

BREVE DESCRIZIONE SULL'USO DI VISUAL BASIC 6

Avviato Visual Basic prima di accedere all'ambiente di lavoro viene richiesto se si desidera aprire un nuovo progetto, uno già esistente o uno utilizzato recentemente. Evidentemente, non essendo ancora stato realizzato nessun progetto, si deve scegliere l'opzione *nuovo* (EXE standard), facendo poi clic sul pulsante *apri*. All'utente si presenta un ambiente grafico simile a quello riportato in fig. 1.

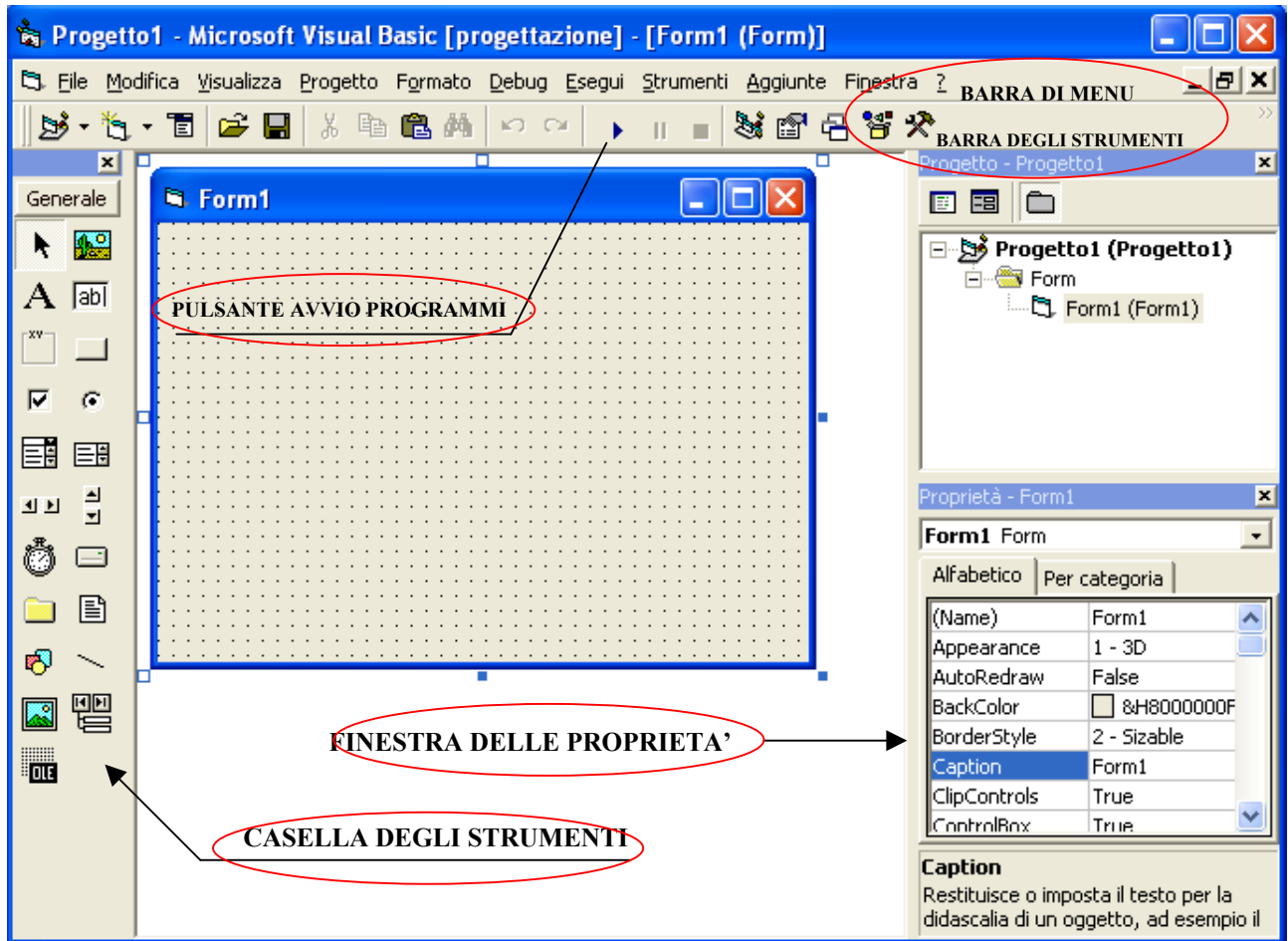


FIG. 1

Sulla sinistra della finestra compare una zona denominata **Casella degli strumenti**. In essa sono visualizzati, sotto forma di icone, gli oggetti disponibili.

Per inserire nuovi oggetti fare clic di destro sulla casella degli strumenti. Si apre una finestra che consente la selezione di nuovi oggetti.

