معتبر برای این مقادیر را در جدول زیر نشان داده شده است .

Boud rate	**110 –300 –600 –1200 – 2400 – 4800 – 9600(** *defult* **) –14400- 19200 – 28800 38400 – 56000 – 57600 - 115200 - 128000 - 256000**
Parity bit	**M (mark) – E (even) – N (none(** *defult* **)) – O (odd) - S (space)**
Data bit	**4 – 5 – 6 – 7 – 8 (** *defult* **)**
Stop bit	**1(** *defult* **) – 1.5 – 2**

۳ – portopen = [value *bolean*]

با set کردن آن به **TRUE** پورت مورد نظر باز شده و در اختیار ما قرار میگیرد .برای قطع ارتباط این خاصیت را **FALS** میکنیم . اگر شماره' پورت در **comport** درست set نشود (موجود نبودن پورت) با باز کردن پورت یک خطا رخ میدهد که میتوان با چک کردن آن درستی عمل را تست نمود.

```
‘ Open the serial port
 MSComm1.CommPort = 2
 MSComm1.Settings = “9600,N,8,1”
 MSComm1.PortOpen = True
```

قدم دوم : <u>سازماندهی و تنظیمات مورد نیاز .</u>

تنظیم بافرهای ورودی وخروجی **:** وقتی که شما یک پورت را باز میکنید دو بافر ورودی و خروجی ایجاد میشود این بافرها نگهدارنده' اطلاعات ورودی وخروجی میباشند .برای سازماندهی این بافرها تعدادی property وجود دارد. **Outbuffersize و Inbuffersize** برای تعیین مقدار حافظه'بافر برای ورودی و خروجی میباشد . افزایش مقدار این حافظه ها باعث احتمال کاهش حافظه برای برنامه میشود البته با وجود کامپیوترهای امروزی نیازی به نگرانی نیست .کاهش این مقدار نیز باعث بروز over flow میشود مگر اینکه از hand shaiking استفاده نمایید .

Rthershold , Sthershold

Sthershold وظیفه' set کردن و برگرداندن حداقل تعداد کاراکترهای قابل قبول در بافر خروجی قبل از رویدادoncomm و تغییر مقدار commevnt به مقدار commevsend را دارد مثلا ۰ کردن آن باعث غیر فعال شدن رویداد oncomm در برابر این مورد میشود و ۱ کردن آن باعث رویدادن آن در زمان خالی شدن بافر میشود .برای توضیحات بیشتر به توضیحات رویداد oncomm مراجعه نمائید .

Rthershold وظیفه' set کردن و برگرداندن تعداد کاراکترهای موجود در بافر ورودی قبل از ایجاد رویداد oncomm و تغییر مقدار commevnt به مقدار commevrecive را دارد .

Inputlen

Set کننده و برگرداننده طول رشته' خوانده شده توسط input میباشد. قرار دادن ♦ در آن باعث خواندن تمام محتویات بافر گیرنده میشود.

Eofenable

نشانگررسیدن به یک رشته end of file (^Z - str$(۲۶)) میباشد و با ۱ کردن آن با رسیدن به eof یک رویداد oncomm رخ خواهد داد.

اداره کردن بافرهای ورودی و خروجی

همانطور که گفته شد بافرهای ورودی و خروجی هنگام ایجاد یک پورت ایجاد میشود. شما با یکسری property میتوانید بر رفتار این بافرها کنترل داشته باشید. شما میتوانید اطلاعات را به صورت binary بهtext دریافت کنید.

Input

این خاصیت برای ذخیره کردن و بازیابی اطلاعات درون بافر ورودی بکار میرود. مثلا اگر میخواهید اطلاعات رسیده را در یک text box نشان دهید:

text۱.text = mscomm۱.input

نکته : ابتدا بایستی طول رشته' مورد نظر را با inputlen معین کنید .

Inputmode

برای تعین فرمت ورودی بکار میرود شما میتوانید اطلاعات را به صورت binary بهtext دریافت کنید این کار را با استفاده از ثابتهای vb به نامهای **comminputmodetext** یا **comminputmodebinary** انجام پذیر است . در این صورت اطلاعات به صورت text و یا باینری در یک آرایه' باینری ذخیره میشود .

