



motion control technologies

# Personalizzazioni

ISaGRAF® Library

ARTECO MOTION TECH SPA. Tutti i diritti riservati.

Le informazioni contenute in questo documento sono soggette a cambiamenti senza comunicazione e senza impegno da parte di Artec. Il software, incluse le informazioni contenute in tutti i database, descritto in questo documento è fornito in base ad un contratto di licenza o un contratto di non divulgazione e può essere utilizzato o copiato solo in accordo con i termini del contratto. E' proibita dalla legge la copia del software se non specificatamente permesso nei termini del contratto. Nessuna parte di questo manuale può essere riprodotta in alcuna forma o mediante alcun mezzo, elettronico o meccanico, incluso fotocopie o registrazioni, per qualsiasi motivo senza la specifica autorizzazione scritta di Artec.

ARTECO MOTION TECH SPA. All right reserved.

Information in this document is subject to change without notice and does not represent a commitment on the part of Artec. The software, which includes information contained in any database, described in this document is furnished under a license agreement or nondisclosure agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the licence or nondisclosure agreement. No part of this manual may be reproduced in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Artec.

## Indice

Utilizzo variabili retentive.....	3
VersEpr.....	4
VersPer.....	4
IORF .....	5
Note .....	6
ReadAna.....	6
Note .....	6
SetAna.....	7
VarIndex.....	8
VarCpy .....	10
VarZip .....	12
MemSt.....	13
ConfSer .....	14
GstFlash .....	18
DecBin8.....	20
DecBin16 .....	21
BinDec8.....	23
BinDec16 .....	24
DecBCD8 .....	26
DecBCD16.....	27
BCDDec8 .....	29
BCDDec16.....	30
GstSer .....	32
ReadSer.....	34
WriteSer .....	36
RdKeyb.....	36
WrKeyb .....	41
WrNumber .....	51
ChgTAssi.....	53
Set_Ip .....	53
Appendice A – Variabili Virtuali .....	54
Appendice B – Utility di lettura e scrittura variabili via seriale.....	55
Appendice C – Libreria I/O .....	58
I/O Digitali .....	58
I/O Analogici .....	61
Stato alimentazione .....	61
Dati gestione Asse.....	62
Appendice D – Errori flash card.....	63
Appendice E – Compilatore Arteco per C167 .....	63

## Utilizzo variabili retentive

Nelle schede SU è riservata una zona di memoria per l'utilizzo delle variabili retentive, diversa a seconda del modello.

L'utilizzo di queste variabili è subordinato all'inserimento dei dati riguardanti la suddetta memoria all'interno delle **Application Run Time Option**. Per indicazioni relative alla compilazione del campo **Retained variables** occorre fare riferimento al capitolo **C.9.2 Application variables backup - Retained variables** del manuale **ISaGRAF User's Guide**.

La memoria riservata all'interno della scheda SU va dall'indirizzo esadecimale 0x114000 all'indirizzo esadecimale 0x11BFFF nel caso di su2xx, cnd51, sumd, mentre per le schede su1xx la memoria va da 0x10C000 a 0x10CFFF.

Per motivi interni occorre tener conto che è possibile allocare solamente un numero di byte pari per ogni variabile, quindi per allocare 10 variabili retentive di tipo booleano non si dovranno allocare 10 byte, ma 20 (10\*2). Per allocare 10 timer non si dovranno allocare 50 byte ma 60 (10\*6).

E' possibile utilizzare le configurazioni suggerite di seguito:

### SU210, SU212, SUMD, CND51

114000,A28,114A30,1454,115EA0,1E78,117D20,4000

Questa configurazione consente di allocare:

1300 variabili booleane  
 1300 variabili analogiche  
 1300 timer  
 64 messaggi (da 256 caratteri)

### SU110, SU112

10C000,C8,10C0D0,258,10C340,258,10C5A0,A00

Questa configurazione consente di allocare:

100 variabili booleane  
 150 variabili analogiche  
 100 timer  
 10 messaggi (da 256 caratteri)

Per qualsiasi informazione relativa all'allocazione in memoria di ulteriori variabili, o per diverse configurazioni contattare l'Arteco spa.

## VersEpr



### Argomenti:

En	BOO	Non utilizzato
versS	MSG	Stringa indicante codice e data della versione della eprom.
versC	INT	Codice della eprom indicato come stringa anche nell'uscita versS

\* Il codice e la data si riferiscono al firmware del kernel, sviluppato dall'Arteco, e non dipendono dall'applicativo.

### Descrizione:

Questo blocco consente di leggere la versione della eprom. Fornisce il codice della eprom e la data di creazione. Sono fornite due uscite per gestire rispettivamente un messaggio indicante tutte le informazioni ed un codice da utilizzare per monitorare ad ogni accensione se la eprom è cambiata. L'ingresso En al momento non svolge nessuna funzione.

Per qualsiasi informazione riguardante la versione in Vostro possesso contattare l'Arteco spa.

## VersPer



### Argomenti:

Typ	INT	Tipo di periferica di cui si vuole sapere la versione firmware.
Opt	INT	Parametro opzionale. Identifica quale periferica se presenti più di una dello stesso tipo.
VersS	MSG	Stringa indicante codice e data della versione del firmware della periferica.
versC	INT	Codice della versione della periferica (se disponibile).