L'icona a forma di punta di freccia (Fig.1) serve per manipolare gli oggetti che vengono posti sul *Form*. I **form** (nella fig.1, *form1*), sono degli oggetti, sui quali vengono posti altri oggetti, ovvero i vari controlli che dovranno essere utilizzati nelle applicazioni.

Alla destra del video è visibile la **finestra delle proprietà** degli oggetti selezionati (si ricordi che le proprietà sono gli attributi che descrivono gli oggetti). Nella figura sono elencate le proprietà dell'oggetto *form1*.

Selezionando una delle *proprietà*, nella parte bassa della finestra viene visualizzata una breve descrizione della stessa. Sopra la finestra delle proprietà è visibile una piccola finestra che descrive la struttura del progetto che si sta realizzando.

Per realizzare un'applicazione in VISUAL BASIC, **deve essere aperto un progetto che raggruppa i vari moduli utilizzati**. I singoli moduli si tradurranno in altrettanti file con appropriate estensioni. Così il file relativo all'intero progetto avrà estensione .vbp, quelli dei moduli dei *form* avranno l'estensione .frm, quelli dei moduli BASIC .bas, e quello relativo all'area di lavoro .vbw. Il file relativo a gruppi di progetto avrà estensione .vbg.

Nel caso più semplice un progetto avrà sempre almeno un *form* con il relativo file .frm e un file di progetto con estensione .vbp. Al progetto poi, a seconda del tipo di applicazione, possono essere aggiunti altri moduli.

Sulla parte alta del video è presente la **barra dei menu** e la **barra degli strumenti**.

Prima di descrivere l'esempio applicativo è bene precisare alcune azioni che vengono abitualmente compiute nell'ambiente grafico di VISUAL BASIC. Fare riferimento alla fig.1.

- Sulla finestra denominata *Form1*, vengono posti gli oggetti prelevati dalla *casella degli strumenti*, denominata *generale* (a sinistra del video).
- Per porre un oggetto su *Form 1* porre il cursore del mouse sull'oggetto e fare clic con il tasto sinistro su di esso. Spostare poi il cursore su *Form1*. il cursore cambia forma assumendo l'aspetto di una piccola croce. Tenendo premuto il tasto sinistro spostare il mouse sul piano di lavoro, fino a quando l'oggetto assume le dimensioni desiderate. Questa azione viene definita *trascinamento*.
- Un qualsiasi oggetto posto su *form1* è selezionato quando sul perimetro che lo delimita compaiono dei quadratini neri (■) (*maniglie di trascinamento*). *Form1* è selezionato quando sul suo perimetro compaiono le *maniglie di trascinamento*.
- Per selezionare un oggetto si deve fare clic con il pulsante sinistro del mouse sull'oggetto stesso. Se esso è selezionato, per togliere la selezione occorre fare clic in un punto qualsiasi della finestra non occupato da altri oggetti.
- Per spostare un oggetto si deve selezionarlo e poi, tenendo premuto il pulsante sinistro del mouse, trascinarlo nella posizione desiderata.
- Per variare le dimensioni di oggetti che presentano le maniglie di trascinamento, si deve porre il cursore su una delle maniglie e, quando esso assume la forma di doppia freccia (↔), si deve tener premuto il tasto sinistro del mouse e spostarlo sul piano di lavoro (*trascinamento*).
- Quando un oggetto posto su *Form1* è selezionato, a destra del video, sulla finestra *Proprietà*, compaiono le **proprietà che l'oggetto possiede**. La finestra *proprietà* è divisa in due colonne: quella a sinistra contiene tutte le proprietà dell'oggetto selezionato; quella a destra permette di modificarle. Se è selezionato *Form1* le proprietà visualizzate sono quelle del *form*. È possibile visualizzare le proprietà di tutti gli oggetti presenti sul *form* selezionandoli nell'apposito elenco a discesa, posto
 - sulla parte superiore della finestra delle *proprietà*, attivato con il pulsante di scorrimento (▼), oppure selezionando direttamente gli oggetti posti sul *form*.
 - Facendo clic con il pulsante destro del mouse si ottiene un menu con alcuni comandi. Esso varia a seconda della zona su cui si è fatto clic.

Interfaccia seriale EIA RS232-C

L'interfaccia seriale americana EIA RS232-C (EIA= Electronic Industries Associates), corrispondente alla V.24/V.28 dell'ITU-T, è uno standard di collegamento seriale che può essere di tipo sincrono o asincrono tra un dispositivo di comunicazione DCE come, ad esempio, il modem (Data Communication Equipment) e un dispositivo terminale DTE, come, ad esempio, il computer (Data Terminal Equipment) con velocità di trasmissione inferiore o uguale a 19.2Kbps (questo limite è oramai superato).

L'interfaccia è costituita da un insieme di 25 linee, non tutte indispensabili, che trasportano i bit di dati, segnali di controllo e la massa.