Inbuffercount

بیانگر تعداد بایتهای رسیده می باشد همچنین با صفر کردن آن بافر ورودی پاک میشود

بافر فرستنده ----------

Output

از این خاصیت برای ارسال اطلاعات یا دستورات استفاده نمود .مانند خاصیت input شما میتوانید binary بهtext را ارسال کنید این کار را میتوانید با استفاده از تعریف مقادیر به صورت binary بهtext انجام دهید .

مثال:

```
MSComm1.Output = "ATDT 111-111111"          Send an AT command

MsComm1.Output = "This is a text string"    Send a text string

MSComm1.Output = Out                        Send Byte array data
```

نکته : برای آشنائی با دستورات **AT** به بخش ارتباط با مودم مراجعه کنید .

Outbuffercount

شما همچنین میتوانید به تعداد کاراکترهای ارسالی از طریق این خاصیت پی ببرید .

Hand Shaiking دستدهی

یک بخش مهم در سازماندهی بافرهای ورودی وخروجی اطمینان از صحت اطلاعات رسیده و اطلاعات ارسالی است و اطمینان از اینکه سرعت در اطلاعات رسیده باعث سرریز (over flow) نشده است. **Hand Shaiking** با استفاده ار بروتکل ارتباطی خود اطلاعات رسیده را بررسی میکنند تا در اثر سرریز از بین نرفته باشند .

شما بایستی که پروتکل مورد نیاز را تعیین کنید. این پروتکل در حالت عادی غیر فعال است(comnone)

Setting	Value	Description
ComNone	0	No handshaking (Default).
ComXOnXOff	1	XOn/XOff handshaking.
ComRTS	2	RTS/CTS (Request To Send/Clear To Send) handshaking.
ComRTSXOnXOff	3	Both Request To Send and XOn/XOff handshaking.

انتخاب پروتکلها بستگی زیادی به دستگاهی که میخواهید به آن متصل شوید دارد با انتخاب comRTSXOnXOff هردو پروتکل حمایت میشوند

نکته : اگر شما این خاصیت را به هر کدام از مقادیر ۱ و ۲ set کنید باید که خاصیت rtsenable را true کنید وکر نه شما قادر به دریافت اطلاعات نخواهید بود .

Rtsenable = [*bolean*]

این خاصیت باعث high شدن پایه'rts میگردد پایه'rts در دستدهی rtc/cts استفاده شده و وجود این خاصیت به شما اجازه میدهد که به صورت دستی این کار را انجام دهید

Dsrholding

.

تشخیص دهنده' وضعیت پایه' dsr است این سیگنال از طرف فرستنده برای اعلام آمادگی خود فرستاده میشود .

Dtrenable

با set کردن آن به true خط **Dtr** با باز شدن پورت high و با بسته شدن آن low میشود

Setting	Description
True	Enable the Data Terminal Ready line.
False	(Default) Disable the Data Terminal Ready line.

نکته: در اکثر مواقع با ۰ کردن این خاصیت باعث hangup مودم میشویم .

Ctsholding

این خاصیت تشخبص دهنده' آمادگی برای ارسال اطلاعات میباشد که با چک کردن پایه' cts این کار را انجام میدهد. این پایه ورودی است و ار طرف فرستنده ارسال میشود

Setting	Description
True	Clear To Send line high.
False	Clear To Send line low.

Parityreplace ----------

این خلصیت set کننده و برگرداننده'کاراکتر جایگزین در هنگام رخداد خطای parity است. این کاراکتر در حالت معمول برابر ؟ است ولی شما میتوانید به هر کاراکتر دیگر آنرا set کنید ویا با مقدار دهی "" آنرا غیر فعال کنید .

رخداد Oncomm و خاصیت Commevent

در هنگام اجرای برنامه شما بایستی به رویدادهای مختلف یا خطاهای ایجاد شده توجه نمایید رخداد **Oncomm** و خاصیت **Commevent** به شما اجازه' بدام انداختن و چک کردن رخدادها و خطاهای گوناگون را میدهد. در هنگام رخ دادن خطا یک رویداد oncomm اجرا شده و مقدار خاصیت commevent تغییر میکند. بنابراین شما میتوانید با تست این خاصیت متوجه عملکرد برنامه شوید

. جدول زیر مقادیر **خطاها** در **commevent** نشان میدهد .