\* Il codice e la data si riferiscono al firmware sviluppato dall'Arteco, e non dipendono dall'applicativo.

## Descrizione:

Questo blocco consente di leggere la versione del firmware di una qualsiasi periferica collegata al sistema SU. Fornisce il codice del firmware e la data di creazione. Sono fornite due uscite per gestire rispettivamente un messaggio indicante tutte le informazioni ed un codice da utilizzare per monitorare ad ogni accensione se la versione è cambiata. Quest'ultimo codice per alcune periferiche può non essere disponibile.

Il parametro Typ può valere:

- 0) Espansione assi bus.
- 1) Espansione dsp.
- 2) Espansione can.

Il parametro Opt rappresenta un parametro opzionale. Nel caso di espansione can indica il numero del nodo di cui si vuole conoscere la versione del firmware.

Per qualsiasi informazione riguardante la versione in Vostro possesso contattare l'Arteco spa.

## IORF



## Argomenti:

En	BOO	Se TRUE ad ogni chiamata del blocco avviene un rinfresco delle variabili Input con gli attuali valori degli Input fisici, oppure il rinfresco delle uscite fisiche con i valori delle variabili Output. L'operazione svolta dipende dal parametro Modo.
Modo	INT	Indica il tipo di rinfresco richiesto: 0 => Rinfresca le uscite. 1 => Rinfresca gli ingressi. 2 => Rinfresca ingressi e uscite.
Q	BOO	Non utilizzato. Indica sempre TRUE.

## Descrizione:

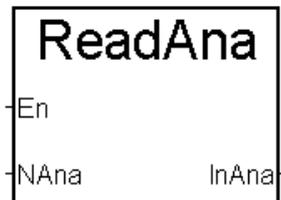
Questo blocco consente il rinfresco di ingressi e uscite in un qualsiasi punto del programma applicativo. Se questo blocco non viene utilizzato il rinfresco delle variabili Input avviene all'inizio di ogni scansione del programma PLC, mentre le uscite vengono rinfrescate alla fine di ogni scansione del programma PLC.

## Note

Il risultato di ogni "ramo" e' immediatamente disponibile per i "rami" successivi, sia un risultato interno o di output. I rami successivi quindi terranno immediatamente conto del valore teorico che le uscite assumono nei "rami" precedenti. L'attivazione "fisica" dell'uscita invece avviene al termine del ciclo PLC.

Quindi: Se un ramo attiva un output ed un altro "ramo" disattiva lo stesso output, tutti i rami intermedi verranno "elaborati" considerando l'output attivo, ma l'uscita fisica non sar  mai attivata, a meno di usare il blocco IORF.

## ReadAna



## Argomenti:

En	BOO	Se TRUE legge l'ingresso analogico fornendo la lettura all'uscita InAna. Lavora sul livello e non sul fronte, quindi mantenendo alto il livello di En si avr� il dato aggiornato ad ogni chiamata del blocco.
NAna	INT	Indica l'ingresso analogico di cui si vuole conoscere la tensione in ingresso. Al momento ammesso 0, 1 o 2.
InAna	INT	Lettura della tensione presente all'ingresso analogico selezionato espressa in mvolt. Es. 2.5volt => 2500.

## Descrizione:

Questo blocco consente la lettura degli ingressi analogici della scheda SU210, restituendo il valore della tensione espresso in mvolt.

Sono disponibili tre ingressi analogici, pertanto l'ingresso Nana pu  assumere i valori 0, 1 o 2.

## Note

Il convertitore analogico digitale ha una risoluzione di 10 bit, quindi il valore di tensione letto avrà una risoluzione di 19mvol risultante dalla divisione tra il range della tensione (+10v, -10v quindi un range di 20v) e il numero massimo restituito dal convertitore ( $2^{10}$  quindi 1024).

Il valore letto da questo blocco è disponibile anche mediante la libreria nS210Ana (vedi Appendice C Libreria I/O).

## SetAna



### Argomenti:

En	BOO	Se TRUE imposta l'uscita analogica. Lavora sul livello e non sul fronte, quindi mantenendo alto il livello di En ad ogni chiamata del blocco viene impostato il valore.
Chan	INT	Indica l'uscita analogica di cui si vuole impostare la tensione.
Vreq	INT	Indica il valore in millivolt da impostare sull'uscita analogica
Vout	INT	Valore dell'uscita analogica espresso in millivolt. Es. 2.5volt => 2500.

### Descrizione:

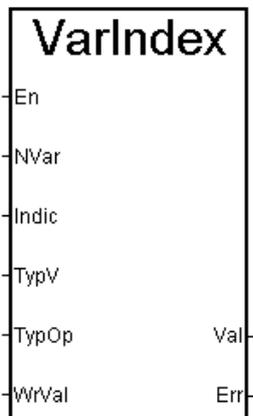
Questo blocco consente l'impostazione sulle uscite analogiche di un valore di tensione espresso in millivolt.