Nel collegamento tra un computer ed un dispositivo periferico vengono adoperati dei connettori miniatura tipo D a 25 poli. Sul DTE (computer, ad esempio) si trova la spina (connettore maschio) mentre sul DCE (modem) si trova la presa (connettore femmina). In alcuni DCE (ad esempio, il mouse seriale) manca la presa esterna poiché il cavo di collegamento entra direttamente nell'apparecchiatura.

I tipici dispositivi periferici che si possono collegare ad un computer via RS232 sono: stampante, modem, mouse, ecc.

Caratteristiche meccaniche ed elettriche della porta seriale RS232

In figura a) si mostra il connettore posto sul retro del computer per la RS-232C le cui caratteristiche meccaniche sono normalizzate secondo lo standard ISO 2110 della International Standard Organization. In molte applicazioni pratiche non si utilizzano tutte le linee ma solo una piccola parte di esse. In tal caso si fa uso di un connettore ridotto a 9 poli come quello in figura b).

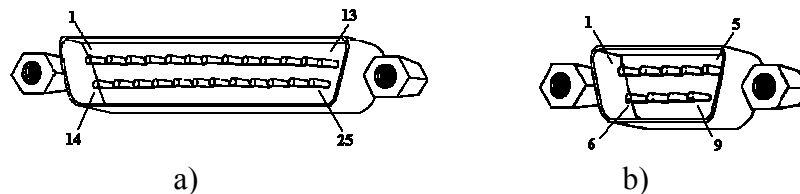
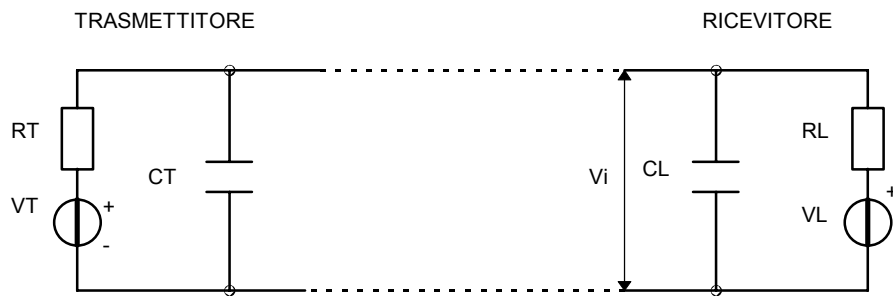


Fig. 2. Connettore per la RS-232C: a) di tipo a 25 poli; b) di tipo a 9 poli.

Il significato di tali linee è descritto nella Tabella 1.

Qualunque sia la linea (dati, clock o controlli), il circuito elettrico equivalente di tale interfaccia tra il trasmettitore e il ricevitore è quello indicato in figura. In tale standard si definisce *mark* la tensione V_i di valore inferiore a $-3V$ e si definisce *space* quella superiore a $+3V$. Durante la trasmissione si associa il livello logico 1 a *mark* e 0 a *space*. Si osserva subito che i livelli logici sono bipolari e in logica negativa (tensione positiva=0; tensione negativa=1). Tipicamente i valori di tensione assunti sono $\pm 12V$.



Circuito equivalente tra trasmettitore e ricevitore nello standard RS232-C.

V_T = f.e.m. del trasmettitore a circuito aperto;
 C_T = Capacità equivalente del trasmettitore;
 R_L = Resistenza d'ingresso del ricevitore;
 V_i = d.d.p. all'interfaccia.

R_T = Resistenza interna del trasmettitore;
 C_L = Capacità equivalente del ricevitore;
 V_L = f.c.e.m. del ricevitore a circuito aperto;

La resistenza di carico del ricevitore R_L deve essere compresa tra $3K\Omega$ e $7K\Omega$, la capacità C_L in parallelo al carico deve essere inferiore a $2.5nF$ e la f.c.e.m. V_L non deve superare i $2V$.

La f.e.m. V_T del driver del trasmettitore non deve superare i $25V$, R_T e C_T non sono specificati ma devono essere tali da evitare una corrente di corto circuito superiore a $0.5A$ e da consentire una V_i compresa tra $5V$ e $15V$.

Poiché la capacità per unità di lunghezza di un cavo è di circa $200pF/metro$ si evince che la massima distanza tra i dispositivi collegati in tale standard non deve superare i $12-15$ metri.