Constant	Value	Description
ComEventBreak	1001	A Break signal was received.
ComEventFrame	1004	Framing Error. The hardware detected a framing error.
ComEventOverrun	1006	Port Overrun. A character was not read from the hardware before the next character arrived and was lost.
ComEventRxOver	1008	Receive Buffer Overflow. There is no room in the receive buffer.
ComEventRxParity	1009	Parity Error. The hardware detected a parity error.
ComEventTxFull	1010	Transmit Buffer Full. The transmit buffer was full while trying to queue a character.
ComEventDCB	1011	Unexpected error retrieving Device Control Block (DCB) for the port.

جدول زیر مقادیر **رویدادها** را در **commevent** نشان میدهد .

Constant	Value	Description
ComEvSend	1	There are fewer than Sthreshold number of characters in the transmit buffer.
ComEvReceive	2	Received Rthreshold number of characters. This event is generated continuously until you use the Input property to remove the data from the receive buffer.
ComEvCTS	3	Change in Clear To Send line.
ComEvDSR	4	Change in Data Set Ready line. This event is only fired when DSR changes from 1 to 0.
ComEvCD	5	Change in Carrier Detect line.
ComEvRing	6	Ring detected. Some UARTs (universal asynchronous receiver-transmitters) may not support this event.
ComEvEOF	7	End Of File (ASCII character 26) character received.

شما با تست این مقادیر در هنگام رویداد oncomm میتوانید اطلاعات مفیدی در مورد اجرای برنامه کسب کنید.

ارتباط با مودم ----------

یکی از کاربردهای جالب پورت سریال استفاده از آن برای ارتباط با مودم میباشد مثلا میتوانید با آن شماره گیری کنید و یا اطلاعات را از طریق خط تلفن ارسال و یا دریافت نمایید. مثلا برای شماره گیری با آن میتوانید از این دستور استفاده کرد.

```
MSComm1.Output = "ATDT 0571-2642332" & vbcr
```

در این مثال **AT** اغازگر ارتباط است **D** نشاندهنده'dialing و**T** شماره گیری به روش تن است. در حالت ارتباط با با یک شبکه بایستی از یک کاراکتر برگشتی نیز استفاده نمود .اگر شماره گیری درست انجام شود برگشتی تابع **"ok"** است.

دیگر دستورات همراه **AT** به صورت زیر است .

D	شماره گیری (dialing)
T	استفاده از سیستم تن
P	استفاده از سیستم پالس
,	وقفه در شماره گیری
!	وقفه در شماره گیری به مقدار ۵. ثانیه
H	باعث Hang up مودم میشود

مثال : ارسال و دریافت یک فایل از طریق مودم

بخش فرستنده :

```
 Dim Offset As Long
 Dim FileData As Byte
 Dim FileName As String
Private Sub Form_Load()
    MSComm1.Settings = "28800,N,8,1"
    MSComm1.InputLen = 1
    MSComm1.CommPort = 2
    MSComm1.PortOpen = True
    Offset = 1
    Open FileName For Binary Access Read As #1
End Sub

Private Sub MSComm1_OnComm()
    If MSComm1.CommEvent = 2 Then
        temp = MSComm1.Input
        If Offset <= FileLen(FileName) Then
            Get #1, Offset, FileData
            q = FileData
            MSComm1.Output = Format(q, "000")
            Offset = Offset + 1
        End If
    End If
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Close #1
End Sub
```

