



Quality in Electronic
Manufacturing

www.qem.it

QEM



Q-PAINT 1.0

Documentazione tecnica D1-Qpaint 1.0

Versione 1.2 - Gennaio 2003

QEM® e QMOVE® sono marchi registrati.

Il presente manuale è pubblicato dalla QEM srl senza alcun tipo di garanzia e si riserva di apportare modifiche ad errori tipografici, imprecisioni nei contenuti e miglioramenti (anche ai prodotti cui il presente manuale fa riferimento). Le eventuali modifiche saranno comunque inserite nelle edizioni successive di questo manuale.

Nessuna parte di questo manuale può essere riprodotta, indipendentemente dal formato e dal mezzo, senza autorizzazione scritta della QEM srl.

È fatta riserva di tutti i diritti.

QEM srl
S.S. 11, Km 339
Località Signolo
36054 Montebello Vic.
Vicenza - Italy
Tel. 0444 440061 r.a.
Fax 0444 440229
e-mail: info@qem.it
www.qem.it

Sommario

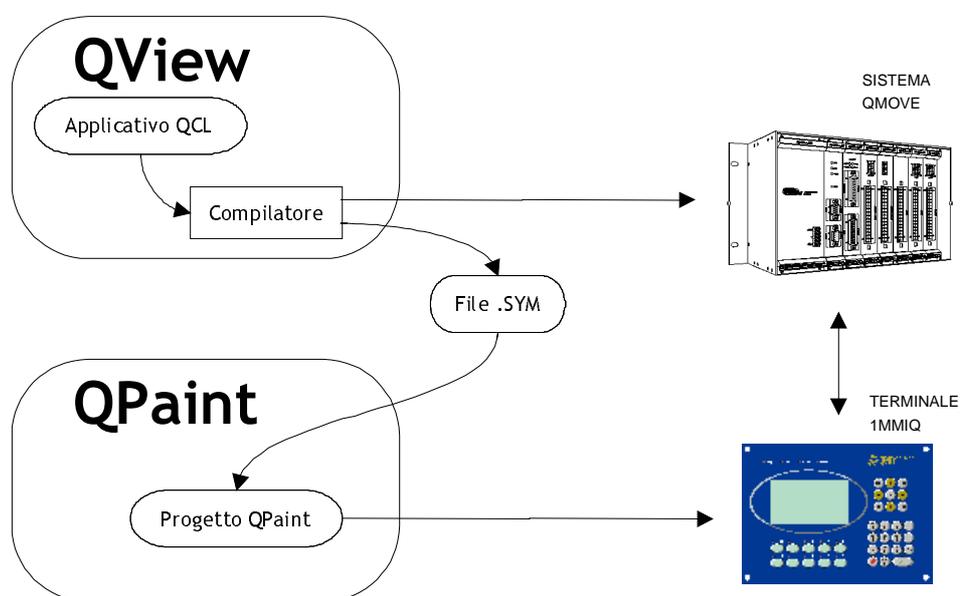
Introduzione	4
Installazione	5
Avvio del programma	6
Caratteristiche del programma	8
Descrizione del programma	9
Barra dei menù	9
Barra di gestione della pagina	45
Barra di impostazione della password di pagina	46
Barra delle funzioni di progetto	47
Barra delle funzioni di disegno	48
Barra degli strumenti di disegno	48
I font del terminale	57
Le variabili di riferimento	58
Appendice	59
Tabelle riassuntive regole variabile di riferimento	59
Stati funzionali visualizzati tramite interfaccia standard	61
Esempi applicativi Qpaint 1.0	70
Scroll personalizzato pagine di visualizzazione	70
Scelta della lingua	74
Cambio pagina forzato da QCL	76
Introduzione con diverse unità di misura	77
Introduzione dati guidata	79
Segnalazione di batteria	85
Controllo comunicazione attiva	87
Introduzione alfanumerica	90
Gestione eventi temporali	95
Pagina di presentazione	99
Animazione	106
Backup e Restore	117

Introduzione

QPaint 1.0 è un ambiente di sviluppo grafico per la programmazione del terminale 1MMIQ. In questo documento sono riportate le caratteristiche principali del programma QPaint. Esso è stato progettato per lavorare in stretta relazione con un sistema QMove e per questo motivo durante la descrizione del prodotto si faranno molti riferimenti ai concetti della programmazione in QCL.

Il terminale è un dispositivo in grado di comunicare con sistemi QMove configurati con applicativi diversi. Per questo esso deve conoscere le informazioni relative a ciascuna delle variabili dell'applicativo QMove con il quale dovrà colloquiare. Tali informazioni si trovano in un particolare file (con estensione SYM) generato dalla compilazione dell'applicativo QMove utilizzando l'ambiente di sviluppo Qview. QPaint utilizza queste informazioni che una volta trasferite al terminale rendono possibile la connessione con il sistema QMove.

I cablaggi e collegamenti dell'apparecchiatura descritta nel presente manuale, devono rispettare la Normativa Europea EN60204-1.



Con i termini "oggetto" ed "oggetto grafico" ci si riferisce alla stessa cosa. Con il termine "font" si intende font terminale e/o font utente. Con il termine "font terminale" o "font interno" si intende il font memorizzato nella ROM del terminale. Con il termine "font utente" si intende il font creato dall'utente e memorizzato nei file QTF e FNT. Con il termine "font file" si intendono i file QTF e FNT in cui sono memorizzate le immagini dei font terminale ed utente.

Installazione

L'installazione, come per tutti i recenti prodotti QEM, é stata affidata al programma InstallShield 5.0. Essa si occupa di creare una directory in cui depositare i files del programma, creare una cartella in cui porre le varie icone di collegamento, preparare le operazioni per la disinstallazione del pacchetto e associare il programma QPaint10.EXE ai file con estensione QTP. Informazioni piú dettagliate inerenti l'installazione sono fornite nel file README.TXT presente nel primo floppy disk del pacchetto. A fine installazione avremo per default in C:\Programmi la directory QPaint10 cosí formata:

C:\Programmi\QPaint10\		
qpaint10	exe	eseguibile
readme	txt	documento
uninst	isu	informazioni per disinstallazione
C:\Programmi\QPaint10\Help		<directory lasciata volutamente vuota>
C:\Programmi\QPaint10\Examples		
tutorial	qtp	progetto esempio uso QPaint
C:\Programmi\QPaint10\Fonts		
qem system large font	bmp	file bitmap sorgente utilizzato per creare il font
qem system large font	fnt	font di Windows® per QPaint
qem system large font	qtf	font terminale per QPaint
qem system medium font	bmp	file bitmap sorgente utilizzato per creare il font
qem system medium font	fnt	font di Windows® per QPaint
qem system medium font	qtf	font terminale per QPaint
qem system small font	bmp	file bitmap sorgente utilizzato per creare il font
qem system small font	fnt	font di Windows® per QPaint
qem system small font	qtf	font terminale per QPaint

L'associazione dell'estensione QTP al programma QPaint10.EXE permetterà di avviare un progetto, tramite il doppio clic sull'icona dello stesso, direttamente dalla finestra di Explorer (Gestione Risorse nella versione italiana).

La finestra di Logo, comunemente chiamata splash screen, ha lo scopo di intrattenere l'utente durante le operazioni di avvio del programma. Il tempo di permanenza va da un minimo fisso di 3 secondi ad un massimo che varia in funzione delle caratteristiche hardware del PC utilizzato e dal numero di font da caricare. Inoltre se viene specificato nella linea di comando anche il nome di un progetto, al normale tempo di avvio, viene aggiunto anche il tempo di caricamento del progetto stesso.

Caratteristiche del programma

Nella fase di stesura delle specifiche di programma si é cercato di delineare quali fossero le caratteristiche funzionali indispensabili per rendere il QPaint un prodotto valido ed efficiente. Grazie all'esperienza accumulata nel tempo, per mezzo di prodotti come QView, si é giunti a definire la seguente lista di argomenti:

1. Eventuali nuove versioni del programma dovranno assicurare la piena compatibilit  dei progetti sviluppati con le versioni precedenti. Per ottenere ci  si é adottato un nuovo standard di memorizzazione delle informazioni in protocollo binario con cui risulta molto semplice gestire la compatibilit  di versione.
2. Per rendere portabile e facilmente mantenibile un progetto non vi dovranno essere collegamenti a file esterni. Lo standard di memorizzazione adottato permette di archiviare, nello stesso file, dati di qualsiasi formato come ad esempio testi, immagini grafiche, data base, etc. Il programma QPaint memorizza nel file di progetto QTP tutto ci  di cui abbisogna. In questo modo per archiviare, trasportare su floppy o distribuire un progetto, sar  sufficiente operare con il solo file QTP.

Per limitare significativamente la dimensione dei file di progetto si é ritenuto utile non inserire nello stesso tutte le informazioni dei font utilizzati, perci  bisogna assicurarsi che nella directory C:\Programmi\QPaint10\Fonts del proprio PC siano presenti tutti i font file QTP richiesti. I tre font distribuiti con il programma QPaint, QEM System Small, Medium e Large Font, corrispondono ai font interni del terminale per cui sono sempre disponibili ed utilizzabili. Ulteriori informazioni sui font sono disponibili nella sezione dedicata.

3. L'interfacciamento grafico all'utente dovr  essere semplice ed intuitivo, rispettando soprattutto le convenzioni e gli standards dettati dai pi  diffusi programmi per Windows[®]. Un'analisi approfondita di prodotti Windows[®] di largo consumo e l'uso attento degli oggetti di costruzione standard ha reso il QPaint un programma dall'interfaccia utente immediata ed intuitiva.
4. Il programma dovr  rappresentare con la maggior corrispondenza possibile il risultato finale nel terminale. In questo caso si potrebbe parlare di " What You See is What You Get ", ci  che vedi é ci  che ottieni. L'uso delle capacit  grafiche fornite da Windows[®] combinate con gli algoritmi utilizzati nel terminale ha fatto s  che quanto visualizzato sul monitor sia perfettamente corrispondente a quello che apparir  nel display del terminale. Funzionalit  grafiche aggiuntive come zoom dell'immagine e cambio del colore di sfondo permettono inoltre di semplificare le operazioni di gestione del progetto.
5. La funzionalit  del programma non dovr  essere vincolata dalle dimensioni dell'attuale display terminale. Attualmente il terminale utilizza un display grafico monocromatico da 240x128 pixel. In futuro vi potrebbero essere nuove versioni con altri modelli di display. Il programma dovr  poter essere facilmente modificato per lavorare con ognuno di essi.

Descrizione del programma

Come si può vedere nella fig. 1, il desktop di QPaint é composto dalle seguenti componenti:

1. Barra dei menú.
2. Barra di gestione della pagina.
3. Barra di impostazione della password di pagina.
4. Barra delle funzioni di progetto.
5. Barra delle funzioni di disegno.
6. Barra degli strumenti di disegno.
7. Barra di stato.
8. Area di disegno rappresentante il display del terminale.

Barra dei menú

Tramite la barra dei menú é possibile accedere a gran parte delle operazioni disponibili in QPaint. Segue ora una descrizione dettagliata delle funzionalità fornite dalle voci presenti nei vari menú.

File New Project

Con questa voce di menú si richiede a QPaint di abbandonare il progetto eventualmente attivo per crearne uno di nuovo. Se durante la creazione del nuovo progetto si riscontra che quello precedente é stato modificato verrà visualizzata una finestra di dialogo con il testo "Save changes to *nome file* ?" e i bottoni "Yes", "No" e "Cancel". Se si conferma l'operazione con "Yes", verrà automaticamente richiamata la finestra di dialogo "Save Project" o "Save Project As...". La pressione di "No" farà proseguire l'operazione senza salvare niente, mentre "Cancel" abortirà l'intera operazione. Il nuovo progetto avrà per default il nome "Untitled.QTP" ed il suo percorso di memorizzazione sarà quello della directory corrente.

File Open Project...

Con questa voce di menú si richiede a QPaint di aprire un progetto precedentemente salvato. Per far ciò viene richiesto di segnalare la posizione ed il nome del progetto tramite l'utilizzo della finestra di dialogo standard di Windows® per l'apertura dei files. Sul display comparirà la seguente finestra di dialogo:

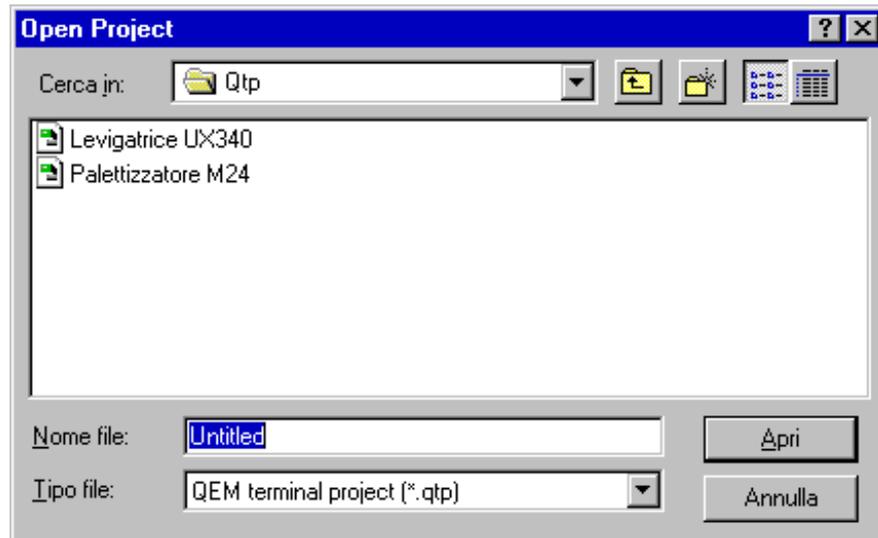


Figura 2 : finestra di dialogo Open Project

Le funzionalità espletate dalla finestra di dialogo sono quelle standard normalmente fornite dallo stesso sistema operativo Windows®. In "Nome file:" comparirà il nome del progetto attualmente in uso. A conferma dell'operazione con il tasto "Apri" verrà caricato il progetto specificato. Se durante il caricamento del progetto si riscontra che quello precedente é stato modificato verrà visualizzata una finestra di dialogo con il testo "Save changes to *nome file* ?" e i bottoni "Yes", "No" e "Cancel". Se si conferma l'operazione con "Yes", verrà automaticamente richiamata la finestra di dialogo "Save Project" o " Save Project As...". La pressione di "No" farà proseguire l'operazione senza salvare niente, mentre "Cancel" abortirà l'intera operazione.

File Save Project

Con questa voce di menù si richiede a QPaint di salvare il progetto attualmente in uso. Se il progetto non era mai stato salvato verrà automaticamente richiamata la finestra di dialogo "Save Project As...".

File Save Project As...

Con questa voce di menù si richiede a QPaint di salvare il progetto attualmente in uso. Per far ciò viene richiesto di segnalare la posizione ed il nome del progetto tramite l'utilizzo della finestra di dialogo standard di Windows® per il salvataggio di files. Sul display comparirà la seguente finestra di dialogo:

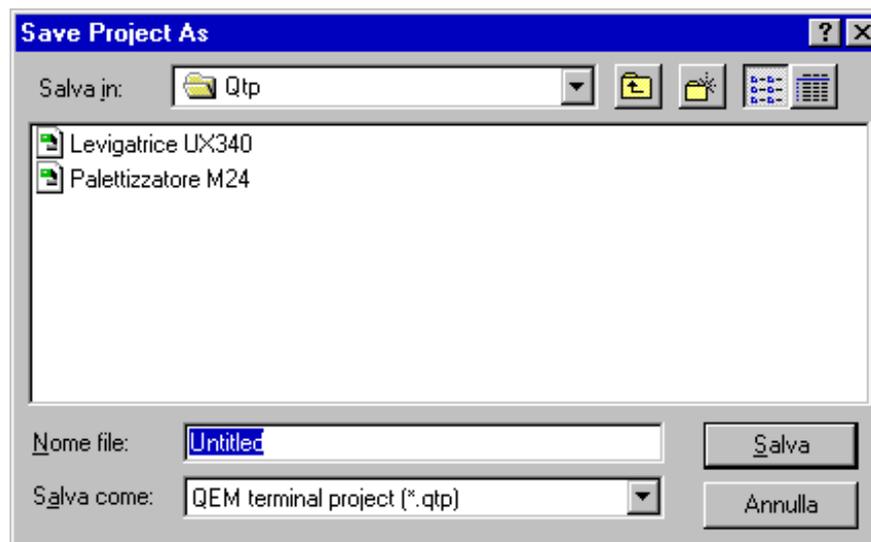


Figura 3 : finestra di dialogo Save Project As

Le funzionalità espletate dalla finestra di dialogo sono quelle standard normalmente fornite dallo stesso sistema operativo Windows®. In "Nome file:" comparirà il nome del progetto attualmente in uso. A conferma dell'operazione con il tasto "Salva" il progetto specificato verrà salvato.

File Recent Files 1, 2, 3 e 4

QPaint mantiene traccia degli ultimi quattro progetti aperti mettendo il loro nome, completo di percorso nella barra dei menù subito sotto alla voce "File | Save Project As...". Tale operazione è chiamata gestione dei "Recent Files" e segue le stesse modalità di funzionamento adottate da tutti i programmi Windows® che la implementano. Selezionando e confermando, con il mouse o la tastiera, un recent file, si richiede a QPaint di riaprire il progetto associato. I recent files sono utili per velocizzare le operazioni di apertura dei progetti usati più frequentemente o ancora in fase di sviluppo. Nel caso si tenti di aprire un recent file che non esiste più o che è memorizzato su un disco non disponibile apparirà il messaggio "Error in opening file "nome file" !" e i bottoni "OK" e "Cancel". Alla pressione di "OK" QPaint ritenterà l'apertura del file, mentre "Cancel" abortirà l'intera operazione. Prima di aprire il recent file selezionato, se si riscontra che il progetto attualmente in uso è stato modificato, verrà visualizzata una finestra di dialogo con il testo "Save changes to "nome file" ?" e i bottoni "Yes", "No" e "Cancel". Se si conferma l'operazione con "Yes", verrà automaticamente richiamata la finestra di dialogo "Save Project" o "Save Project As...". La pressione di "No" farà proseguire il programma senza salvare niente, mentre "Cancel" abortirà l'intera operazione.

File Exit

Con questa voce di menù si richiede a QPaint di concludere la propria esecuzione. Se si riscontra che il progetto attualmente in uso è stato modificato verrà visualizzata una finestra di dialogo con il testo "Save changes to "nome file" ?" e i bottoni "Yes", "No" e "Cancel". Se si conferma l'operazione con "Yes", verrà automaticamente richiamata la finestra di dialogo "Save Project" o "Save Project As...". La pressione di "No" terminerà il programma senza salvare niente, mentre "Cancel" abortirà l'intera operazione.

Edit Delete Object

Con questa voce di menú si richiede a QPaint di cancellare l'oggetto selezionato nell'area di disegno. A termine della cancellazione, se vi sono ancora oggetti a disposizione, verrà automaticamente selezionato l'oggetto successivo.

Condizioni di attivazione: un oggetto selezionato nell'area di disegno.

Edit Property Object

Con questa voce di menú si richiede a QPaint di visualizzare l'editor delle proprietà dell'oggetto selezionato nell'area di disegno. Ulteriori informazioni sugli oggetti e i relativi editor delle proprietà sono disponibili nella sezione dedicata.

Condizioni di attivazione: un oggetto selezionato nell'area di disegno.

Edit Bring Up Object

Con questa voce di menú si richiede a QPaint di portare ad un livello superiore di vista l'oggetto selezionato nell'area di disegno. Ulteriori informazioni sugli oggetti e le loro caratteristiche di disegno sono disponibili nella sezione dedicata.

Condizioni di attivazione: un oggetto selezionato nell'area di disegno di livello superiore a 1/n ed almeno due oggetti nella pagina.

Edit Bring Down Object

Con questa voce di menú si richiede a QPaint di portare ad un livello inferiore di vista l'oggetto selezionato nell'area di disegno. Ulteriori informazioni sugli oggetti e le loro caratteristiche di disegno sono disponibili nella sezione dedicata.

Condizioni di attivazione: un oggetto selezionato nell'area di disegno di livello inferiore a n/n ed almeno due oggetti nella pagina.

View Zoom In

Con questa voce di menú si richiede a QPaint di incrementare il fattore di zoom utilizzato per la visualizzazione dell'area di disegno. I fattori disponibili sono X1, X2, X3 e X4. Se l'area di disegno risultasse essere piú grande dell'area visualizzabile sarà possibile agire solo sulla parte visualizzata.

Condizioni di attivazione: fattore di zoom attuale inferiore a 4.

View Zoom Out

Con questa voce di menú si richiede a QPaint di decrementare il fattore di zoom utilizzato per la visualizzazione dell'area di disegno. I fattori disponibili sono X1, X2, X3 e X4. Se l'area di disegno risultasse essere piú grande dell'area visualizzabile sarà possibile agire solo sulla parte visualizzata.

Condizioni di attivazione: fattore di zoom attuale maggiore di uno.

View Previous Page

Con questa voce di menú si richiede a QPaint di visualizzare la pagina precedente a quella attualmente visualizzata. A cambio pagina avvenuto nessun oggetto comparirà selezionato nell'area di disegno.

Condizioni di attivazione: numero della pagina attuale maggiore di 1.

View Next Page

Con questa voce di menú si richiede a QPaint di visualizzare la pagina successiva a quella attualmente visualizzata. A cambio pagina avvenuto nessun oggetto comparirà selezionato nell'area di disegno.

Condizioni di attivazione: numero della pagina attuale inferiore al numero totale pagine.

View Background Mode

Con questa voce di menú si richiede a QPaint di cambiare il colore di sfondo utilizzato nell'area di disegno. Il terminale utilizza il bianco come colore di sfondo ed il nero per i testi, rettangoli ed altro. Per migliorare le operazioni di composizione delle pagine QPaint permette di scegliere il colore di sfondo utilizzabile tra bianco e verde. Quando il colore di sfondo selezionato é il bianco la pagina visualizzata da QPaint rispecchierà fedelmente ciò che apparirà sul terminale. Quando il colore di sfondo é verde si potranno individuare piú facilmente gli oggetti depositati sulla pagina. Il colore di sfondo utilizzato nell'area di disegno non influenzerá in alcun modo quanto inviato al terminale. Il colore di default all'avvio di QPaint é il verde. Questa informazione non viene memorizzata nel file del progetto e nelle impostazioni del programma. Vediamo ora un esempio di Background Mode:

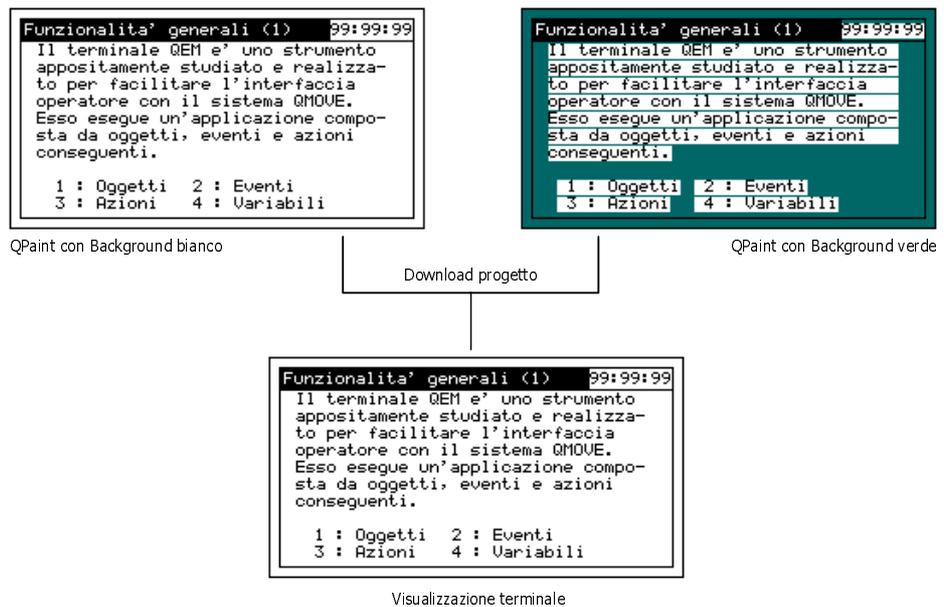


Figura 4 : esempio Background Mode con risultato nel terminale dopo il download

Come si può vedere nella fig.4, quando il Background Mode é verde, risulta molto piú facile individuare il campo di competenza (area) di ogni singolo oggetto. In questo modo si possono selezionare piú agevolmente gli oggetti, trascinarli, ed evitare sovrapposizioni indesiderate. Impostando il Background Mode su bianco si ha invece la rappresentazione di come apparirà la pagina quando sará visualizzata sul terminale. Si può inoltre notare che indipendentemente dal Background mode utilizzato, il risultato sul terminale sará lo stesso.

View Report Window

Con questa voce di menú si richiede a QPaint di visualizzare la finestra di Report, che é una finestra di tool e quindi come tale presenta alcune caratteristiche intrinseche : ha una barra del titolo ridotta, non puó essere ridotta ad icona o ingrandita a tutto schermo, é sempre sovrapposta alla finestra principale del programma e condivide con essa il flusso di input della tastiera e del mouse. All'avvio di QPaint é sempre invisibile e presenta posizione e dimensioni di default. Puó quindi essere visualizzata, spostata, ridimensionata e nascosta. Alla sua riattivazione la posizione e le dimensioni precedenti verranno ripristinate. Alla chiusura di un progetto viene sempre ripulita del suo contenuto e nascosta. La posizione e le dimensioni non vengono memorizzate nel file del progetto e nelle impostazioni del programma. In essa sono rappresentati i risultati ottenuti dalla procedura di controllo del progetto. Ulteriori informazioni inerenti ai contenuti della Report Window sono disponibili nella sezione dedicata al Check Project.

Project Symbol Reference

Con questa voce di menú si richiede a QPaint di aprire la finestra di dialogo Symbol Reference dalla quale é possibile importare nel progetto un nuovo file simboli o visualizzare le informazioni di quello attualmente caricato. Nella Symbol Reference é possibile visualizzare i valori di checksum e l'originaria posizione con nome del file simboli importato precedentemente. Nel caso di un progetto nuovo o di un progetto in cui non é mai stato importato un file simboli la dialog visualizzata sará la seguente :

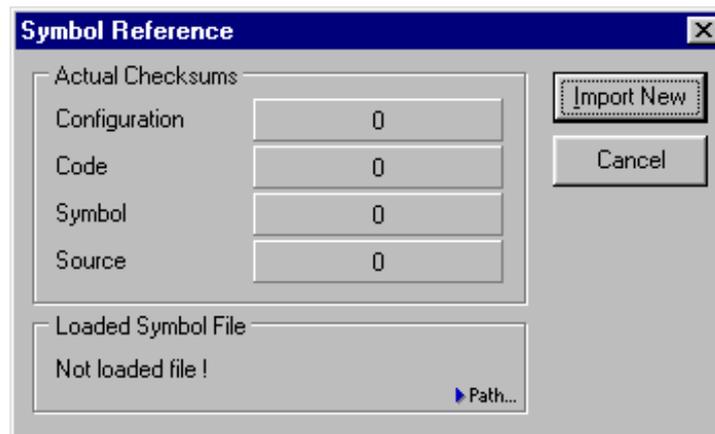


Figura 5 : finestra di dialog Symbol Reference di un progetto senza file simboli

Come si può vedere nella fig.5, nel gruppo "Actual Checksums" compaiono rispettivamente i checksum registrati nel file simboli importato ed inerenti al progetto QView che lo ha generato. Queste informazioni sono utili per verificare la validità del file simboli. Nel gruppo "Loaded Symbol File" compaiono il nome del file simboli e il percorso da cui è stato prelevato. Per visualizzare il percorso bisogna portare e mantenere il puntatore mouse sopra all'oggetto Path. Dopo alcuni istanti comparirà la finestra di tool tip con il percorso originario del file simboli. La tool tip rimarrà visibile per circa un secondo. In questo caso, non essendo mai stato importato un file simboli, i checksum avranno tutti valore zero, mentre "Not loaded file !" verrà visualizzata sia nel nome che nel percorso dello stesso. Nel caso di un progetto in cui è già stato importato un file simboli la dialog visualizzata sarà la seguente:

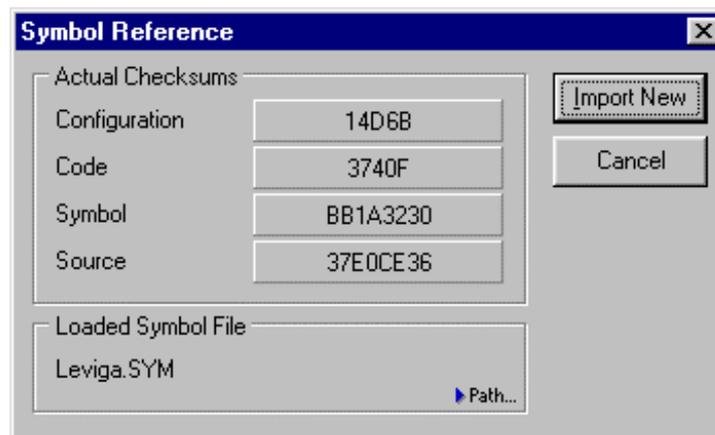


Figura 6 : finestra di dialog Symbol Reference di un progetto con file simboli

La pressione del pulsante "Import New" farà comparire la seguente finestra di dialogo:

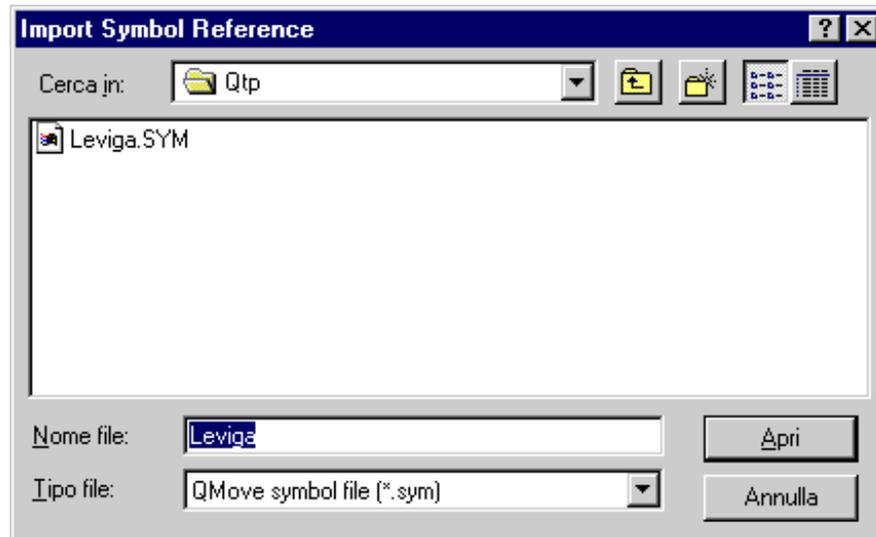


Figura 7 : finestra di dialogo Import Symbol Reference

Le funzionalità espletate dalla finestra di dialogo sono quelle standard normalmente fornite dallo stesso sistema operativo Windows®. Se nel progetto è già stato importato un file simboli, la dialog tenterà di riposizionare il percorso ed il nome del file ai valori originari. A conferma dell'operazione con il tasto "Apri" il nuovo file simboli verrà importato e ne verranno visualizzate le nuove informazioni nella dialog Symbol Reference. A questo punto il pulsante "Cancel" verrà rinominato con "OK" e focalizzato.

Se si modificano solamente i moduli task MOD non è necessaria nessuna operazione di sincronizzazione!

Il file simboli contiene tutte le informazioni necessarie per poter eseguire le operazioni di interfacciamento con l'applicativo QMove presente nella CPU collegata al terminale. Esso viene creato da QView nella fase di compilazione. Ogni compilazione comporta un aggiornamento di tale file per cui è necessario effettuare un aggiornamento dei simboli ogni qual volta si sono apportate delle modifiche al file di configurazione CNF dell'applicativo. La sequenza corretta per essere sempre sincronizzati tra QView, QMove e QPaint è la seguente

1. QView: modificare il file di configurazione CNF dell'applicativo.
2. QView: compilare l'applicativo.
3. QView: effettuare il download dell'applicativo nella CPU del QMove.
4. QView: salvare l'applicativo.
5. QPaint: con "Project | Symbol Reference" importare dalla directory dell'applicativo il nuovo file simboli SYM creato da QView durante la compilazione.

Project Check Project

Con questa voce di menù si richiede a QPaint di eseguire il controllo del progetto. Vediamo ora cosa si intende per controllo del progetto e perché sia necessaria questa operazione. QPaint è stato progettato per lasciare ampia libertà nell'impostazione delle proprietà dei vari oggetti presenti in un progetto, come eventi, azioni ed oggetti grafici. Ciò a beneficio di una maggiore flessibilità nella realizzazione dello stesso. È infatti possibile impostare gli oggetti con simboli ancora da definire, con immagini ancora da importare o creare ed altro ancora. Può anche accadere che simboli presenti inizialmente, nel file simboli importato, cambino di tipo, gruppo di appartenenza o vengano eliminati durante lo sviluppo stesso del progetto. Bene, in qualsiasi momento è possibile eseguire un controllo su quanto scritto e verificare che tutto sia coerente con le regole fissate. Tramite Check Project viene effettuato un controllo su tutti gli oggetti di un progetto e crea nella Report Window una lista con la descrizione dettagliata degli eventuali errori incontrati. Ogni linea identifica un errore rilevato in un determinato oggetto. Lo stesso oggetto può presentare più linee di errore consecutive. Ma questo non è tutto. Per semplificare le operazioni di correzione degli errori la linea selezionata sulla lista della Report Window è sensibile alla pressione del tasto Enter o al doppio clic sul tasto sinistro del mouse, di modo che venga aperto l'editor di proprietà dell'oggetto interessato con il fuoco posto automaticamente sulla causa dell'errore. Inoltre, sulla barra di stato del programma compare un estratto esplicativo dell'errore in correzione. Alla richiesta di controllo del progetto la Report Window viene posta automaticamente in visualizzazione. Il controllo del progetto viene automaticamente eseguito anche durante la fase di download. Ulteriori informazioni a riguardo sono disponibili nella sezione dedicata al download del progetto.

Nel caso in cui il controllo del progetto non riscontri alcun errore comparirà la seguente finestra:

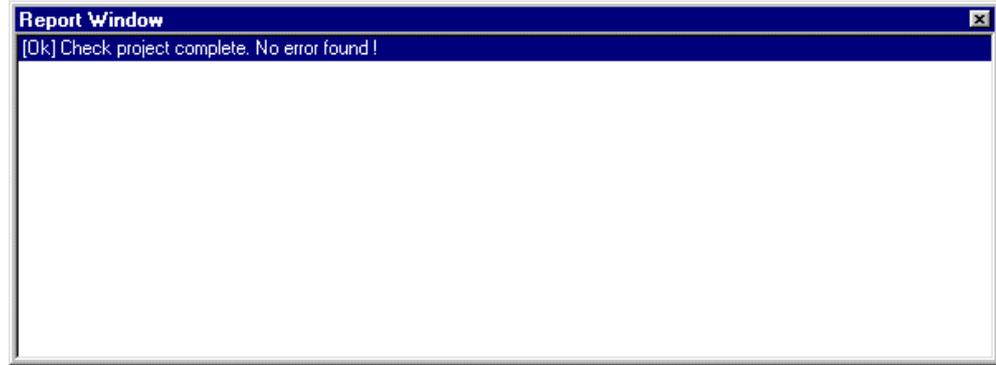


Figura 8: finestra Report Window

Come si può vedere in fig.8, nella lista della Report Window compare una sola linea che notifica il buon esito del controllo sul progetto. La pressione del tasto Enter o il doppio clic sul tasto sinistro del mouse su questa linea non comporrà alcun effetto.

Nel caso in cui il controllo del progetto riscontri degli errori comparirà la seguente finestra:

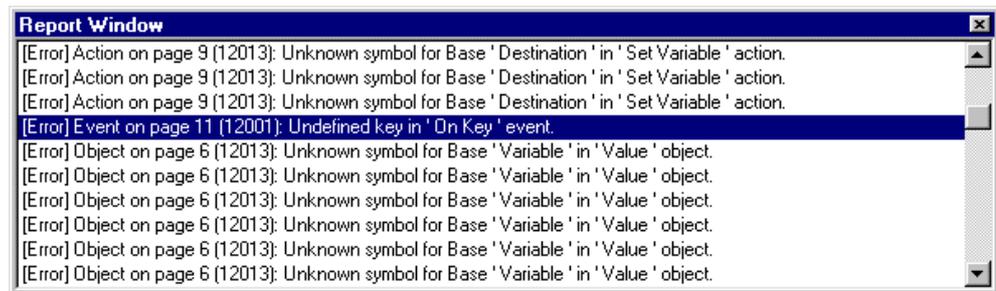


Figura 9 : finestra Report Window

Come si può vedere nella fig.9, la Report Window è composta da una lista a scorrimento verticale contenente gli errori rilevati dal controllo del progetto. Ad ogni linea corrisponde un errore. Ogni oggetto può avere più errori e quindi più linee. In ogni caso gli errori di un determinato oggetto sono sempre raggruppati. Una linea della lista è composta delle seguenti parti:

[<causa>] <collocazione> (<codice errore>): <descrizione>.

causa Definisce il motivo della linea di rapporto. In questo caso [Error] sta per errore nel progetto.

collocazione Definisce la collocazione dell'errore. Può valere :

Event in global events Evento globale

Action in global events Azione globale

Event on page ⁿ Evento sulla pagina, dove ⁿ è il numero della pagina.