Tabella 1

PIN (9)	PIN (25)	NOME V.24 ITU	NOME RS-232	DESCRIZIONE
	1	C101	FG	Frame ground = Massa di protezione
3	2	C103	TxD	Transmitted data = Dati in trasmissione
2	3	C104	RxD	Received data = Dati in ricezione
7	4	C105	RTS	Request to send = Richiesta di trasmissione
8	5	C106	CTS	Clear to send = Pronto a trasmettere
6	6	C107	DSR	Data set ready = DCE pronto
5	7	C102	GND	Ground = Massa dei segnali
1	8	C109	DCD	Data carrier detector = Portante in ricezione presente
	9			Riservato per apparecchi di collaudo
	10			Riservato per apparecchi di collaudo
	11	C126	CK	Scelta frequenza in trasmissione
	12	C122	SCF	Segnale di ricezione presente sul canale ausiliario
	13	C121	SCB	Pronto per la trasmissione sul canale ausiliario
	14	C118	SBA	Dati in trasmissione del canale ausiliario
	15	C114	TC	Transmit clock = Clock di trasmissione dal modem
	16	C119	SBB	Dati in ricezione del canale ausiliario
	17	C115	RC	Received clock = Clock di ricezione
	18			Non connesso
	19	C120	SCA	Richiesta di trasmissione del canale ausiliario
4	20	C108	DTR	Data terminal ready = DTE pronto
	21	C110	CG	Rivelatore della qualità del segnale
9	22	C125	RI	Ring indicator = Chiamata in arrivo
	23	C111	CI	Selezione velocità di trasmissione da DTE
	24	C113	DA	Clock di trasmissione da DTE
	25			Non connesso

Caratteristiche funzionali dell'interfaccia ITU-T V.24

Si descrivono le caratteristiche funzionali delle linee (circuiti) dell'interfaccia V.24 esaminate nella precedente tab.1. Nella trasmissione dati tra un DTE e un DCE il numero di linee utilizzate dipende dal tipo di collegamento e dai modem impiegati.

C101= Massa di protezione.

Linea collegata alla massa dei segnali C102 all'interno del modem.

C102=Massa dei segnali.

Linea comune di riferimento per tutti i circuiti di interfaccia.

C103=Dati in trasmissione (DTE → DCE)

I dati binari in forma seriale generati dal DTE viaggiano verso il DCE. In assenza di trasmissione tale linea è nello stato MARK (tensione negativa).

C104=Dati in ricezione (DCE → DTE)

I dati binari in forma seriale generati dal DCE viaggiano verso il DTE.

C105=Richiesta di trasmissione (DTE → DCE)

Tale segnale obbliga il modem a trasmettere la portante in linea entro 2ms. Se la linea è nello stato di riposo il modem interrompe la trasmissione entro 2ms.

C106=Pronto a trasmettere (DCE → DTE)

Tale segnale indica che il DCE è pronto a trasmettere in linea e rappresenta la risposta alla linea C105 purché il modem sia connesso alla linea telefonica.

Il tempo di risposta in corrispondenza della prima richiesta di trasmettere è compreso tra 750 e 1400ms. ed è compreso tra 22 e 40ms. nelle successive richieste di trasmissione.

C107=Modem pronto (DCE → DTE)

Questo segnale indica al DTE che il modem è collegato alla linea telefonica e che non è in condizioni di prova. Nei modem in banda base tale linea è sempre attiva.

C108=DTE pronto (DTE → DCE)

L'attivazione di questa linea provoca il collegamento del modem alla linea telefonica qualunque sia la condizione delle altre linee. Questa linea, inoltre, indica che il DTE è pronto sia a trasmettere che a ricevere i dati.

C109=Portante in ricezione (DCE → DTE)

Il DCE informa il DTE che la portante in linea è ad un livello superiore alla soglia di ricezione.

C110=Rivelatore della qualità del segnale (DCE → DTE)

Il DCE informa il DTE che non ci sono ragioni per credere che i dati ricevuti siano errati.

C111=Selezione di velocità (DTE → DCE)

L'attivazione di tale linea obbliga il modem a scegliere la velocità più elevata. Il livello logico opposto imposta la velocità più bassa (per modem 1200bps / 600bps).

C112=Selezione velocità (DCE → DTE)

Se la scelta della velocità è fatta dal modem, quest'ultimo avverte il DTE della velocità selezionata : lo stato ON sceglie la velocità più alta, lo stato OFF quella più bassa. Questa linea utilizza lo stesso pin del C111 ed è ad esso alternativa.

C113=Clock di trasmissione da DTE (DTE → DCE)

Questa linea rappresenta il clock generato dal DTE. Le transizioni ON-OFF devono coincidere con la posizione centrale su ciascun bit della linea C103.

C114=Clock di trasmissione da DCE (DCE → DTE)

Questa linea rappresenta il clock generato dal DCE. Durante le transizioni OFF-ON il DTE deve generare il bit successivo su C103.

C115=Clock in ricezione (DCE → DTE)

Questa linea rappresenta il clock per i dati che il DTE riceve su C104. La forma d'onda è quadra e la transizione ON-OFF indica il centro del bit ricevuto su C104.

C118=Trasmissione dati sul supervisore (DTE → DCE)

Questa linea svolge le stesse funzioni del C103 ma si riferisce al canale supervisore. Per i modem a 1200bps il canale supervisore è a 75 bps.

C119=Ricezione dati sul supervisore (DCE → DTE)

Questa linea svolge le stesse funzioni del C104 ma si riferisce al canale supervisore. Per i modem a 1200bps il canale supervisore è a 75 bps.

C120=Richiesta di trasmissione sul canale supervisore (DTE → DCE)

Questa linea svolge le stesse funzioni del C105 ma si riferisce al canale supervisore.