بخش گیرنده :

```
Dim ByteCount As Long
Dim FileData As Byte

Private Sub Form_Load()
    MSComm1.Settings = "28800,N,8,1"
    MSComm1.InputLen = 3
    MSComm1.CommPort = 1
    MSComm1.PortOpen = True
    Open "c:\sample.txt" For Binary Access Write As #1
End Sub

Private Sub MSComm1_OnComm()
    If MSComm1.CommEvent = 2 And MSComm1.InBufferCount > 0 Then
        FileData = CInt(MSComm1.Input)
        ByteCount = ByteCount + 1
        Put #1, ByteCount, FileData
        MSComm1.Output = Chr$(26)
    End If
End Sub

Private Sub Command1_Click()
    MSComm1.Output = Chr$(26)
End Sub

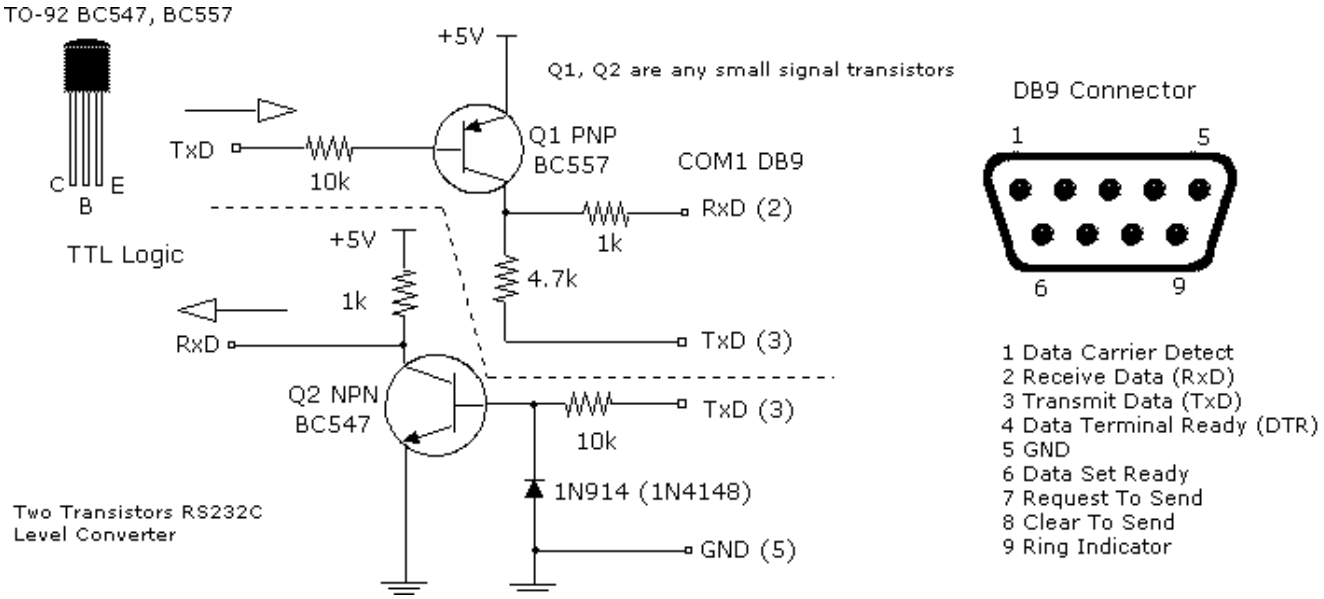
Private Sub Form_Unload(Cancel As Integer)
    Close #1
End Sub
```

توضیح پایه های پورت سریال

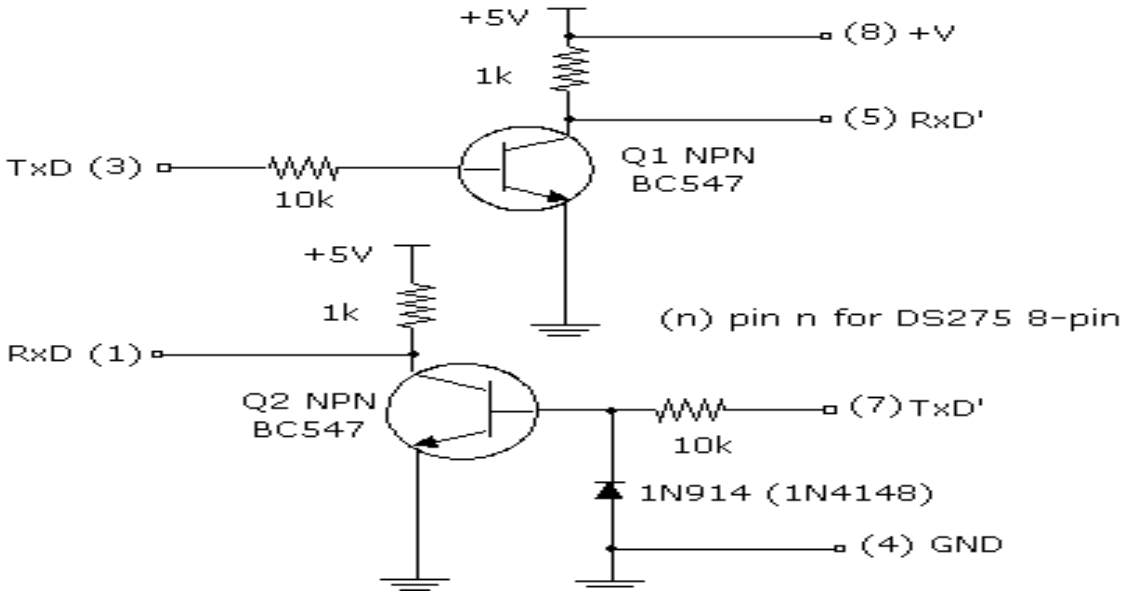
9-pin	25–pin	نام پایه	توضیح پایه
3	2	**TD** Transmit data	ارسال کننده' اطلاعات
2	3	**RD** Receive data	گیرنده' اطلاعات
7	4	**RTS** Request to send	آماده برای ارسال
8	5	**CTS** Clear to send	آماده' دریافت اطلاعات
6	6	**DSR** Data set ready	گیرنده متصل شده و آماده است
5	7	**GND** Signal ground	زمین
4	20	**DTR** Data terminal ready	فرستنده آماده است