Action on page ⁿ Azione sulla pagina, dove ⁿ è il numero della pagina.

Object on page ⁿ Oggetto grafico sulla pagina, dove ⁿ è il numero della pagina.

codice errore Codice identificativo dell'errore. Può valere :

- 12001 Nell'evento **On Key** non é stato definito il tasto di attivazione.
- 12002 Nell'evento **On Press** non é stato definito il tasto di attivazione.
- 12003 Nell'evento **On Release** non é stato definito il tasto di attivazione.
- 12004 Nell'evento **On Time** non é stato definito il tempo di attivazione.
- 12005 Nell'azione **Goto Page** non é stata definita la pagina a cui saltare.
- 12006 Nell'azione **Goto Page** é stato impostato il nome di una pagina inesistente.
- 12007 Nell'azione **Send Command** non é stato definito il comando da eseguire.
- 12008 Nell'azione **Send Command** é stato definito un comando non riconosciuto.
- 12009 Nell'azione **Led On** non é stato definito il led interessato all'operazione.
- 12010 Nell'azione **Led Off** non é stato definito il led interessato all'operazione.
- 12011 Nell'azione **Led Blink** non é stato definito il led interessato all'operazione.
- 12012 Nella variabile di riferimento il campo **Base** non é stato definito.
- 12013 Nella variabile di riferimento il campo **Base** contiene un simbolo sconosciuto.
- 12014 Nella variabile di riferimento il campo **Base** contiene un simbolo di un gruppo dati non valido.
- 12015 Nella variabile di riferimento il campo **Base** contiene un simbolo di un tipo dati non valido.
- 12016 Nella variabile di riferimento il campo **Base** contiene un simbolo con accesso a sola lettura.
- 12017 Nella variabile di riferimento il campo **First Index** non é stato definito.
- 12018 Nella variabile di riferimento il campo **First Index** contiene un simbolo sconosciuto.
- 12019 Nella variabile di riferimento il campo **First Index** contiene un simbolo di un gruppo dati non valido.
- 12020 Nella variabile di riferimento il campo **First Index** contiene un simbolo di un tipo dati non valido.
- 12021 Nella variabile di riferimento il campo **First Index** é impostato anche se non richiesto.
- 12022 Nella variabile di riferimento il campo **Second Index** non é stato definito.
- 12023 Nella variabile di riferimento il campo **Second Index** contiene un simbolo sconosciuto.
- 12024 Nella variabile di riferimento il campo **Second Index** contiene un simbolo di un gruppo dati non valido.
- 12025 Nella variabile di riferimento il campo **Second Index** contiene un simbolo di un tipo dati non valido.
- 12026 Nella variabile di riferimento il campo **Second Index** é impostato anche se non richiesto.

descrizione Descrive in modo dettagliato l'errore riscontrato. Vedere descrizione del campo *codice errore* per una migliore comprensione dell'errore.

Quando la Report Window é attiva, é possibile tramite i tasti cursore eseguire lo scroll dell'intera lista e utilizzare il tasto Enter per attivare l'editor proprietà dell'oggetto descritto nella linea selezionata. Tramite il mouse si possono ottenere gli stessi risultati, anche se piú agevolmente. Una volta attivato, l'editor proprietà dell'oggetto posizionerà automaticamente il fuoco sul campo dati che ha generato l'errore. Questo semplifica notevolmente le operazioni di correzione dello stesso. La Report Window modifica il contenuto della lista errori solo sulla richiesta di un controllo del progetto, per cui anche quando un errore viene corretto, la relativa linea di descrizione continuerá ad esistere. Lo stesso vale se un oggetto con uno o piú errori o la pagina in cui é contenuto vengono rimossi. A questo punto la Report Window si limita a segnalare che l'uno o l'altro oggetto non sono piú disponibili e che quindi sarebbe buona cosa ripetere il controllo del progetto per lavorare con una lista errori aggiornata. Vedere la lista messaggi all'utente per una descrizione dettagliata delle segnalazioni generate. Durante le operazioni di scaricamento del progetto attivo, la Report Window verrà automaticamente nascosta e tutti i dati in essa contenuta cancellati.

Nel caso in cui la larghezza della Report Window risulti essere insufficiente per la visualizzazione completa dei messaggi presenti sulla lista, nella parte inferiore della finestra apparirá la barra di scorrimento orizzontale.

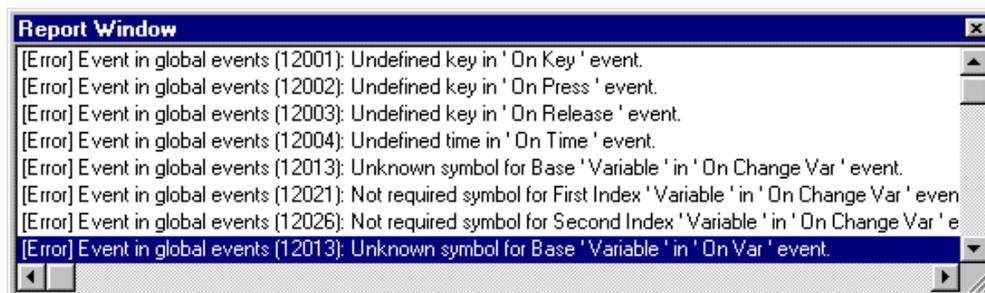


Figura 10 : finestra Report Window

Condizioni di attivazione: nel progetto deve essere stato importato almeno una volta un file simboli.

Project Project Setup

Con questa voce di menú si richiede a QPaint di aprire la finestra di dialogo Project Setup dalla quale é possibile accedere ai parametri di configurazione del progetto in uso. La finestra di dialogo visualizzata sará la seguente:

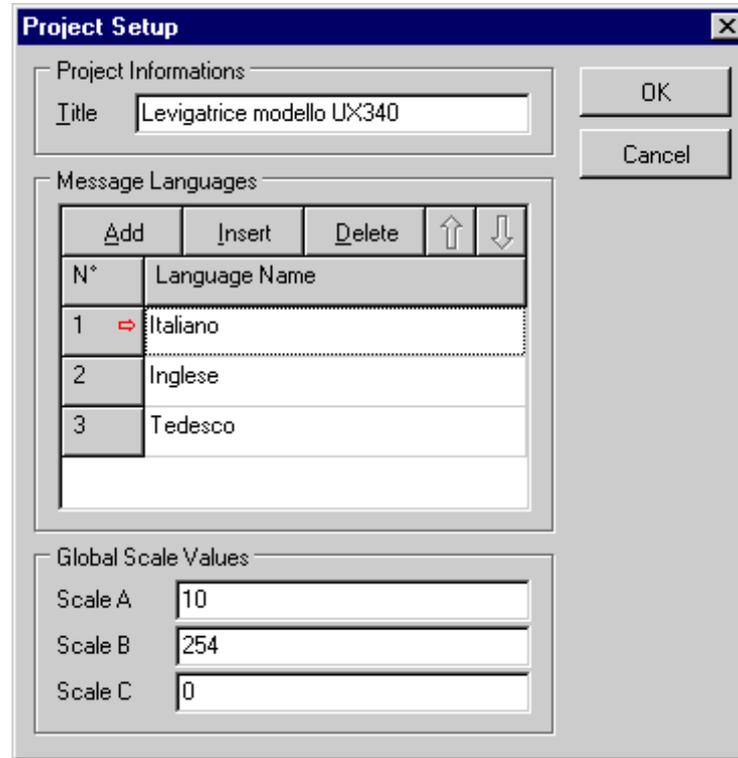


Figura 11 : finestra di dialogo Project Setup

Ogni progetto ha la possibilitá di definire i seguenti parametri di configurazione:

Title: titolo dell'applicazione. Il titolo dell'applicazione é composto da una stringa ASCII lunga al massimo 25 caratteri. Viene memorizzato nel terminale durante la fase di download ed é visualizzabile sullo stesso nella pagina INFO.

L'oggetto Value ha la possibilitá di non eseguire nessun scaling, eseguire lo scaling globale o definire un proprio fattore di scaling. Vedere la documentazione sull'oggetto Value per maggiori informazioni.

Scale A, Scale B e Scale C : valori di scaling globali. Tramite la combinazione di tali valori si ottiene un fattore di scala globale che permette la conversione tra diverse unitá di misura, come ad esempio millimetri e pollici. Il fattore di scala risultante puó essere applicato all'oggetto grafico "Value". Value ha il compito di interfacciare un campo di introduzione/visualizzazione del terminale con le variabili interne del QMove. La regola applicata é la seguente:

$$Y = X \cdot \frac{ScaleA}{ScaleB} + ScaleC$$

dove: X é il contenuto della memoria CPU QMove indirizzata dall'oggetto Value.

Y é il valore risultante visualizzato sul terminale.

Nell'esempio di fig. 11 si vede come ponendo Scale A a 10, Scale B a 254 e Scale C a 0 si ottenga il fattore di scala necessario per visualizzare un dato originario in mm nel suo corrispondente in pollici.

Message Languages: lingue messaggi. Tramite i bottoni "Add", "Insert", "Remove" e la griglia di edit sottostante, é possibile definire fino a 10 diverse lingue messaggi per tutti i testi che compariranno sul terminale. Per ogni lingua é possibile definire un nome identificativo lungo al massimo 12 caratteri. Nomi uguali per piú lingue sono ammessi. Gli oggetti interessati dalle lingue messaggi sono gli oggetti grafici "String" e "ValString". Nel progetto vi deve essere sempre almeno una lingua attiva perciò il pulsante "Remove" si disabiliterá quando si arriverá ad avere una sola lingua. Le lingue consentite sono al massimo 10, per cui i bottoni "Add" ed "Insert" si disabiliteranno quando si raggiungerá tale valore. Durante la fase di download verranno memorizzati nel terminale oltre a tutti i messaggi in lingua anche le loro descrizioni. In questo modo tramite la pagina CONFIGURATION del terminale sará possibile scorrere la lista lingue disponibili e scegliere quale rendere attiva sul display. Quando si rimuove una lingua esistente, tramite il pulsante "Remove", tutti i messaggi associati a quella lingua in tutti gli oggetti String e ValString presenti nel progetto verranno automaticamente rimossi. In QPaint é possibile definire quale lingua messaggi usare per le varie visualizzazioni nell'area di disegno. Ciò si ottiene effettuando un clic con il tasto sinistro del mouse sul numero posto a sinistra del nome identificativo della lingua interessata. Una freccia rossa "" fará da segnaposto alla lingua selezionata.

Il pulsante "Cancel" non ha alcun effetto sulle modifiche apportate alle lingue messaggi per cui tutte le impostazioni effettuate non potranno essere annullate.

In questa sezione si é parlato delle pagine di terminale INFO e CONFIGURATION, senza però descriverne la modalitá di utilizzo. Tutte le informazioni a riguardo sono disponibili nel manuale tecnico del terminale.

Project Internal Variables

Con questa voce di menú si richiede a QPaint di aprire la finestra di dialogo Internal Variables dalla quale é possibile aggiungere, rimuovere e visualizzare le variabili interne. Durante lo sviluppo di un progetto per il terminale sono disponibili tre diversi insiemi di variabili : le variabili QMove, le variabili interne e le variabili di terminale. Le variabili QMove sono quelle messe a disposizione del terminale tramite l'importazione di un file simboli. Esse risiedono nella CPU del QMove a cui é collegato il terminale e vi si accede tramite la comunicazione seriale. Le variabili interne sono variabili definite dall'utente a livello di progetto terminale. Vengono ricreate nel terminale durante la fase di download e risultano disponibili nello stesso per tutta la vita del progetto. Infine, le variabili di terminale sono variabili messe a disposizione dal costruttore e rendono disponibili all'utente informazioni specifiche del terminale come stato dei tasti e timers di esecuzione. Ulteriori informazioni sui gruppi di variabili e sulla loro collocazione nei vari dispositivi hardware e software sono disponibili nell'apposita sezione.

In un progetto possono essere create tante variabili interne quanta è la memoria disponibile nel terminale. Sono definibili solo i tipi di dati semplici flag, byte, word, long e single. Vengono automaticamente azzerate ad ogni avvio del progetto. Non vi possono essere due variabili con lo stesso nome. Ogni nome variabile è composto dal prefisso "@" seguito da una stringa di 1 ÷ 11 caratteri ASCII in cui viene rilevata la differenza tra caratteri minuscoli e maiuscoli.

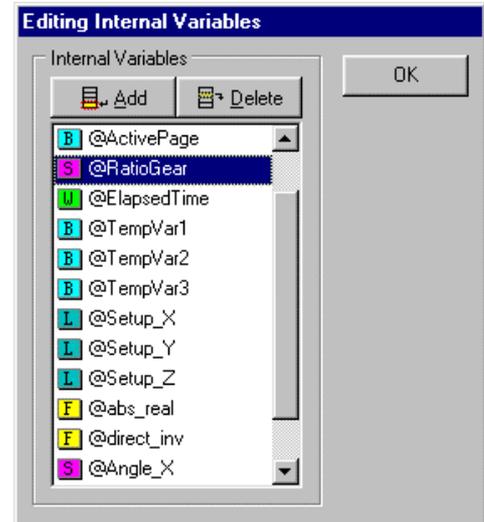
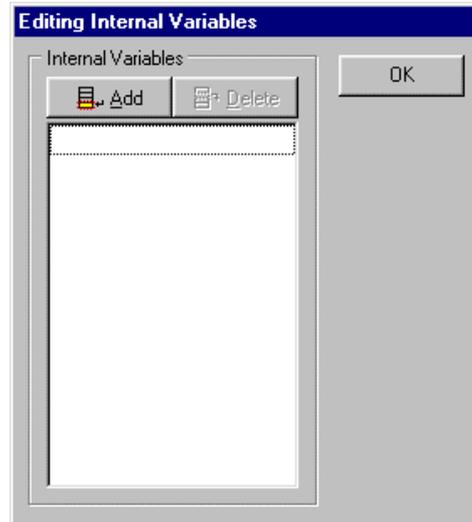


Figura 12 : finestra di dialog Internal Variables Figura 13 : finestra di dialog Internal Variables

Nella fig.12 è possibile vedere lo stato della finestra di dialogo Internal Variables quando nessuna variabile interna è stata definita. Il pulsante "Add" è abilitato indicando che è possibile definire una nuova variabile, mentre il pulsante "Delete" è disabilitato poiché non vi è nessuna variabile da rimuovere. Nella fig.13, invece, è raffigurata la stessa dialog in cui sono state definite svariate variabili interne di più tipi di dato. Si può notare che il pulsante "Delete" risulta ora abilitato, è comparsa una barra a scorrimento verticale con la quale visualizzare tutte le variabili, ed è presente una linea di selezione per identificare la linea soggetta all'azione del tasto "Delete".

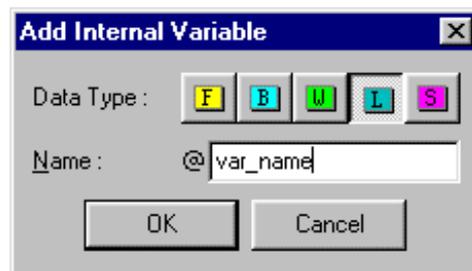


Figura 14 : finestra di dialog Add Internal Variable

Premendo il pulsante "Add" viene visualizzata la dialog di fig.14. In essa sono disponibili 5 bottoni con i quali selezionare il tipo di dato desiderato: F = flag, B = byte, W = word, L = long e S = single. I tipi di dati sono gli stessi presenti nel QMove. In "Name : " é possibile impostare il nome della variabile. Il prefisso "@" non é rimuovibile poiché obbligatorio per ogni variabile interna. Nel caso in cui si tenti di dichiarare una variabile interna con un nome già usato comparirà un box di messaggio con la scritta "Name "@var_name" already used by another Internal Variable !". Premendo il tasto "OK" si ritornerà alla dialog di figura 14.

Project Global Events Editor

Con questa voce di menú si richiede a QPaint di aprire la finestra di dialogo Global Events Editor dalla quale é possibile visualizzare ed introdurre gli eventi attivi a livello globale.

Un evento é una particolare condizione che si verifica nel tempo e che viene rilevata dal terminale, come ad esempio la pressione di un tasto o il cambiamento del valore di una variabile nella CPU del QMove. Ad un determinato evento é possibile associare una lista di azioni che il terminale andrà a svolgere quando l'evento viene rilevato. Gli eventi si possono dividere in due gruppi di appartenenza: eventi globali ed eventi di pagina. Gli eventi globali vengono sempre elaborati prima di quelli di pagina. Un evento definito sia negli eventi globali che negli eventi di pagina, vedrà eseguite prima le azioni relative all'evento globale e poi quelle relative all'evento di pagina.

La finestra di dialogo visualizzata sarà la seguente:

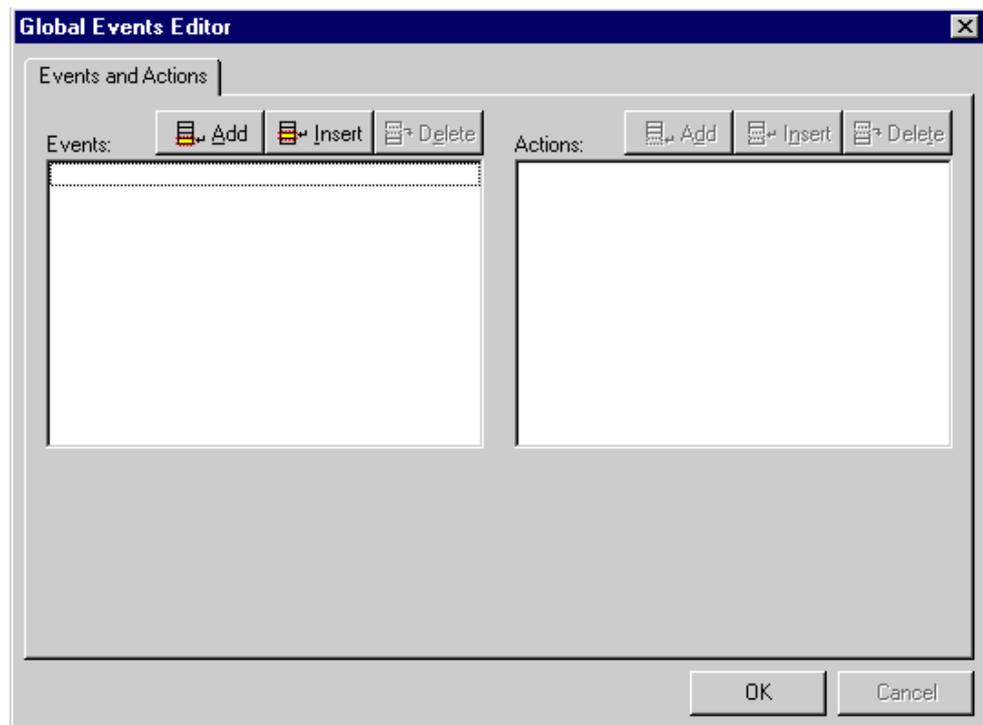


Figura 15 : finestra di dialogo Global Events Editor

Come si può vedere nella fig.15, la dialog è divisa in tre aree principali:

1. la zona di sinistra, dedicata alla gestione degli eventi.
2. la zona di destra, dedicata alla gestione delle azioni associate agli eventi.
3. l'area sottostrante le due, riservata all'introduzione dei parametri specifici della prima o della seconda (in questo caso risulta vuota). Tale area viene anche chiamata "pannello impostazioni".

In un progetto possono essere creati tanti eventi quanta è la memoria disponibile nel terminale.

Definizione di un nuovo evento

Nella zona di sinistra, dedicata alle operazioni sugli eventi, sono presenti i bottoni "Add", "Insert", "Delete" ed una casella di riepilogo contenente la lista degli eventi già definiti. Per definire un nuovo evento vi sono due metodi :

1. utilizzare il pulsante "Add" per aggiungere un nuovo evento in coda alla lista degli eventi.
2. selezionare nella casella di riepilogo la posizione ove inserire il nuovo evento. Utilizzare il pulsante "Insert" per inserire un nuovo evento nella lista degli eventi e alla posizione selezionata.

Alla pressione del pulsante "Add" o "Insert" comparirà la seguente finestra di selezione:



Figura 16 : finestra di selezione Select Event

Come si può vedere nella fig.16, la finestra Select Event è completamente occupata dalla casella di riepilogo contenente le definizioni degli eventi disponibili a livello globale. La selezione avviene come per una normale casella di riepilogo : effettuando un doppio clic con il tasto sinistro del mouse sulla voce interessata o selezionandola con i tasti UP & DOWN seguiti da una conferma con ENTER. Una volta confermato l'evento, sul pannello impostazioni, compariranno i campi di impostazione dei parametri dell'evento.

OnKey

L'evento OnKey a livello globale non è associabile ai tasti PGUP, PGDN e INS per cui i corrispettivi bottoni sul pannello impostazioni risulteranno sempre disabilitati.

L'evento OnKey viene generato, ad ogni scansione di pagina, fintanto che il tasto del terminale associato risulta premuto. Il tempo di ripetizione dell'evento dipende dalla complessità della pagina attiva. Più sono gli oggetti dinamici visualizzati, più lungo è il tempo tra la generazione di un evento e l'altro. Per associare la generazione di un evento OnKey ad un tasto del terminale basta fare clic con il pulsante sinistro del mouse sul pulsante rappresentante tale tasto nel pannello impostazioni.

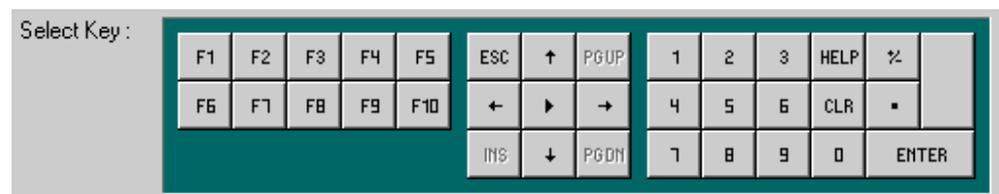


Figura 17: pannello impostazioni evento OnKey

Se un tasto del terminale é già stato utilizzato o non é disponibile poiché già definito dal sistema, il suo pulsante rappresentante, visualizzato nel pannello impostazioni, comparirá disabilitato.

OnPress

L'evento OnPress viene generato alla pressione di un tasto del terminale. Per l'evento OnPress valgono le stesse procedure di impostazione descritte nell'evento OnKey.

L'evento OnPress a livello globale non é associabile ai tasti PGUP, PGDN e INS per cui i corrispettivi bottoni sul pannello impostazioni risulteranno sempre disabilitati.

OnRelease

L'evento OnRelease viene generato al rilascio di un tasto del terminale. Per l'evento OnRelease valgono le stesse procedure di impostazione descritte nell'evento OnKey.

L'evento OnRelease a livello globale non é associabile ai tasti PGUP, PGDN e INS per cui i corrispettivi bottoni sul pannello impostazioni risulteranno sempre disabilitati.

OnAlways

L'evento OnAlways viene generato fintanto che si é nella sua area di competenza. In questo caso trattandosi di eventi di livello globale verrá generato in continuazione. Il tempo di ripetizione dell'evento dipende dalla complessitá della pagina attiva. Piú sono le variabili della CPU QMove visualizzate, piú lungo é il tempo tra la generazione di un evento e l'altro. L'evento non ha campi di impostazione e ne puó essere definito solo uno a livello globale, ed uno per ogni pagina. A questo scopo QPaint fará sí che l'evento, se già in uso, scompaia dalla finestra di selezione Select Event.

OnPageIn

L'evento OnPageIn viene generato al primo ingresso in una nuova pagina. L'evento non ha campi di impostazione e ne puó essere definito solo uno a livello globale, ed uno per ogni pagina. A questo scopo QPaint fará sí che il nome dell'evento, se é già in uso, scompaia dalla finestra di selezione Select Event.

OnTime

L'evento OnTime viene generato quando il calendario del terminale coincide con i valori di comparazione impostati. L'evento viene generato solamente una volta. Perché venga generato l'evento, il terminale dovrà prima trovare dei valori non corrispondenti seguiti da dei valori corrispondenti. Ciò significa, inoltre, che se il terminale viene spento quando l'evento OnTime é soddisfatto, alla successiva riaccensione esso non verrá rigenerato. Se al momento in cui l'evento diventa valido, il terminale si trova spento, esso verrá perso.

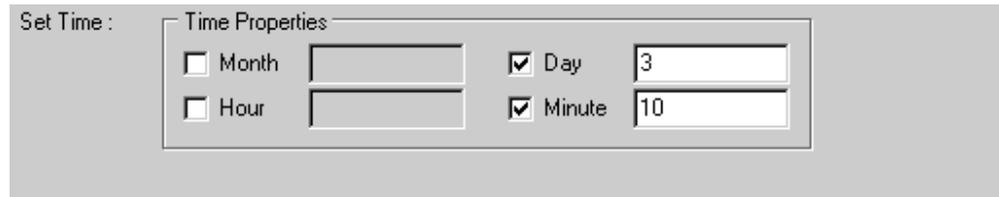


Figura 18 : pannello impostazioni evento OnTime

Nell'esempio di fig. 18 si è impostato l'evento OnTime per venir generato al decimo minuto di ogni ora del giorno 3 di ogni mese. Come si può vedere è possibile definire il mese, giorno, ora e minuto di occorrenza dell'evento. Una casella di controllo posta davanti al campo di introduzione di mese, giorno, ora e minuto ne abilita o disabilita la comparazione. Se tale casella è disabilitata si intende che l'evento viene generato per tutti i valori possibili del campo in esame. Almeno un campo deve essere impostato per assicurare la correttezza dell'introduzione.

OnChangeVar

L'evento OnChangeVar viene generato ogni qual volta la variabile di riferimento, definita nel pannello delle impostazioni, cambia il proprio valore.

Nel documento è presente il capitolo "Impostazione variabili di riferimento" nel quale sono descritte in dettaglio tutte le operazioni necessarie per definire una variabile di riferimento.



	Base	First Index	Second Index
Variable	\$PAGE		

Figura 19: pannello impostazioni evento OnChangeVar

Nell'esempio di fig. 19 si è impostato che l'evento OnChangeVar venga generato ad ogni variazione della variabile di terminale \$PAGE.

OnVar

L'evento OnVar viene generato ogni qual volta il valore della variabile di riferimento corrisponde ai parametri di comparazione impostati. Tali parametri sono composti da due campi:

1. Compare Mode, la modalità di comparazione.
2. Compare Value, il valore in comparazione.

Compare Mode può avere i seguenti valori :

1. < less than minore di
2. > greater than maggiore di
3. < = less or equal minore o uguale a
4. > = greater or equal maggiore o uguale a
5. = equal uguale a

Compare Value può contenere solamente valori interi compresi tra -2147483648 e 2147483647.

	Base	First Index	Second Index
Variable	\$PAGE		
Comp.Mode	= equal		
Comp.Value	10		

Figura 20 : pannello impostazioni evento OnVar

Nell'esempio di fig.20 si è impostato che l'evento OnVar venga generato quando la variabile di terminale \$PAGE assume il valore 10.

Definizione di una nuova azione

In un progetto possono essere create tante azioni quanta è la memoria disponibile nel terminale.

Nella zona di destra della finestra di dialogo "Global Events Editor", dedicata alle operazioni sulle azioni, sono presenti i bottoni "Add", "Insert", "Delete" ed una casella di riepilogo contenente la lista delle azioni già definite. Ogni evento, rappresentato nella casella di riepilogo a sinistra, è collegato ad una lista di azioni che verranno eseguite quanto l'evento stesso viene generato. Non vi può essere una lista di azioni senza che vi sia un evento collegato. Per questo motivo quando non vi sono eventi definiti non sarà possibile definire alcuna azione e i bottoni "Add", "Insert" e "Delete" appariranno disabilitati. È importante anche ricordare che la lista di azioni visualizzata è quella relativa all'evento selezionato nella casella di riepilogo degli eventi e come tale cambia il suo contenuto ogni volta che si cambia l'evento selezionato. Per definire una nuova azione vi sono due metodi :

1. utilizzare il pulsante "Add" per aggiungere una nuova azione in coda alla lista delle azioni.
2. selezionare nella casella di riepilogo la posizione ove inserire la nuova azione. Utilizzare il pulsante "Insert" per inserire la nuova azione nelle lista delle azioni e alla posizione selezionata.

Alla pressione del pulsante "Add" o "Insert" comparirá la seguente finestra di selezione :

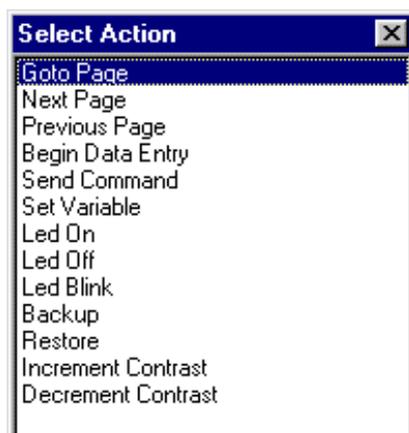


Figura 21 : finestra di selezione Select Action

Come si può vedere nella fig.21, la finestra Select Action é completamente occupata dalla casella di riepilogo contenente le definizioni delle azioni disponibili a livello globale. La selezione avviene come per una normale casella di riepilogo : effettuando un doppio clic con il tasto sinistro del mouse sulla voce interessata o selezionandola con i tasti UP & DOWN seguiti da una conferma con ENTER. Una volta confermata l'azione, sul pannello impostazioni, compariranno i campi di impostazione dei parametri dell'azione.

Goto Page

Tramite l'azione Goto Page si richiede al terminale l'attivazione della pagina specificata. In QPaint vi sono due modi per identificare univocamente una pagina : tramite il suo numero ordinale e tramite il nome. Tutte le pagine hanno un numero che le identifica all'interno del progetto. Questo significa che se nel progetto vi sono 54 pagine, avremo che la prima pagina della lista avrà il numero 1, la seconda il 2 e così via fino all'ultima che avrà il 54. Quando si aggiunge una pagina, questa verrà creata in fondo alla lista ed acquisirà il numero successivo all'ultima pagina presente. Quando si inserisce una pagina, per esempio, in dodicesima posizione, tutte le pagine dalla 12 all'ultima incrementeranno il loro numero di una unità e la nuova pagina prenderà il numero 12. Da questo si capisce che riferirsi ad una determinata pagina tramite il numero può risultare pericoloso. Se inizialmente alla pagina 32 abbiamo le visualizzazioni di produzione, non è detto che in futuro, dopo inserimenti o rimozioni di altre pagine prima di quella interessata, tale pagina sia ancora la dodicesima nella lista delle pagine. Ciò comporterebbe quindi un continuo riaggiustamento delle azioni che utilizzano il numero 12 per identificare la pagina. Per semplificare le operazioni di identificazione di una pagina c'è la possibilità di associare ad essa anche un nome che la rappresenti in modo univoco per tutto il progetto. Il nome di una pagina è una stringa ASCII da 1 ÷ 20 caratteri in cui le lettere maiuscole sono ritenute diverse dalle minuscole. Una pagina ha sempre un numero, ma può non avere un nome. Questo avviene quando il nome della pagina è composto da soli caratteri spazio. Quando si usa il nome per identificare una pagina, questo renderà valida la sua identificazione per tutta la durata dello sviluppo senza dover mai riaggiornare gli oggetti che la usano. L'azione Goto Page permette di specificare la pagina a cui eseguire il salto sia con una variabile di riferimento che con un nome scelto dalla lista dei nomi delle pagine disponibili. Nel primo caso la variabile potrebbe contenere una costante per il numero della pagina o una variabile QMove o di terminale in cui sia contenuto il numero della pagina desiderata. Questo rende possibile decidere la pagina a cui saltare direttamente da linguaggio QCL sulla CPU del QMove collegato al terminale. Nel secondo caso sarà utilizzato il nome stesso della pagina desiderata scelto dalla lista dei nomi pagina rilevati nel progetto.

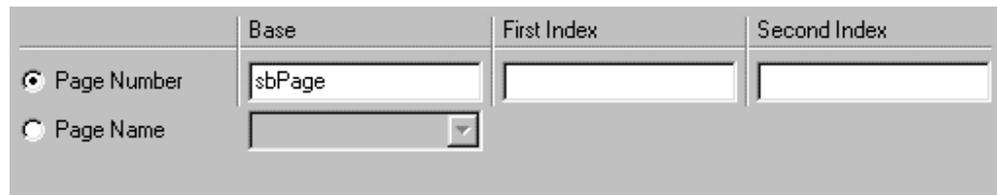


Figura 22 : pannello impostazioni azione Goto Page

Come si può vedere nella fig.22, selezionando il pulsante di opzione "Page Number" si abilita l'azione Goto Page ed eseguire il salto sulla pagina specificata da una variabile di riferimento. In questo caso si tratta di una variabile QCL di nome sbPage presente nella CPU del QMove collegato al terminale.

Nel documento è presente il capitolo "Impostazione variabili di riferimento" nel quale sono descritte in dettaglio tutte le operazioni necessarie per definire una variabile di riferimento.

	Base	First Index	Second Index
<input type="radio"/> Page Number	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input checked="" type="radio"/> Page Name	Available Chars <input type="text"/>		

Figura 23: pannello impostazioni azione Goto Page

Come si può vedere nella fig.23, selezionando il pulsante di opzione "Page Name" si abilita l'azione Goto Page ad eseguire il salto alla pagina specificata dal nome presente nella casella combinata contenente la lista dei nomi delle pagine del progetto.

Nel caso in cui una pagina con nome venga rimossa, per esempio "Available Chars", e vi siano una o più azioni Goto Page che la indirizzavano tramite il campo Page Name, accadrà che : nella casella di riepilogo della zona di destra, dedicata alle operazioni sulle azioni, comparirà la linea **Goto Page: " Available Chars "**. Sebbene "Available Chars" non esista più, per agevolare le operazioni di correzione dell'azione, nella casella di riepilogo verrà mantenuto il nome della pagina in cui veniva richiesto il salto. Diversamente, la casella combinata del pannello impostazioni apparirà vuota e scomparirà ogni riferimento alla pagina "Available Chars".

Next Page

Tramite l'azione Next Page si richiede al terminale l'attivazione della pagina successiva a quella attualmente attiva. Nel caso in cui la pagina attiva sia l'ultima della lista pagine di progetto, l'azione non effettuerà alcun cambio di pagina.

Previous Page

Tramite l'azione Previous Page si richiede al terminale l'attivazione della pagina precedente a quella attualmente attiva. Nel caso in cui la pagina attiva sia la prima della lista pagine di progetto, l'azione non effettuerà alcun cambio di pagina.

Begin Data Entry

Tramite l'azione Begin Data Entry si richiede al terminale di passare in modalità introduzione dati. Tale modalità si abiliterà solamente se sono presenti nella pagina attiva oggetti grafici di tipo Value o ValString con la proprietà "Enable Modify Mode" abilitata.

Send Command

Tramite l'azione Send Command si richiede al terminale di inviare un comando ad un device presente nella CPU del QMove. In questo caso le fasi eseguite sono : recupero informazioni azione, invio comando al QMove tramite comunicazione seriale, verifica corretta ricezione comando con eventuali ritentativi, bufferizzazione del comando ricevuto nel QMove e infine analisi prossima azione presente nella lista azioni del terminale. Da quanto appena visto si capisce che il terminale non attende la reale esecuzione del comando inviato. Infatti il QMove si limiterà a bufferizzarlo per poi eseguirlo nel momento più opportuno. Il tempo di ritardo nell'esecuzione del comando sarà compreso tra 0 e il tempo di campionamento del device.



Figura 24: pannello impostazioni azione Send Command

Come si può vedere nella fig.24, l'azione Send Command è stata programmata per inviare il comando START al device AsseX. Per semplificare le operazioni di impostazione dei parametri dell'azione, è stata introdotta una finestra di dialogo chiamata Symbol Object Selection. La dialog può essere richiamata premendo il tasto F5 quando il fuoco è sul campo di introduzione "Device Command".

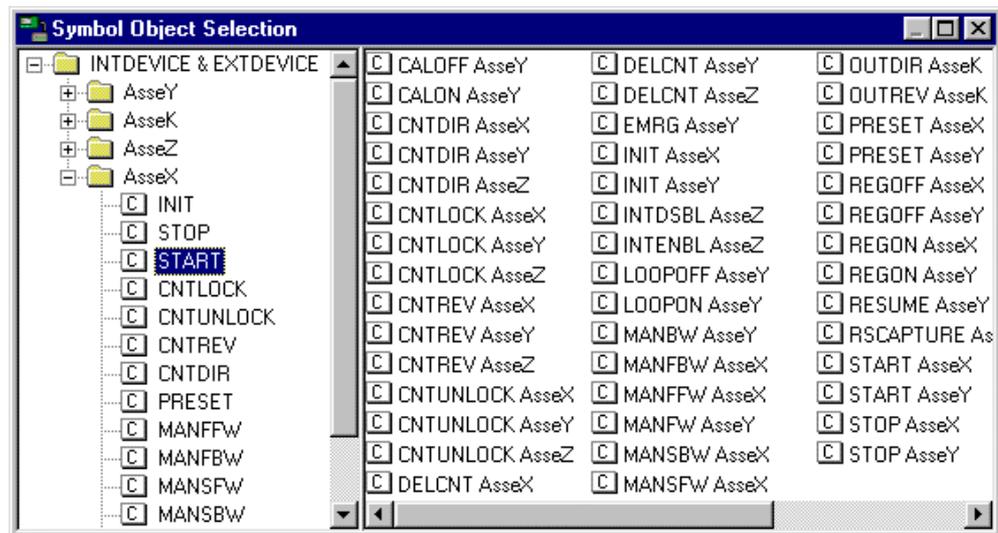


Figura 25 : finestra di dialogo Symbol Object Selection

Come si può vedere nella fig.25, la dialog Symbol Object Selection è divisa in due aree distinte. Quella di sinistra raggruppa in una struttura ad albero i vari device disponibili nella sezione INTDEVICE / EXTDEVICE del file simboli importato. Quella di destra è invece occupata dalla lista completa di tutti i comandi disponibili. Per confermare un comando è sufficiente eseguire un doppio clic con il tasto sinistro del mouse o premere il tasto ENTER sulla voce desiderata.