C121=Pronto a trasmettere sul canale supervisore (DCE → DTE)

Questa linea svolge le stesse funzioni del C106 ma si riferisce al canale supervisore.

C122=Rivelatore segnale di linea dati supervisore (DCE → DTE)

Questa linea svolge le stesse funzioni del C109 ma si riferisce al canale supervisore.

C123=Rivelazione qualità segnale (DCE → DTE)

Questa linea svolge le stesse funzioni del C110 ma si riferisce al canale supervisore.

C125=Chiamata in arrivo (DCE → DTE)

L'attivazione di questa linea indica al DTE che il modem ha ricevuto una chiamata telefonica. La risposta del DTE sulla linea C108 provoca la connessione del modem alla linea (risposta automatica).

C126=Scelta della frequenza in trasmissione (DTE → DCE)

Linea che serve alla scelta della frequenza del modem quando questo prevede due canali.

Le linee dei segnali sono dette *sbilanciate* perché sono tutte riferite ad una massa comune (circuito C102).

Trasmissione seriale

Nella trasmissione seriale il collegamento tra trasmettitore e ricevitore si può realizzare con un minimo di due fili: il primo rappresenta la linea su cui viaggiano i bit, l'altro è il filo della massa. Le trasmissioni seriali si dividono in:

- 1) Sincrone;
- 2) Asincrone.

Nelle prime il trasmettitore invia degli impulsi di clock contemporaneamente ai bit di informazione in modo da consentire al ricevitore la corretta lettura dei bit di informazione in modo da consentire al ricevitore la corretta lettura dei dati in arrivo ad intervalli regolari di tempo scanditi dal trasmettitore. Il collegamento si realizza con tre fili (clock, bit e massa) come si mostra nella fig.3.

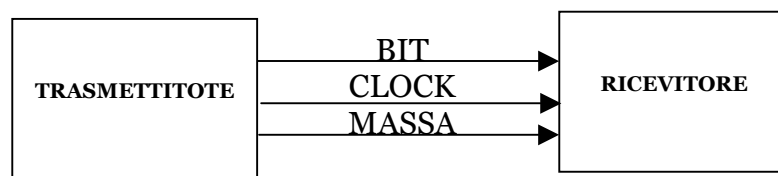


Fig. 3 – Schema a blocchi di una trasmissione seriale sincrona

Nel collegamento seriale asincrono non si trasmette il clock ma il ricevitore genera un clock locale della stessa frequenza. Affinché i due clock risultino in fase occorre che il ricevitore sappia quando ha inizio la trasmissione di un carattere in modo da sincronizzare la lettura dei vari bit. In pratica, un carattere di trasmissione è preceduto da un bit di start e seguito da uno o più bit di stop. Il bit di start è costituito dal livello logico 0 mentre il bit di stop dal livello logico 1. In assenza di trasmissione si ha il livello logico 1; quando la trasmissione ha inizio, l'applicazione del bit di start genera un fronte negativo che sincronizza, ad intervalli regolari di tempo, il bit del carattere da trasmettere nel codice ASCII a 7 o 8 bit di seguito, eventualmente, da un bit di parità pari o dispari e da uno o due bit di stop. Le modalità di trasmissione, ovviamente, devono essere noti prima che questa sia attivata. Si definisce velocità di trasmissione e si indica con *baut rate* il numero di bit trasmessi in un secondo. Un sistema che trasmette a 9600 baut con 8bit per carattere senza alcun bit di parità ed un solo bit di stop può trasferire fino a 960 bit di dato, 1 bit di stop. In fig. 4 si mostra l'analisi temporale della trasmissione del byte 6B.

Dal punto di vista hardware il trasmettitore presenta nello stadio di uscita un circuito in grado di effettuare la conversione di un carattere dalla forma parallela e quella seriale in modo del tutto simile ad un registro a scorrimento con caricamento parallelo ed uscita seriale (PISO). Il ricevitore, invece, presenta nello stadio di ingresso un circuito in grado di trasformare un carattere ricevuto in forma seriale in parallelo. Ciò si realizza con una soluzione simile ad un registro a scorrimento con caricamento seriale e uscita parallela (SIPO).

Vi sono circuiti integrati in grado di comportarsi, all'occorrenza, sia da SIPO che da PISO, presentano un clock locale che può essere selezionato sul valore richiesto. Essi prendono il nome di USART (Universal Synchronous Asynchronous Receiver Transmitter = ricevitore

trasmettitore sincrono o asincrono universale) e sono utilizzati sia nei ricevitori che nei trasmettitori. Gli USART risultano convenienti nelle trasmissioni bidirezionali dove i ruoli tra trasmettitore e ricevitore sono intercambiabili. Le velocità permesse nelle trasmissioni seriali asincrone hanno i seguenti valori espressi in baud:

75, 110, 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600.