Sulla scheda SU210 sono disponibili due uscite analogiche corrispondenti a Chan = 5 e 6, due uscite analogiche pilotate in pwm corrispondenti a Chan = 7 e 8. È inoltre possibile utilizzare l'uscita relativa all'asse selezionando Chan=0 purché l'asse non sia stato parametrizzato con il blocco Parasse.

Sulla scheda CND51 è presente una uscita analogica selezionabile mettendo Chan=5. È inoltre possibile utilizzare le uscite relative agli assi selezionando Chan = 0, 1, 2, 3 e 4 purché i corrispondenti assi non siano stati parametrizzati con il blocco Parasse.

Sulla scheda SUMD è presente un'uscita analogica selezionabile con Chan=8.

## VarIndex



### Argomenti:

En	BOO	Se TRUE restituisce il valore della variabile il cui indice è la somma degli ingressi Nvar e Indic.
NVar	INT	Rappresenta il primo addendo che determina il Virtual Address della variabile di cui si vuole leggere il valore. Vedi descrizione.
Indic	INT	Rappresenta il secondo addendo che determina il Virtual Address della variabile di cui si vuole leggere il valore. Vedi descrizione.
TypV	INT	Indica se la variabile richiesta è ISaGRAF o Virtuale. 0 indica variabile ISaGRAF regolarmente dichiarata. 1 Indica variabile virtuale.
TypOp	INT	Indica il tipo di operazione da eseguire. 0 lettura della variabile. 1 scrittura della variabile.
WrVal	INT	Valore da scrivere nella variabile. Utilizzato esclusivamente se il parametro TypOp indica la scrittura (1).
Val	INT	Valore della variabile il cui virtual Address è indicato dalla somma degli indici rappresentati dagli ingressi Nvar e Indic.
Err	BOO	Indica se la lettura della variabile richiesta è avvenuta correttamente o se non è stata possibile. FALSE indica lettura corretta, TRUE indica lettura fallita.

## Descrizione:

Questo blocco consente la lettura o la scrittura del valore di una variabile mediante l'utilizzo del Virtual Address.

Sono presenti due ingressi (Nvar e Indic) la cui somma costituisce il Virtual Address utilizzato per raggiungere la variabile da leggere.

Questo blocco può essere utilizzato per la lettura o scrittura di una variabile mediante indicizzazione, dove Nvar può rappresentare una variabile di partenza e Indic un indice che si incrementa progressivamente.

Questo blocco gestisce una sola variabile per volta. Per il trasferimento di interi blocchi di variabili fare riferimento al blocco VarCpy.

E' possibile leggere sia una variabile ISaGRAF che una variabile virtuale, utilizzando l'ingresso TypV per indicare la scelta. Per le variabili Virtuali l'indice deve essere contenuto tra 1 e 8000 decimale.

Il Virtual Address viene inserito in esadecimale, mentre l'ingresso di questo blocco utilizza valori decimali. Quindi la variabile con Virtual Address 1000 deve essere raggiunta con indirizzo 4096 (corrispettivo decimale dell'indirizzo esadecimale 1000). Se il Virtual Address si riferisce ad una variabile analogica il risultato Val è un numero intero decimale, se si riferisce ad una variabile booleana Val può valere 0 (FALSE) oppure 1 (TRUE), mentre se si riferisce ad un timer è un numero intero decimale che rappresenta dei millisecondi.

Nel caso di scrittura il valore assegnato a WrVal sarà considerato come intero se la variabile di destinazione è un intero. Sarà considerato in msec se la variabile di destinazione è un timer. Nel caso di variabile di destinazione booleana se WrVal è uguale a zero sarà assegnata a FALSE, in tutti gli altri casi (qualsiasi numero diverso da zero) sarà assegnata uguale a TRUE.

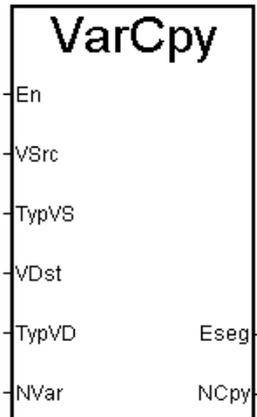
**N.B. Nonostante il blocco consenta di gestire variabili booleane, intere e timers l'uscita del blocco è comunque intera. Per scrivere detto valore in variabili booleane o timers è necessario utilizzare rispettivamente i blocchi Boo e Tmr.**

## Attenzione:

E' possibile introdurre numeri esadecimali nelle costanti utilizzando il prefisso 16#.

Es. Volendo leggere le 30 variabili successive al Virtual Address 2A00 si può fornire all'ingresso NVal una variabile inizializzata a 16#2A00 e utilizzare Indic dando i dieci valori da 1 a 30. Gli indirizzi letti andranno da 2A01 a 2A1E.