Set Variable

Tramite l'azione Set Variable si richiede al terminale di copiare il valore della variabile di riferimento Source nella variabile di riferimento Destination.



Figura 26 : pannello impostazioni azione Set Variable

Come si può vedere nella fig.26, l'azione Set Variable è stata programmata affinché copi il contenuto della variabile terminale \$HOUR nella variabile del QMove sLOre.

Nel documento è presente il capitolo "Impostazione variabili di riferimento" nel quale sono descritte in dettaglio tutte le operazioni necessarie per definire una variabile di riferimento.

Led On

Tramite l'azione Led On si richiede al terminale di attivare uno dei led associati ai tasti funzione disponibili nella tastiera dello stesso.



Figura 27 : pannello impostazioni azione Led On

Come si può vedere nella fig.27, l'azione Led On è stata impostata affinché attivi il led associato al tasto funzione F4.

Led Off

Tramite l'azione Led Off si richiede al terminale di disattivare uno dei led associati ai tasti funzione disponibili nella tastiera dello stesso.



Figura 28 : pannello impostazioni azione Led Off

Come si può vedere nella fig.28, l'azione Led Off è stata impostata affinché disattivi il led associato al tasto funzione F4.

Led Blink

Tramite l'azione Led Blink si richiede al terminale di attivare in modalità di blink uno dei led associati ai tasti funzione disponibili nella tastiera dello stesso.

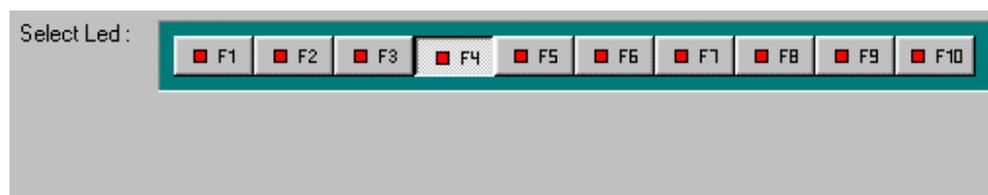


Figura 29 : pannello impostazioni azione Led Blink

Come si può vedere nella fig.29, l'azione Led Blink è stata impostata affinché attivi in modalità di blink il led associato al tasto funzione F4.

Backup

Tramite l'azione Backup si richiede alla CPU del QMove di avviare la procedura di backup dei dati. Durante l'esecuzione dell'azione la CPU passa in stato READY e vi rimane anche al completamento dell'operazione.

Restore

Tramite l'azione Restore si richiede alla CPU del QMove di avviare la procedura di restore dei dati. Durante l'esecuzione dell'azione la CPU passa in stato READY e vi rimane anche al completamento dell'operazione.

L'azione Restore viene eseguita solo se in precedenza è stata effettuata almeno una procedura di Backup.

Increment Contrast

Tramite l'azione Increment Contrast si richiede al terminale di incrementare il contrasto di visualizzazione del display LCD.

Nella condizione di temperatura ottimale (vedere caratteristiche terminale) il tempo di warm-up del display è dell'ordine di 5 ÷ 10 secondi. Si consiglia comunque di attendere almeno 5 minuti dall'accensione a freddo del terminale prima di procedere alla modifica del contrasto.

Decrement Contrast

Tramite l'azione Decrement Contrast si richiede al terminale di decrementare il contrasto di visualizzazione del display LCD.

Nella condizione di temperatura ottimale (vedere caratteristiche terminale) il tempo di warm-up del display è dell'ordine di 5 ÷ 10 secondi. Si consiglia comunque di attendere almeno 5 minuti dall'accensione a freddo del terminale prima di procedere alla modifica del contrasto.

Attenzione !

Quando il terminale è in modalità introduzione dati sono disabilitati gli eventi On Press, On Key e On Release sia a livello globale che di pagina. Inoltre sono disabilitate anche le azioni Goto Page, Next Page e Previous Page. Nel caso in cui si volesse realizzare un campo introduzione che permetta di impostare il numero della pagina in cui saltare, è consigliato l'uso dell'evento On Change Var abbinato all'azione Goto Page.

Page Insert Page

Con questa voce di menù si richiede a QPaint di inserire una nuova pagina nel progetto. In un progetto vi possono essere al massimo 999 pagine, per cui una volta raggiunto tale limite la voce verrà disabilitata.

Condizioni di attivazione: nel progetto vi devono essere meno di 999 pagine.

Page Add Page

Con questa voce di menù si richiede a QPaint di aggiungere una nuova pagina nel progetto. In un progetto vi possono essere al massimo 999 pagine, per cui una volta raggiunto tale limite la voce verrà disabilitata.

Condizioni di attivazione: nel progetto vi devono essere meno di 999 pagine.

Page Delete Page

Con questa voce di menù si richiede a QPaint di rimuovere la pagina attiva dal progetto. Prima di effettuare la rimozione verrà visualizzata una finestra di dialogo con il testo "Page will be lost ! Are you sure of to want to continue ?". Se si conferma l'operazione con "Yes", la pagina verrà rimossa, mentre "No" abortirà l'intera operazione.

Condizioni di attivazione : nel progetto vi devono essere almeno 2 pagine.

Page Toggle Lock Page

Con questa voce di menù si richiede a QPaint di abilitare o disabilitare l'impostazione del "controllo della password" relativo alla pagina attiva. Ulteriori informazioni sono disponibili sulla sezione riguardante la "barra di impostazione della password di pagina".

Page Page Events Editor

Con questa voce di menù si richiede a QPaint di aprire la finestra di dialogo Page Events Editor dalla quale é possibile visualizzare ed introdurre gli eventi relativi alla pagina di progetto attiva. Sostanzialmente gli eventi di pagina sono eguali a quelli globali per cui quanto descritto nella sezione "Project | Global Events Editor" é valido anche per la sezione "Page | Page Events Editor". Tra i due editor esistono però alcune differenze:

- nel titolo della finestra di dialogo compare il numero, ed eventualmente anche il nome, della pagina interessata.
- negli eventi On Key, On Press e On Release é possibile creare una lista di azioni anche per i tasti INS, PG DN e PG UP.

La finestra di dialogo visualizzata sarà la seguente :

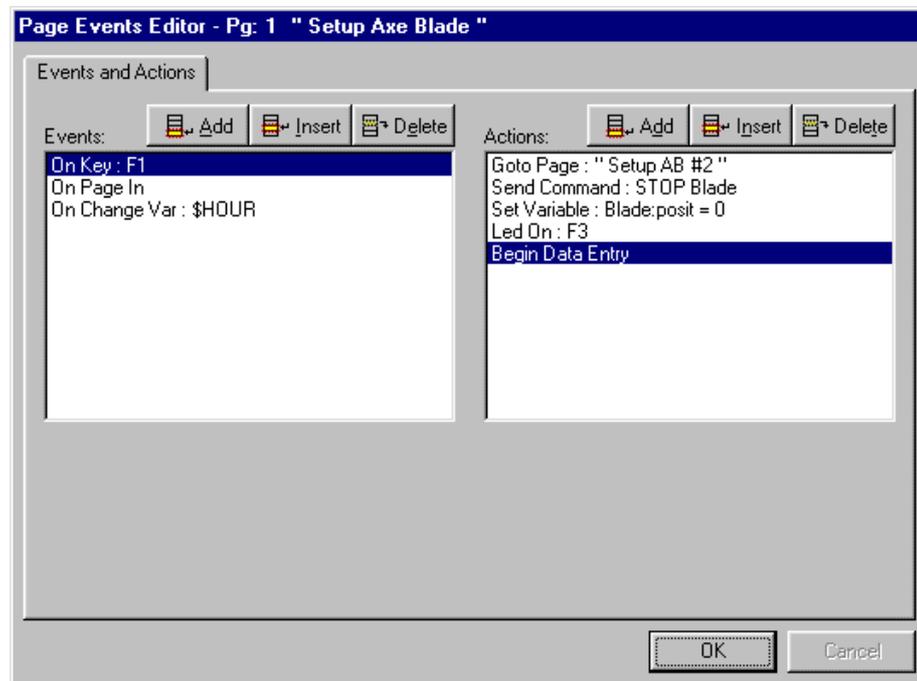


Figura 30 : finestra di dialogo Page Events Editor

Come si può vedere in fig. 30, l'editor eventi di pagina è esattamente lo stesso utilizzato per gli eventi globali e differisce solamente per il testo visualizzato nel titolo della finestra di dialogo.

Transfer Download

Con questa voce di menù si richiede a QPaint di effettuare il download del progetto nel terminale. La procedura di download si divide in cinque fasi:

1. Il controllo del progetto.
2. La connessione seriale con il terminale.
3. La compilazione del progetto.
4. La cancellazione della memoria nel terminale.
5. Il download del progetto nel terminale.
6. Lo start del progetto scaricato nel terminale.

Il controllo del progetto

Durante questa fase il QPaint esegue un controllo del progetto per verificare che lo stesso sia coerente con le regole fissate. Ciò permette di evitare che vengano effettuati trasferimenti nel terminale di progetti non corretti, impedendo quindi malfunzionamenti e comportamenti indesiderati. La procedura seguita è la stessa utilizzata per la voce di menù "Project | Check Project" a cui si rimanda per ulteriori informazioni. Unica differenza tra il controllo di progetto effettuato durante il download e quanto fatto durante il Check Project di "Project | Check Project" è che durante il download la finestra di dialogo Report Window non viene visualizzata automaticamente, ma la si potrà richiamare manualmente una volta usciti dalla dialog del download. Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK".

La connessione seriale con il terminale

Durante questa fase il QPaint verifica la comunicazione seriale con il terminale che deve trovarsi in modalità "Download Configuration". I parametri di comunicazione sono quelli impostati in "Options | Communication" e a cui si rimanda per ulteriori informazioni. Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK".

La compilazione del progetto

Durante questa fase il QPaint effettua la compilazione del progetto e crea il file da trasferire al terminale. Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK".

La cancellazione della memoria nel terminale

Durante questa fase il QPaint effettua l'azzeramento della memoria del terminale in cui verrà memorizzato il progetto compilato. Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK". Le operazioni effettuate nel terminale non verranno annullate.

Il download del progetto nel terminale

Durante questa fase il QPaint effettua il download del progetto compilato nella memoria del terminale. A questo punto nella dialog Download Project comparirà un indicatore che visualizzerà la progressione dei dati trasferiti istante per istante. Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK". Le operazioni effettuate nel terminale non verranno annullate.

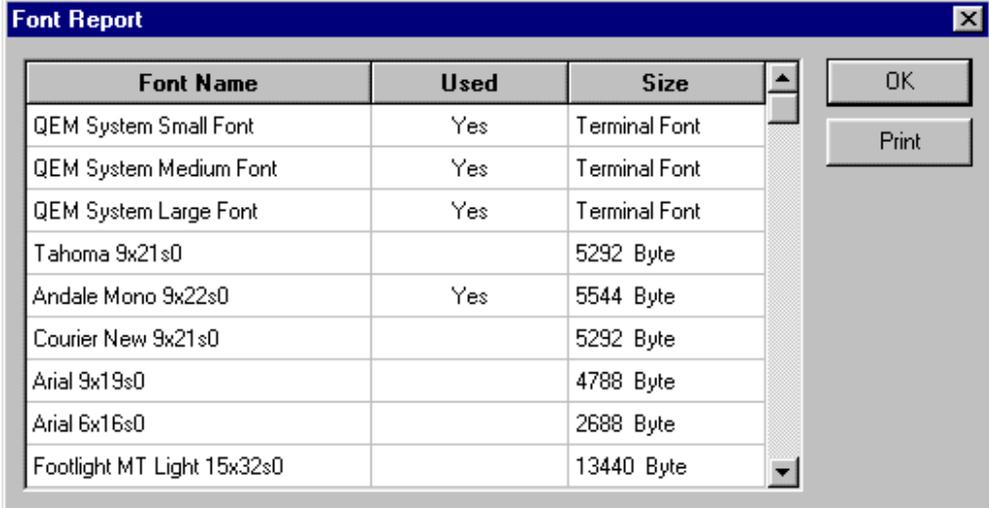
Lo start del progetto scaricato nel terminale

Durante questa fase il QPaint invia al terminale il comando di start progetto. Tramite questo comando il terminale avvierà il progetto scaricato uscendo dalla modalità "Download Configuration". Se durante questa fase viene riscontrato un errore la procedura di download termina e sarà possibile uscire dalla dialog Download Project tramite il pulsante "OK". Le operazioni effettuate nel terminale non verranno annullate.

Graphics Font Report

Con questa voce di menù si richiede a QPaint di aprire la finestra di dialogo Font Report dalla quale è possibile visualizzare e stampare un rapporto dettagliato sui font disponibili, inclusa l'indicazione di quali sono in uso nel corrente progetto. La finestra di dialogo visualizzata sarà la seguente:

Ulteriori informazioni riguardo ai font e la loro implementazione nel sistema QPaint/Terminale sono disponibili nella sezione dedicata.



Font Name	Used	Size
QEM System Small Font	Yes	Terminal Font
QEM System Medium Font	Yes	Terminal Font
QEM System Large Font	Yes	Terminal Font
Tahoma 9x21s0		5292 Byte
Andale Mono 9x22s0	Yes	5544 Byte
Courier New 9x21s0		5292 Byte
Arial 9x19s0		4788 Byte
Arial 6x16s0		2688 Byte
Footlight MT Light 15x32s0		13440 Byte

Figura 31 : finestra di dialogo Font Report

Come si può vedere nella fig.31, per ogni font viene visualizzato il nome, lo stato di utilizzo nel corrente progetto e la quantità di memoria terminale che impegnerà se utilizzato. I font con il campo "Size" impostato a "Terminal Font" sono residenti nella ROM interna del terminale, per cui non vengono scaricati durante la fase di download e sono sempre disponibili in qualsiasi progetto.

Graphics New Font From... Bitmap File

Con questa voce di menú si richiede a QPaint di creare un nuovo font utente a partire da un file Bitmap monocromatico. La finestra di dialogo visualizzata sarà la seguente:

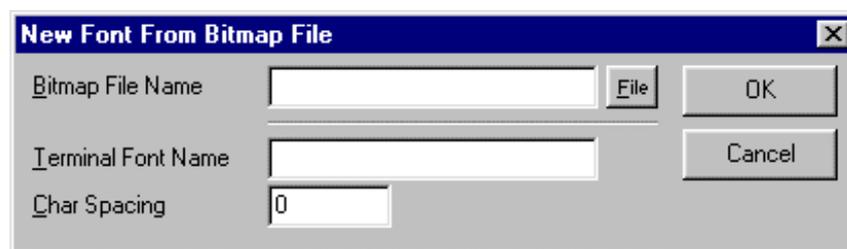


Figura 32 : finestra di dialogo New Font From Bitmap File

Come si può vedere nella fig.32, nella dialog sono presenti tre campi di introduzione dati e rispettivamente : la casella "Bitmap File Name", nella quale si specifica il nome del file Bitmap comprensivo di percorso ed estensione (es: "C:\Progetti\Terminale\Bitmap per Font\Display 7 segmenti.BMP"), la casella "Terminal Font Name", nella quale si definisce il nome del font utente in creazione (es: "Display 7 segmenti") e la casella "Char Spacing", nella quale si imposta il valore di spacing carattere desiderato. Tramite il pulsante "OK" si avvia l'operazione di creazione del font, mentre "Cancel" abortisce l'intera operazione. É presente, inoltre, il pulsante "File", con il quale é possibile selezionare il file Bitmap utilizzando la finestra di dialogo standard di Windows® per la selezione delle immagini.

Alla pressione del pulsante "OK" il QPaint effettuerá una serie di controlli sui dati introdotti. Tali controlli verificheranno la bontá dei dati nei campi di introduzione e le caratteristiche del file Bitmap specificato. In caso vengano riscontrate irregolaritá QPaint visualizzerá un box di dialogo con le indicazioni del caso.

Graphics New Font From... True Type Font

Con questa voce di menú si richiede a QPaint di creare un nuovo font utente a partire da un font True Type® di Windows®. La finestra di dialogo visualizzata sarà la seguente:

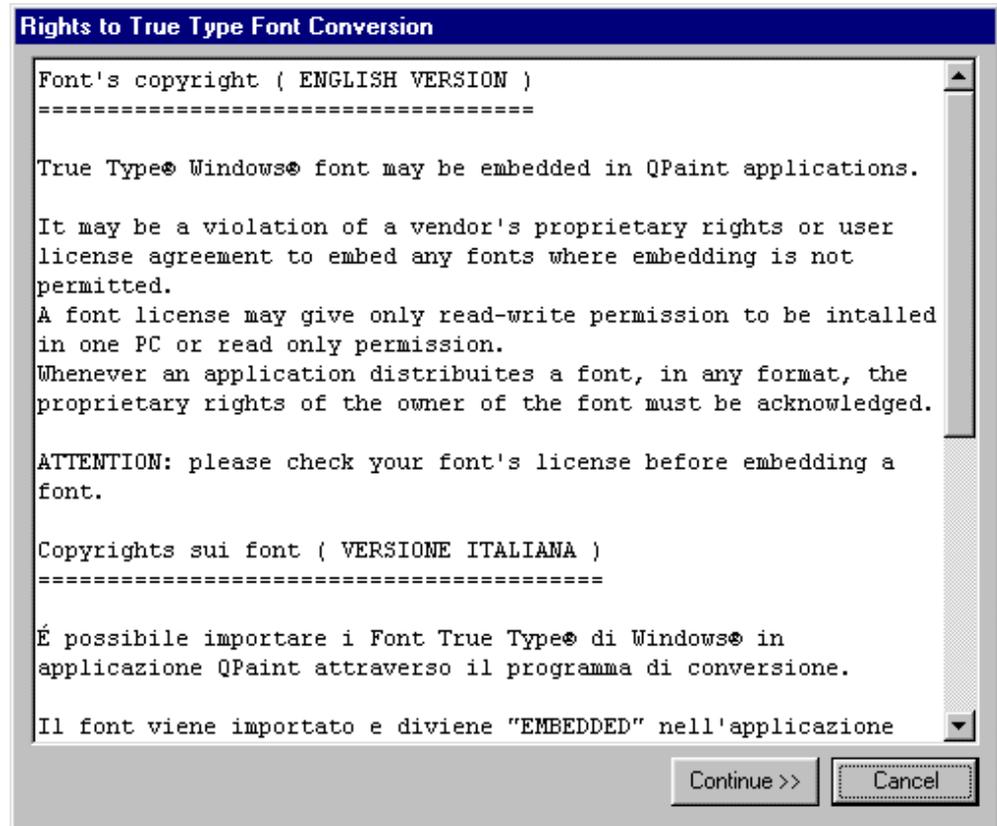


Figura 33 : finestra di dialogo Rights to True Type® Font Conversion

Come si può vedere nella fig.33, prima di poter creare un nuovo font utente a partire da un font True Type® di Windows® è necessario leggere attentamente le istruzioni per i diritti di autore inerenti l'utilizzo, il trasporto e la modifica dei font True Type® installati nel computer. È piena responsabilità dell'utente far sì che quanto ivi descritto venga rispettato. La pressione del pulsante "Continue >>" permetterà la creazione del nuovo font, mentre "Cancel" abortirà l'intera procedura.

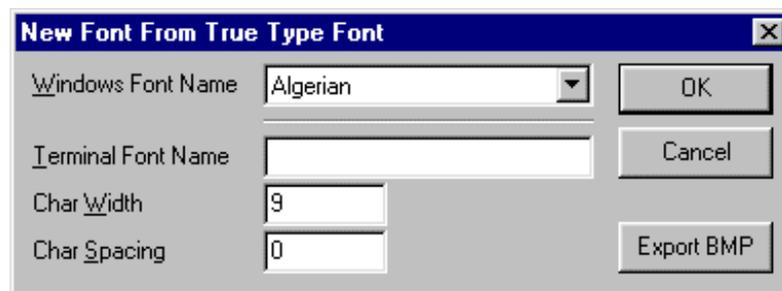


Figura 34 : finestra di dialogo New Font From True Type® Font

Come si può vedere nella fig.34, la dialog è composta dai seguenti oggetti : la casella di riepilogo "Windows Font Name", con la quale è possibile selezionare il font True Type® dal quale derivare il nuovo font utente, la casella "Terminal Font Name", nella quale si definisce il nome del font utente in creazione (es: "Display 7 segmenti"), la casella "Char Width", nella quale si definisce la larghezza in pixel dei caratteri nel font in creazione e la casella "Char Spacing", nella quale si imposta il valore di spacing carattere desiderato. Tramite il pulsante "OK" si avvia l'operazione di creazione del font, mentre "Cancel" abortisce l'intera procedura. È presente, inoltre, il pulsante "Export BMP", con il quale è possibile creare un file Bitmap monocromatico con il font risultante dalle impostazioni effettuate. Quest'ultima operazione risulta utile quando si intende generare la Bitmap di un font partendo da un font True Type® di Windows®, modificare tale Bitmap con un editor grafico quale il PaintBrush di Windows® e creare il nuovo font utente con la Bitmap modificata tramite la voce di menù "Graphics | New Font From... | New Font From Bitmap File".

Attenzione !

La qualità del font utente creato tramite la dialog "New Font From True Type Font" dipende da molti aspetti tecnici che di solito non sono visibili ad un normale utente. Di regola si può però assumere che, maggiore è la dimensione in pixel del carattere definita tramite il parametro "CharWidth", migliore è la qualità dello stesso. Importante è anche il tipo di font True Type® utilizzato in quanto maggiore è la presenza di piccoli dettagli grafici nel font, minore è la qualità del font utente ottenibile.

Alla pressione del pulsante "OK" il QPaint effettuerà una serie di controlli sui dati introdotti. Tali controlli verificheranno la bontà dei dati nei campi di introduzione. In caso vengano riscontrate irregolarità QPaint visualizzerà un box di dialogo con le indicazioni del caso.

Graphics Image Manager

Con questa voce di menù si richiede a QPaint di aprire il gestore immagini. Nel terminale è stato implementato un oggetto grafico, chiamato Image, con il quale è possibile visualizzare, su di una pagina, immagini monocromatiche. Le immagini visualizzate possono avere qualsiasi dimensione entro la grandezza del display in uso. Al fine di ridurre la quantità di memoria utilizzata dal terminale per mantenere i dati delle immagini, si è pensato di adottare un'architettura a libreria di immagini. Con questo si intende che, invece di memorizzare i dati dell'immagine direttamente nell'oggetto Image, essi verranno inseriti nella libreria e ad ogni immagine verrà assegnato un nome identificativo composto da una stringa ASCII di lunghezza massima 16 caratteri. Tale nome sarà utilizzato dall'oggetto Image per definire quale immagine della libreria si vuole visualizzare. Se si pensa che normalmente un'immagine appare in più pagine si capisce immediatamente l'enorme risparmio di memoria ottenuto con questo sistema. Inoltre durante la procedura di download verranno trasferite nel terminale solo le immagini utilizzate dagli oggetti Image. La libreria immagini del terminale viene creata per mezzo del gestore di immagini Image Manager. Con esso l'utente gestisce la libreria inserendo, rimuovendo e rinominando le immagini in formato BMP che verranno incorporate nel progetto. Il fatto di incorporare le immagini direttamente nel file del progetto permette di eliminare ogni collegamento con i file BMP originali, facilitando le operazioni di trasporto dello stesso su altri computer o media di archiviazione.

Nel caso in cui nel progetto non siano mai state inserite immagini comparirà la seguente finestra:

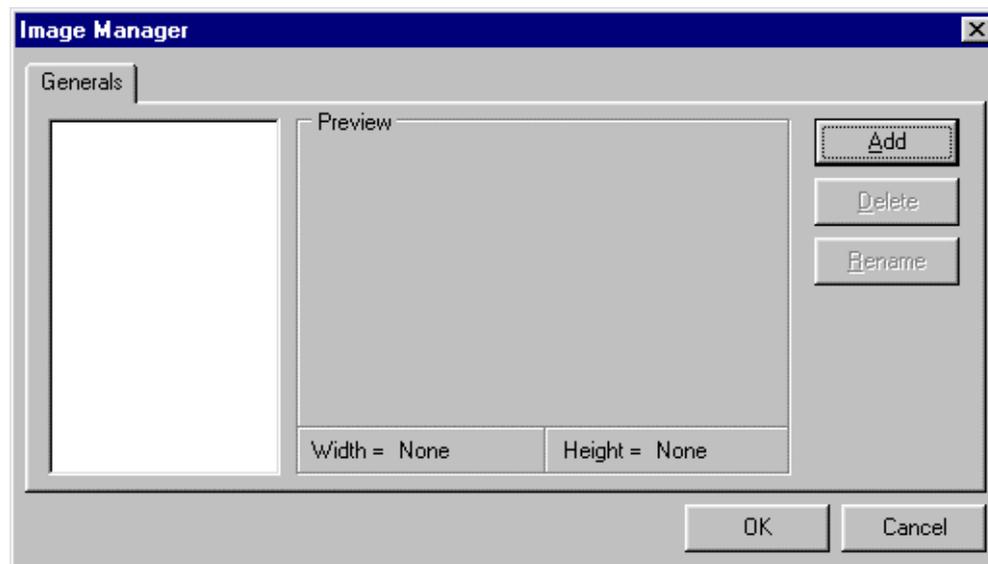


Figura 35 : finestra di dialogo Image Manager di un progetto senza immagini

Come si può vedere in fig.35, nella finestra di dialogo Image Manager abbiamo : una casella di riepilogo in cui vengono visualizzati i nomi delle immagini presenti in libreria, il box "Preview" in cui viene visualizzata l'immagine selezionata nella casella di riepilogo comprensiva dei campi Width e Height per le dimensioni in pixel della stessa, più i bottoni "Add" per l'aggiunta di una nuova immagine, "Delete" per la rimozione dell'immagine selezionata e "Rename" per la modifica del nome dell'immagine selezionata. I bottoni "Delete" e "Rename" sono abilitati solo quando esiste almeno un'immagine nella lista immagini di progetto.

La pressione del pulsante "Add" farà comparire la seguente finestra di dialogo:

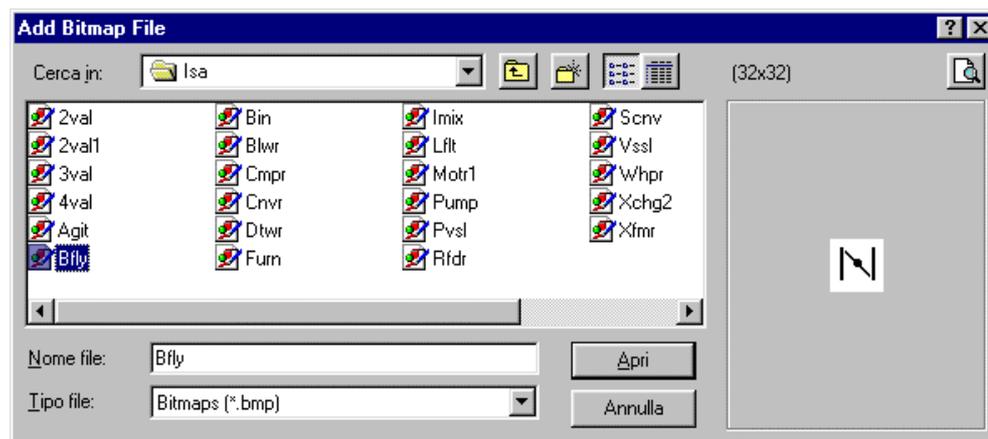


Figura 36: finestra di dialogo Add Bitmap File

Le funzionalità espletate dalla finestra di dialogo sono quelle standard normalmente fornite dallo stesso sistema operativo Windows®. A conferma dell'operazione con il tasto "Apri", Image Manager analizzerà le caratteristiche del file bitmap BMP selezionato. Nel caso in cui venga rilevata una bitmap non monocromatica comparirà un box di messaggio con la scritta "The BMP can be only monochrome !". Premendo il tasto "OK" l'intera procedura di aggiunta immagine verrà abortita. Nel caso in cui venga rilevata una bitmap di dimensioni non compatibili con il display in uso comparirà un box di messaggio con la scritta "Invalid BMP size !". Premendo il tasto "OK" l'intera procedura di aggiunta immagine verrà abortita. Una volta verificato che le caratteristiche dell'immagine selezionata sono corrette, essa verrà aggiunta in fondo alla lista immagini di progetto e le verrà assegnato un nome di default. Nella dialog Add Bitmap File non é possibile selezionare piú di un'immagine da aggiungere alla volta. Questo comunque non é un problema poiché é sufficiente ripetere l'operazione con il pulsante "Add" quante volte si desidera.

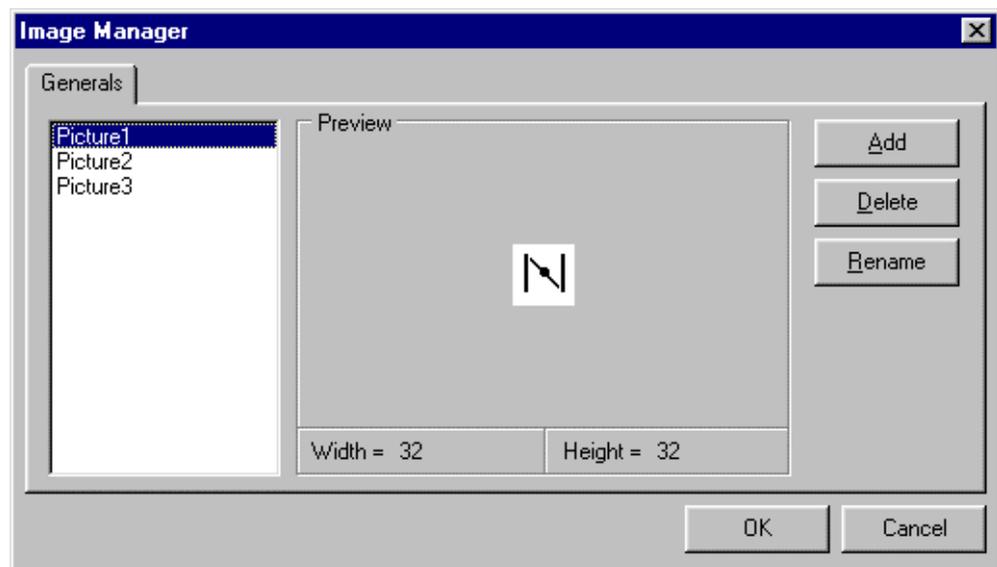


Figura 37 : finestra di dialogo Image Manager di un progetto con tre immagini

Come si può vedere nella fig.37, abbiamo un esempio in cui sono state inserite tre immagini nella lista immagini di progetto. Image Manager ha impostato per default il loro nome in : Picture1, Picture2 e Picture3. Per semplificare l'identificazione di una specifica immagine, Image Manager rende possibile modificarne il nome tramite il pulsante "Rename".

La pressione del pulsante "Rename" farà comparire la seguente finestra di dialogo:

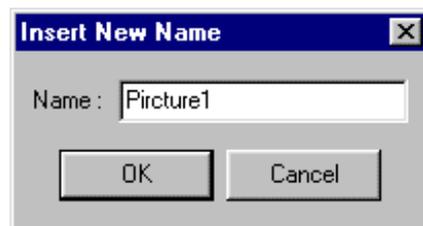


Figura 38: finestra di dialogo Insert New Name

Tramite la dialog Insert New Name é possibile introdurre un nuovo nome per l'immagine selezionata. Vi sono però alcune regole da rispettare. Il nome deve avere una lunghezza compresa tra 1 e 16 caratteri ASCII. Le lettere minuscole sono considerate differenti da quelle maiuscole. Non vi possono essere due immagini con lo stesso nome. Image Manager si occuperá di controllare che quanto introdotto sia corretto, attivando un opportuno box di messaggio nel caso non si rispetti quanto appena descritto.

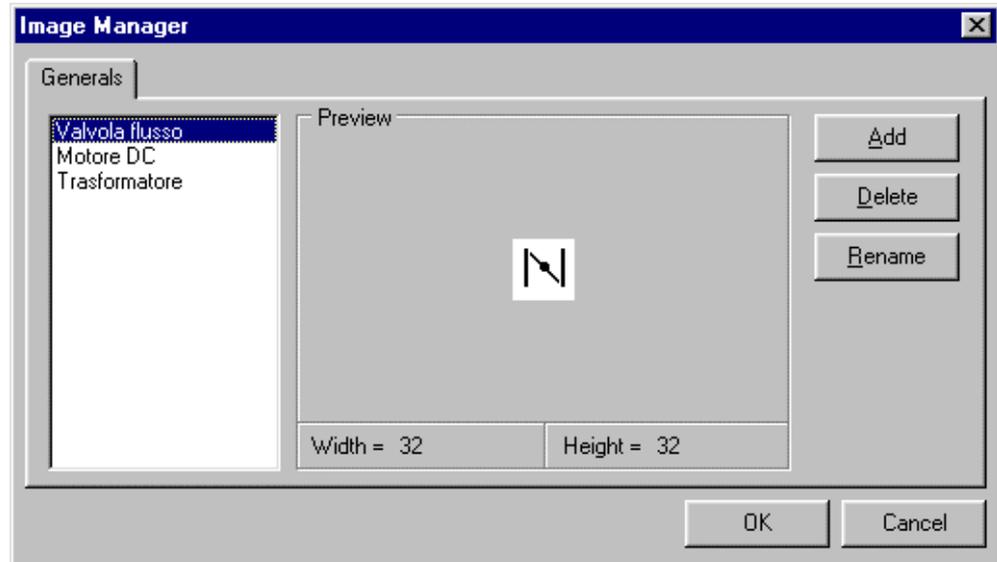


Figura 39: finestra di dialogo Image Manager di un progetto con tre immagini

Come si può vedere nella fig.39, i nomi delle tre immagini "Picture1", "Picture2" e "Picture3" sono stati modificati in "Valvola flusso", "Motore DC" e "Trasformatore". In questo modo diventerá piú semplice identificarli quando usati nell'oggetto Image.

Quando si esce da Image Manager tramite il pulsante "OK", viene verificato se le modifiche apportate comporteranno il riallineamento di alcuni oggetti Image presenti nel progetto. In questo caso comparirá un box di messaggio con la scritta "The acceptance of the changes will involve the realignment of some objects image of the project ! Are you sure of to want to continue ?" e i bottoni "Yes" e "No". Se si conferma l'operazione con "Yes", gli oggetti Image verranno riallineati e tutte le impostazioni fatte con Image Manager confermate, mentre "No" terminerá la procedura di uscita dalla dialog.

Options Communication

L'utilizzo della COM3 o della COM4 richiede che per essa non vengano utilizzati gli stessi IRQ impegnati da altri dispositivi hardware come mouse, COM1, COM2, etc.

Con questa voce di menù si richiede a QPaint di visualizzare la finestra di impostazione dei parametri di comunicazione seriale. Il QPaint utilizza la comunicazione seriale, durante la procedura di download, per trasferire il progetto al terminale utilizzando il ponte fornito dal QMove. Ulteriori informazioni inerenti i collegamenti e le configurazioni hardware sono disponibili nel manuale tecnico del terminale e del QMove. Tramite la finestra di dialogo Serial Port Settings é possibile effettuare tutte le impostazioni necessarie ad ottenere il collegamento seriale con il terminale.

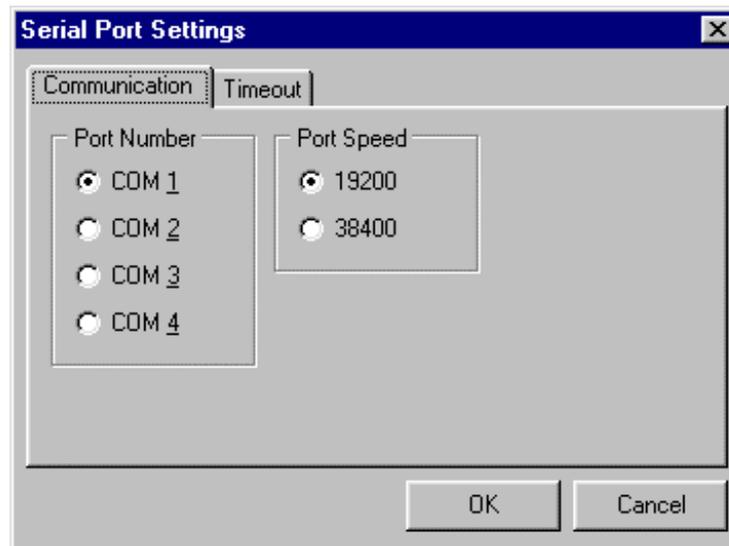


Figura 40: finestra di dialogo Serial Port Settings

Come si può vedere nella fig.40, la finestra di dialogo presenta due pagine di impostazione: Communication e Timeout. Con la prima si definiscono la porta di comunicazione seriale e la velocità di trasferimento dei dati. Con la seconda si definiscono i tempi di timeout per le varie fasi del download.