Collegamento null-modem

Per collegare tra di loro due computer con l'interfaccia seriale RS-232 si deve preparare un cavo, noto come cavo **null modem**, con almeno tre fili (RxD, TxD e GND) e due connettori a 25 o a 9 poli femmine da porre alle due estremità del cavo. Occorre, però, fornire i corretti potenziali alle linee di ingresso che non si intendono utilizzare.

In fig.5, si mostra un tipico collegamento ove il pin 5 (linea di ingresso CTS) è pilotato dallo stesso DTE attraverso il pin 4 (linea di uscita RTS). Analogamente la linea di uscita DTR (pin 20) va collegata ai pin 6 e 8 di ingresso (DSR e DCD, rispettivamente). Ciò vale per entrambi i computer.

In questo modo si "imbrogia" il computer ma si deve rinunciare al collegamento in *handshake*. La tecnica dell'*handshake* (letteralmente stretta di mano) consiste nel sincronizzare il collegamento attraverso due linee di controllo: il trasmettitore invia un livello logico di richiesta di invio (uscita RTS, pin 4) ed il ricevitore risponde con un livello logico di pronto a trasmettere (ingresso CTS pin 5); in pratica si autorizza il trasmettitore ad inviare i dati.

L'imbroglio (collegamento pin 4 e 5 dello stesso connettore) consiste nel fatto che il trasmettitore "autorizza se stesso" ad inviare dati.

Il collegamento dei pin 6, 8 e 20 dello stesso connettore realizza un "imbroglio" simile.

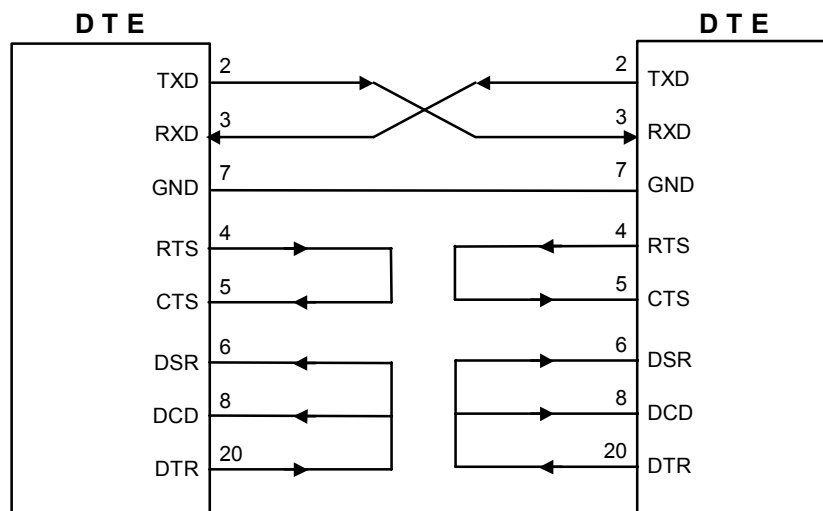


Fig. 5 - Collegamento tra due computer via RS-232C con cavo NULL-MODEM.

Volendo un collegamento con handshake si devono utilizzare almeno 5 fili. In fig.6 si mostra un possibile collegamento.

I due computer, essendo collegati direttamente e non via modem, possono comunicare ad elevata velocità.

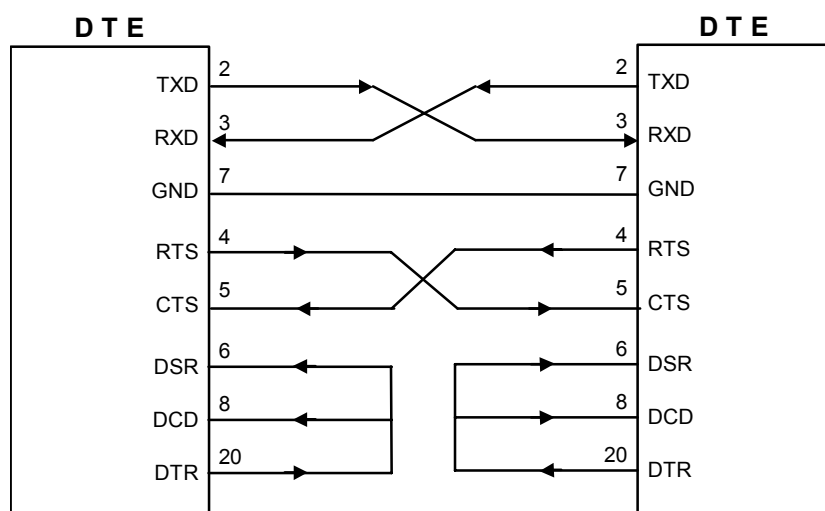
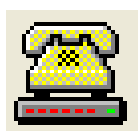


Fig.6 - Collegamento tra due computer via RS-232C con cavo NULL-MODEM con possibilità di handshake.