## VarCpy



### Argomenti:

En	BOO	Sul fronte positivo avvia la copia delle variabili.
VSrc	INT	Virtual Address della prima variabile sorgente.
TypVS	INT	Indica se la variabile sorgente è ISaGRAF o Virtuale. 0 indica variabile ISaGRAF regolarmente dichiarata. 1 Indica variabile virtuale. In caso di copia di più variabili il tipo si riferisce a tutte le variabili sorgente.
VDst	INT	Virtual Address della prima variabile destinazione.
TypVD	INT	Indica se la variabile di destinazione è ISaGRAF o Virtuale. 0 indica variabile ISaGRAF regolarmente dichiarata. 1 Indica variabile virtuale. In caso di copia di più variabili il tipo si riferisce a tutte le variabili di destinazione.
NVar	INT	Numero di variabili <u>consecutive</u> da copiare. Vedi descrizione.
Eseg	BOO	Indica se l'operazione di copiatura delle variabili è in corso. TRUE indica operazione in corso. FALSE indica operazione terminata.
NCpy	INT	Indica il numero di variabili effettivamente copiato. Normalmente vale NVar. Qualora fosse inferiore a NVar significa che alcune variabili non sono state copiate. Vedi descrizione.

### Descrizione:

Questo blocco consente di copiare un blocco di variabili BOOLEANE, ANALOGICHE INTERE o TIMER in un altro gruppo, anche di tipo diverso, purchè appartenente alle categorie appena citate. L'ingresso VSrc indica il Virtual Address della prima variabile da copiare, mentre l'ingresso VDst indica il Virtual Address della prima variabile da sovrascrivere con i nuovi valori. NVar indica il numero di variabili consecutive che devono essere copiate. Con consecutive ci si riferisce al valore del Virtual Address.

E' inoltre possibile copiare variabili da e nella zona di variabili virtuali. Utilizzando TypVS e TypVD è possibile indicare rispettivamente il tipo delle variabili sorgente e destinazione (ISaGRAF o Virtuali). Per le variabili Virtuali l'indice deve essere contenuto tra 1 e 8000 decimale.

Il Virtual Address viene inserito in esadecimale, mentre l'ingresso di questo blocco utilizza valori decimali. Quindi la variabile con Virtual Address 1000 deve essere raggiunta con indirizzo 4096 (corrispettivo decimale dell'indirizzo esadecimale 1000).

Siccome la copia di grossi blocchi di memoria potrebbe richiedere un certo tempo l'operazione viene suddivisa in più cicli di PLC in modo da non richiedere per ogni ciclo di PLC un tempo superiore a 4 millisecondi. Se le variabili da copiare sono poche, l'operazione sarà terminata al ciclo successivo del PLC, viceversa potrebbero essere necessari più cicli. Le uscite Eseg e NCpy consentono di sapere lo stato di avanzamento dell'operazione di copiatura.

Sul fronte positivo dell'ingresso En l'uscita Eseg si porta a livello alto per indicare l'operazione in corso. Quando l'uscita Eseg si porta a livello basso l'operazione è terminata e NCpy indica il numero di variabili effettivamente copiate. Questo numero deve essere uguale a NVar, viceversa nell'intervallo dei blocchi mancano dei Virtual Address (non dichiarati) oppure vi sono delle variabili di tipo non ammesso.

Ad ogni giro di PLC è possibile sapere le variabili effettivamente copiate mediante l'uscita NCpy , anche se l'operazione e' ancora in corso.

E' possibile copiare variabili di tipo diverso. Di seguito le conversioni effettuate nel caso di variabili sorgente e destinazione di tipo diverso:

Destinazione

- 1) Booleana. Se la variabile sorgente e' intera uguale a zero la destinazione vale FALSE, viceversa vale TRUE. Se la variabili sorgente è timer e uguale a zero vale FALSE, viceversa vale TRUE.
- 2) Intera. Se la variabile sorgente e' booleana uguale a FALSE la destinazione vale zero, viceversa vale 1. Se la variabili sorgente è timer e uguale a zero vale zero, viceversa vale il valore del timer espresso in millisecondi.
- 3) Timer. Se la variabile sorgente e' booleana uguale a FALSE la destinazione vale zero, viceversa vale 1millisecondi. Se la variabili sorgente è intera e uguale a zero vale zero, viceversa vale il valore della sorgente espresso in millisecondi.

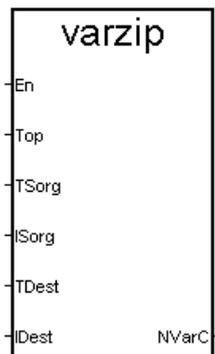
## Attenzione:

E' possibile introdurre numeri esadecimali nelle costanti utilizzando il prefisso 16#.

Es. Volendo copiare le 30 variabili successive al Virtual Address 2A00 a partire dalla variabile il cui Virtual Address è 3000 si può fornire all'ingresso VSrc una variabile inizializzata a 16#2A00 (oppure il numero esadecimale 10752), all'ingresso VDst il valore 16#3000 (oppure il numero decimale 12288) e indicare in NVar 30. I valori delle variabili con indirizzi compresi tra gli indirizzi 2A00 e 2A1E saranno copiati nelle variabili comprese tra gli indirizzi 3000 e 301E.

Se non sono state dichiarate le variabili da 3003 a 3005, al termine della copia NCpy indicherà 28.