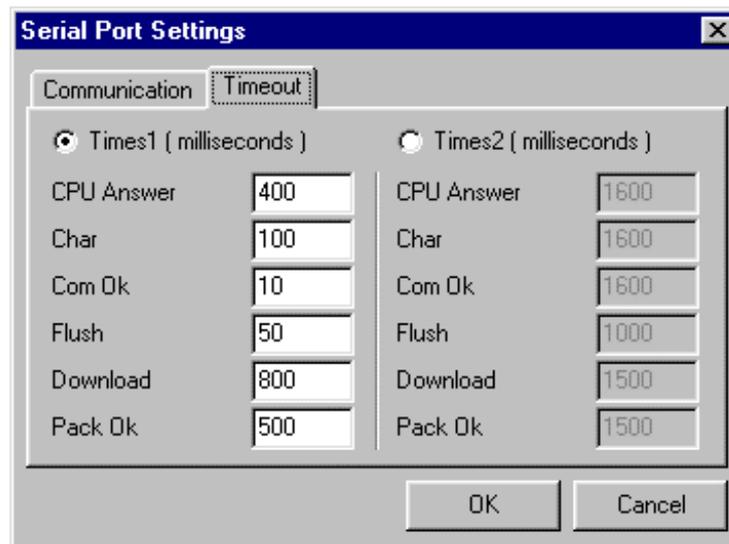


Figura 41 : finestra di dialogo Serial Port Settings

Come si può vedere in fig.41, nella pagina Timeout sono presenti due gruppi di tempi timeout. Il gruppo Times1 é da utilizzarsi quando si effettua un collegamento diretto tra PC, QMove e terminale. Il gruppo Times2 é da utilizzarsi quando si effettua un collegamento remotato tra PC, modem con QModem, QMove e terminale. Nel secondo caso sono necessari tempi di timeout maggiori dovuti ai ritardi di trasferimento introdotti dalla comunicazione con modem. La possibilitá di modificare i tempi di timeout, permette all'utente di adattare il trasferimento dei dati alle proprie condizioni di utilizzo. Segue ora una breve descrizione dei parametri di timeout:

CPU Answer timeout sulla risposta del terminale
 Char timeout fra carattere e carattere
 Com Ok timeout sulla verifica di connessione seriale
 Flush timeout su invio richiesta al terminale
 Download timeout su conclusione procedura download al terminale
 Pack Ok timeout ricezione pacchetto informazioni

Help Technical Info

Con questa voce di menú si richiede a QPaint di visualizzare la finestra delle informazioni tecniche. In essa sono elencati tutti gli oggetti OLE, librerie dinamiche e drivers utilizzati dal QPaint per il proprio funzionamento. Può essere utilizzata, su richiesta della QEM srl, nell'eventualitá di un supporto tecnico al prodotto.

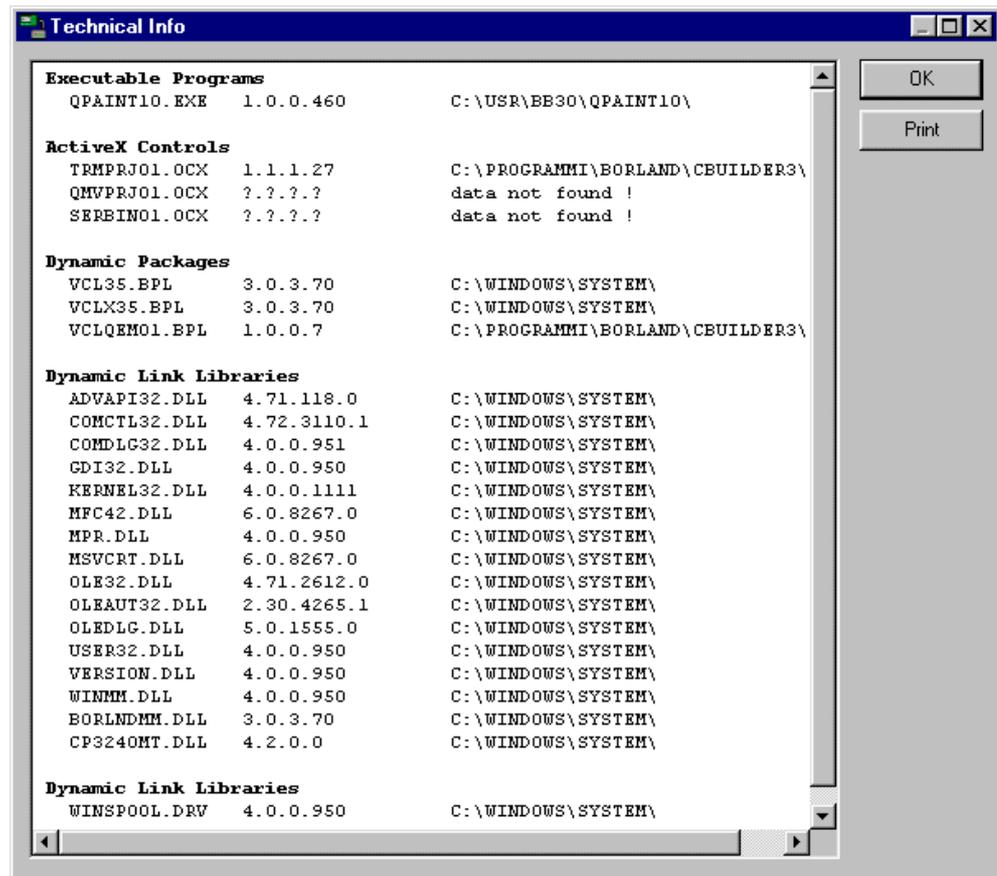


Figura 42: finestra di dialogo Technical Info

Come si può vedere in fig.42, nella dialog è presente il pulsante "Print" con il quale si potranno stampare le informazioni tecniche visualizzate. Prima di avviare la procedura di stampa assicurarsi che in Windows® vi sia installata almeno una stampante.

Help Technical Info

Con questa voce di menù si richiede a QPaint di visualizzare la finestra di About. In essa sono presenti le informazioni inerenti il prodotto, la QEM srl e gli indirizzi internet disponibili all'utente. La casella info@qem.it può essere utilizzata per richiedere informazioni generali e commerciali sui prodotti QEM. La casella soft@qem.it può essere utilizzata per richiedere informazioni ed assistenza al supporto tecnico QEM.



Figura 43 : finestra di dialogo About QPaint

Barra di gestione della pagina

Tramite la barra di gestione della pagina è possibile:

1. Scegliere la pagina da attivare specificandone il numero.
2. Scegliere la pagina da attivare scegliendola dalla lista pagine di progetto.
3. Modificare il nome associato alla pagina attiva.



Figura 44: barra di gestione della pagina

Come si può vedere in fig.44, nella barra di gestione della pagina sono presenti un campo di introduzione numerico ed una casella combinata. Nel campo di introduzione viene visualizzato il numero della pagina attiva, mentre nella casella combinata compare il nome ad essa associato. Nel campo di introduzione è possibile specificare il numero della pagina che si desidera attivare. Se viene introdotto un valore inferiore a 1 o maggiore del numero complessivo delle pagine verrà attivata l'ultima pagina del progetto. Tramite la casella combinata è possibile indicare il nome della pagina che si vuole attivare scegliendola dalla lista dei nomi pagine di progetto. Con essa si può inoltre modificare il nome associato alla pagina attiva.

N.B:

Ulteriori informazioni riguardo la gestione delle pagine sono reperibili nella sezione dedicata all'evento Goto Page.

Barra di impostazione della password di pagina

Quando si realizza un'interfaccia verso una macchina utensile vi sono alcune parametrizzazioni, come la velocità massima di lavoro attrezzo o le impostazioni degli assi di movimento, il cui accesso deve essere consentito solo a determinati operatori (tecnico manutenzione, installatore, ecc...) in modo da evitare che un inserimento di dati errati possa portare ad alterazioni delle caratteristiche di funzionamento o influire sulla sicurezza stessa della macchina. Per la protezione dei dati critici il terminale rende possibile attivare una password che impedisca l'accesso a determinate pagine del progetto da parte di utenti non autorizzati. Tale password è composta da una stringa di lunghezza massima 8 caratteri compresi nel set alfanumerico 0 ÷ 9 e A ÷ Z. Ogni pagina del progetto ha la possibilità di richiedere una propria password. Quando l'utente richiama una pagina protetta il terminale visualizza al suo posto una speciale pagina di richiesta della password. Se l'utente introduce la password corretta il terminale visualizza la pagina protetta, altrimenti resta nella pagina speciale di richiesta password. Per uscire dall'introduzione è sufficiente premere il tasto ESC, il che comporta il ritorno alla pagina precedente. Quando vi sono più pagine collegate con la stessa password di protezione l'utente deve inserire la password solo per il primo accesso, poi può muoversi tra le varie pagine protette senza altre introduzioni di password.

Tramite la barra di impostazione della password di pagina è possibile :

1. Abilitare / disabilitare la password di pagina.
2. Impostare la password della pagina.



Figura 45 : barra di impostazione della password di pagina

Come si può vedere in fig.45, nella barra di impostazione della password di pagina sono presenti un pulsante ed un campo di introduzione. Il pulsante può assumere due stati stabili. Lo stato ON abilita l'introduzione e la modifica della password di pagina. Quando la password non contiene alcun carattere il terminale non esegue nessuna richiesta di introduzione password. Lo stato OFF è utilizzato durante lo sviluppo del progetto per agevolare il debug dello stesso. Infatti, in questo stato il valore della password rimane impostato ma il terminale non ne richiede l'introduzione, agevolando quindi le operazioni di debug. Per default il pulsante è in stato di OFF. Nella sezione esempi è presentata una dimostrazione di come utilizzare la password di pagina.

Barra delle funzioni di progetto

Tramite la barra delle funzioni di progetto è possibile:

1. Salvare il progetto aperto.
2. Richiamare la finestra di dialogo "Global Events Editor" per la gestione degli eventi/azioni globali.
3. Richiamare la finestra di dialogo "Page Events Editor" per la gestione degli eventi/azioni di pagina.
4. Avviare la procedura di download del progetto nel terminale.



Figura 46 : barra delle funzioni di progetto

-  Questo pulsante è la scorciatoia alla voce di menù "File | File Save" o "File | File Save As...".
-  Questo pulsante è la scorciatoia alla voce di menù "Project | Global Events Editor".
-  Questo pulsante è la scorciatoia alla voce di menù "Project | Page Events Editor".
-  Questo pulsante è la scorciatoia alla voce di menù "Transfer | Download".

Barra delle funzioni di disegno

Tramite la barra delle funzioni di disegno é possibile:

1. Aumentare il fattore di zoom dell'area di disegno.
2. Diminuire il fattore di zoom dell'area di disegno.
3. Portare ad un livello superiore di vista l'oggetto selezionato nell'area di disegno.
4. Portare ad un livello inferiore di vista l'oggetto selezionato nell'area di disegno.
5. Cambiare il colore di sfondo utilizzato nell'area di disegno.



Figura 47: barra delle funzioni di disegno

-  Questo pulsante é la scorciatoia alla voce di menú "View | Zoom In".
-  Questo pulsante é la scorciatoia alla voce di menú "View | Zoom Out".
-  Questo pulsante é la scorciatoia alla voce di menú "Edit | Bring Up Object".
-  Questo pulsante é la scorciatoia alla voce di menú "Edit | Bring Down Object".
-  Questo pulsante é la scorciatoia alla voce di menú "View | Background Mode".

Barra degli strumenti di disegno

La barra degli strumenti di disegno é utilizzata dall'utente per creare la parte grafica delle pagine di un progetto. Con essa é possibile depositare nell'area di disegno, rappresentante il contenuto della pagina attiva, oggetti quali campi di visualizzazione/introduzione dati alfanumerici, messaggi in lingua, rettangoli ed immagini. L'inserimento di un nuovo oggetto nella pagina attiva é un'operazione molto semplice. Per prima cosa bisogna selezionare sulla barra degli strumenti di disegno, facendo click con il tasto sinistro del mouse, il tipo di oggetto che si intende aggiungere. Poi basta fare di nuovo click con il mouse sull'area di disegno nella zona dove si intende depositare il nuovo oggetto. A questo punto alcuni oggetti visualizzano automaticamente la pagina delle loro proprietá, mentre altri come i rettangoli non richiedono alcun dato aggiuntivo. Appena aggiunto il nuovo oggetto sulla pagina, il pulsante precedentemente attivo sulla barra degli strumenti di disegno ritorna in posizione di disattivazione e viene selezionato il *Pointer Object*. Questo non é propriamente un oggetto di disegno bensí un strumento di manipolazione degli oggetti già presenti sulla pagina. Con esso é infatti possibile spostare, ridimensionare, selezionare e deselezionare, etc, un oggetto sull'area di disegno. Per depositare piú di un oggetto dello stesso tipo sulla pagina, senza dover rileszionare ogni volta il corrispondente pulsante, é sufficiente mantenere premuto il tasto SHIFT sinistro della tastiera. In questo modo al termine dell'operazione di inserimento oggetto nella pagina non verrà attivato il *Pointer Object* ma rimarrá attivo il pulsante precedentemente selezionato.

In una pagina vi sono due tipi di oggetti depositabili: gli oggetti statici e gli oggetti dinamici. Per oggetti statici si intendono tutti quegli oggetti che vengono disegnati sul display solamente una volta al cambio della pagina attiva. Per oggetti dinamici si intendono tutti quegli oggetti che sono in continuo aggiornamento sul display. Oltre alla modalità di aggiornamento tra gli oggetti statici e dinamici esiste un'altra importante differenza. Ogni oggetto occupa per la propria visualizzazione una area grafica ben definita. Quando due oggetti statici sovrappongono le loro aree di ingombro, anche se parzialmente, avviene che uno è sempre visualizzato sopra all'altro. Questo è dovuto al fatto che gli oggetti statici posseggono ciascuno una proprietà detta livello che identifica l'ordine di disegno dello stesso. Un oggetto con livello 3 viene sempre visualizzato sotto ad un oggetto di livello 4. Gli oggetti dinamici invece sono sempre prioritari rispetto a quelli statici per cui verranno sempre visualizzati ad un livello superiore. Nel QPaint sono presenti due voci di menù che permettono di modificare il livello di disegno di un oggetto statico e più precisamente "Bring Up Object" e "Bring Down Object". Inoltre quando sull'area di disegno è selezionato un oggetto statico, sarà presente nella Status Bar l'indicazione del livello di disegno ad esso associato. Come si può vedere in fig.1, la barra degli strumenti di disegno è composta dai seguenti pulsanti:

-  Pulsante *Pointer Object*.
-  Pulsante *Value Object*.
-  Pulsante *ValString Object*.
-  Pulsante *String Object*.
-  Pulsante *Image Object*.
-  Pulsante *Box Object*.

Pulsante Pointer Object

Questo è l'unico pulsante della barra strumenti di disegno che non viene utilizzato per inserire un nuovo oggetto nella pagina attiva ma bensì è utilizzato per la selezione di un oggetto già presente nella pagina. L'oggetto oltre che essere selezionato può essere spostato, ridimensionato, modificato nelle proprietà e così via.

Pulsante Value Object

Con questo pulsante si richiede a QPaint di depositare sulla pagina attiva un oggetto dinamico di tipo Value. Tramite l'oggetto Value è possibile visualizzare e/o introdurre un valore numerico o alfanumerico in una variabile di riferimento. Le proprietà impostabili sono le seguenti:

Proprietá	Descrizione	Limiti e condizioni
Dec.Point (Decimal Point)	VR per la posizione del punto decimale	Se non definita disabilita la funzionalità di punto decimale.
Variable	VR per la variabile CPU QMove o terminale da visualizzare / introdurre	Se viene selezionata una variabile del gruppo DEVPAR a sola lettura la proprietá Enable Modify Mode dovrà essere OFF.
Max Variable	VR per il controllo del limite superiore in introduzione	Se non definita non viene effettuato alcun controllo sul limite superiore.
Min Variable	VR per il controllo del limite inferiore di introduzione	Se non definita non viene effettuato alcun controllo sul limite inferiore.
Enable Lead Zero Blank Mode	Flag che se ON sostituisce tutti i caratteri spazio non rappresentativi presenti in testa alla visualizzazione col carattere '0'	
Enable Sign Mode	Flag che se ON abilita l'introduzione del segno	Se il flag é ON vengono disabilitate le seguenti proprietá :
Enable Alpha Mode	Flag che se ON abilita l'oggetto a trattare dati in modalitá alfanumerica	<ul style="list-style-type: none"> - Max Variable - Min Variable - Scale Type - Scale A/B/C - Enable Sign Mode - Enable Lead Zero Blank Mode <p>La proprietá Variable potrà selezionare solamente simboli del gruppo ARRSYS e ARRGL di tipo BYTE.</p>

Proprietá	Descrizione	Limiti e condizioni
Enable Modify Mode	Flag che se ON abilita la modifica del valore indirizzato dalla proprietá Variable	Se il flag é ON la proprietá Enable Sign Mode viene abilitata.
Length	Definisce il numero di caratteri che compone l'oggetto comprendendo i caratteri utilizzati dal segno e dal punto decimale	Range = 1 ÷ numero massimo caratteri visualizzabili su una linea. Questo valore dipende principalmente dal tipo di font selezionato e dal valore di Spacing.
Scaling Mode	Definisce il tipo di scaling utilizzato dall'oggetto	<p>La proprietá Scaling puo assumere i seguenti valori :</p> <ul style="list-style-type: none"> - None : In questo caso il valore di Variable non é soggetto a nessuna correzione di scaling. I campi di Scale A/B/C sottostanti sono automaticamente disabilitati. - Global : In questo caso il valore di Variable é soggetto alla correzione con lo scaling globale definito nella voce di menú "Project Project Setup". I campi sottostanti di Scale A/B/C sono automaticamente disabilitati ma visualizzano i valori di scaling globale. - Local : In questo caso il valore di Variable é soggetto alla correzione con lo scaling definito nelle proprietá di Scale A/B/C sottostanti.
Scale A	Valore di scale A	<p>La correzione di scaling viene ottenuta con la seguente formula :</p> $Y = (Scale A / Scale B) \cdot X + Scale C$ <p>Range = $-2^{31} \div 2^{31} - 1$</p>

Scale B	Valore di scale B	La correzione di scaling viene ottenuta con la seguente formula : $Y = (\text{Scale A} / \text{Scale B}) \cdot X + \text{Scale C}$ Range = $-2^{31} \div 2^{31} - 1$
Scale C	Valore di scale C	La correzione di scaling viene ottenuta con la seguente formula : $Y = (\text{Scale A} / \text{Scale B}) \cdot X + \text{Scale C}$ Range = $-2^{31} \div 2^{31} - 1$
Left	Attributo posizione lato sinistro oggetto	Range = 0 ÷ 239 pixel
Top	Attributo posizione lato superiore oggetto	Range = 0 ÷ 127 pixel
Font Name	Nome del font utente o terminale in uso	
Spacing	Spacing di visualizzazione del font	Il range di valori dipende dal font in uso
Reverse Mode	Flag che se ON inverte il campo di visualizzazione dell'oggetto	
Blink	Flag che se ON abilita il lampeggio del campo di visualizzazione dell'oggetto	

N.B: VR sta a significare "variabile di riferimento".

Pulsante ValString Object

Con questo pulsante si richiede a QPaint di depositare sulla pagina attiva un oggetto dinamico di tipo ValString. Tramite l'oggetto ValString é possibile visualizzare un array di messaggi in lingua indicizzati da una variabile di riferimento. Le proprietà impostabili sono le seguenti:

Proprietá	Descrizione	Limiti e condizioni
First Message	Definisce l'indice associato al primo messaggio nell'array messaggi in lingua	Range = $-2^{31} \div 2^{31} - \text{MaxMessages}$
Max Messages	Definisce il numero di messaggi presenti nell'array messaggi in lingua	Range = $1 \div 999$
Active Language	Definisce la lingua messaggi in uso nella griglia di introduzione messaggi	
Active Message	Definisce l'indice del messaggio in uso nella griglia introduzione messaggi	Range = $1 \div$ numero massimo caratteri visualizzabili su una linea. Questo valore dipende principalmente dal tipo di font selezionato e dal valore di Spacing.
Length	Definisce il numero di caratteri che compone l'oggetto comprendendo i caratteri utilizzati dal segno e dal punto decimale	
Variable	VR per la variabile CPU QMove o terminale da visualizzare / introdurre	
Enable Modify Mode	Flag che se ON abilita la modifica del valore indirizzato dalla proprietà Variable	Range = $0 \div 239$ pixel
Left	Attributo posizione lato sinistro oggetto	Range = $0 \div 127$ pixel
Top	Attributo posizione lato superiore oggetto	
Font Name	Nome del font utente o terminale in uso	Il range di valori dipende dal font in uso
Spacing	Spacing di visualizzazione del font	

Proprietá	Descrizione	Limiti e condizioni
Reverse Mode	Flag che se ON inverte il campo di visualizzazione dell'oggetto	
Blink	Flag che se ON abilita il lampeggio del campo di visualizzazione dell'oggetto	

Nella finestra di dialogo "ValString Object Property Editor" sono presenti i pulsanti "Message" e "Language" con i quali é possibile scegliere due modalitá di introduzione dei messaggi. Facendo click sul primo si abilita l'introduzione con prioritá sui messaggi. Ció sta a significare che nella griglia di introduzione messaggi vi sono tutti i messaggi disponibili per la lingua selezionata nella casella di riepilogo "Active Language". In questo modo si hanno sotto controllo tutti i messaggi disponibili per la lingua attiva. Quando si crea un oggetto nuovo questa é la modalitá di introduzione impostata di default. Premendo il pulsante "Language" si abilita l'introduzione messaggi in modalitá lingua per cui nella griglia verrá visualizzato un messaggio alla volta ma in tutte le lingue disponibili. In cooperazione con i pulsanti appena citati sulla dialog compaiono alternativamente i campi "Active Language" e "Active Message". Nella dialog é presente anche il pulsante "Unlock" o "Lock". Questo pulsante ha il compito di sbloccare il controllo della lunghezza messaggio in introduzione e porla alla massima dimensione visualizzabile. In questo modo é possibile introdurre dei messaggi senza preoccuparsi anticipatamente della lunghezza in caratteri che occuperanno.

Pulsante String Object

Con questo pulsante si richiede a QPaint di depositare sulla pagina attiva un oggetto statico di tipo String. Tramite l'oggetto String é possibile visualizzare un messaggio in lingua. Le proprietá impostabili sono le seguenti:

Proprietá	Descrizione	Limiti e condizioni
Length	Definisce il numero di caratteri che compone l'oggetto comprendendo i caratteri utilizzati dal segno e dal punto decimale	Range = 1 ÷ numero massimo caratteri visualizzabili su una linea. Questo valore dipende principalmente dal tipo di font selezionato e dal valore di Spacing.
Left	Attributo posizione lato sinistro oggetto	Range = 0 ÷ 239 pixel
Top	Attributo posizione lato superiore oggetto	Range = 0 ÷ 127 pixel
Font Name	Nome del font utente o terminale in uso	
Spacing	Spacing di visualizzazione del font	Il range di valori dipende dal font in uso

Proprietà	Descrizione	Limiti e condizioni
Transparent Mode	Flag che se ON abilita la trasparenza dello sfondo di visualizzazione dell'oggetto	
Reverse Mode	Flag che se ON inverte il campo di visualizzazione dell'oggetto	

Nella finestra di dialogo "String Object Property Editor" è presente anche il pulsante "Unlock" o "Lock". Questo pulsante ha il compito di sbloccare il controllo della lunghezza messaggio in introduzione e porla alla massima dimensione visualizzabile. In questo modo è possibile introdurre dei messaggi senza preoccuparsi anticipatamente della lunghezza in caratteri che occuperanno.

Pulsante Image Object

Con questo pulsante si richiede a QPaint di depositare sulla pagina attiva un oggetto statico di tipo Image. Tramite l'oggetto Image è possibile visualizzare una immagine bitmap monocromatica. Le proprietà impostabili sono le seguenti:

Proprietà	Descrizione	Limiti e condizioni
Image Name	Definisce l'immagine da visualizzare specificando il nome di una immagine presente nella libreria immagini del progetto creata con la voce di menù "Graphics Image Manager"	Se il nome dell'immagine specifica non è presente nella libreria immagini verrà visualizzata un'immagine di default.
Left	Attributo posizione lato sinistro oggetto	Range = 0 ÷ 239 - Width immagine.
Top	Attributo posizione lato superiore oggetto	Range = 0 ÷ 127 - Height immagine.

Pulsante Box Object

Con questo pulsante si richiede a QPaint di depositare sulla pagina attiva un oggetto statico di tipo Box. Tramite l'oggetto Box é possibile visualizzare una serie di rettangoli. Le proprietà impostabili sono le seguenti:

Proprietà	Descrizione	Limiti e condizioni
Left	Attributo posizione lato sinistro oggetto	Range = 0 ÷ 239 - Width
Top	Attributo posizione lato superiore oggetto	Range = 0 ÷ 127 - Height
Width	Attributo larghezza oggetto	Range = 5 ÷ 240
Height	Attributo altezza oggetto	Range = 5 ÷ 128
Back Color	Attributo colore sfondo oggetto	
Color	Attributo colore disegno bordo oggetto	
Style	Attributo modalità disegno bordo oggetto	

I font del terminale

Il terminale é stato progettato per lavorare con due gruppi di font: i font di sistema e i font utente. I font di sistema risiedono nella ROM del terminale e sono sempre disponibili. I font utente risiedono nella directory "\Fonts" del QPaint sotto forma di file QTF ed FNT e vengono trasferiti nel terminale durante la procedura di download. Anche i font di sistema hanno i loro file QTF e FNT nella directory "\Fonts" del QPaint ma questi vengono utilizzati solo per le operazioni interne di visualizzazione ed introduzione dati. L'utente può facilmente creare nuovi tipi di font partendo da file Bitmap o derivandone di nuovi direttamente dai font TTF di Windows®. La possibilità di creare un font utente prelevando le informazioni direttamente da un file Bitmap espande notevolmente le possibilità grafiche del terminale stesso. Infatti é possibile realizzare delle animazioni realizzando un font in cui i caratteri sono in realtà le varie parti dell'animazione che si vuole visualizzare. Ulteriori informazioni riguardo le animazioni effettuate tramite font utente sono disponibili negli esempi allegati.

I font utente disponibili sono presenti sotto forma di file QTF e FNT nella directory "\Fonts" del QPaint. Quando si trasporta un progetto che utilizza font utente in un altro PC é necessario copiare nella directory "\Fonts" di tale PC i file QTF ed FNT dei font utilizzati dal progetto. Una volta copiato il progetto e i file QTF e FNT dei font utente richiesti é necessario uscire da Windows® e riavviare il computer per rendere attive le nuove impostazioni.

La rimozione di un font utente é una operazione che deve essere fatta solo in caso di reale necessità. Le operazioni da eseguire sono le seguenti:

1. Chiudere tutte le applicazioni avviate.
2. Con il Gestore risorse eliminare i file QTF e FNT del font da rimuovere.
3. Uscire da Windows®.
4. Riavviare il computer.

Le variabili di riferimento

Tramite le variabili di riferimento QPaint permette all'utente di accedere ai vari gruppi di variabili disponibili nella CPU del QMove e nel terminale. Una variabile di riferimento é composta da tre campi introduzione : il campo Base, il campo First Index ed il campo Second Index. Il campo Base ha lo scopo di definire il nome stesso della variabile a cui si vuole accedere. Il campo First Index ha lo scopo di definire il primo indice mentre Second Index definisce il secondo. Avremo quindi che:

Data Types	Base	First Index	Second Index
SYSTEM	variabile CPU QMove		
GLOBAL	variabile CPU QMove		
ARRSYS	array CPU QMove	indice elemento array	
ARRGBL	array CPU QMove	indice elemento array	
TIMER	timer CPU QMove		
STATIC	datagroup CPU QMove	indice programma	
INDEX	datagroup CPU QMove	indice programma	indice passo di programma
DEVPAR	parametri device CPU QMove		
IO	input / output CPU QMove		
INTERNAL	variabile interne		
TERMINAL	variabile terminale		
CONST	valore costante		

I campi introduzione per variabili di riferimento sono presenti in molti oggetti come Value e ValString, eventi ed azioni. I valori introducibili per tali oggetti sono regolati da apposite tabelle specificate in dettaglio nell'appendice A di questo documento. Per semplificare le operazioni di introduzione dei campi Base, First Index e Second Index, QPaint mette a disposizione un'apposita dialog di selezione dei simboli richiamabile tramite la pressione del tasto F5 quando l'introduzione é attiva su uno dei tre campi di una variabile di riferimento.

	Base	First Index	Second Index
Variable	<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 48 : campi introduzione variabile di riferimento

Appendice

Tabelle riassuntive regole variabile di riferimento

In queste tabelle sono riportate le regole da applicare per la costruzione delle variabili di riferimento. Ad ogni tabella é associato un numero che dovrà essere utilizzato per riferirsi all'oggetto interessato.

Oggetto Value

Dec.Point	tabella 4
Variable	con Enable Alpha Mode ON = tabella 3 con Enable Alpha Mode OFF = tabella 2
Max Variable	tabella 5
Min Variable	tabella 5

Oggetto ValString

Variable	tabella 1
----------	-----------

Evento OnChangeVar tabella 2

Evento OnVar tabella 1

Azione GotoPage tabella 4

Azione SetVariable

Source	tabella 6
Destination	tabella 7

Data Types	1	F	B	W	L	S	2	F	B	W	L	S	3	F	B	W	L	S
SYSTEM		X	X	X	X		X	X	X	X	X							
GLOBAL		X	X	X	X		X	X	X	X	X							
ARRSYS			X	X	X			X	X	X	X							
ARRGBL			X	X	X			X	X	X	X				X			
TIMER															X			
STATIC		X	X	X	X		X	X	X	X	X							
INDEX		X	X	X	X		X	X	X	X	X							
DEVPAR		X	X	X	X		X	X	X	X	X							
IO		X	X	X	X		X	X	X	X								
INTERNAL		X	X	X	X		X	X	X	X	X							
TERMINAL		X	X	X	X		X	X	X	X	X							
CONST																		

Data Types	4	F	B	W	L	S	5	F	B	W	L	S	6	F	B	W	L	S
SYSTEM		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
GLOBAL		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
ARRSYS			X	X	X			X	X	X	X			X	X	X	X	X
ARRGBL			X	X	X			X	X	X	X			X	X	X	X	X
TIMER																		
STATIC		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
INDEX		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
DEVPAR		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
IO		X	X	X	X		X	X	X	X			X	X	X	X	X	
INTERNAL		X	X	X	X		X	X	X	X	X		X	X	X	X	X	X
TERMINAL											X	X	X	X	X	X	X	X
CONST					X					X	X					X	X	

Data Types	7	F	B	W	L	S	8	F	B	W	L	S
SYSTEM		X	X	X	X	X		X	X	X	X	
GLOBAL		X	X	X	X	X		X	X	X	X	
ARRSYS			X	X	X	X						
ARRGBL			X	X	X	X						
TIMER												
STATIC		X	X	X	X	X						
INDEX		X	X	X	X	X						
DEVPAR		X	X	X	X	X		X	X	X	X	
IO		X	X	X	X							
INTERNAL		X	X	X	X			X	X	X	X	
TERMINAL		X	X	X	X	X						
CONST												X

Stati funzionali visualizzati tramite interfaccia standard

Per semplificare l'utilizzo di QPaint é stato introdotta un'interfaccia standard con la quale il programma comunica all'utente l'ingresso in particolari stati funzionali. Gli stati notificati si dividono in 5 gruppi:

- | | |
|--------------------------|---|
| 1. <i>Informazioni</i> | Tramite questo tipo di stato il QPaint comunica all'utente informazioni riguardanti l'operazione in esecuzione. |
| 2. <i>Richiesta dati</i> | Tramite questo tipo di stato il QPaint richiede all'utente informazioni riguardanti l'operazione in esecuzione. |
| 3. <i>Pericolo</i> | Tramite questo tipo di stato il QPaint indica all'utente che l'operazione in esecuzione potrebbe portare a risultati non desiderati. |
| 4. <i>Errore</i> | Tramite questo tipo di stato il QPaint indica all'utente che é stato riscontrato un errore nell'operazione in esecuzione. |
| 5. <i>Errore fatale</i> | Tramite questo tipo di stato il QPaint indica all'utente che é stato riscontrato un grave errore funzionale il quale non permette un corretto funzionamento del programma e ne precede, quindi, la conclusione forzata. |

Viene ora descritta l'interfaccia standard dei messaggi fornita da QPaint:

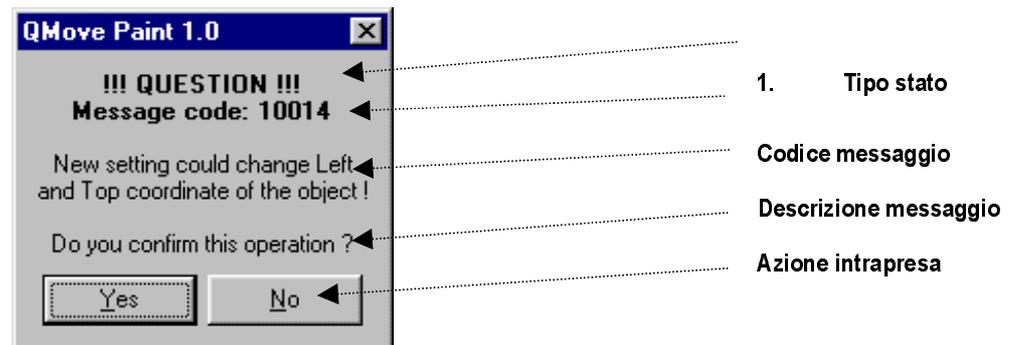


Figura xx : interfaccia standard messaggi

Come si può vedere nella fig.xx, l'interfaccia standard dei messaggi altro non é che un box di dialogo di Windows® opportunatamente organizzato. Vi sono 4 campi di testo che servono per la descrizione delle varie informazioni, piú una serie di bottoni per le operazioni che l'utente può effettuare. Vediamo ora in dettaglio il significato di quanto presente:

- | | |
|----------------------------|--|
| 1. <i>Tipo stato</i> | <p>Descrive il tipo di stato in cui é entrato QPaint. Può valere:</p> <p>!!! INFORMATION !!! per un stato informazione.</p> <p>!!! QUESTION !!! per un stato di richiesta dati.</p> <p>!!! WARNING !!! per un stato di pericolo.</p> <p>!!! ERROR !!! per un stato d'errore.</p> <p>!!! FATAL ERROR !!! per un stato d'errore fatale.</p> |
| 2. <i>Codice messaggio</i> | <p>Contiene il numero identificativo del messaggio intervenuto. Tramite questo numero é possibile accedere alla documentazione approfondita dello stato presente in questa appendice.</p> |

- 3. *Descrizione messaggio* Descrive in modo sintetico lo stato intervenuto.
- 4. *Azione intrapresa* Viene descritta l'azione intrapresa di risposta allo stato in cui si trova QPaint. Può contenere anche una estensione alla descrizione messaggio o una richiesta specifica all'utente. Questo campo non è disponibile in tutti gli stati.
- 5. *Operazioni possibili* Sono disponibili uno o più bottoni per le scelte dell'utente.

Nella tabella a seguire sono riportati e dettagliatamente descritti tutti i messaggi visualizzati da QPaint tramite il box di dialogo.