COLLEGAMENTO TRA DUE PC CON INTERFACCIA SERIALE RS232 E SOFTWARE DI GESTIONE IN VISUAL BASIC 6

Nel seguito sarà descritto il software per il collegamento di due PC con seriale RS 232-C. Si deve preparare un cavo con due connettori femmine con un collegamento null-modem come mostrato nella precedente figura (fig.5, fig.6). Su un computer si dovrà scrivere il software per la trasmissione e nell'altro quello per la ricezione. Nell'esempio si mostrerà la semplice possibilità di trasmettere un testo. Collegati i due PC si potranno in esecuzione i due programmi. Scrivendo del testo sul PC di trasmissione esso sarà trasferito su quello in ricezione.



Per la realizzazione della rete con l' RS-232 ci serviamo del programma Visual Basic 6, al cui interno, viene fornito un componente per la gestione seriale, l'**MSCOMM.OCX**, presente nella figura a sinistra. Il componente deve essere inserito nella casella degli strumenti. Fare clic di destro su tale casella e attivare la scelta relativa a "**MICROSOFT COMM CONTROLL**". Successivamente tale componente deve essere inserito sul form di lavoro.

L'**MSCOMM** dispone di diverse proprietà, di un solo evento (**OnComm()**) e nessun metodo.

SOFTWARE PER LA TRASMISSIONE



In figura si mostra la schermata relativa al form del PC trasmettitore. Nella TextBox si scrive il testo da trasmettere che, in tempo reale, è trasferito al PC ricevente.

La parte fondamentale del software riguarda le impostazioni del controllo MSCOMM attivata al caricamento del form.

Si decide di impostare la porta seriale con 9600 bit/s (*velocità di trasferimento*), n (*nessuna parità*), 8 (*il numero di Bit del carattere*), 1 (*1 bit di stop*).

Ovviamente il PC ricevitore deve avere le stesse impostazioni.

```
Private Sub Form_Load()
MSComm1.CommPort = 2 ' Porta COM2
MSComm1.Settings = "9600,n,8,1"
On Error Resume Next
MSComm1.PortOpen = True
If Err Then
MsgBox "Impossibile aprire la porta" & MSComm1.CommPort & vbCrLf & Error$
End If
End Sub
```

Impostiamo anche *PortOpen* a *True* come rilevazioni di errori, qualora la porta *COM* non fosse disponibile. Se non lo fosse, apparirebbe un errore con l'istruzione *OnError*.

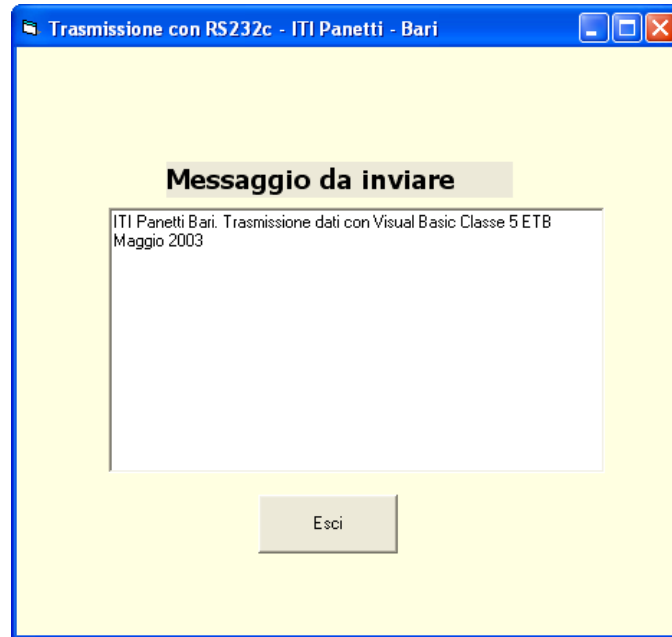
Per chiudere la porta e rilasciarne il controllo, è sufficiente impostare *PortOpen* a *False*. In tutti i casi quando si chiude il programma automaticamente viene chiusa la porta seriale. La Subroutine della TextBox consente di inviare il singolo carattere nel codice ASCII al computer ricevitore, nel momento stesso in cui il carattere è digitato.

```
Private Sub txtMessaggio_KeyPress(KeyAscii As Integer)
MSComm1.Output = Chr$(KeyAscii)
End Sub
```

Infine impostiamo il *CommandButton* "Esci", in modo che al clic del mouse, cessi l'invio del testo e si chiude la trasmissione.