## VarZip



### Argomenti:

En	BOO	Se l'ingresso En è TRUE si esegue l'operazione di compattazione o estrazione delle variabili.
TOp	INT	Indica il tipo di operazione che deve essere eseguita: 0 → Compattazione di 32 variabili (Booleane, Intere o Virtuali) in una variabile di 32 bit (Intera o Virtuale). 1 → Estrazione di 32 variabili (Booleane, Intere o Virtuali) da una variabile di 32 bit (Intera o Virtuale).
TSorg	INT	Indica il tipo di variabile sorgente: 0 → Variabile ISaGRAF® (Intera, Booleana o Timer). 1 → Variabile Virtuale.
ISorg	INT	Valore dell'indirizzo (o indice) della variabile sorgente (se TOp=1) o della prima variabile del gruppo di 32 variabili da compattare (TOp=0). I valori ammessi sono 1-FFFF esadecimale per le variabili ISaGRAF® e 1-8000 decimale per le variabili Virtuali.
TDest	INT	Indica il tipo di variabile destinazione: 0 → Variabile ISaGRAF® (Intera, Booleana o Timer). 1 → Variabile Virtuale.
IDest	INT	Valore dell'indirizzo (o indice) della variabile destinazione (se TOp=0) o della prima variabile del gruppo di 32 variabili da estrarre (TOp=1). I valori ammessi sono 1-FFFF esadecimale per le variabili ISaGRAF® e 1-8000 decimale per le variabili Virtuali.
NVarC	INT	Indica il risultato dell'operazione richiesta. Un suo <b>valore positivo</b> indica che l'operazione di compattazione o estrazione delle variabili è stata eseguita con successo. Il suo valore rappresenta il numero di variabili effettivamente compattate o estratte. Un suo <b>valore negativo</b> indica una condizione di errore. In particolare si possono avere i seguenti codici d'errore: -1 : L'indirizzo sorgente è errato (eccede dai valori massimi consentiti). -2 : La variabile sorgente è di tipo errato (es. Booleana con TOp=1). -3 : L'indirizzo destinazione è errato (eccede dai valori massimi consentiti). -4 : Variabile destinazione non definita. -5 : Variabile sorgente non definita. -6 : La variabile destinazione è di tipo errato (es. Booleana con Top=0).

## Descrizione:

Questo blocco consente di compattare 32 variabili con indirizzo consecutivo in un'unica variabile di 32 bit, oppure di scomattare una variabile di 32bit in 32 variabili con indirizzo consecutivo. Tali variabili possono essere sempre variabili ISaGRAF® o variabili virtuali.

Se TOP=0, il blocco consente di compattare 32 variabili con indirizzo consecutivo in un'unica variabile di 32. Se le variabili sorgente sono di tipo Booleano, la compattazione viene eseguita nel seguente modo:

- i-sima variabile=TRUE → i-simo bit della var. destinazione=1
- i-sima variabile=FALSE → i-simo bit della var. destinazione=0

Se le variabili sorgente NON sono di tipo Booleano (Virtuali, Intere o Timer), la compattazione viene eseguita nel seguente modo:

- i-sima variabile ≠ 0 → i-simo bit della var. destinazione =1
- i-sima variabile = 0 → i-simo bit della var. destinazione =0

Le variabili sorgente possono anche essere di tipo diverso (es. sia Booleano che Intero), ma debbono essere tutte definite e con indirizzi consecutivi. Se nel gruppo di 32 variabili sorgente una o più variabili non sono definite (e quindi gli indirizzi non sono consecutivi) il blocco ritorna errore.

Se la variabile destinazione è di tipo Boolean il blocco ritorna errore.

Se TOP=1, il blocco consente di estrarre 32 variabili con indirizzo consecutivo da un'unica variabile di 32 bit (o Virtuale o Intera di ISaGRAF). Se le variabili destinazione sono di tipo Booleano, la decompattazione viene eseguita nel seguente modo:

- i-simo bit della var. sorgente =1 → i-sima variabile destinazione =TRUE
- i-simo bit della var. sorgente =0 → i-sima variabile destinazione =FALSE

Se le variabili destinazione NON sono di tipo Boolean (Virtuali, Intere o Timer), la decompattazione viene eseguita nel seguente modo:

- i-simo bit della var. sorgente =1 → i-sima variabile destinazione =1
- i-simo bit della var. sorgente =0 → i-sima variabile destinazione =0

Le variabili destinazione possono anche essere di tipo diverso (es. sia Booleano che Intero), ma debbono essere tutte definite e con indirizzi consecutivi. Se nel gruppo di 32 variabili destinazione una o più variabili non sono definite (e quindi gli indirizzi non sono consecutivi) il blocco ritorna errore.

Se la variabile sorgente è di tipo Booleano il blocco ritorna errore.

L'operazione avviene ad ogni chiamata del blocco VarZip se l'ingresso En è TRUE, mentre se l'ingresso è FALSE non si esegue alcuna operazione e l'uscita NvarC è mantenuta uguale a zero.

## MemSt



## Argomenti:

Type	INT	Indica il tipo di memoria di cui si vuole sapere l'utilizzo. 1 => Dimensione codice applicativo. 2 => Dimensione dati applicativo.
Used	INT	Memoria utilizzata in KB.
Avail	INT	Memoria disponibile in KB.

## Descrizione:

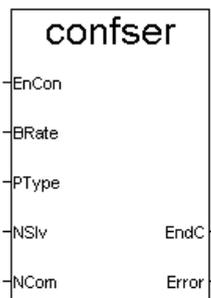
Questo blocco consente la lettura della memoria utilizzata per l'applicativo e la memoria ancora disponibile, sia per i dati dell'applicativo, sia per il codice dell'applicativo.