Codice stato	Tipo stato	Descrizione stato
10000	QUESTION	<p><i>Vuoi veramente modificare il nome della pagina attiva ?</i></p> <p>Si entra in questo stato quando l'utente introduce un nuovo nome per la pagina attiva.</p>
10001	WARNING	<p>Nome pagina introdotto invalido poiché già utilizzato da un'altra pagina ! Introdurre un nuovo nome !</p> <p>Si entra in questo stato quando l'utente introduce un nuovo nome per la pagina attiva e questo risulta essere già stato utilizzato da un'altra pagina.</p>
10002	QUESTION	<p>La pagina attiva verrà persa ! Sei sicuro di volere continuare ?</p> <p>Si entra in questo stato quando l'utente esegue la procedura per la rimozione della pagina attiva.</p>
10003	FATAL ERROR	<p>Errore interno fatale e non recuperabile ! Il programma verrà terminato !</p> <p>Si entra in questo stato quando in QPaint si verifica un errore non gestibile e che comporta una inevitabile conclusione dello stesso. Tutte le modifiche apportate al progetto successive all'ultimo salvataggio verranno perse.</p>
10004	ERROR	<p>Errore nella creazione del file !</p> <p>Si entra in questo stato quando QPaint non riesce a salvare il file di progetto. Nella maggior parte dei casi, questo errore, è dovuto al tentativo di scrivere un file protetto in scrittura o dall'accesso ad un disco non disponibile o in modalità a sola lettura.</p>

Codice stato	Tipo stato	Descrizione stato
10005	ERROR	<p>Errore nell'apertura del file !</p> <p>Si entra in questo stato quando QPaint non riesce a leggere il file di progetto. Nella maggior parte dei casi, questo errore, è dovuto al tentativo di leggere un progetto inesistente o dall'accesso ad un disco non disponibile.</p>
10006	INFORMATION	<p>File di progetto non valido ! Il progetto non verrà caricato !</p> <p>Si entra in questo stato quando QPaint rileva che il file di progetto in lettura non è valido.</p>
10008	INFORMATION	<p>Versione file di progetto non compatibile ! Il progetto non verrà caricato !</p> <p>Si entra in questo stato quando QPaint rileva che la versione del file di progetto in lettura non è compatibile.</p>
10009	FATAL ERROR	<p>Errore durante il caricamento dei font terminale ! Prova a reinstallare QPaint ! Il programma verrà terminato !</p> <p>Si entra in questo stato quando QPaint, durante la procedura di avvio, rileva la mancanza dei file font terminale nella directory Fonts. In questo caso si consiglia di reinstallare QPaint nella stessa directory in cui era stato precedentemente installato.</p>
10013	QUESTION	<p>Le nuove impostazioni potrebbero troncare il testo in eccesso di alcuni messaggi in lingua ! Vuoi confermare questa operazione ?</p> <p>Si entra in questo stato quando QPaint rileva la necessità di troncare il testo in eccesso di alcuni messaggi in lingua. Tale condizione si verifica durante la modifica della lunghezza testo o delle proprietà font degli oggetti String e ValString.</p>

Codice stato	Tipo stato	Descrizione stato
10014	QUESTION	<p>Le nuove impostazioni potrebbero modificare le proprietà Left e Top dell'oggetto ! Vuoi confermare questa operazione ?</p> <p>Si entra in questo stato quando QPaint rileva la necessità di modificare le proprietà Left e Top dell'oggetto attivo affinché rimanga entro la dimensione disponibile nell'area di disegno.</p>
10019	WARNING	<p>Nome del font utente non valido ! Valore vuoto o presenti i seguenti caratteri non validi ' < > : " / \ ' . Introduci un nome valido.</p> <p>Si entra in questo stato quando QPaint rileva che il nome del font utente appena introdotto è vuoto o presenta uno o più caratteri non validi.</p>
10020	WARNING	<p>Nome del font utente già utilizzato da un font di Windows® ! Introduci un nome nuovo !</p> <p>Si entra in questo stato quando QPaint rileva che il nome del font utente appena introdotto è già utilizzato da un font di Windows®.</p>
10021	WARNING	<p>Nome del font utente già utilizzato ! Introduci un nome nuovo !</p> <p>Si entra in questo stato quando QPaint rileva che il nome del font utente appena introdotto è già presente nella lista dei nomi font utente.</p>
10023	ERROR	<p>File BMP non valido ! L'immagine non verrà aggiunta !</p> <p>Si entra in questo stato quando QPaint rileva che il file grafico in lettura non è di tipo bitmap BMP.</p>

Codice stato	Tipo stato	Descrizione stato
10024	ERROR	<p>Dimensione file BMP non valida ! L'immagine non verrà aggiunta !</p> <p>Si entra in questo stato quando QPaint rileva che le dimensioni in altezza e/o larghezza del file grafico in lettura superano le misure dell'area di disegno. La larghezza e l'altezza massima sono fissate rispettivamente a 240 x 128 pixel.</p>
10025	ERROR	<p>Il file BMP può essere solamente di tipo monocromatico. L'immagine non verrà aggiunta !</p> <p>Si entra in questo stato quando QPaint rileva che il file grafico in lettura non è di tipo monocromatico.</p>
10026	WARNING	<p>Nome per l'immagine non valido poiché vuoto ! Introduci un nome valido !</p> <p>Si entra in questo stato quando QPaint rileva che il nome dell'immagine appena introdotto è vuoto.</p>
10027	WARNING	<p>Nome già utilizzato da un'altra immagine ! Introduci un nome nuovo !</p> <p>Si entra in questo stato quando QPaint rileva che il nome dell'immagine appena introdotto è già presente nella lista dei nomi immagine.</p>
10028	QUESTION	<p>L'accettazione delle modifiche effettuate potrebbe comportare il riallineamento di alcuni oggetti immagine presenti nel progetto ! Sei sicuro di volere continuare la procedura ?</p> <p>Si entra in questo stato quando QPaint rileva che la modifica di una o più immagini nella finestra di dialogo "Image Manager" potrebbe comportare la necessità di spostare uno o più immagini presenti nel progetto.</p>

Codice stato	Tipo stato	Descrizione stato
10029	QUESTION	<p>Vuoi veramente modificare la password della pagina attiva ?</p> <p>Si entra in questo stato quando l'utente introduce una nuova password per la pagina attiva.</p>
10030	INFORMATION	<p>Correntemente per l'esecuzione di questo programma sono necessari almeno 256 colori di sistema !</p> <p>Il programma verrà terminato !</p> <p>Si entra in questo stato quanto QPaint, durante la procedura di avvio, rileva che la configurazione video di Windows® é impostata a meno di 256 colori di sistema.</p>
10031	ERROR	<p>File di progetto non trovato !</p> <p>Si entra in questo stato quando QPaint, durante la procedura di avvio, rileva che il nome del file di progetto impostato nella linea di comando non corrisponde ad un file realmente esistente.</p>
10032	WARNING	<p>Nome variabile interna non valido poiché vuoto !</p> <p>Introduci un nome valido !</p> <p>Si entra in questo stato quando QPaint rileva che il nome della variabile interna appena introdotto é vuoto.</p>
10033	WARNING	<p>Nome già utilizzato da un'altra variabile interna !</p> <p>Introduci un nome nuovo !</p> <p>Si entra in questo stato quando QPaint rileva che il nome della variabile interna appena introdotto é già presente nella lista dei nomi variabili interne.</p>
10034	QUESTION	<p>Vuoi salvare le modifiche del progetto ?</p> <p>Si entra in questo stato quando QPaint rileva che si sta scaricando un progetto modificato e richiede all'utente se desidera salvare le modifiche apportate.</p>

Codice stato	Tipo stato	Descrizione stato
10036	INFORMATION	<p>Impossibile accedere ai dati di una pagina rimossa ! Esegui la procedura di Check Project per aggiornare la Report Window !</p> <p>Si entra in questo stato quando si seleziona un item della report Window riferito ad una pagina che é stata rimossa.</p>
10037	INFORMATION	<p>Impossibile accedere ai dati di un evento rimosso ! Esegui la procedura di Check Project per aggiornare la Report Window !</p> <p>Si entra in questo stato quando si seleziona un item della report Window riferito ad un evento che é stato rimosso.</p>
10038	INFORMATION	<p>Impossibile accedere ai dati di un'azione rimossa ! Esegui la procedura di Check Project per aggiornare la Report Window !</p> <p>Si entra in questo stato quando si seleziona un item della report Window riferito ad un'azione che é stata rimossa.</p>
10039	INFORMATION	<p>Impossibile accedere ai dati di un oggetto rimosso! Esegui la procedura di Check Project per aggiornare la Report Window!</p> <p>Si entra in questo stato quando si seleziona un item della report Window riferito ad un oggetto che é stato rimosso.</p>
10040	ERROR	<p>Il programma QPaint ha rilevato che nel progetto in caricamento sono utilizzati font non disponibili in questo computer! I font richiesti hanno i seguenti type name:</p> <p>Si entra in questo stato quando QPaint rileva che il progetto in lettura necessita di uno o piú font utente non disponibili nel computer. Nella dialog vengono riportate le caratteristiche di ogni font mancante comprensive di nome font, altezza, larghezza e spacing carattere.</p>

Codice stato	Tipo stato	Descrizione stato
10042	ERROR	<p>Il file bitmap del font non é stato trovato ! Seleziona un nome file valido !</p> <p>Si entra in questo stato quando QPaint non riesce a trovare il file bitmap specificato.</p>
10043	ERROR	<p>Il file bitmap per il font non é valido ! La bitmap deve essere monocromatica !</p> <p>Si entra in questo stato quando QPaint rileva che il file bitmap in lettura non é monocromatico.</p>
10044	ERROR	<p>La dimensione del file bitmap per il font non é valida! Il range di valori validi per la larghezza della bitmap va da 96 a 512 pixel ! Il range di valori validi per l'altezza della bitmap va da 84 a 784 pixel !</p> <p>Si entra in questo stato quando QPaint rileva che la dimensione del file bitmap in lettura non é entro i limiti consentiti.</p>
10045	INFORMATION	<p>Il file bitmap per il font é stato salvato !</p> <p>Si entra in questo stato quando QPaint termina la creazione del file bitmap per il font.</p>

Esempi applicativi Qpaint 1.0

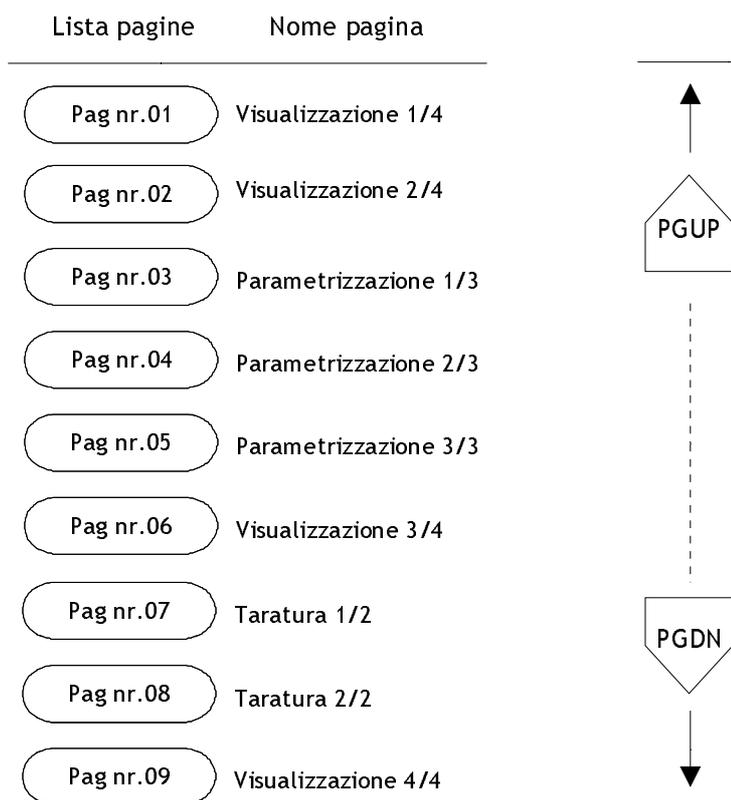
In questo documento sono riportati alcuni esempi applicativi.

Scroll personalizzato pagine di visualizzazione

Per default il terminale utilizza i tasti PGUP e PGDN per cambiare la pagina di visualizzazione. Se il progetto prevede molte pagine questi tasti forniscono uno scroll sequenziale e ordinato di tutte le pagine appartenenti al progetto.

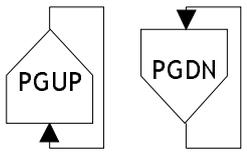
Molte volte esiste la necessità di vincolare lo scroll all'interno di solo determinate pagine ad esempio le pagine di parametrizzazione e le pagine di taratura. Inoltre in alcuni casi si deve proteggere l'accesso a queste pagine con una password. Vedremo in questo esempio come realizzare tutto questo.

Supponiamo di avere un progetto composto da 4 pagine di visualizzazione normale, 3 pagine di parametrizzazione e 2 pagine di taratura. Supponiamo inoltre che la posizione delle pagine di parametrizzazione e taratura non coincida con l'inizio o la fine della lista pagine. Se eseguiamo un "esploso" del progetto risulta:



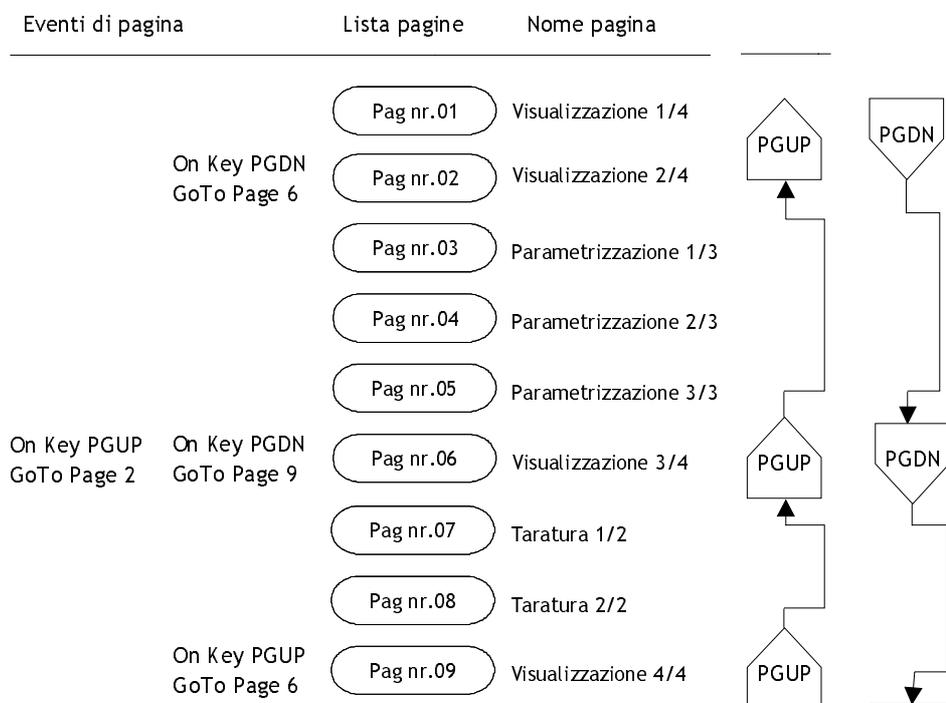
Vogliamo che una volta entrati nella pagina nr.03 "Parametrizzazione 1/3" i tasti PGDN e PGUP eseguano lo scroll solo tra le pagine di parametrizzazione cioè le nr.03 , nr.04 e nr.05. Se invece si entra nella pagina nr.07 i tasti PGDN e PGUP devono eseguire lo scroll solo tra le pagine di taratura cioè la nr.07 e nr.08. Su questo gruppo di pagine vogliamo inoltre che la successione sia circolare cioè che raggiunta l'ultima pagina si passi, con un ulteriore incremento, alla prima e viceversa.

Per fare questo é sufficiente "sovraccaricare" il funzionamento dei tasti PGUP e PGDN nelle pagine in cui si intende modificare il funzionamento di default descritto in precedenza. In questo caso le pagine in cui si deve modificare il funzionamento sono: la pagina nr.03 perché il tasto PGUP non deve eseguire nulla, la pagina nr.05 perché tasto PGDN deve essere disabilitato, la pagina nr.07 perché il tasto PGUP deve richiamare la nr.08 e la pagina nr.08 perché il tasto PGDN deve richiamare la nr.07. Per sovraccaricare il funzionamento si imposta nella pagina interessata un evento di pagina di tipo "On Key" al quale si associa una azione "Goto Page" alla pagina interessata. Per disabilitare il funzionamento invece basta definire l'evento di pagina senza associare nessuna azione. La programmazione degli eventi diventa così:

Eventi di pagina	Lista pagine	Nome pagina	
	Pag nr.01	Visualizzazione 1/4	
	Pag nr.02	Visualizzazione 2/4	
On Key PGUP	Pag nr.03	Parametrizzazione 1/3	
	Pag nr.04	Parametrizzazione 2/3	
On Key PGDN	Pag nr.05	Parametrizzazione 3/3	
	Pag nr.06	Visualizzazione 3/4	
On Key PGUP GoTo Page 8	Pag nr.07	Taratura 1/2	
On Key PGDN GoTo Page 7	Pag nr.08	Taratura 2/2	
	Pag nr.09	Visualizzazione 4/4	

Per uscire dalle pagine di parametrizzazione o taratura bisogna programmare un tasto apposito che esegua un salto alle pagine di visualizzazione.

Se si vuole mantenere una sequenza ordinata di scroll tra le pagine di visualizzazione (che invece non hanno una disposizione ordinata), bisogna "sovraccaricare" il funzionamento dei tasti PGUP e PGDN nel seguente modo:



É importante ricordare che nella definizione dell'azione "Goto Page" é possibile utilizzare anche il nome della pagine anziché il numero in modo da non dover modificare l'azione in caso di inserimento o rimozione di una pagina.

Nel caso in cui si voglia controllare l'accesso al gruppo di pagine di parametrizzazione e taratura con password bisogna procedere nel seguente modo: per tutte le pagine apparteneti al gruppo (nr.03, 04 e 05 per la parametrizzazione) bisogna abilitare e introdurre lo stesso valore di password nell'apposito campo. Questo perché una volta introdotta la password essa rimane valida per tutte le pagine in cui la password é uguale. Vediamo come dovrebbe essere programmato il campo password per il nostro esempio:

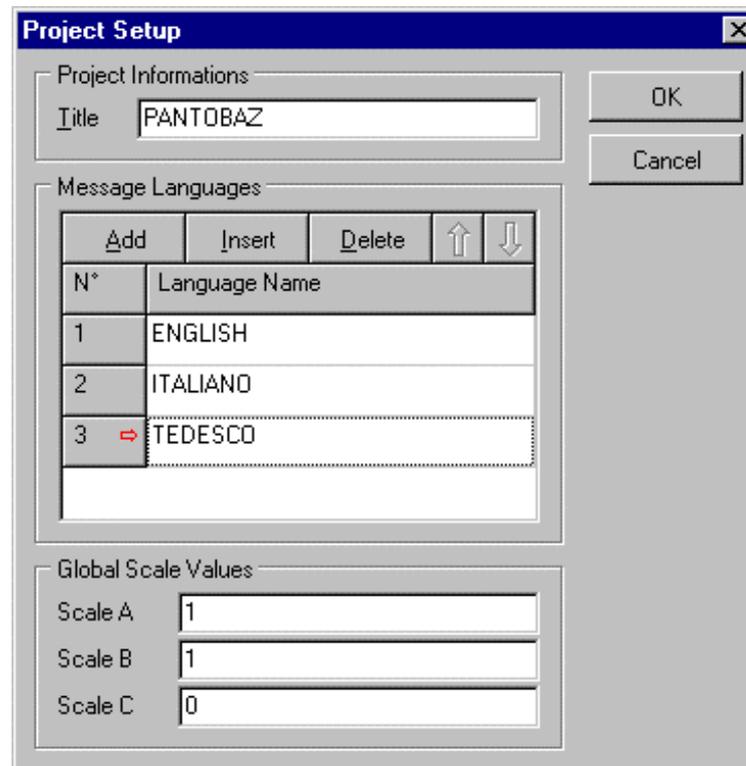
Password	Lista pagine	Nome pagina
	Pag nr.01	Visualizzazione 1/4
	Pag nr.02	Visualizzazione 2/4
"1234"	Pag nr.03	Parametrizzazione 1/3
"1234"	Pag nr.04	Parametrizzazione 2/3
"1234"	Pag nr.05	Parametrizzazione 3/3
	Pag nr.06	Visualizzazione 3/4
"TAR01"	Pag nr.07	Taratura 1/2
"TAR01"	Pag nr.08	Taratura 2/2
	Pag nr.09	Visualizzazione 4/4

Scelta della lingua

Quando un progetto viene realizzato utilizzando più lingue esistono vari modi per eseguire la scelta sul terminale.

Il modo più semplice e immediato è quello di entrare nelle pagine di "set-up" del terminale e scegliere il nome della lingua utilizzata. Questa procedura è in genere consigliata per l'installatore del terminale visto che nelle pagine di set-up sono disponibili anche altre configurazioni che se erroneamente toccate (vedi velocità seriale) possono interrompere il normale funzionamento del sistema e richiedere l'intervento di personale specializzato. La lingua è però selezionabile anche direttamente nella normale applicazione utente. Esiste la variabile terminale \$LANGUAGE il cui valore rappresenta il numero della lingua attualmente selezionata.

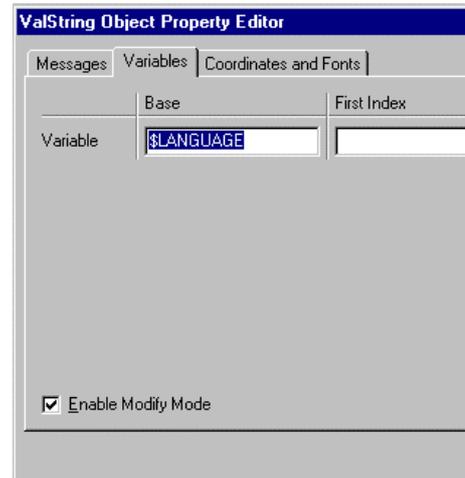
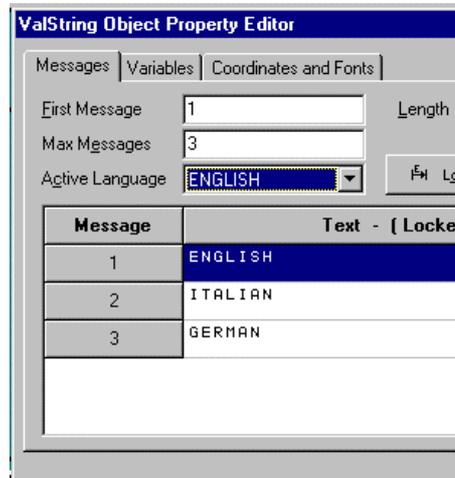
Si consideri il seguente progetto di esempio che è stato sviluppato con 3 lingue:



per la lingua "ENGLISH" la variabile \$LANGUAGE assume valore 1, per la lingua "ITALIANO" assume valore 2, per la lingua "TEDESCO" assume valore 3.

Per realizzare la scelta della lingua da applicazione, senza utilizzare le pagine di "Set-up", basta creare un oggetto VALUE assegnando la variabile \$LANGUAGE con limiti di introduzione minimo 1 e massimo 3. Per questa soluzione si deve in qualche modo però indicare la relazione tra nome lingua e valore numerico che non è immediata per l'utente finale. Una soluzione più elegante e funzionale è quella di definire un oggetto VALSTRING con proprietà "Enable Modify Mode" attiva dove i messaggi corrispondono ai nomi assegnati alla lingua. In questo modo l'utilizzatore sceglie la lingua mediante lo scroll di tutti i nomi delle lingue mediante il tasto "Next Char" e quando conferma viene scritto il nuovo valore sulla variabile di terminale.

Vediamo un esempio:



É importante introdurre i testi per tutte le lingue esistenti nel progetto magari con le appropriate traduzioni.

Cambio pagina forzato da QCL

In molte applicazioni é consueto che una determinata condizione verificatasi nell'applicativo QCL debba segnalare un'informazione all'operatore tramite terminale. Solitamente si riserva una determinata pagina per segnalare all'operatore questa situazione e bisogna forzare il cambio pagina in corrispondenza di tale condizione. Per fare questo il QCL deve poter modificare la pagina corrente nel terminale. Facciamo un esempio: un determinato ingresso del sistema segnala la mancanza olio nella macchina. In tale condizione il ciclo di lavoro si arresta ed il terminale visualizza la pagina numero 25 di avviso controllo olio. Vediamo come implementare questo: si definisce in QCL la variabile `gbNewPage` di tipo GLOBAL e dimensione BYTE. Si definisce nel terminale un evento globale di tipo "OnChangeVar" associato alla variabile `gbNewPage`. A tale evento si definisce l'azione "GoToPage" associata al valore della stessa variabile `gbNewPage`. Cosí facendo ad ogni variazione della variabile `gbNewPage` corrisponderá il salto alla pagina definita dal valore della variabile stessa. In altre parole, per il QCL, basterá scrivere il nuovo valore nella variabile ed il terminale cambierà pagina automaticamente senza dover separare i comandi di cambio pagina e numero nuova pagina. In QCL si dovrà implementare il seguente codice:

```
..
..
    IF (ioMancaOlio)
        sbStopCiclo = 1      ;Arresto del ciclo automatico
        gbNewPage = 25      ;Terminale in pagina nr.25!
    ENDF
..
..
```

dove `ioMancaOlio` é la variabile di tipo INPUT che rappresenta il valore dell'ingresso. Questa programmazione non modifica il comportamento dei tasti PGUP e PGDN.

In altre applicazioni é invece necessario il trasferimento opposto e cioé si verifica quando l'applicativo, per come é stato progettato, deve eseguire operazioni diverse a seconda della pagina che il terminale sta visualizzando. Per fare questo é il terminale che ad ogni cambio pagina deve aggiornare l'applicativo sul nuovo valore della pagina in esecuzione. Vediamo come fare questo: si definisce in QCL la variabile `gbTerPage` di tipo GLOBAL e dimensione BYTE. Si definisce nel terminale un evento globale di tipo "OnChangeVar" associato alla variabile di terminale `$PAGE`. A tale evento si definisce l'azione "Set Var" con associata la variabile `$PAGE` come source e `gbTerPage` come destination. Cosí facendo ad ogni variazione della variabile `$PAGE` corrisponderá la scrittura del nuovo valore nella variabile dell'applicativo `gbTerPage`.

N.B. Se l'applicativo é formato da piú di 127 pagine le variabili definite in QCL dovranno essere di dimensione WORD anziché BYTE.

Introduzione con diverse unità di misura

In alcune applicazioni esiste la necessità di visualizzare i valori numerici in diverse unità di misura. Ad esempio in una pagina di terminale il cliente potrebbe scegliere se utilizzare la rappresentazione in millimetri oppure in pollici oppure in metri anziché decimetri. Tutte le rappresentazioni numeriche del progetto dovrebbero così commutare i valori ed adeguarsi alla nuova scelta. Il terminale mette a disposizione una serie di strumenti per permettere questa possibilità. Come premessa bisogna dire che esistono molti modi per realizzare la conversione, il terminale implementa questa caratteristica mantenendo invariato il valore originale del dato e introducendo un fattore di scala tale da eseguire la conversione dell'unità di misura.

Supponiamo di voler sviluppare un progetto terminale dove alcune visualizzazioni/introduzioni possono essere effettuate sia in valori metrici che in valori anglosassone. Supponiamo che la parametrizzazione della macchina sia stata effettuata per rappresentare il decimo di millimetro e che si voglia visualizzare un punto decimale per facilitare la lettura del valore. Consideriamo anche che per la rappresentazione in unità anglosassone bisogna rappresentare il valore in pollici senza però perdere precisione. Così è necessario modificare la posizione del punto decimale (ad esempio 100.4mm permette una precisione del decimo di mm, ma 4.6inch significa una precisione di 2.54mm!).

Procediamo per passi: tutte le introduzioni numeriche dove vogliamo ottenere due rappresentazioni fanno capo ad oggetti VALUE in cui bisogna abilitare la voce "Global" nel campo "Scaling Mode". In questo modo il dato prima di essere visualizzato sarà convertito per il fattore di scala composto dai valori ScaleA, ScaleB e ScaleC presenti nella finestra richiamata dal menù "Project Setup". Questi valori sono modificabili durante l'esecuzione dell'applicazione tramite la scrittura nelle variabili di terminale \$SCALEA, \$SCALEB e \$SCALEC. Inoltre definiamo la variabile interna @PPDec di dimensione BYTE. Tutti gli oggetti VALUE sopra indicati devono avere associata la variabile @PPDec nel campo Dec.Point. Supponiamo che la parametrizzazione sia stata eseguita perché l'applicativo lavori con la precisione del decimo di millimetro e quindi per questa unità di misura si visualizzano i valori con un punto decimale. Per ottenere la visualizzazione metrica bisogna impostare nel terminale i seguenti valori:

```
$SCALEA = 1
```

```
$SCALEB = 1
```

```
$SCALEC = 0
```

```
@PPDec = 1
```

Per la rappresentazione anglosassone:

```
$SCALEA = 393700
```

```
$SCALEB = 100000
```

```
$SCALEC = 0
```

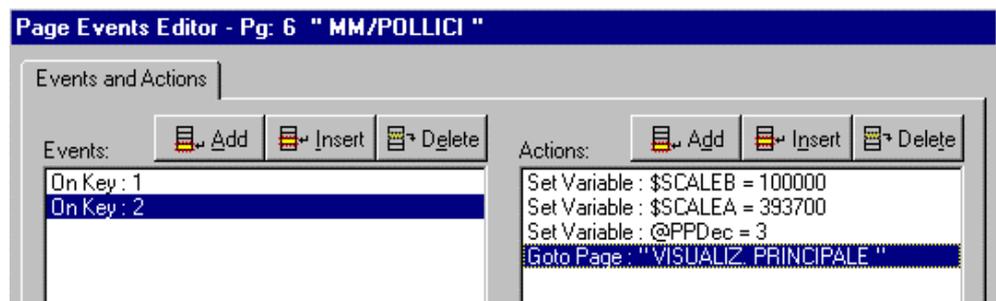
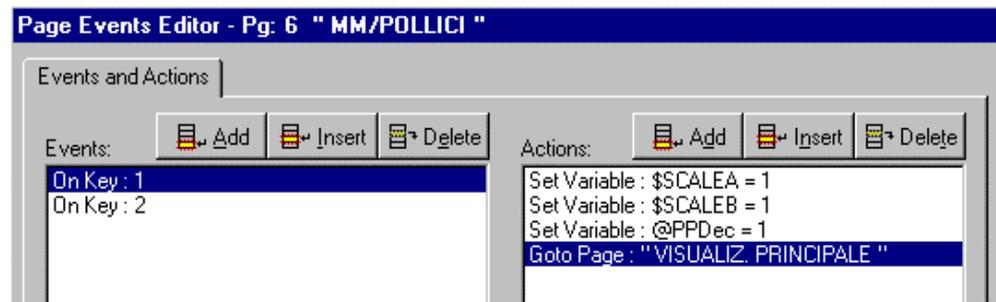
```
@PPDec = 3
```

Con questa impostazione si intende visualizzare il millesimo di pollice che corrisponde a 0.0393700 mm e quindi mantiene la precisione del decimo di millimetro. Per convertire da decimo di millimetro a millesimo di pollice si deve moltiplicare per:

$$\text{valore_in_millesimi_di_pollice} = \frac{\text{valore_originale}}{10} \cdot \frac{1000}{25.4} = 3.93700$$

siccome la variabile \$SCALEA è di dimensione LONG può contenere solamente valori interi. Allora si moltiplica il valore per 100000 per renderlo intero e si programma \$SCALEB.

Ora sarà sufficiente realizzare una pagina di scelta unità di misura come in figura:

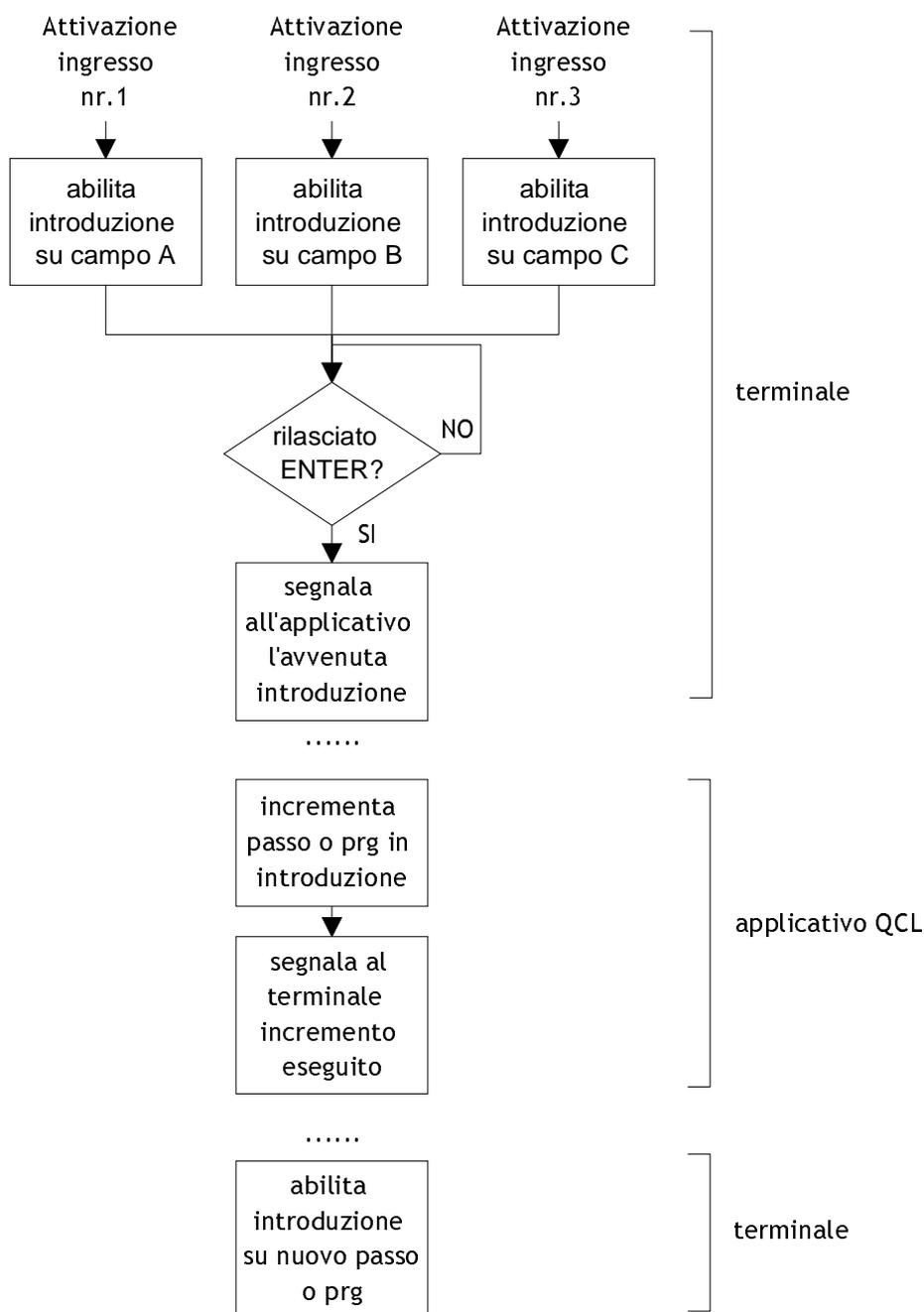


Fare attenzione perché le variabili interne all'accensione del terminale hanno valore zero mentre le variabili \$SCALEA, \$SCALEB e \$SCALEC hanno il valore fissato nell'introduzione "Global Scale Values" nella finestra nella finestra richiamata dal menù "Project Setup" di QPaint.

Introduzione dati guidata

Si vuole in questo esempio creare una pagina di introduzione dati contenuti in un datagroup. L'introduzione viene definita guidata per due motivi: 1) perché essa viene abilitata non dalla pressione del tasto INS ma dall'attivazione di una serie di ingressi sul sistema 2) perché l'incremento del passo o programma in introduzione viene eseguito automaticamente dopo ogni conferma del dato.

Le operazioni vengono svolte sia dal terminale che dall'applicativo come mostrato in figura:



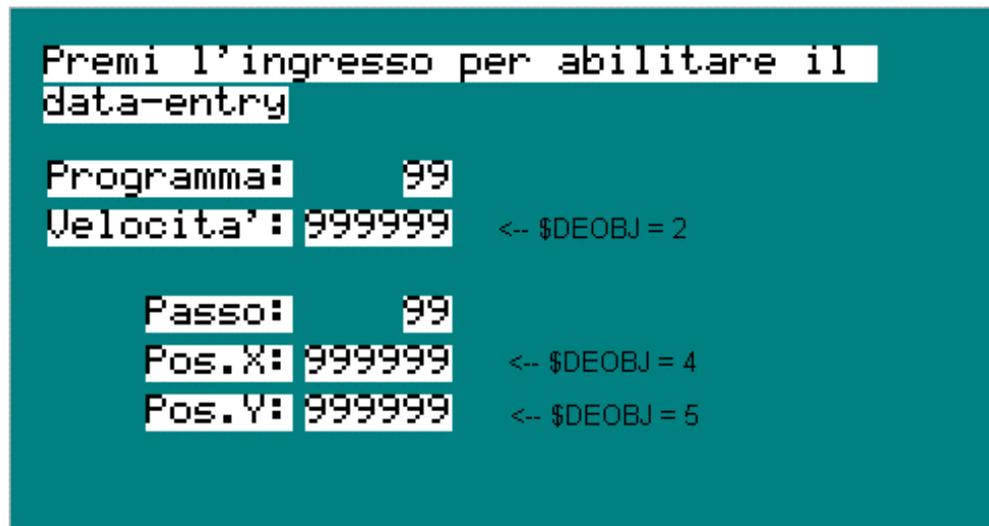
Questo esempio può comunque essere preso come spunto per realizzare altri tipi di introduzioni.

Si sfruttano per questo esempio le variabili terminale \$SINGLEDE e \$OBJDE. Queste variabili nascono per modificare la normale introduzione dati del terminale. In dettaglio la variabile \$SINGLEDE (Single data-entry) se viene impostata a uno il terminale alla conferma di un valore non abilita il campo di introduzione successivo ma esce dal data-entry. Si comporta come se nella pagina fosse presente solamente un valore in introduzione. La variabile \$OBJDE rappresenta il numero dell'oggetto dinamico al quale il data-entry fa riferimento. Infatti nel terminale gli oggetti dinamici (VALUE e VALSTRING) sono identificati con un numero progressivo. Questa variabile avendo accesso in scrittura e lettura può essere utilizzata con il duplice scopo di far conoscere all'applicativo in quale oggetto l'utente sta introducendo un valore e per imporre il data-entry del terminale su un determinato campo. Per maggiori dettagli su queste variabili vedere la sezione dedicata.