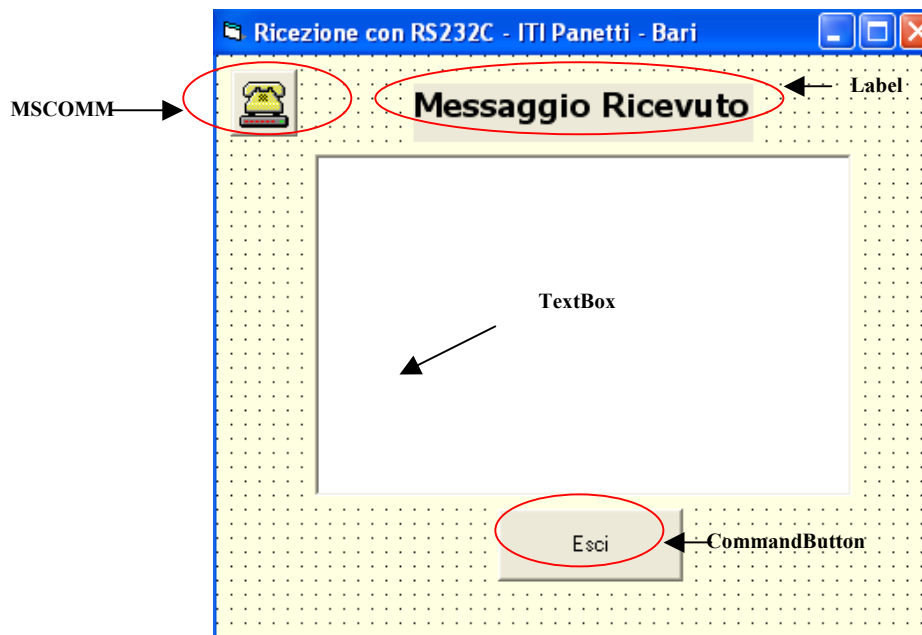
```
Private Sub cmdEsci_Click()
End
End Sub
```

Si mostra la schermata del programma in esecuzione.



SOFTWARE PER LA RICEZIONE

Nella figura si mostra la schermata relativa al form del PC utilizzato come ricevitore.



Il programma è quasi identico a quello della trasmissione, vanno solo sostituiti alcuni parametri. Al caricamento del form si attiva la seguente procedura. In particolare la proprietà `RTHRESHOLD` posta a 1 informa il controllo `MSCOMM` che è stato ricevuto un carattere. Viene attivato l'evento `OnComm ()`

Aggiungiamo al *Form* un controllo testo denominato `txtRicezione` con proprietà *MultiLine a True* in modo da avere il testo su più righe.

Ricevuto il carattere si scatena l'evento OnComm().

```
Private Sub MSComm1_OnComm()  
Dim Messaggio$  
Messaggio$ = MSComm1.Input 'Legge il buffer di ricezione  
If Len(Messaggio$) Then  
txtRicezione = txtRicezione & Messaggio$  
End If  
End Sub  
,
```

Infine aggiungiamo il Comando per il pulsante Esci:

```
Private Sub cmdEsci_Click()  
End  
End Sub
```

Si mostra la schermata del programma in ricezione quando è in esecuzione.



SOFTWARE COMPLETO PER LA RICEZIONE E L'INVIO DI MESSAGGI

È possibile configurare ciascun PC sia da trasmettitore che da ricevitore. In tal modo si realizza un collegamento Full-Duplex.

Su ciascun PC si deve installare il software per la gestione della trasmissione e della ricezione. Il software è ottenuto da una sorta di fusione dei programmi descritti nei paragrafi precedenti. Si riporta lo schermata del form del programma per la rice-trasmissione con RS232C e il relativo software.



```
Private Sub cmdEsci_Click()
Dim uscita As String
uscita = MsgBox("Sei sicuro?", vbInformation + vbOKCancel, "STAI USCENDO")

If uscita = vbOK Then: MSComm1.PortOpen = False: End
End Sub

Private Sub Form_Load()
MSComm1.CommPort = 2 ' Seleziona COM2
MSComm1.Settings = "115200,E,8,1" ' Imposta RS232
On Error Resume Next
MSComm1.PortOpen = True
If Err Then
MsgBox "Impostazione errata della porta COM " & MSComm1.CommPort & vbCrLf & Error$, vbCritical, "Attenzione"
End If
MSComm1.RThreshold = 1
End Sub

Private Sub MSComm1_OnComm()
Dim Messaggio$
Messaggio$ = MSComm1.Input 'Legge il buffer di ricezione
If Len(Messaggio$) Then
txtMessaggio.Text = txtMessaggio.Text & Messaggio$
End If
End Sub
```

```
Private Sub txtMessaggio_KeyPress(KeyAscii As Integer)
carattere = Chr$(KeyAscii)
'Codice ASCII del carattere
asciicarattere = KeyAscii
'Controlla se si preme il tasto INVIO (ASCII 13)
'In questo caso viene inviato al ricevitore
'un INVIO (ASCII 13) e un Line Feed (ASCII 10)
If asciicarattere = 13 Then
MSComm1.Output = Chr(13) & Chr(10)
End If
MSComm1.Output = carattere
End Sub
```

Nel programma è stato modificato la routine relativa al pulsante ESCI e quella relativa alla trasmissione del messaggio.

I codici ASCII 13 e ASCII 10 consentono il ritorno a capo del testo inviato quando si preme il tasto INVIO.