La dimensione indicata sia per la memoria utilizzata che per la memoria disponibile è espressa in Kbyte.

La somma della memoria utilizzata e di quella disponibile forniscono la dimensione totale ammessa.

Qualora l'applicativo dovesse essere superiore allo spazio ammesso contattare l'Ufficio Tecnico dell'Arteco SpA.

## ConfSer



## Argomenti:

EnCon	BOO	Sul fronte positivo avvia l'operazione di configurazione del protocollo ModBus e l'inizializzazione delle tre seriali disponibili per i protocolli del WorkBench, ModBus e del Modulo Passo-Passo. Questo ingresso deve rimanere attivo almeno fino a quando l'uscita EndC diventa attiva.
BRate	INT	Indica il valore desiderato per il BaudRate per il protocollo ModBus. I codici da inserire sono rispettivamente: 1 → BaudRate=57600 bit/sec 2 → BaudRate=38400 bit/sec 3 → BaudRate=19200 bit/sec 4 → BaudRate=9600 bit/sec

5 → BaudRate=4800 bit/sec

PType	INT	Indica la versione(*) ModBus desiderata. Le versioni attualmente a disposizione sono le seguenti: 1 → Versione "ModBus-ISaGRAF" 2 → Versione "ModBus 32Bit" 3 → Versione "ModBus Standard" (*) vedi documentazione sui protocolli disponibili.
NSlv	INT	Rappresenta l'identificatore dello slave (cioè della SU210/CND51) nella rete ModBus. I valori utilizzabili sono compresi tra 01 e F7 Hex.
NCom	INT	Permette di assegnare a ciascuno dei protocolli seriali disponibili (quello del WorkBench, il ModBus e quello del Modulo Passo-Passo) una delle porte seriali del PLC SU210/CND51. Si veda la tabella successiva per le configurazioni possibili ed i corrispondenti valori da attribuire a tale ingresso.
EndC	BOO	Il valore TRUE indica la fine dell'operazione di configurazione (se l'operazione è terminata correttamente).
Error	INT	Codice degli errori occorsi durante la fase di configurazione: 0 → Nessun Errore 1 → Errore nell'impostazione del BaudRate 2 → L'identificatore assegnato allo slave non è corretto. 3 → La configurazione selezionata non è disponibile. 4 → È stata selezionata una versione non esistente.

## Descrizione:

Il blocco *ConfSER* consente di configurare le tre porte seriali del PLC SU210/CND51.

Nel caso del PLC **SU210** le porte seriali COM1 e COM2 non sono direttamente presenti sul PLC, ma sull'espansione SU-COM-MOD. Al momento tale espansione **non consente di utilizzare congiuntamente le seriali COM1 e COM2**, ma solamente l'una in alternativa all'altra.

Nel caso di PLC CND51 questa limitazione non c'è.

Più nel dettaglio il blocco *ConfSER* consente di:

1. Assegnare a ciascuno dei protocolli seriali implementati (quello del WorkBench ISaGRAF, il ModBus e quello per il collegamento con il Modulo Passo-Passo) una delle tre seriali disponibili. Le configurazioni possibili sono elencate nella tabella seguente.

Configurazione	Seriale COM0	Seriale COM1	Seriale COM2
<b>1</b>	ModBus	WorkBench ISaGRAF	Inutilizzata
<b>2</b>	ModBus	Inutilizzata	WorkBench ISaGRAF

3	WorkBench ISaGRAF	Inutilizzata	ModBus
4	WorkBench ISaGRAF	ModBus	Inutilizzata
5	ModBus	Colleg.Seriale mod.passo	Inutilizzata
6	ModBus	Inutilizzata	Colleg.Seriale mod.passo
7	WorkBench ISaGRAF	Colleg.Seriale mod.passo	Inutilizzata
8	WorkBench ISaGRAF	Inutilizzata	Colleg.Seriale mod.passo
9*	ModBus	WorkBench ISaGRAF	Colleg.Seriale Mod. Passo-Passo
10*	ModBus	Colleg.Seriale Mod. Passo-Passo	WorkBench ISaGRAF
11*	WorkBench ISaGRAF	Colleg.Seriale Mod. Passo-Passo	ModBus
12*	WorkBench ISaGRAF	ModBus	Colleg.Seriale Mod. Passo-Passo
13*	Inutilizzata	ModBus	WorkBench ISaGRAF
14*	Inutilizzata	WorkBench ISaGRAF	ModBus

\* disponibile solo su alcune configurazioni hardware.

## 2. Impostare i parametri di configurazione del protocollo ModBus.

Tali parametri sono il BaudRate, l'identificatore dello "slave" e la versione del protocollo (Standard, "32Bit Data ModBus" o "ModBus-ISaGRAF").

Modificando la velocità di trasmissione, viene automaticamente modificato anche il TimeOut che definisce l'avvenuta ricezione di un messaggio ModBus.