Supponiamo di voler introdurre i dati sul seguente datagroup definito nel file CNF:

```
-----  
; Definizione DATAGROUP  
-----  
DATAGROUP  
Sequenza  
    DATAPROGRAM  
    0003  
        dplVeloc                L  
  
    STEP  
    0005  
        dslQPosX                L    ;Quota di posizionamento asse X  
        dslQPosY                L    ;Quota di posizionamento asse Y
```

Si costruisce la seguente pagina di terminale:

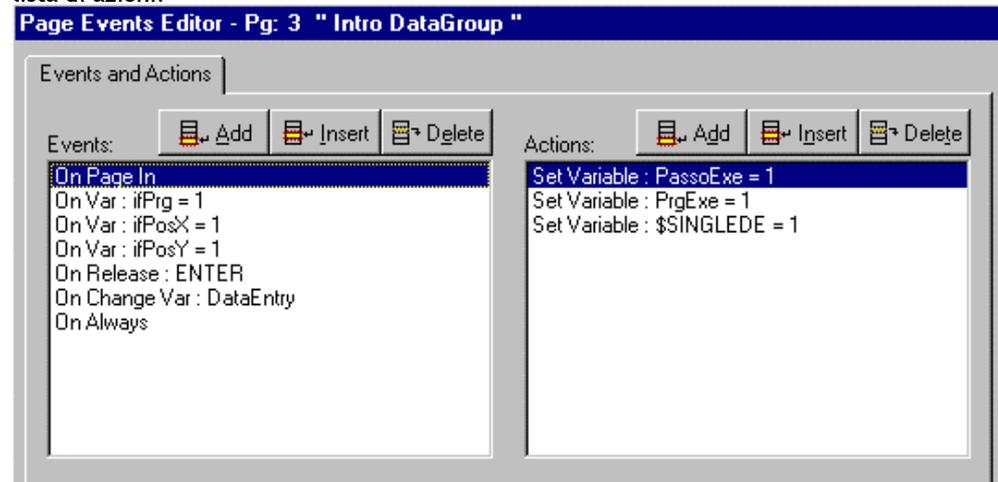


Le variabili "Programma:" e "Passo:" visualizzano solo il numero del programma/passò in introduzione. Non è possibile modificare il valore. Le variabili "Velocità:", "Pos.X:" e "Pos.Y:" invece rappresentano il valore del datagroup e sono modificabili. Normalmente in una pagina realizzata in questo modo alla pressione del tasto INS si abilita il data-entry sul primo oggetto modificabile ed alla conferma di un valore il data-entry si abilita sul campo successivo in maniera ciclica tra "Velocità:", "Pos.X:" e "Pos.Y:". Nel nostro caso invece l'abilitazione è comandata da QCL in corrispondenza dell'attivazione di un ingresso. Inoltre alla conferma di un valore non si deve abilitare il campo successivo ma si deve uscire dal data-entry.

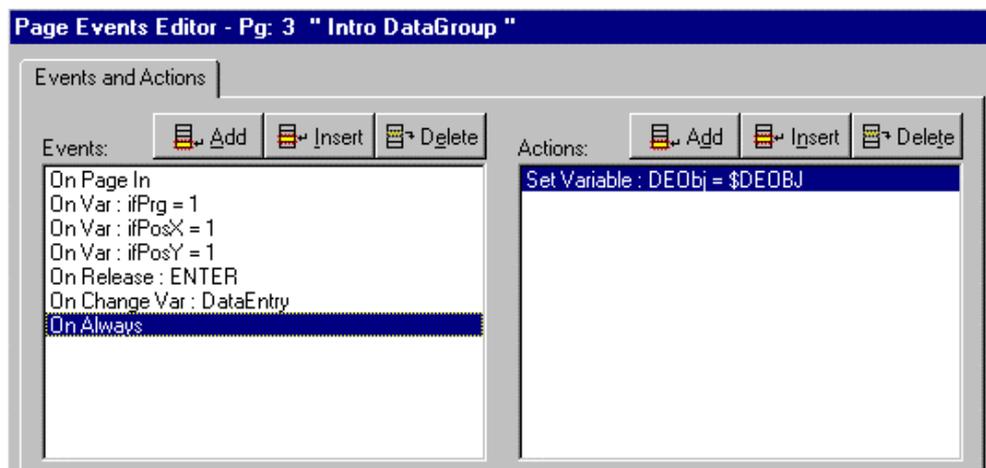
Prima di tutto bisogna sapere che ogni oggetto dinamico (VALUE o VALSTRING) è identificato da un numero che viene segnalato durante il data-entry dalla variabile terminale \$DEOBJ. Per conoscere il numero identificativo dell'oggetto bisogna visualizzare il valore di \$DEOBJ durante il data-entry. Nel nostro esempio i valori sono visualizzati nella figura precedente. Vogliamo realizzare un'introduzione con le seguenti caratteristiche. Abbiamo tre ingressi del sistema "ifPrg", "ifPosX" e "ifPosY" se attivati abilitano l'introduzione nei rispettivi campi "Velocità:", "Pos.X:" e "Pos.Y:". Introdotto il valore, il data-entry rimane sullo stesso campo ma visualizza il valore del programma o passo successivo. Se viene raggiunto l'ultimo passo o programma, l'introduzione ritorna sul primo valore. Queste operazioni vengono eseguite parte sul terminale e parte sull'applicativo. Più precisamente il terminale si occupa di fornire le informazioni all'applicativo il quale, avendo un linguaggio di programmazione, riesce ad eseguire tutte le computazioni e a rendere disponibili al terminale i risultati.

Procediamo per passi:

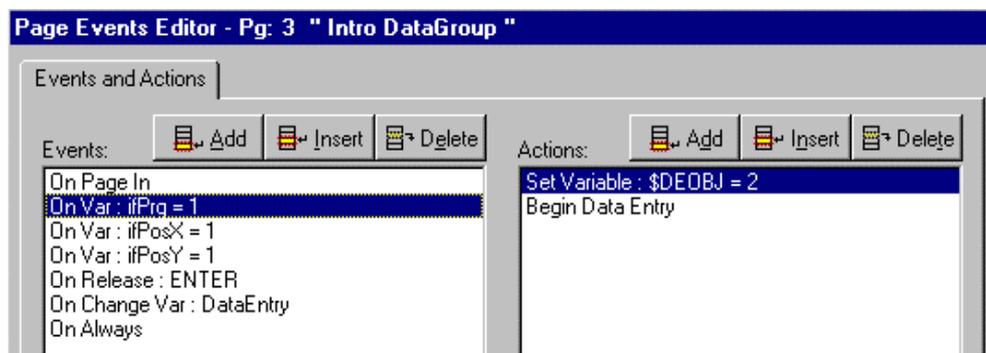
Bisogna utilizzare ed inizializzare due variabili QCL di appoggio per contenere il numero di passo e programma in introduzione. Poi dobbiamo impostare ad 1 la variabile di terminale \$SINGLEDE per disabilitare, dopo l'introduzione di un valore, il passaggio automatico al campo successivo. Per questo definiamo un evento di pagina "OnPageIn" con la seguente lista di azioni:



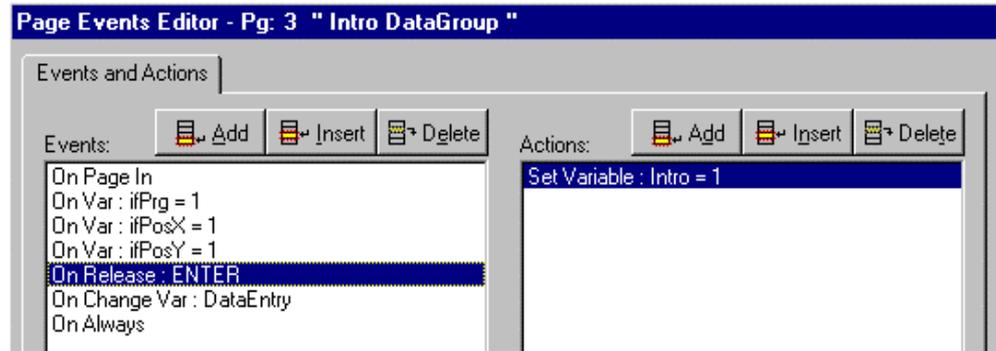
L'applicativo deve eseguire operazioni diverse a seconda del campo di introduzione che l'utente ha abilitato. Ad esempio una introduzione di "Velocità:" deve incrementare il programma in introduzione, una introduzione di "Pos.X:" deve incrementare il passo. Ciò significa che l'applicativo deve conoscere sempre il numero del campo in introduzione per svolgere l'operazione corretta. Per questo bisogna definire la variabile QCL "DEObj" di tipo GLOBAL e dimensione Word. L'evento "On Always" si occupa di copiare il valore della variabile terminale nella variabile QCL come in figura:



Alla attivazione dei tre ingressi si deve scegliere l'oggetto in data-entry e abilitare il data-entry come in figura:



L'utente imposta il valore ed al termine deve indicare all'applicativo di incrementare il passo/programma in introduzione. A questo compito pensa l'evento "On Release" associato al tasto ENTER che imposta un valore su una nuova variabile QCL "Intro". Vediamo in figura:



A questo punto l'applicativo deve riconoscere l'introduzione dato ed eseguire l'incremento. Deve essere utilizzata una ulteriore variabile QCL che chiameremo "DataEntry" che indica al terminale che l'applicativo ha eseguito l'incremento. In totale tutte le variabili da definire in QCL sono:

```

;-----
; Definizione Variabili GLOBAL
;-----
GLOBAL
...
...
    PassoExe    B                ;Numero del passo in introduzione nel
                                ;terminale
    PrgExe      B                ;Numero del programma in introduzione
                                ;nel terminale
    Intro       B                ;Segnalazione per l'applicativo di
                                ;introduzione avvenuta
    DataEntry   B                ;Segnalazione per il terminale di
                                ;incremento avvenuto
    DEObj       W                ;Copia valore variabile $OBJDE
...
..

```

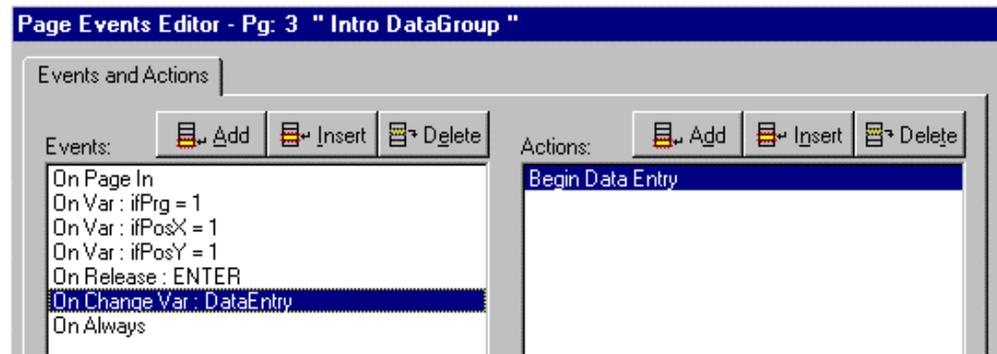
Su di un task bisogna introdurre il seguente codice:

```

;Introduzione dati guidata -----
IF (Intro EQ 1)
  IF ((DEObj EQ 4) OR (DEObj EQ 5))
    IF (PassoExe LT 5)
      PassoExe = PassoExe + 1
    ELSE
      PassoExe = 1
    ENDIF
  DataEntry = DataEntry + 1
ENDIF
IF (DEObj EQ 2)
  IF (PrgExe LT 3)
    PrgExe = PrgExe + 1
  ELSE
    PrgExe = 1
  ENDIF
  DataEntry = DataEntry + 1
ENDIF
Intro = 0
ENDIF

```

Vediamo che tutto il codice é condizionato dal valore 1 della variabile Intro e cioé dal rilascio da parte dell'utente del tasto Enter. Notare la riga di istruzioni di incremento variabile DataEntry che serve solamente a scatenare l'evento "On change Var" programmato nel terminale. Questo evento serve per riabilitare il data-entry con il programma/passo incrementato.



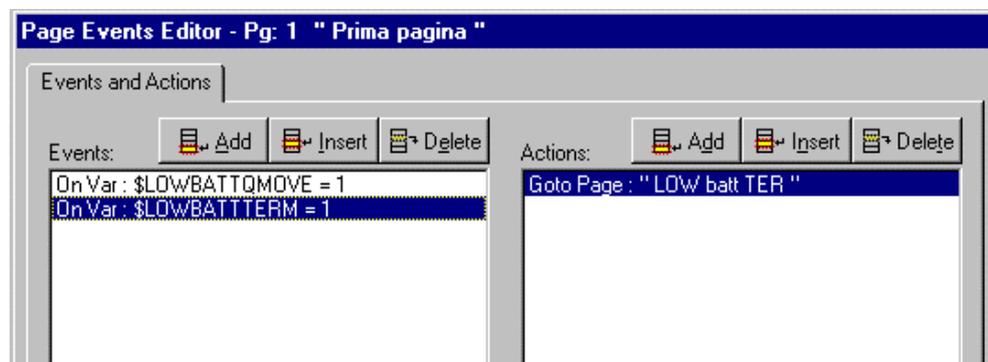
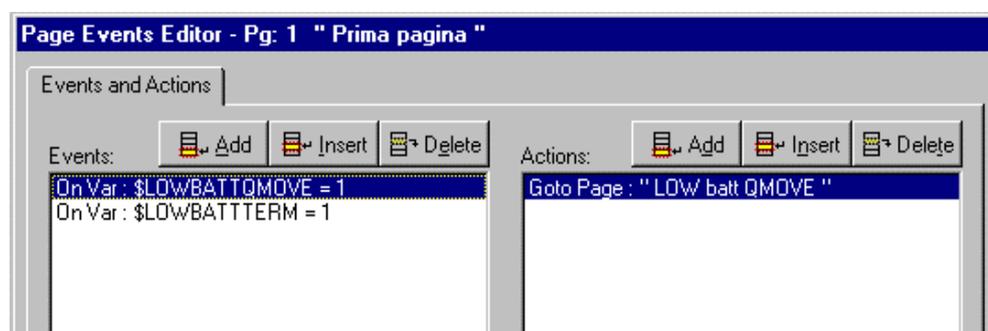
Attenzione perché il valore 1 alla variabile di terminale \$SINGLEDE può non essere adatta per le altre pagine. In questo caso deve essere resettata all'ingresso di qualsiasi nuova pagina.

Segnalazione di batteria

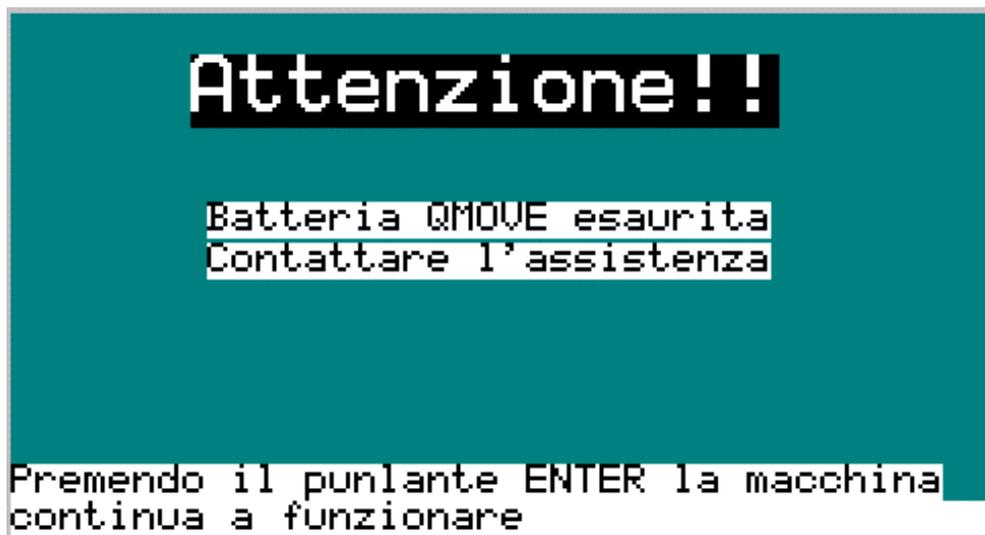
Nel terminale e nel sistema QmoveÒ si trova una batteria tampone per la memorizzazione dei dati ritentivi. In caso di esaurimento di tale componente esiste una segnalazione nelle apparecchiature. Per il terminale la segnalazione si trova nella pagina Info, per il sistema QmoveÒ esiste una spia luminosa sul frontalino. In molti casi però questa segnalazione non può essere sufficiente perché non è direttamente accessibile all'utente, ad esempio il sistema QmoveÒ si trova all'interno di un quadro elettrico o nel terminale non viene mai richiamata la pagina di set-up. Per questi motivi è utile segnalare l'esaurimento delle batterie sulla pagina di terminale in modo che l'utente si accorga e possa prendere i provvedimenti necessari. Esistono due variabili di terminale \$LOWBATTTERM e \$LOWBATTQMOVE rispettivamente che rappresentano la batteria esaurita nel terminale e nel QmoveÒ.

In questo esempio vediamo come realizzare un applicativo in cui il terminale, se una batteria è esaurita, segnala all'utente questo stato con una pagina dedicata:

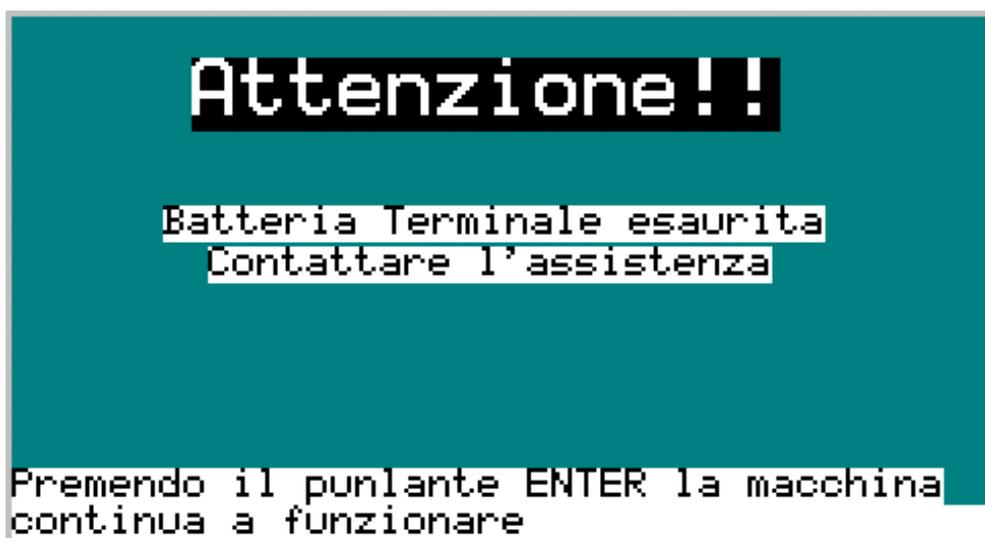
Nella prima pagina di terminale inseriamo i seguenti eventi di pagina:



La pagina "LOW batt QMOVE" deve essere:



La pagina "LOW batt Ter" deve essere:



Controllo comunicazione attiva

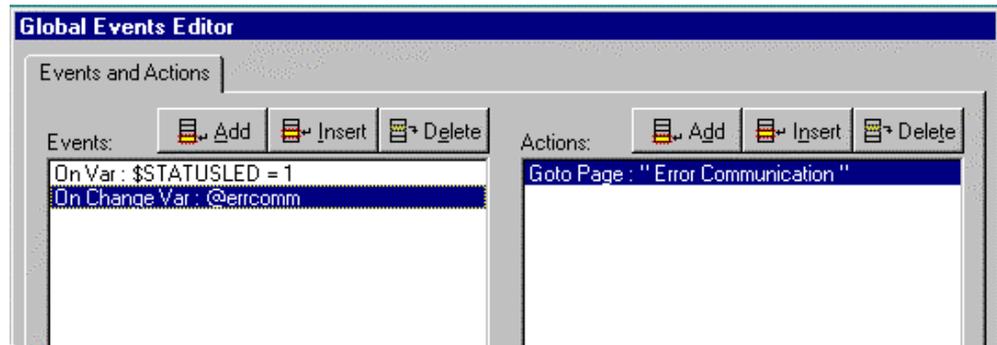
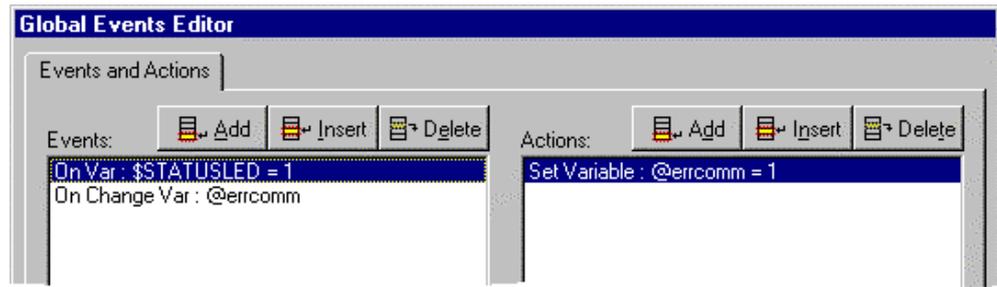
Nel terminale esiste la possibilità di segnalare una anomalia durante la comunicazione tra il terminale e il sistema QmoveÒ. Tale anomalia può essere di origine hardware (mancanza di collegamento, disturbi alla linea) oppure determinata da situazioni relative al funzionamento dell'applicazione (letture non corrette di variabili QmoveÒ) ed è segnalata dal lampeggio del led "status" posto nella parte superiore del pannello terminale. Quando si verifica un errore di questo tipo è opportuno verificare quale è stata la causa che lo ha prodotto; ciò può essere fatto semplicemente entrando nella "Info Page" del Setup alla voce "Comunic. Status". È importante sottolineare che l'errore e quindi il lampeggio vengono cancellati solamente allo spegnimento del terminale o dopo che si è usciti dal Setup per ritornare all'applicativo.

Questa segnalazione può essere ottenuta anche da programma tramite l'uso della variabile di terminale \$STATUSLED; in questo esempio vediamo come realizzare un applicativo in cui il terminale, se avviene un errore di status, segnala questo all'utilizzatore andando in una pagina dedicata.

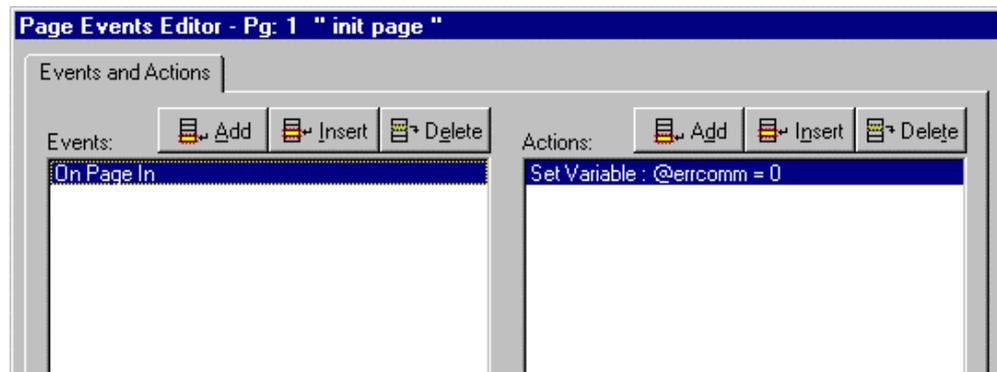
Il nostro esempio ha 4 pagine: la pagina iniziale in cui non vi è alcuna comunicazione con il QmoveÒ ma si visualizza una variabile di terminale, due pagine in cui si vanno a fare delle operazioni di lettura di variabili QmoveÒ, e infine la pagina di allarme. Ciò che vogliamo ottenere dall'applicativo è andare alla pagina di allarme ogni volta che si verifica un qualsiasi errore di comunicazione tra terminale e Qmove.

La cosa più semplice è quella di definire un evento globale OnVar sulla variabile di terminale \$STATUSLED, verificando quando questa assume il valore 1 (che indica l'errore di comunicazione) ed effettuando il salto alla pagina di allarme. Questo però ha un inconveniente: infatti quando si arriva alla pagina di allarme, l'evento OnVar, essendo globale e agendo sulla var \$STATUSLED, è ancora attivo e quindi il salto alla pagina di allarme si effettua sempre. Questo implica il continuo rinfresco della pagina di allarme, e questo produce uno sfarfallio fastidioso, e non è esattamente quello che si voleva.

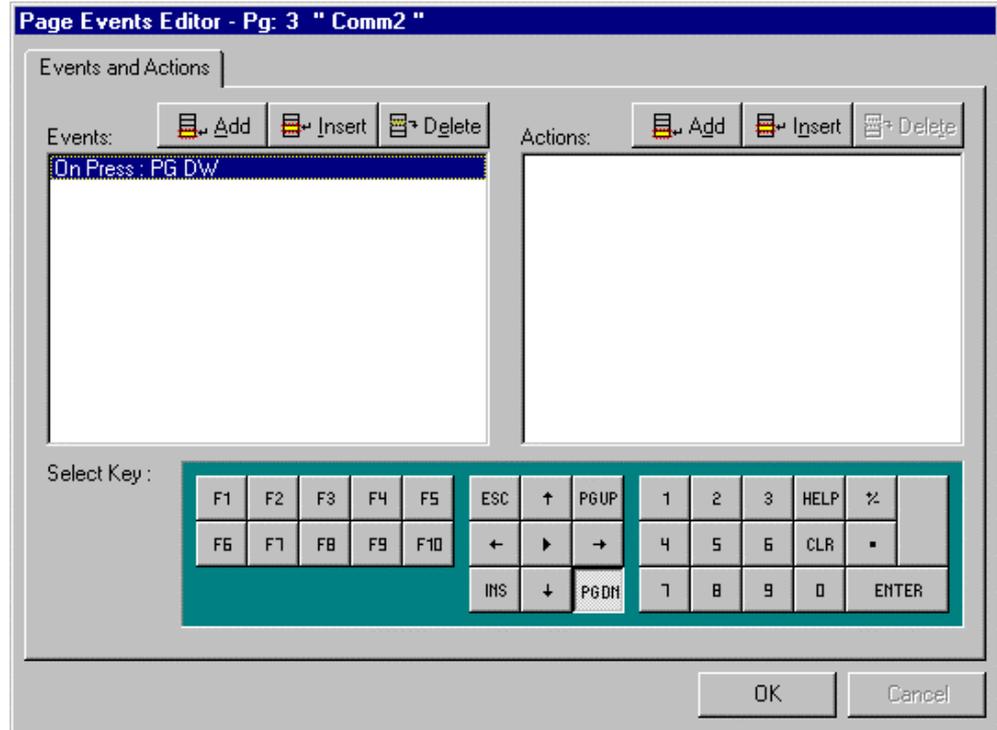
Bisogna procedere in altro modo; definiamo una variabile interna @errcomm, che utilizzeremo per il salto pagina e solo quando c'è un cambiamento nel suo valore, in modo da non saltare nuovamente alla pagina di allarme una volta avvenuto l'errore. Quando la var \$STATUSLED assume il valore 1 allora anche la var @errcomm verrà impostata ad 1. Quindi si crea un evento OnChangeVar sul cambio di valore della variabile @errcomm con azione collegata Goto Page "Error Communication". È conveniente usare il nome della pagina piuttosto che il suo numero, perché questo ci rende completamente indipendenti da eventuali modifiche nella posizione della pagina di allarme. Quindi la pagina degli Eventi Globali appare così:



Per far sí che una volta verificato l'errore si torni ad una situazione normale, in cui si é ancora sensibili ad un nuovo errore di comunicazione, si deve resettare il valore di @errcomm nella pagina iniziale, in quanto una volta usciti dalla Info di Setup Page si entra in questa pagina. Allora si definisce un evento di pagina OnPageIn in questo modo



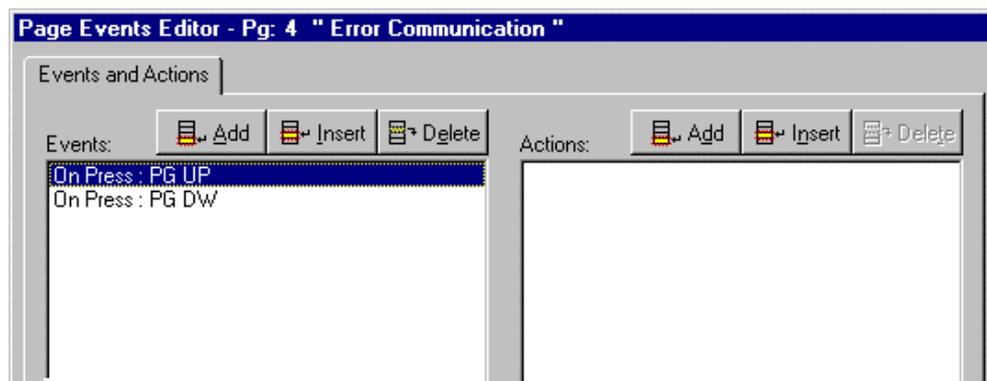
Infine, per fare in modo di non accedere direttamente alla pagina di allarme con l'uso dei tasti PGUP o PGDN occorre sovraccaricare il tasto PGDN nella pagina precedente e sovraccaricare il tasto PGUP nella pagina successiva. Nel nostro caso la pagina di allarme é l'ultima e quindi si lavora solo sulla pagina precedente.



La pagina di allarme, che abbiamo chiamato "Error Communication", potrebbe essere così disegnata:



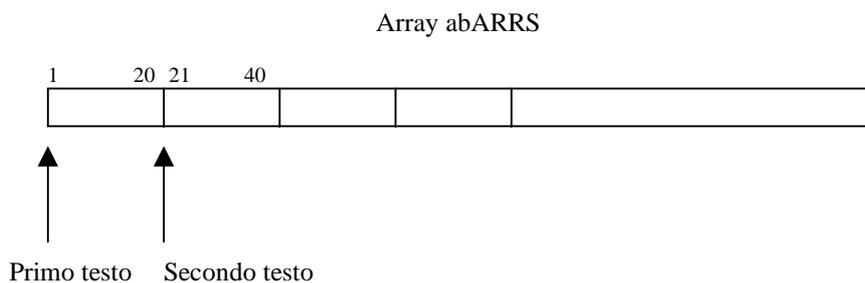
Anche nella pagina di allarme occorre sovraccaricare i tasti PGUP e PGDN per fare in modo che l'uscita avvenga solo dalla pagina di Setup/Info, in modo anche da verificare il tipo di errore che si é verificato.



Introduzione alfanumerica

In questo esempio vediamo come poter trattare dati alfanumerici in maniera dinamica usando una variabile QCL Array di tipo byte. In particolare simuleremo un piccolo archivio di testi (45 stringhe di 20 caratteri ciascuna) usando un array di 900 elementi. Vedremo come poter selezionare uno qualsiasi di questi testi, modificarlo, ecc...

Il modo in cui vogliamo gestire l'array byte (variabile abARRS in QCL) è raffigurato nel disegno seguente:

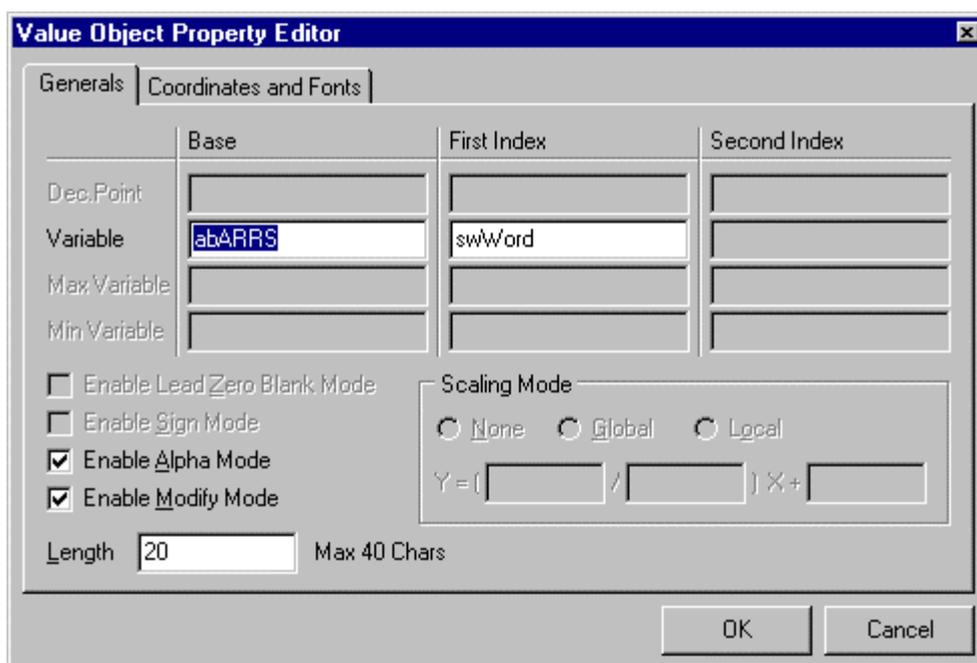


Creiamo un oggetto VALUE in cui definiamo come variabile associata proprio la abARRS e in cui usiamo come First Index una variabile QCL, nel nostro esempio swWord che è una variabile SYSTEM di tipo WORD; questo per poterlo modificare da QMoveÒ, così ci possiamo spostare lungo l'array e selezionare quella parte che ci interessa. Poi definiamo degli eventi su tasti funzione in modo da poter scorrere la lista dei testi contenuti nell'array; vi saranno due tasti di scorrimento rapido e due tasti che si spostano di un testo alla volta. E' da sottolineare che la suddivisione dell'array in singole stringhe è puramente logica in quanto in realtà i caratteri si susseguono uno dopo l'altro; tuttavia quando si esegue il programma nel terminale l'idea che ci si fa è quella di stare trattando realmente un archivio di stringhe, perfettamente separate tra di loro. Ed è ciò che vogliamo ottenere con questo esempio.

Vediamo come é organizzata la pagina del terminale:



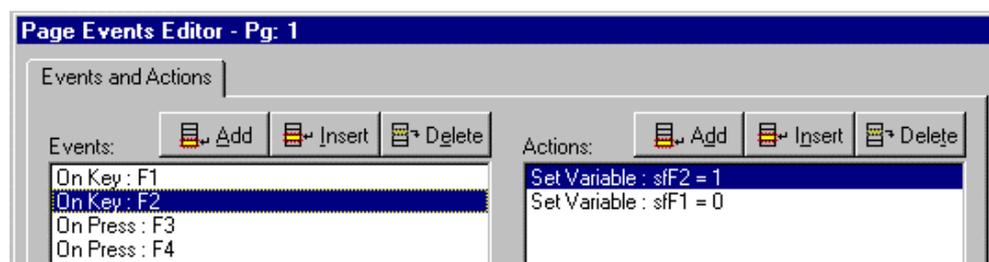
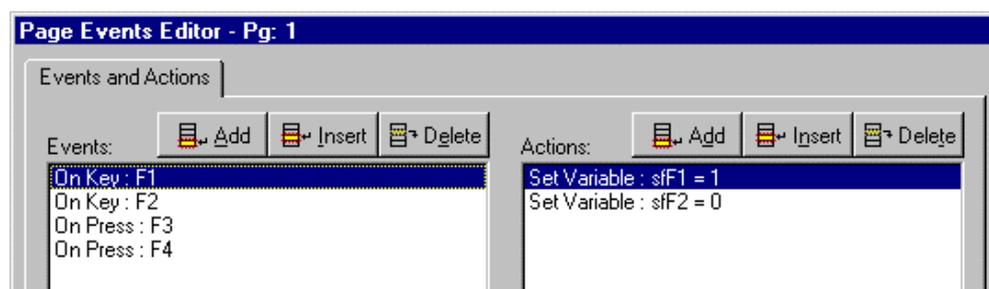
Vediamo piú in particolare l'oggetto che rappresenta il testo. Come detto é un VALUE con queste proprietá:



E' stata usata la variabile QCL abARRS, in quanto l'uso dei Value alfanumerici é possibile solo con array a byte, l'indice di inizio lettura del valore é dato dalla variabile swWord, e quindi tale indice di inizio é modificabile dall'applicativo QCL, e sono state abilitate le proprietà Alpha Mode, per poter trattare i valori dei singoli elementi dell'array come caratteri, e il Modify Mode, che ci consente di poter introdurre il testo direttamente dalla pagina di terminale. Infine nel campo Length abbiamo specificato che la lunghezza dell'oggetto Value é di 20 caratteri.

Per la visualizzazione della posizione si é usato un oggetto VALUE con variabile gbPosStr, che rappresenta il numero d'ordine del testo all'interno della nostra lista.