ارتباط میکروکنترلر و کامپیوتر از طریق پورت سریال .

برای ارتباط با میکرو از طریق پورت سریال **RS۲۳۲** بایستی از یک رابط سخت افزاری برای تطابق سطح سیگنال**RS۲۳۲** با سطح **TTL** استفاده نمود برای این کار می توان از تراشه هایی مانند **MAX۲۳۲** یا **MAX۲۳۳** و یا **DS۲۷۵** استفاده نمود طرح ارائه شده در زیر را میتوان برای ارتباط میکرو با **RS۲۳۲** بکار برد .



مدار ارائه شده در زیر را نیز میتوان به راحتی جایگزین تراشه **DS۲۷۵** نمود



با کامل شدن ارتباط سخت افزاری میتوان شروع به ایجاد نرم افزار لازم نمود . برای ارتباط با پورت سریال کامپیوتر نیاز به تنظیم **BOUD RATE** و دیگر تنظیمات لازم داریم که توسط **میکرو** و **کامپیوتر** حمایت شده باشد

.برای تنظیم این مقدار در کامپیوتر با توجه به توضیحات قبلی بایستی:

```
MSComm1.Settings = "9600,N,8,1"
```

البته میتوان مقادیر **Parity bit - Data bit - Stop bit** را به مقدار دلخواه در **میکرو** و **کامپیوتر** set نمود..

برای تنظیمات لازم در میکرو کنترلر به مثال زیر توجه نمائید .

```
        ORG     00
        DB    1, 2 , 3, 4, 5, 0
        ORG     100
        MOV     TMOD,#20H       timer1 , mode 2 ( auto reload )
        MOV     TH1,#0FAH       4800 baud rate  ( with xtall = 11.0592 mhz )
        MOV     SCON,#50H       8 bit , 1 stop , ren enable
        SETB    TR1             run timer 1
        MOV     DPTR,#00H       load data pointer to transfer data
LL1:
        CLR     A
        MOVC    A,@A+DPTR       load ACC with data
        JZ      LL2             test for 0 data ( end data )
        CALL    SEND            call sender data function
        INC     DPTR            inc data poniter
        JMP     LL1
SEND:
        MOV     SBUF,A          load SBUF with data and send it
LL3:
        JNB     T1,LL3          wait for end send
        RET
LL2:
        END
```

مثال ارائه شده در بالا طرح ساده ای از ارسال چند عدد به کامپیوتر می باشد . همچنین شما میتوانید اطلاعات را به همین صورت به میکرو منتقل نما ئید .

برای آشنا ئی بیشتر با ارتباط سریال در میکرو کنترلر می توانید به کتابهای موجود در این زمینه و یا أدرس اینجانب مراجعه نمائید .

به امید سودمند بودن این مقاله و ارتقای روز افزون سطح تکنولوژی در کشورمان *ایران* .

بهزاد خزاما دانشجوی سبزواری

b_khazama@yahoo.com

www.khazama.com

TEL = 0571 - 2642332

آدرس : سبزوار – پارک بعثت – کوچه' طالقانی ۲۲ (شهید محسن صانعی)