Se non viene utilizzato il blocco ConfSER, viene mantenuta la seguente **configurazione di default** che si ha direttamente all'accensione:

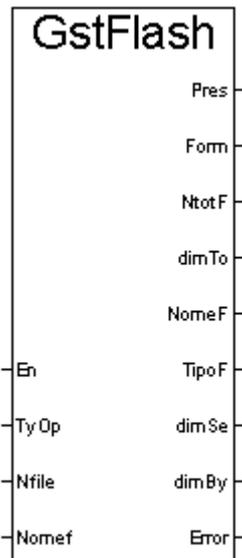
- Utilizzo della versione "ModBus-ISaGRAF" con una velocità di trasmissione di 9600 bit/sec e 01 Hex come identificatore dello slave.
- Utilizzo della porta seriale COM0 per il protocollo ModBus e della seriale COM2 per il WorkBench.  
**Importante.** Il protocollo per il modulo passo-passo non è direttamente disponibile all'accensione. Deve essere attivato (e configurata la seriale da esso utilizzata) tramite il blocco *ConfSER*.

Nel caso in cui il blocco *ConfSER* venga utilizzato per attivare il protocollo per il collegamento seriale con il modulo passo-passo, l'uscita *EndC* potrebbe impiegare alcune centinaia di millisec prima di diventare TRUE (per segnalare la fine dell'operazione di configurazione). Ciò è dovuto al fatto che, in questo caso, deve essere gestita anche una fase "di aggancio" tra il PLC ed il modulo passo, oltre alla inizializzazione della porta seriale. In questo caso l'uscita *EndC* diventa attiva solo a sincronizzazione avvenuta.

Ovviamente durante questo intervallo di tempo l'esecuzione del ciclo PLC continua, NON viene interrotta.

Assegnando le seriali a ciascuno dei protocolli disponibili bisogna tenere presente che il collegamento wireless per il ModBus è disponibile solo sulla COM1, mentre il collegamento via MODEM del WorkBench ISaGRAF è disponibile solamente sulla COM2.

## GstFlash



### Argomenti:

En	BOO	Sul fronte positivo avvia l'operazione da effettuare sulla flash card.
TyOp	INT	Permette di scegliere l'operazione da effettuare. I valori ammissibili sono: 0 → Visualizzazione dei file contenuti nella flash card. 1 → Formattazione della flash card. 2 → Cancellazione dalla flash card di un file di cui si indica il numero. 3 → Cancellazione dalla flash card di un file di cui si indica il nome.
Nfile	INT	Indica il numero del file su cui si deve eseguire l'operazione richiesta: nel caso di <b>visualizzazione</b> il numero 0, come pure il numero 1, indica che deve essere visualizzato il primo file presente nella flash card, il numero 2 indica che deve essere visualizzato il secondo file, il numero n indica che deve essere visualizzato l'n-esimo file presente sulla flash card. Nel caso di <b>cancellazione</b> di un file <b>dato il numero</b> , rappresenta il numero del file che si desidera cancellare dalla flash card. Nel caso di <b>formattazione</b> e di <b>cancellazione</b> di un file <b>dato il nome</b> questo parametro non è significativo.
Nomef	MSG	Nome del file. Questo parametro è significativo solo per l'operazione di cancellazione di un file dato il nome.

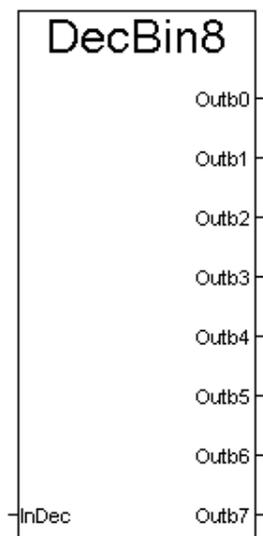
Pres	BOO	Il valore TRUE indica che la flash card è presente.
Form	BOO	Il valore TRUE indica che la flash card è formattata.
NtotF	INT	Indica il numero totale di file presenti sulla flash card.
DimTo	INT	Indica la dimensione totale utile della flash card (differisce dalla dimensione teorica a causa della presenza dei restricted sector).
NomeF	MSG	Nome del file su cui ho eseguito l'operazione. Questo parametro non ha significato nel caso in cui sulla flash non sia presente alcun file.
TipoF	INT	Indica il tipo del file visualizzato o cancellato. I valori ammissibili sono: 1 → Firmware. 2 → Applicativo. 3 → Variabili booleane. 4 → Variabili intere. 5 → Variabili timer. 6 → Variabili messaggio. 7 → Variabili virtuali.
DimSe	INT	Dimensione del file considerato (numero dei settori di flash occupati).
DimBy	INT	Dimensione del file considerato in byte
Error	INT	Indica se l'operazione si è svolta correttamente o se si è verificato un errore: 0 → Operazione terminata correttamente. 1 → Errore nella gestione della flash card. 2 → Il file richiesto non è presente (parametro errato). 3 → Flash card inconsistente (dati errati o non coerenti).

## Descrizione:

Il blocco Gstflash consente di eseguire alcune operazioni sulla flash card esterna. In particolare il blocco verifica se la flash card è presente, se è formattata ed in tal caso visualizza il numero totale dei file presenti e la dimensione utile. In aggiunta è possibile visualizzare il nome e il tipo dei file contenuti, inserendo nel parametro Nfile il numero del file. Inoltre è possibile formattare la flash card e cancellare un file di cui si conosce il numero (nell'ordine in cui compaiono sulla flash card) oppure il nome. Se l'operazione termina correttamente il parametro Error restituisce 0, in caso contrario si ha la segnalazione di un errore.