Vediamo anche come sono stati definiti i tasti funzione e come sono stati usati Eventi e Azioni:



L'evento OnKey viene generato fino a quando il tasto relativo resta premuto; questo ci serve per poter implementare lo scorrimento veloce della lista, verso l'alto e verso il basso. Per questo usiamo i due tasti F1 e F2, che semplicemente impostano le due variabili di tipo Flag sfF1 e sfF2, che poi verranno gestite dall'applicativo QCL. I tasti F3 e F4 operano nella stessa maniera, nel senso che impostano analogamente le due variabili sfF1 e sfF2, solo che l'evento é OnPress, che viene generato una sola volta alla pressione del tasto, e affinché sia generato di nuovo é necessario che il tasto venga rilasciato per poi essere ripremuto. E questo é proprio ciò che ci serve per scorrere la lista passo passo, spostandoci di un elemento per volta. Questo é tutto per quanto riguarda il terminale. Vediamo ora invece come interviene il codice QCL. Nel file .CNF dobbiamo definire:

```
-----  
; Definizione Variabili SYSTEM  
-----  
SYSTEM  
  
...  
    swWord W                ;variabile utilizzata come indice  
                            ;carattere (1÷900)  
    sfdown F                ;Tasto di scorrimento lista 1->45  
                            ;premuto da terminale  
    sfup F                  ;Tasto di scorrimento lista 45->1  
                            ;premuto da terminale  
  
...  
  
-----  
; Definizione Variabili GLOBAL  
-----  
GLOBAL  
  
...  
    gbStep B                ;Indice relativo al testo selezionato  
                            ;nella lista (0÷44)  
    gbPosStr B              ;Indice relativo al testo e visualizzato  
                            ;in terminale (1÷45)  
  
...  
  
-----  
; Definizione Variabili ARRSYS1  
-----  
ARRSYS  
  
abARRS B 900                ;Contiene i caratteri dei testi da  
                            ;gestire  
  
...
```

Mentre nel codice task .MOD abbiamo:

```
-----  
; Programma Principale  
-----  
MAIN:  
  
;gestisce l'avanzamento in avanti verificando la fine  
;sfF1 = 1 significa premuto F1 o F3  
  
IF ((sfF1 EQ 1) AND (gbPosStr LT 45))  
    gbStep = gbStep + 1  
    swWord = gbStep * 20 + 1    ;calcola l'indice all'interno  
                                ;dell'array  
  
ENDIF
```

```
;gestisce l'avanzamento all'indietro verificando l'inizio
;sfF2 = 1 significa premuto F2 o F4

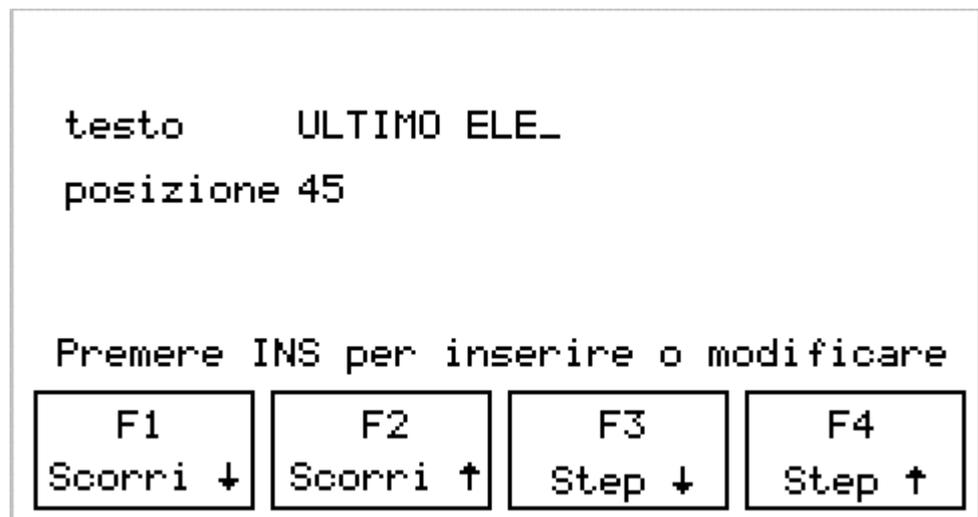
IF ((sfF2 EQ 1) AND (gbPosStr GT 1))
  gbStep = gbStep - 1
  swWord = gbStep * 20 + 1      ;calcola l'indice all'interno
                                ;dell'array
ENDIF

; resetta i tasti
sfF1 = 0
sfF2 = 0

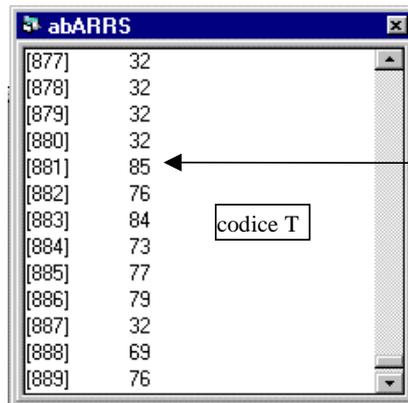
; aggiorna la posizione del testo selezionato
gbPosStr = gbStep + 1
```

Si vede come la cosa venga fatta semplicemente: la variabile `gbStep` contiene il valore dell'indice del testo nella lista, partendo però da valore zero, cui corrisponde il carattere di inizio `swWord` pari a 1. `swWord` contiene l'indice del carattere di inizio dei vari testi, quindi assume solo i valori 1, 21, 41, 61,881. La pressione dei tasti funzione modifica il valore di `sbStep` e quindi di `swWord`; infine viene impostata anche `gbPosStr` che rappresenta la posizione del testo nella lista in modo "naturale" cioè partendo da indice 1.

Come detto il testo selezionato può essere modificato dal terminale: questo è dovuto al fatto che abbiamo selezionato `Modify Mode` nell'oggetto `VALUE`. Per inserire o modificare il testo dalla pagina terminale è sufficiente premere il tasto `INS`, e confermare l'inserimento con il tasto `ENTER`, che fa uscire dalla fase di `Data Entry`. Vediamo come si presenta il terminale durante il funzionamento:



Se andiamo a verificare i valori dell'array abARRS (ad es. con QView) noteremo che:



Index	Value
[877]	32
[878]	32
[879]	32
[880]	32
[881]	85
[882]	76
[883]	84
[884]	73
[885]	77
[886]	79
[887]	32
[888]	69
[889]	76

da qui parte l'ultimo testo della lista

85 é il codice ASCII della lettera U

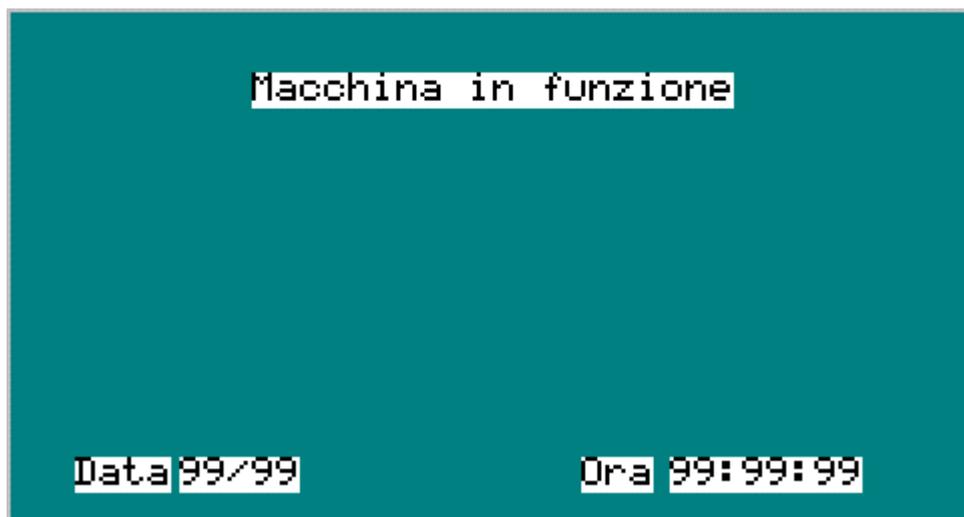
Questo significa che anche da QCL potremmo modificare quanto appare nel terminale. Se ad esempio scriviamo il valore 49 nell'elemento abARRS[886], il testo che appare nel terminale é "ULTIM1 ELE..." in cui il carattere 'O' é stata sostituita dal carattere '1'. Questo dimostra la grande flessibilitá e interoperabilitá tra terminale e QMove.

Gestione eventi temporali

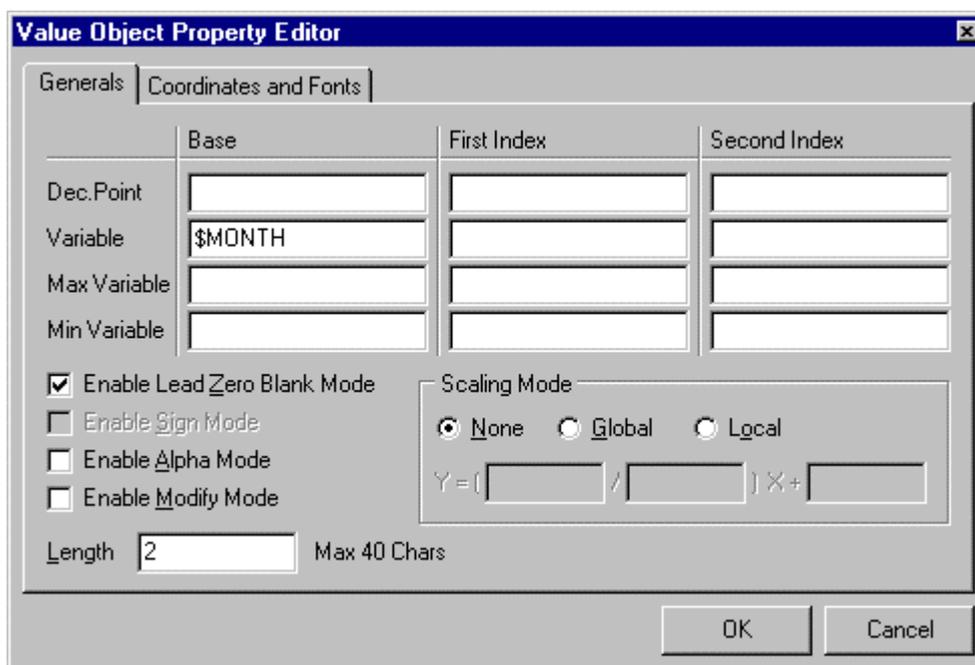
Vediamo con questo esempio come generare degli eventi a scadenze prefissate, facendo uso dell'evento OnTime. Questo é utile per avvisare l'operatore sulla necessitá di effettuare determinati interventi, ad esempio di manutenzione o di controllo, che devono essere eseguiti a scadenze o a date fisse.

Supponiamo, per semplificare le cose, di avere una sola pagina del terminale che descrive il funzionamento in corso, e di avere invece 3 pagine di avviso su operazioni da compiere; queste pagine saranno attivate in determinati momenti. Una pagina (1) indica di provvedere ad operazioni di ingrassaggio motore, una (2) indica di eseguire operazioni di manutenzione e pulizia filtri, la (3) ricorda di effettuare un backup dei dati di lavoro della macchina. Vogliamo che queste pagine vengano attivate rispettivamente: per la pagina (1) ogni mese, al giorno 15, alle ore 12; per la pagina (2) ogni mese 1, 4, 7 e 10 (operazioni trimestrali), per la pagina (3) ogni giorno alle ore 17:30. Tutte le operazioni che coinvolgono l'evento OnTime fanno riferimento al timer del terminale, che é possibile impostare nella pagina Info, ed é possibile leggere usando le variabili terminale \$YEAR, \$MONTH, \$DAY, \$HOUR, \$MIN, \$SEC.

Vediamo come poter fare tutto questo. La pagina di lavoro é cosí strutturata:

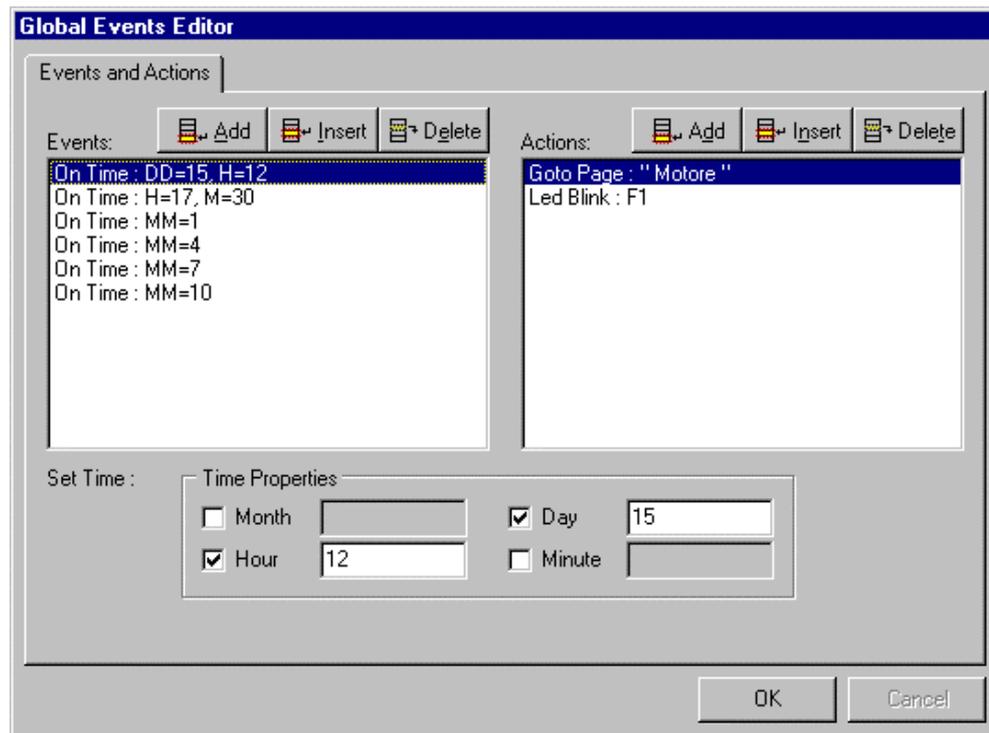


Per visualizzare la data e l'ora abbiamo usato 5 oggetti di tipo VALUE in cui come Base si sono definite rispettivamente le variabili terminale \$DAY, \$MONTH per la data e \$HOUR, \$MIN, \$SEC per l'ora. La pagina delle proprietà di una di queste appare così:



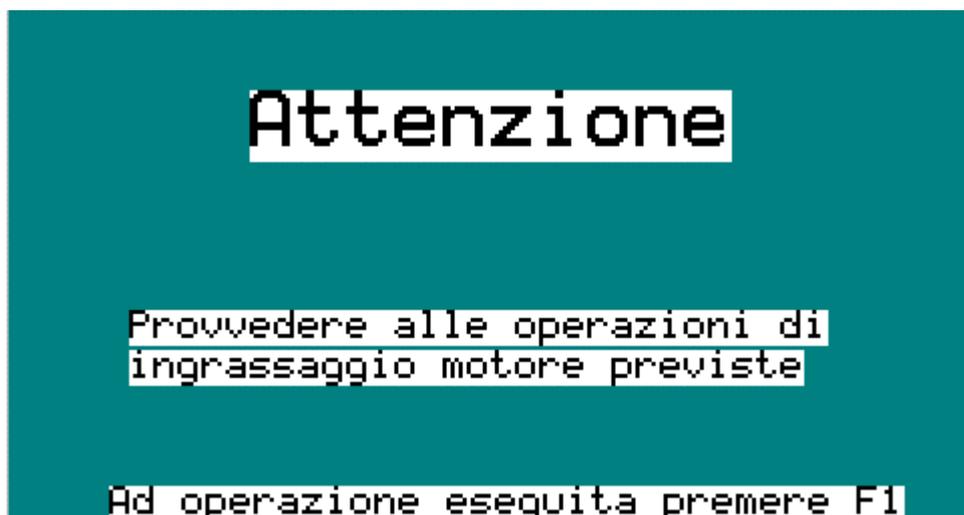
E' stata abilitata la modalità "Enable Lead Zero Blank" in modo da far precedere i numeri con una cifra dallo zero. In questo modo la data viene rappresentata ad es. 15/07 al posto di 15/7 come sarebbe invece se tale modo non fosse stato abilitato. Analogamente si procede per le altre variabili. Questa indicazione ci serve per poi testare il funzionamento del terminale e verificare se quanto abbiamo programmato viene effettivamente eseguito.

Il passaggio alle varie pagine di avviso viene impostato usando eventi globali: questo perché potrebbero esserci più pagine che descrivono il normale funzionamento della nostra macchina e noi vogliamo essere sicuri che gli eventi temporali vengano segnalati sempre, in qualsiasi punto ci troviamo. Vediamo come si presenta la finestra degli eventi globali:

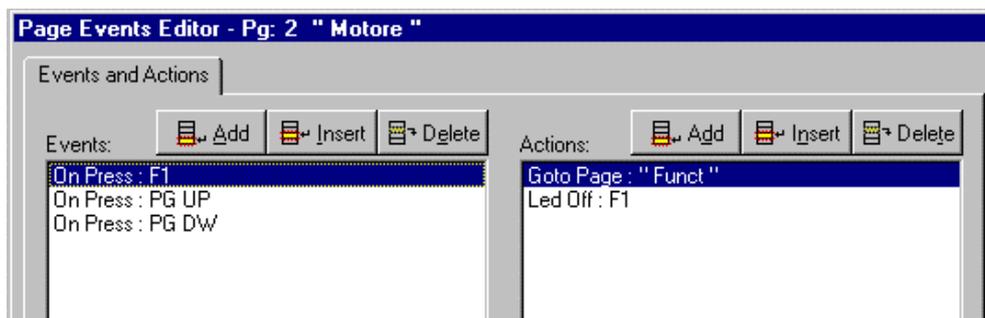


Vediamo che ci sono solo eventi di tipo OnTime; nella figura è rappresentato l'evento che attiva la pagina "Motore" ogni giorno 15, alle ore 12. Analizziamo le proprietà dell'evento OnTime; esse sono: Month, Day, Hour e Minute. Se nessuna di queste è selezionata, l'evento non viene mai generato, è come se non esistesse. Se invece è selezionato qualcosa, l'evento accade quando la proprietà in esame assume per la prima volta il valore impostato. Se vi sono più proprietà selezionate, l'evento è generato quando si verifica la combinazione dei valori delle proprietà stesse. In questo esempio quando Day assume il valore 15 e quando Hour assume il valore 12 l'evento viene generato e come azioni conseguenti si passa alla pagina "Motore" e si fa lampeggiare il led F1. Combinando opportunamente le quattro proprietà possiamo gestire una qualsiasi temporizzazione, con la precisione del minuto. Una cosa da ribadire è che è il valore della proprietà che determina la generazione dell'evento, quindi se vogliamo generare degli eventi che cadono mensilmente, non dobbiamo lavorare sulla proprietà Month, come potrebbe sembrare a prima vista, ma su Day, in quanto l'evento si verificherà ogni volta che tale proprietà assume il valore da noi specificato (e questo vuol dire allo stesso giorno di ogni mese). Così dicasi se vogliamo degli eventi giornalieri: dobbiamo lasciare non selezionata la proprietà Day e lavorare invece su Hour e Minute, in modo che alla stessa ora di ogni giorno si verifichi l'evento. Vediamo gli altri eventi; il passaggio alla pagina di salvataggio dati viene fatto, come si vede, ogni volta che la var H (Hour) assume il valore 17 e la var M (Minute) il valore 30; ciò vuol dire che ogni giorno, alle ore 17:30 apparirà l'avviso di salvare i dati di lavoro. Le altre impostazioni invece riguardano l'avviso di pulizia filtri, che avviene ogni 3 mesi, come si vede dalle impostazioni della variabile MM (Month); l'evento viene generato quando la var Month assume i valori 1, 4, 7 e 10. E' da ricordare inoltre che l'evento è generato nel momento del "passaggio" ai valori impostati, per cui se non si rileva questo passaggio (es. il terminale è spento) l'evento non sarà più rilevabile. Nell'esempio poi ad ogni passaggio pagina viene fatto lampeggiare un led diverso, che servirà nelle pagine di avviso per indicare quale tasto funzione premere dopo aver eseguito l'operazione e tornare nella pagina di "macchina in funzione". Vediamo come si presenta una pagina di avviso (le altre

sono analoghe e non verranno descritte in dettaglio):



Gli eventi di pagina vengono così impostati:



Premendo il tasti F1, il cui led lampeggia, si torna alla pagina di funzionamento e si spegne il led F1. Inoltre, dato che questa é una pagina di avviso a cui vi si arriva solo a scadenze fissate, sono stati sovrascritti i tasti PGUP e PGDN, in modo che l'unico modo per uscire sia quello di usare il tasto funzione. Lo stesso succede per le altre pagine di avviso, di cui vediamo la rappresentazione



Pagina di presentazione

Con questo esempio vediamo come realizzare una pagina iniziale di presentazione temporizzata (ad es. con un logo) in modo che l'avvio del programma su terminale e quello su QMoveÒ si sincronizzino. Ciò che vogliamo ottenere è che la partenza dei due programmi sia contemporanea, indipendentemente dall'ordine di accensione di terminale o QMoveÒ. Vediamo intanto la nostra pagina con il logo; essa potrebbe essere:



Vogliamo che all'accensione appaia questa pagina, che resti visualizzata per 5 secondi e poi passi alla pagina successiva, dove inizia il normale funzionamento del nostro applicativo terminale; nel nostro esempio vi sarà solo una pagina che non fa niente (appare solo un testo), ma la cosa è assolutamente ininfluente sul funzionamento della pagina temporizzata. Intanto vediamo come inserire un'immagine .BMP in una pagina di terminale. Per ogni progetto .QTP c'è la possibilità di creare una galleria di immagini che possono essere utilizzate nel progetto stesso e che vengono scaricate nel terminale durante la fase di download. Per costruire questa raccolta occorre utilizzare la funzionalità offerta dal menu "Graphics \ Image Manager" di QPaint, che fa apparire questa finestra:

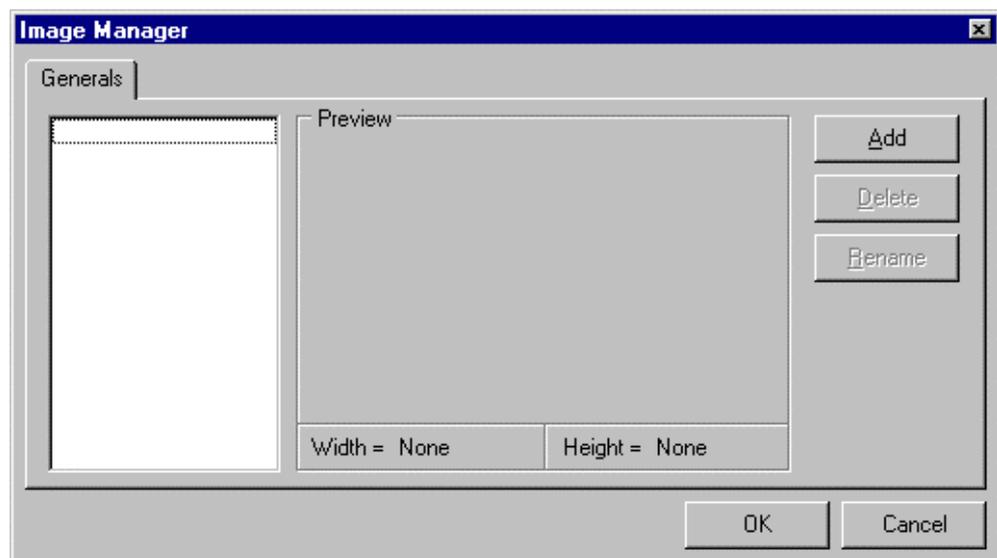
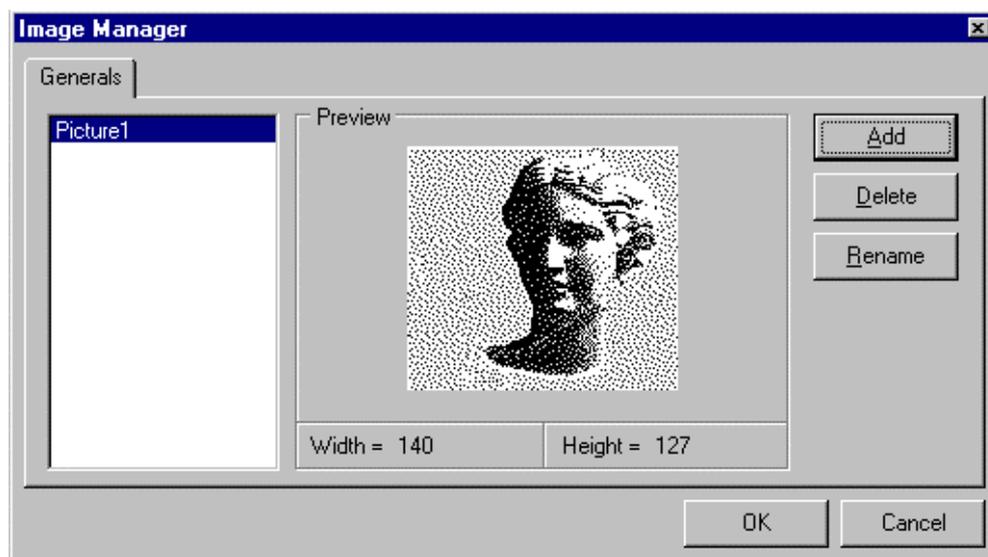
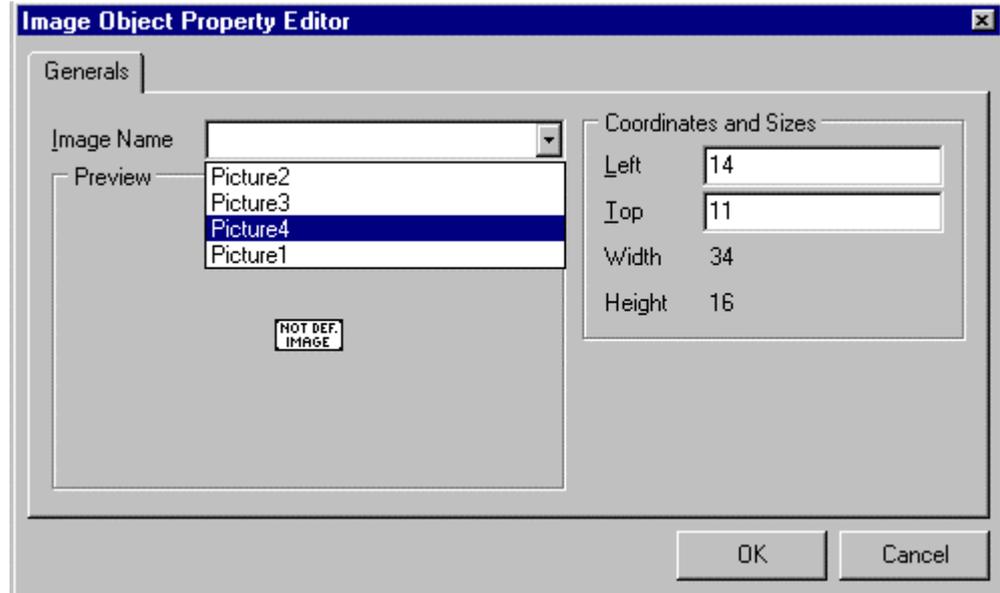


Image Manager, come dice il nome stesso, permette di gestire la libreria di immagini legate al progetto QTP in esame. Inizialmente tale libreria é vuota ed é abilitato il solo pulsante Add, che permette di aggiungere una immagine .BMP alla lista; tale immagine deve essere monocromatica, altrimenti QPaint lo segnala con un errore. Usando la funzione Add, appare una dialog box in cui si può scegliere un file .BMP; una volta selezionato il file la dialog box si chiude e la finestra Image Manager appare così modificata:



Come si vede, a sinistra appare la lista delle immagini presenti; ad ogni immagine viene dato un nome di default (Picture nnn dove nnn é un numero relativo alla prima posizione libera nella lista) che é possibile modificare usando il pulsante Rename. Al centro appare il Preview dell'immagine appena inserita in lista, con le sue dimensioni. Sulla destra i tasti con le funzioni possibili; abbiamo già visto Add e Rename, Delete effettua la rimozione dalla lista dell'immagine selezionata. Con OK si conferma l'inserimento della nuova immagine nella lista. Una volta fatto questo, per poter effettivamente aggiungere l'immagine nella pagina prescelta, occorre selezionare un Image Object e posizionare l'oggetto stesso facendo click nel punto desiderato della pagina. Appare quindi questa finestra:



in cui è possibile selezionare dalla lista di immagini costruita con Image Manager, quella che ci interessa. E' possibile anche impostare la posizione indicando il valore di Left e Top che corrispondono alle coordinate dell'angolo superiore a sinistra dell'immagine.

Quindi abbiamo completato la nostra pagina aggiungendo un box per lo sfondo e un paio di oggetti String per i testi.

Per ottenere la sincronizzazione del funzionamento di terminale e QmoveÒ, dobbiamo avere alcune variabili che contengano lo stato in cui si trovano i due dispositivi. Vediamo come va scritto l'applicativo QCL che ci permette questo.

Usiamo variabili GLOBAL, in modo che ad ogni restart dell'applicativo o riaccensione di QmoveÒ vengano inizializzate a zero.

```
-----  
; Definizione Variabili GLOBAL  
-----  
...  
gfTerminale F ;Verifica la presenza del terminale  
gfRun F ;Indica che il terminale non é nella  
 ;pagina di presentazione  
gfQMove F ;Utilizzata per l'uscita dalla pagina  
 ;iniziale di logo  
...  
-----  
; Definizione Variabili TIMER  
-----  
...  
tmLogo ;Timer per tempo pagina di  
 ;presentazione  
...
```

E viene anche definita una variabile TIMER tmLogo che ci servirá per impostare il timeout di cambio pagina.

Nel codice task avremo le seguenti istruzioni

```
-----  
; Operazioni di Inizializzazione TASK  
-----  
  
WAIT gfTerminale ;Verifica la presenza del terminale  
tmLogo = 5000 ;imposta la temporizzazione della  
 ;pagina  
  
-----  
; Programma Principale  
-----  
MAIN:  
  
IF (tmLogo AND NOT gfRun) ;verifica il timeout  
 gfQMove = 1 ;serve per il cambio pagina in QPaint  
 gfRun = 1 ;evita di rientrare in questa parte  
 ;dopo il tempo tmLogo  
  
ENDIF  
  
IF (gfRun EQ 1) ;una volta scaduto il timeout  
 tmLogo = 5000 ;serve per tenere tmLogo a zero  
 ;e poter effettivamente vedere il  
 ;cambio di gfQMove  
  
ENDIF  
  
IF (gfRun EQ 1) ;Condizione per stabilisce se il  
 ;terminale
```

```
...                                     ;ha terminato la presentazione
...                                     ;Codice task sincronizzato con il
...                                     ;terminale

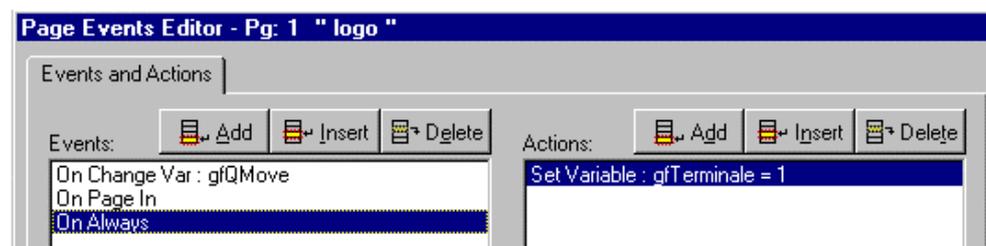
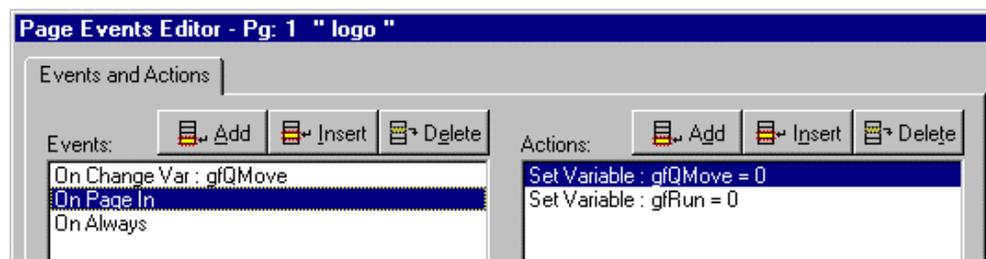
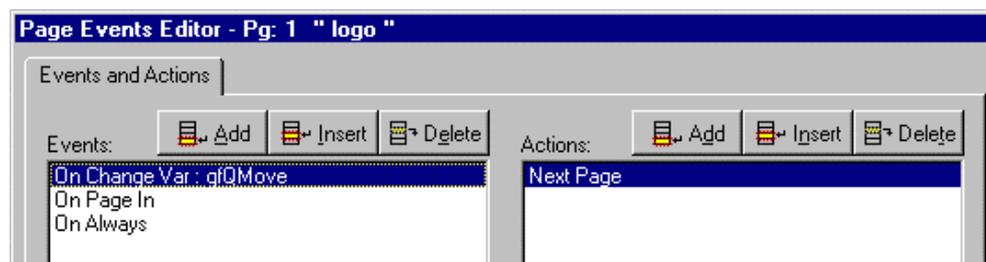
ENDIF

WAIT 1
JUMP MAIN
END
```

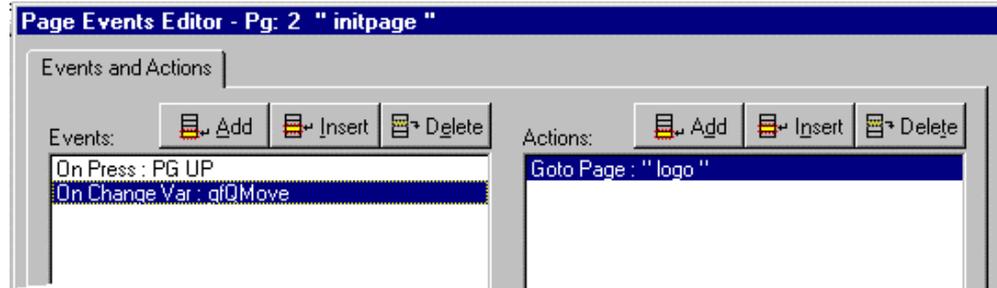
Durante la fase di inizializzazione l'applicativo QCL attende che il terminale sia nella pagina di presentazione verificando il valore di `gfTerminale`. Vedremo che il valore di questa variabile viene impostato a 1 da un evento `OnAlways` sulla pagina iniziale; quindi si imposta la variabile `TIMER tmLogo` con valore di 5000 (5 sec) che é quello che abbiamo scelto prima di passare alla pagina successiva. Nel programma principale poi avviene il controllo sia sullo scadere del timeout sia sullo stato dell'applicativo, in modo da non ripetere piú tale controllo se é già stato fatto una volta. All'inizio la variabile `gfRun` (essendo `GLOBAL`) é a zero, quindi dato che il controllo é fatto sul suo valore negato, l'istruzione `IF` dipende solo dal valore di `tmLogo`, che diventa 1 solo quando il timeout é scaduto, cioè dopo 5 sec dall'accensione del terminale. Una volta scaduto il timeout viene impostata a 1 la variabile `gfQMove`, che determina il cambio pagina nel terminale, e anche la var `gfRun` in modo che al ciclo successivo non si entri piú in questo ramo dell'`IF` e in modo da indicare all'applicativo che il terminale non sia piú in pagina di presentazione. Poi si deve sempre impostare `tmLogo` finché `gfRun` é a 1, in modo da essere pronti nel caso in cui si faccia il Restart dell'applicativo e si spenga il terminale. (Nota: non é possibile impostare una variabile `TIMER` da `QPaint`, per questo lo abbiamo fatto qui).

Vediamo ora come il terminale gestisce queste variabili. Le condizioni che si verificano sono: `gfQMove = 0` quando il terminale si trova nella pagina "logo", `gfQMove = 1` quando il terminale si trova in un'altra pagina. Quando la var `gfQMove` passa a zero (e questo vuol dire che la CPU é stata reinizializzata) si vuole che il terminale si riporti alla pagina "logo", parta il timeout e si ricominci.

Nella pagina "logo" avremo pertanto questi eventi di pagina, con le relative azioni:



All' accensione il valore di gfQMove é zero, e il suo passaggio a 1 avviene quando é scaduto il timeout e l'applicativo QCL ne assegna il valore; questo cambio di valore fa passare il terminale alla pagina successiva. Quando invece si entra nella pagina "logo" si resettano i valori di gfQMove e gfRun, in modo da verificare lo scadere del timeout e poter far scattare il cambio pagina con il cambio di valore di gfQMove. Infine l'evento OnAlways imposta la var gfTerminale sempre a 1 quando ci si trova nella pagina iniziale; questo dá il segnale all'applicativo QCL che il terminale é OK e si può far partire il timeout. Nella pagina di lavoro, e in tutte le pagine da cui si vuole uscire per tornare alla pagina "logo" ogni volta che si fa il restart della CPU, occorre verificare il cambio valore della variabile gfQMove e associare un salto pagina alla pagina "logo". Infine, per impedire il passaggio dalla pagina "di lavoro" alla pagina di logo con il tasto PgUp basta sovrascrivere la pressione del tasto PGUP non associando alcuna azione. Così l'unico modo per arrivare alla pagina di logo é o all'accensione o ad ogni restart della CPU.

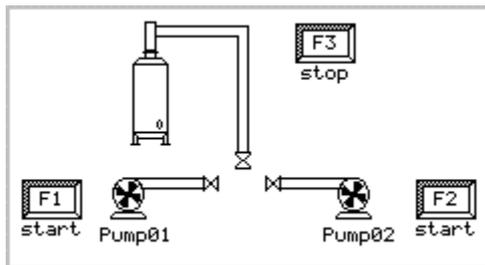


Vediamo ora il comportamento in alcune condizioni:

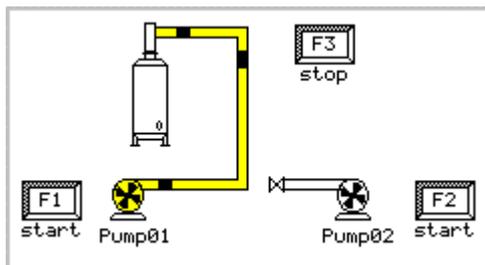
- Accensione del sistema QmoveÒ prima del terminale. In questo caso il task rimane in attesa dell'accensione del terminale per l'istruzione di WAIT sulla variabile gfTerminale. Poi allo scadere dei 5 sec. la variabile gfRun viene impostata e l'applicazione utente inizia ad essere eseguita.
- Accensione del terminale prima del sistema QmoveÒ. In questo caso il terminale non potendo comunicare con il sistema, continua a tentare la connessione. Non appena il sistema viene acceso il terminale trovando la connessione attiva esegue la scrittura della variabile gfTerminale e poi il funzionamento rimane quello descritto precedentemente.
- Durante il normale funzionamento il sistema viene spento e riacceso. In questo caso il terminale torna nella pagina di presentazione perché la variabile gfQMove si trova ad assumere il valore zero. Pio il comportamento rimane il medesimo.
- Durante il normale funzionamento il terminale viene spento e riacceso. Alla riaccensione del terminale la variabile gfRun viene posta a zero interrompendo ogni operazione nell'applicativo (o segnalando all'applicativo che il terminale è stato riattivato). Dopo 5 sec. il terminale si porta nella pagina di funzionamento.

Animazione

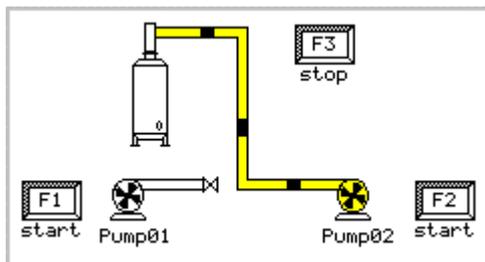
Con questo esempio vediamo come realizzare una semplice animazione, sfruttando gli oggetti VALSTRING e VALUE (con Array a Byte) e nuovi font creati appositamente, ottenuti a partire da copie dei file bitmap contenenti i font di sistema esistenti. Inoltre sfrutteremo alcune immagini bitmap create ad hoc. L'esempio richiede alcuni passi, che descriveremo in dettaglio uno per uno. Ciò che si vuole ottenere è un semplice sinottico, che visualizza il funzionamento di 2 pompe che, alternativamente, sono collegate ad un serbatoio di raccolta. L'animazione consiste nel visualizzare le pompe in funzione, il flusso di liquido attraverso le tubazioni, l'impianto in riposo con le tubazioni chiuse da valvole. Ecco subito come risulta l'effetto finale e poi vediamo come realizzarlo.



Impianto in condizioni di STOP



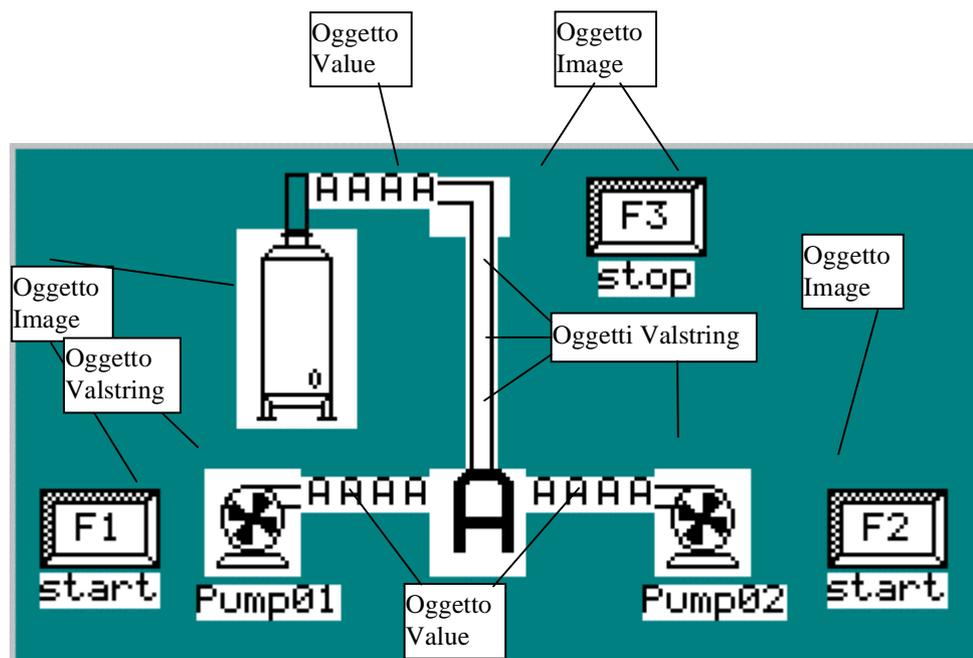
Impianto con pompa 1 in funzione
In evidenza le parti che si vedranno "in movimento" nel pannello terminale.



Impianto con pompa 2 in funzione
In evidenza le parti che si vedranno "in movimento" nel pannello terminale

La realizzazione dell'animazione è fatta nell'applicativo QCL, che descriveremo in seguito, mentre in QPaint c'è soltanto la gestione dei tasti F1, F2 e F3, insieme, ovviamente, all'utilizzo dei vari oggetti sia statici che dinamici che compongono questa pagina.

Vediamo ora come appare la pagina nel progetto QPaint, indicando quali oggetti sono stati utilizzati per creare l'effetto finale; come si nota appare leggermente diversa da quella che si vede in fase di esecuzione!

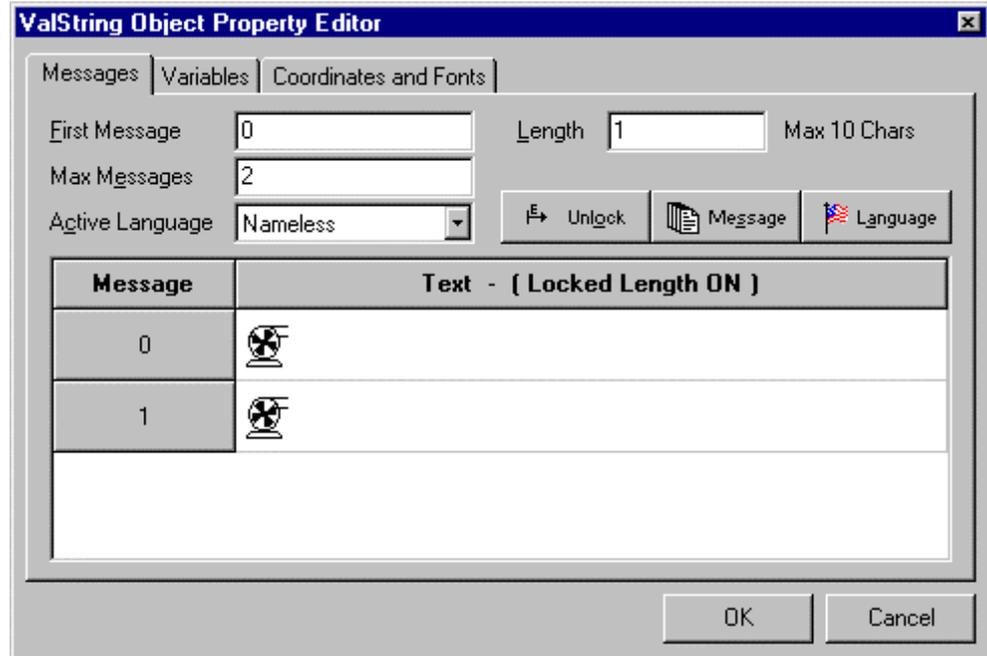


Per ottenere un'animazione sono utilizzabili piú oggetti diversi; procediamo con ordine.

Pompa: é stato utilizzato un oggetto Valstring che, si ricorda, é un oggetto dinamico che ha la possibilitá di visualizzare stringhe di testo diverse in funzione del valore posseduto dalla variabile collegata all'oggetto stesso. In questo caso viene visualizzato un solo carattere. Rispettivamente le variabili associate ai due oggetti ValString rappresentanti le pompe si chiamano gfPump01ON e gfPump02ON, sono di tipo Flag e dovranno essere definite nell'applicativo QCL. Per ottenere il movimento della pompa si associano 2 caratteri diversi ai 2 valori che decidiamo potrà assumere la variabile collegata e in fase di esecuzione si assegnano alternativamente questi due valori. Come si vede però questi 2 caratteri sono in realtà due simboli grafici. Come fare per ottenere ciò? Occorre definire un proprio font personalizzato, dove al posto dei soliti caratteri alfanumerici di tastiera possiamo sostituire dei disegni, come appunto in questo caso. La posizione in cui saranno introdotti nel font, determinerà quale carattere della tastiera premere affinché siano visualizzati.

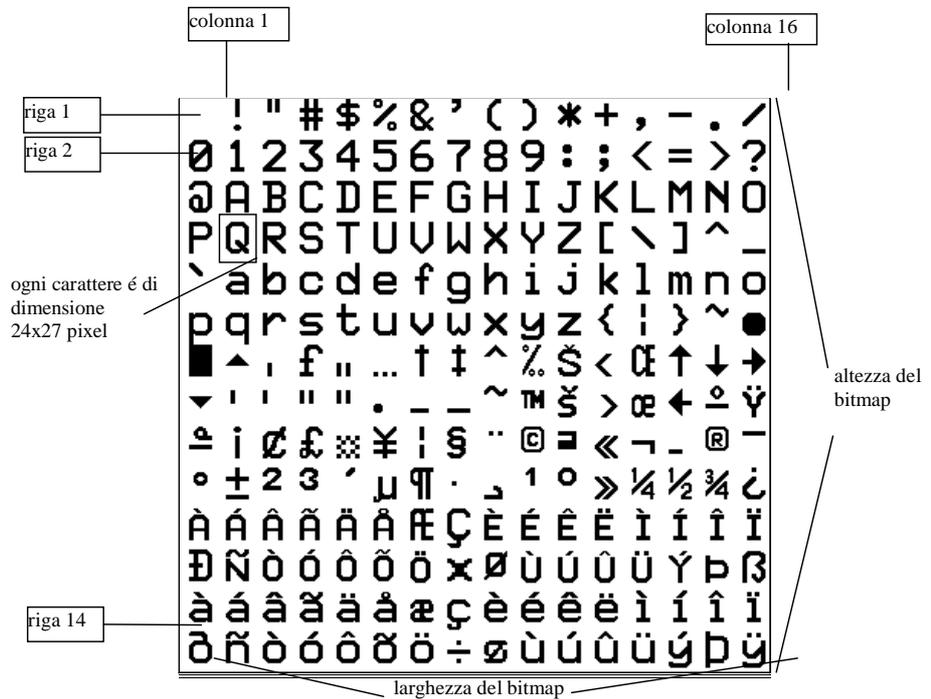
Vediamo intanto la finestra delle proprietà dell'oggetto ValString per la pompa 1 (é assolutamente identico per la pompa 2 a parte la variabile collegata), con il carattere per il valore 0 e per il valore 1.

Durante l'esecuzione gli oggetti ValString vengono continuamente rinfrescati e quindi, alternando il valore tra 0 e 1 della variabile gfPump01ON, si visualizzano alternativamente i due "testi" relativi ai due valori. Questo continuo refresh rende l'animazione.



Come detto sopra, per far sí che al posto di un carattere appaia un simbolo grafico, occorre usare un font personalizzato, derivato da un file bitmap, e nel quale si modificano alcuni caratteri. Per la pompa 1 sono stati "sovrascritti" i caratteri '\$' e '%', mentre per la pompa 2 i caratteri '! e '#'. Tutto qui. Vediamo come creare un font personalizzato; innanzitutto partiamo da un file bitmap. Utilizziamo la copia di uno di quelli che si trovano nella directory Fonts di QPaint10, e scegliamo "QEM System Large Font.bmp" che visualizza dei caratteri di dimensione 24 x 27 pixel. Se volessimo utilizzare un altro bitmap, con dimensione dei caratteri diversa e magari di nostra ideazione, possiamo farlo a patto di suddividere il bitmap in 14 righe e 16 colonne, perché questo é il formato richiesto dalla funzione di menú "Graphics\New Font From" presente in QPaint. Ogni elemento di questa suddivisione rappresenta un carattere. Inoltre dobbiamo ricordarci che il bitmap deve essere monocromatico (bianco e nero). Ad esempio, se volessimo costruire un font con caratteri di dimensione 12 x 15 pixel dovremmo creare un bitmap di larghezza 12*16 (n. colonne) = 192 e di altezza 15 x 14 (n. righe) = 210.

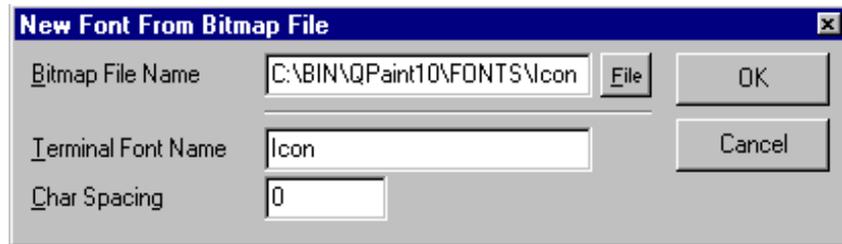
Il bitmap rappresentante i Large Font, che utilizzeremo per creare i caratteri visti sopra appare così:



C'è infine da dire che i caratteri sono disposti nel bitmap in ordine di codice, a partire dal codice 32 che corrisponde allo spazio bianco, fino al codice 255 che corrisponde a ÿ. Questo codice ci sarà utile in seguito. Creiamo il nostro font personalizzato partendo da questo bitmap. Ne facciamo una copia e lo chiamiamo "Icon Font.bmp"; ora lo editiamo e disegniamo, al posto dei caratteri !, #, \$ e % i nostri nuovi simboli. Il bitmap alla fine apparirà così:



Abbiamo anche aggiunto sue simboli che ci serviranno in seguito per il deviatore della tubazione. Una volta creato il bitmap occorre includerlo nel progetto QPaint tramite la funzione "Graphics\New Font From\Bitmap File". La finestra che appare é questa:



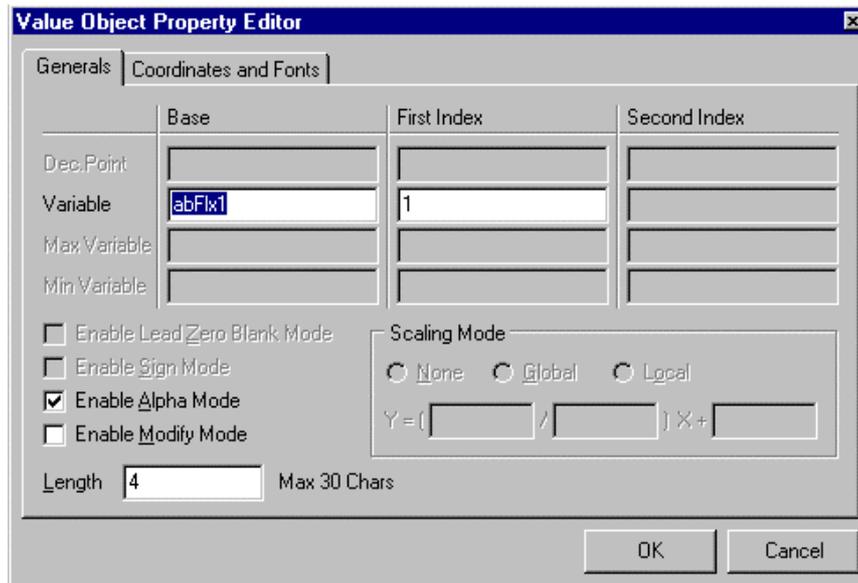
Nella casella "Bitmap File Name" va indicato il nome del file BMP che si vuole usare come nuovo Font. Il tasto a destra "File" permette di selezionarlo tramite la finestra di dialogo di Windows. Nella casella "Terminal Font Name" va invece scritto il nome che vogliamo dare al font e che apparirá sempre in questo progetto nella lista dei font disponibili. Una volta scelto OK viene fatta la conversione e da questo momento in poi ogni volta che apriremo la finestra "Font Properties", ci sará in lista anche "Icon 24x27s0" che é il nome interno dato da QPaint. "Icon" é il nome che avevamo digitato in fase di creazione, "24x27s0" ci ricorda la dimensione del font e lo spacing impostato.

Ritornando alla finestra "ValString Object Property Editor" vista sopra ora é facile capire che il font scelto per la stringa é proprio il nostro "Icon" e che la stringa scritta per il valore 0 é il carattere di tastiera '\$' e per il valore 1 é il carattere '%'.
\$ %

Tubazione in mandata delle pompe: per ottenere l'animazione di questa parte, che consiste nel visualizzare il fluido che si sposta, utilizziamo invece un oggetto Value, con una variabile Array byte con proprietá Alpha Mode attiva. Questo ci permette di visualizzare una stringa di caratteri variabile durante l'esecuzione e in cui é possibile modificare runtime anche un solo carattere. Trattandosi di una var array, il valore del singolo elemento rappresenta il carattere che sará visualizzato. Ad esempio il valore 32 visualizzerá uno spazio, il valore 48 il carattere '0' e cosí via. Per rappresentare questi caratteri creiamo un nuovo font, partendo dal bitmap "QEM System Small Font.bmp" e seguiamo la stessa procedura vista sopra. Il bitmap del nuovo font sará questo (é stato ingrandito per maggiore chiarezza):



Si vede che i primi caratteri sono stati sostituiti da quelli che ci serviranno per realizzare l'animazione delle tubazioni e le valvole. I codici che visualizzano tali simboli sono quelli da 32 a 37 compresi. Creiamo il nuovo font, partendo da questo bitmap, nel modo giá visto e lo chiamiamo "Flusso"; il nome completo che QPaint assegna a questo font é pertanto "Flusso 8x9s0".



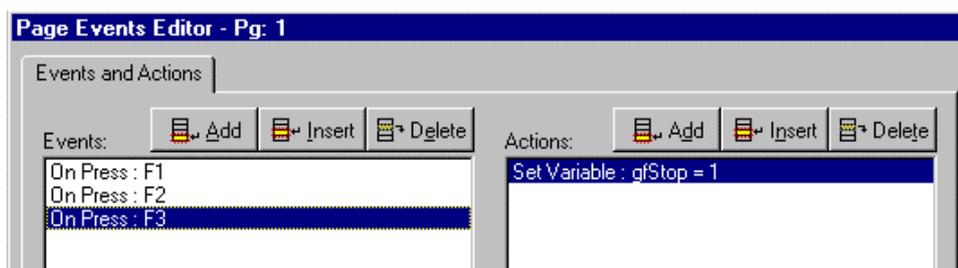
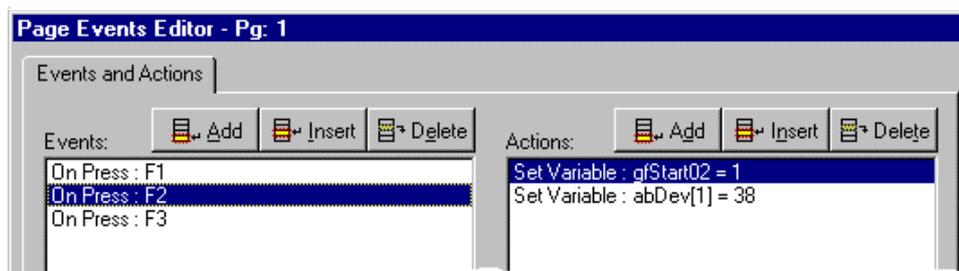
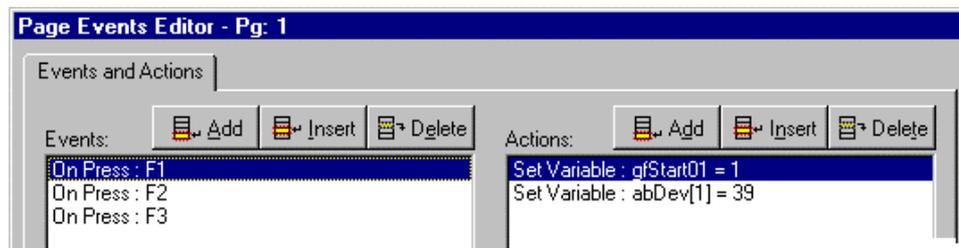
Per visualizzare la tubazione, come già detto, si crea un oggetto Value, e lo si collega ad una variabile QCL di tipo Array Byte (in questo caso abFlx1) e si imposta la proprietà Alpha Mode. Il font prescelto è "Flusso" e la lunghezza dell'oggetto è di 4 caratteri.

Per la tubazione in uscita dalla pompa 2 si userà in maniera analoga un altro oggetto Value legato alla variabile abFlx2 dell'applicativo QCL. Anche in questo caso l'animazione è controllata dal codice task QCL, che analizzeremo tra breve. Infine anche la tubazione orizzontale che confluisce nel serbatoio è fatta nello stesso modo, utilizzando la variabile abFlx3.

Deviatore: dato che nel vostro esempio le due pompe lavorano in alternativa c'è bisogno di rappresentare un raccordo che colleghi la pompa che sta funzionando alla tubazione verticale. In questo caso si ottiene con un oggetto Value, con la modalità Alpha Mode, di dimensione 1 e che è collegato alla variabile QCL abDev, che è un Array Byte. Il font utilizzato è "Icon" e i valori che assegnati alla variabile abDev saranno 39 per la pompa 1 e 38 per la pompa 2. Questi visualizzeranno i due "caratteri" corrispondenti agli opportuni bitmap rappresentanti le due posizioni del deviatore.

Tubazione verticale: per rappresentare questa parte è stata utilizzata una serie di oggetti ValString, disposti uno sull'altro, e collegati ciascuno ad un elemento di una variabile QCL di tipo array a byte che si chiama abTuboV. Il font usato è "Flusso"; il valore 0 rappresenta il carattere "tubo vuoto" (codice 34) mentre il valore 1 rappresenta il carattere "tubo pieno" (codice 35). La scelta di utilizzare un array deriva dal fatto che facilita la gestione dell'animazione da parte del codice task QCL. In tutto si tratta di 7 oggetti ValString; per il primo oggetto, il più in basso nel disegno, si associa un carattere "valvola" (codice 37) al valore 2 in modo da visualizzare la situazione di impianto in Stop.

Con questo abbiamo creato e impostato tutti gli oggetti responsabili dell'animazione. Vediamo ora come usare eventi ed azioni per dare il via all'animazione. Usiamo i tasti F1 per dare lo start alla pompa1 e posizionare il deviatore, F2 per dare lo start alla pompa 2 e posizionare il deviatore, F3 per fermare l'impianto. Il resto viene lasciato all'applicativo QCL.



Come si vede vengono impostate 3 variabili QCL per lo Start delle 2 pompe e per lo Stop dell'impianto. Vediamo ora come il codice QCL si occupa di gestire l'animazione.

Innanzitutto occorre definire le variabili necessarie nel file .CNF

```
-----  
; Definizione Variabili GLOBAL  
-----  
GLOBAL  
...  
...  
gfPump01ON    F           ;Gestisce l'animazione della pompa 1  
gfPump02ON    F           ;Gestisce l'animazione della pompa 2  
gfStart01     F           ;Pompa 1 in funzione  
gfStart02     F           ;Pompa 2 in funzione  
gfStop        F           ;Stop impianto  
gbStep        B           ;Var ausiliaria per cicli FOR  
glSecondi     L           ;Var per temporizzazione  
...  
...  
  
-----  
; Definizione Variabili GLOBAL  
-----  
ARRGBL  
...  
...  
  
abDev         B 5         ;Deviatore  
abFlx1        B 10        ;Tubo in mandata pompa 1  
abFlx2        B 10        ;Tubo in mandata pompa 2  
abFlx3        B 10        ;Tubo collettore serbatoio  
abTuboV       B 10        ;Tubo verticale  
...  
...  
  
-----  
; Definizione Variabili TIMER  
-----  
TIMER  
...  
...  
tmSecondi     ;Timer secondi  
...  
...
```

Sono stati usati degli array di dimensione maggiore a quella utilizzata dagli oggetti QPaint; questo non provoca alcuna conseguenza nel loro uso. Nel task vengono anche utilizzate due variabili (tmSecondi e glSecondi) per realizzare un contatore dei secondi di esecuzione dell'applicativo.

Nel task .MOD avremo invece:

```
.....
;ANIMAZIONE POMPA 1
IF (gfStart01 AND (abDev[1] EQ 39)) ;se la pompa1 é ON e "possiede"
                                     ;il deviatore

gfPump02ON = 0
gfPump01ON = glSecondi%2           ;Animazione pompa alternativamente
                                     ;imposta il valore di gfPump01ON a 0 e
                                     ;1, con ciclo di 1 secondo. Questo
                                     ;porta al refresh dell'oggetto
                                     ;ValString rappresentante la pompa1,
                                     ;ogni volta con 2 caratteri diversi.
                                     ; questo ciclo fa in modo che gli array
                                     ;rappresentanti le tubazioni assumano
                                     ;valori corrispondenti a caratteri
                                     ;"tubo vuoto"

FOR (gbStep = 1, gbStep LE 10, 1)
  abFlx1[gbStep] = 32             ;Oggetto Value tubazione pompa1
  abFlx2[gbStep] = 32             ;Oggetto Value tubazione pompa2
  abFlx3[gbStep] = 32             ;Oggetto Value tubazione serbatoio
  abTuboV[gbStep] = 0             ;Oggetto ValString tubo verticale
NEXT
abFlx1[1+glSecondi%5] = 33
;imposta quale elemento dell'array deve rappre sentare la parte di
;"tubo pieno" nella tubazione pompa1
abFlx3[5-glSecondi%5] = 33
;imposta quale elemento dell'array deve rappresentare la parte di
;"tubo pieno" nella tubazione serbatoio
abTuboV[1+glSecondi%8] = 1
;imposta quale elemento dell'array deve rappresentare la parte di
;"tubo pieno" nella tubazione verticale
abFlx2[1] = 36
;imposta la visualizzazione della valvola nella parte di tubazione
:pompa2

ENDIF

; GESTIONE DI UN CONTASECONDI
IF tmSecondi                       ;Timer da 1 Sec. terminato ?
  tmSecondi = 1000                 ;Reload timer
  IF glSecondi LT 59               ;Controllo secondi
    glSecondi = glSecondi + 1     ;Incremento secondi
  ELSE
    glSecondi = 0
  ENDIF
ENDIF
.....
```

Per simulare lo scorrere del liquido nella tubazione occorre impostare in maniera sequenziale e ciclica, in questo caso ogni secondo, un elemento diverso dell'array in modo che visualizzi il carattere "tubo pieno", mentre il resto dell'array deve rappresentare il "tubo vuoto". Quindi prima si effettua un ciclo FOR, che serve per "pulire" le tubazioni e poi si passa ad impostare il solo elemento che contiene il carattere di "tubo pieno". Con l'espressione `1+glSecondi%5` usata come indice di `abFlx1`, ad esempio, ad ogni secondo si incrementa di 1 il valore dell'indice con la sequenza : 1, 2, 3, 4, 5. Nel nostro caso l'elemento 5 non è visualizzato e questo fa sì che una volta che il fluido è arrivato alla destra del tubo, non riparta subito da sinistra, ma attenda un altro secondo. Questo rende migliore l'animazione. Dato che il fluido di questo tratto di tubo deve andare da sinistra verso destra, l'indice dell'array deve essere incrementato. Invece per la tubazione che confluisce nel serbatoio il fluido deve viaggiare da destra verso sinistra e quindi l'indice degli array deve decrescere; ecco che per `abTuboV` la sequenza di aggiornamento sarà: 5, 4, 3, 2, 1. Anche in questo caso l'elemento 5 non è compreso nella visualizzazione. La stessa cosa accade per il tubo verticale: qui occorre avere un ciclo di lunghezza 8, dato che il tubo è composto da 7 oggetti `ValString`. Si noti che non si imposta il codice del carattere, ma il valore che la variabile deve assumere; 0 = tubo vuoto, 1 = tubo pieno. L'uso dell'array dá la possibilità di gestire correttamente l'animazione. Infine per visualizzare la valvola nel tratto di tubo collegato alla pompa 2 e non utilizzato, si imposta il codice 36 che corrisponde appunto al carattere "valvola" nel primo elemento dell'array (visualizzato come il piú a sinistra).

Analogamente per il funzionamento della pompa2.

```
;ANIMAZIONE POMPA 2
IF (gfStart02 AND (abDev[1] EQ 38))
;se la pompa2 é ON e "possiede" il deviatore
    gfPump01ON = 0
    gfPump02ON = glSecondi%2
;Animazione pompa2: alternativamente imposta il valore di gfPump02ON a
;0 e 1, con ciclo di 1 secondo. Questo porta al refresh dell'oggetto
;ValString rappresentante la pompa2, ogni volta con i 2 caratteri
;diversi.
;questo ciclo fa in modo che gli array rappresentanti le tubazioni
;assumano valori corrispondenti a caratteri "tubo vuoto"

FOR (gbStep = 1, gbStep LE 10, 1)
    abFlx1[gbStep] = 32      ;Oggetto Value tubazione pompa1
    abFlx2[gbStep] = 32      ;Oggetto Value tubazione pompa2
    abFlx3[gbStep] = 32      ;Oggetto Value tubazione serbatoio
    abTuboV[gbStep] = 0      ;Oggetto ValString tubo verticale
NEXT

    abFlx2[5-glSecondi%5] = 33
;imposta quale elemento dell'array deve rappresentare la parte di
;"tubo pieno" nella tubazione pompa2
    abFlx3[5-glSecondi%5] = 33
;imposta quale elemento dell'array deve rappresentare la parte di
;"tubo pieno" nella tubazione serbatoio
```

```
abTuboV[1+glSecondi%8] = 1
;imposta quale elemento dell'array deve rappresentare la parte di
;"tubo pieno" nella tubazione verticale
abFlx1[4] = 36
;imposta la visualizzazione della valvola nella parte di tubazione
;pompa1 (va nell'elemento piú a destra)
ENDIF
```

Infine quando l'impianto é in Stop si resetta tutto

```
; STOP IMPIANTO
IF (gfStop)
  gfStop = 0
  gfPump01ON = 0
  gfPump02ON = 0
  gfStart02 = 0
  gfStart01 = 0
  abDev[1] = 32           ;al posto del deviatore si scrive uno
                          ;spazio

  FOR (gbStep = 1, gbStep LE 10, 1)
    abFlx1[gbStep] = 32
    abFlx2[gbStep] = 32
    abFlx3[gbStep] = 32
    abTuboV[1+glSecondi%8] = 0
  NEXT

  abFlx1[4] = 36         ;valvola alla fine della tubazione
                          ;pompa1
  abFlx2[1] = 36        ;valvola alla fine della tubazione
                          ;pompa2
  abTuboV[1] = 2        ;valvola nella parte bassa tubo
                          ;verticale

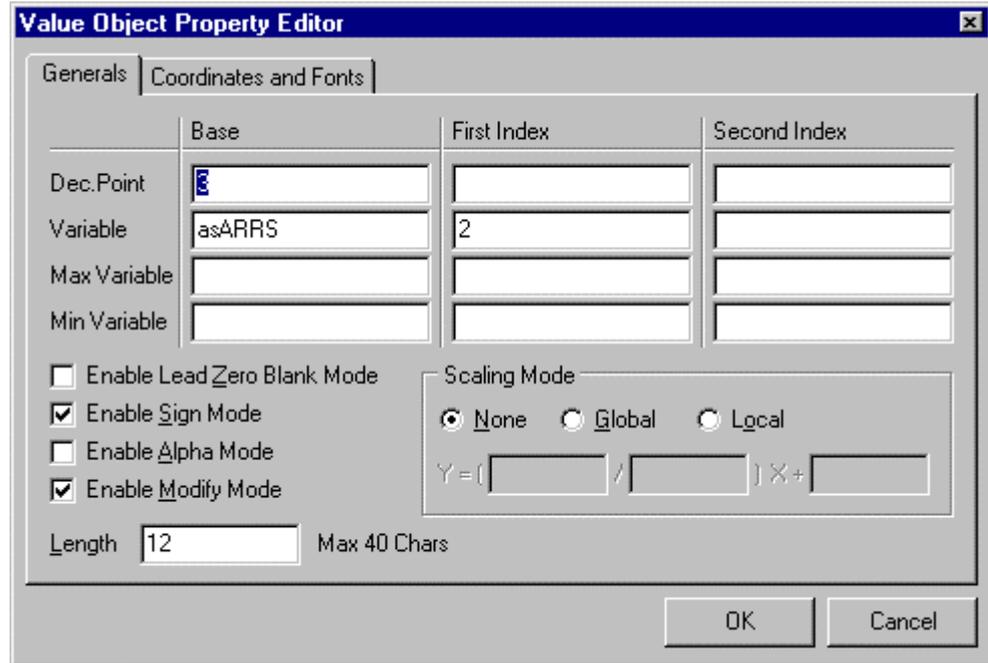
ENDIF
```

Backup e Restore

In questo esempio vediamo come realizzare una semplice pagina in cui poter eseguire le operazioni di Backup e Restore dei dati ritenitivi di QMove. L'unica cosa da ricordare é che l'operazione di Restore può essere fatta solo dopo aver effettuato l'operazione di backup e fino a che non si effettua un nuovo download dell'applicativo, dopo di che il Restore provoca un errore. Questo viene segnalato dal terminale con il lampeggio del led "status" e con l'informazione "Restore error" nella Info Page alla voce "Comunic. status". L'impossibilità di poter fare il Restore dopo un download deriva dal fatto che potremmo aver scaricato un applicativo diverso da quello cui si riferiscono i dati salvati con il Backup, che quindi potrebbero non avere più senso con la nuova situazione. Detto questo costruiamo una semplice pagina, con alcuni oggetti VALUE che servono per verificare la correttezza delle operazioni:

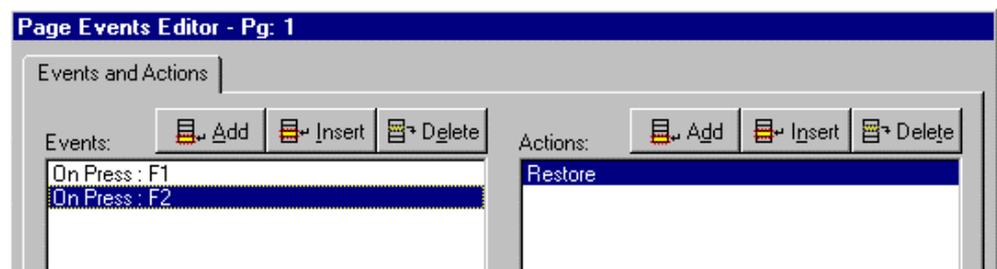
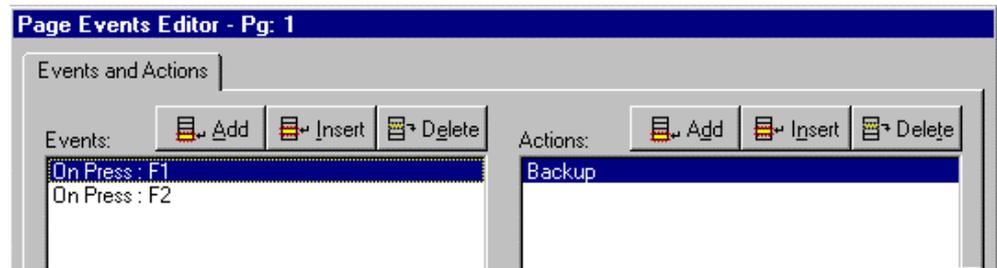


I 3 oggetti VALUE hanno tutti la proprietà "Enable Modify Mode" abilitata, in modo da poterli modificare da terminale; esaminiamo in particolare solo la variabile asARRS[2], perché si tratta di un array di tipo Single, in cui vediamo come visualizzare variabili con decimali.



Come si vede é stato impostato il valore relativo al decimal point che rappresenta il numero di cifre decimali che vogliamo visualizzare per la variabile. E' stato abilitato il segno e, come già anticipato, la possibilità di poter modificare il valore. Ora dobbiamo associare opportunamente le azioni di Backup e Restore a due tasti funzione.

La cosa é veramente semplice: nella finestra relativa agli eventi locali creiamo degli eventi OnPress sui tasti F1 e F2 ed associamo rispettivamente le azioni di Backup e Restore. Il passaggio alla fase di Data Entry con il tasto INS é invece di default e quindi non serve alcun evento collegato a tale tasto.



Una volta scaricata l'applicazione su terminale, impostiamo dei valori nelle 3 variabili, premendo INS per entrare in Data Entry; il valore relativo a swWord lampeggia e ciò indica che è possibile modificarlo. Con il tasto ENTER si conferma l'introduzione e si passa al campo successivo, e così fino all'ultimo. Da qui con ENTER si ripassa al primo campo e così via in maniera ciclica. E' possibile spostarsi tra i vari campi di inserimento anche con i tasti cursore: in questo caso però i dati inseriti vengono persi. Una volta completata l'impostazione delle variabili, possiamo salvarli effettuando un Backup, con il tasto F1. Quindi li modifichiamo nuovamente, impostando dei nuovi valori; a questo punto premendo il tasto F2, viene eseguito il Restore e i dati ritorneranno quelli impostati in precedenza.

Sia per l'operazione di backup sia per quella di restore, la mancata esecuzione del comando viene segnalata dal terminale con il lampeggio del led "status" e con l'informazione rispettivamente "Backup error" e "Restore error" nella Info Page alla voce "Comunic. status". Il comando di restore trova la sua giusta utilità in caso di sprogrammazione dati nel sistema QMoveÒ. In questa situazione infatti il comando restore permette di ripristinare il normale funzionamento del sistema (se ovviamente era stato fatto preventivamente il backup).