Per informazioni sugli errori fare riferimento anche all'Appendice D.

## DecBin8



### Argomenti:

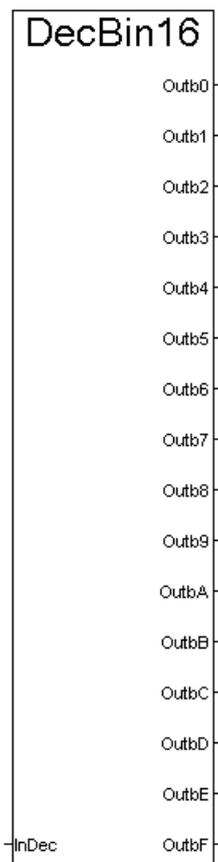
InDec	INT	Numero intero a 8 bit da convertire in 8 bit.
Outb0	BOO	Valore del Bit 0 del numero intero presente all'ingresso InDec. Bit meno significativo.
Outb1	BOO	Valore del Bit 1 del numero intero presente all'ingresso InDec.
Outb2	BOO	Valore del Bit 2 del numero intero presente all'ingresso InDec.
Outb3	BOO	Valore del Bit 3 del numero intero presente all'ingresso InDec.
Outb4	BOO	Valore del Bit 4 del numero intero presente all'ingresso InDec.
Outb5	BOO	Valore del Bit 5 del numero intero presente all'ingresso InDec.
Outb6	BOO	Valore del Bit 6 del numero intero presente all'ingresso InDec.
Outb7	BOO	Valore del Bit 7 del numero intero presente all'ingresso InDec. Bit più significativo.

### Descrizione:

Questo blocco consente di convertire una variabile intera considerandone gli 8 bit meno significativi, presente all'ingresso InDec, in 8 variabili booleane rappresentante ciascuna il relativo bit.

Se il bit è a "zero" il valore della rispettiva uscita è FALSE, viceversa se è "uno" l'uscita sarà TRUE. Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso negli 8 dati in uscita.

## DecBin16



### Argomenti:

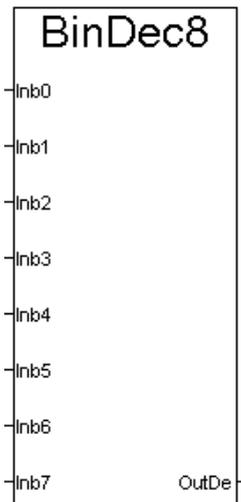
InDec	INT	Numero intero a 16 bit da convertire in 16 bit.
Outb0	BOO	Valore del Bit 0 del numero intero presente all'ingresso InDec. Bit meno significativo.
Outb1	BOO	Valore del Bit 1 del numero intero presente all'ingresso InDec.
Outb2	BOO	Valore del Bit 2 del numero intero presente all'ingresso InDec.
Outb3	BOO	Valore del Bit 3 del numero intero presente all'ingresso InDec.
Outb4	BOO	Valore del Bit 4 del numero intero presente all'ingresso InDec.
Outb5	BOO	Valore del Bit 5 del numero intero presente all'ingresso InDec.
Outb6	BOO	Valore del Bit 6 del numero intero presente all'ingresso InDec.
Outb7	BOO	Valore del Bit 7 del numero intero presente all'ingresso InDec.
Outb8	BOO	Valore del Bit 8 del numero intero presente all'ingresso InDec.
Outb9	BOO	Valore del Bit 9 del numero intero presente all'ingresso InDec.
OutbA	BOO	Valore del Bit 10 del numero intero presente all'ingresso InDec.
OutbB	BOO	Valore del Bit 11 del numero intero presente all'ingresso InDec.
OutbC	BOO	Valore del Bit 12 del numero intero presente all'ingresso InDec.
OutbD	BOO	Valore del Bit 13 del numero intero presente all'ingresso InDec.
OutbE	BOO	Valore del Bit 14 del numero intero presente all'ingresso InDec.
OutbF	BOO	Valore del Bit 15 del numero intero presente all'ingresso InDec. Bit più significativo.

## **Descrizione:**

Questo blocco consente di convertire una variabile intera considerandone i 16 bit meno significativi, presente all'ingresso InDec, in 16 variabili booleane rappresentante ciascuna il relativo bit.

Se il bit è a "zero" il valore della rispettiva uscita è FALSE, viceversa se è "uno" l'uscita sarà TRUE. Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso nei 16 dati in uscita.

## BinDec8



### Argomenti:

Inb0	BOO	Valore del Bit 0 del numero intero da presentare sull'uscita OutDe. Bit meno significativo.
Inb1	BOO	Valore del Bit 1 del numero intero da presentare sull'uscita OutDe.
Inb2	BOO	Valore del Bit 2 del numero intero da presentare sull'uscita OutDe.
Inb3	BOO	Valore del Bit 3 del numero intero da presentare sull'uscita OutDe.
Inb4	BOO	Valore del Bit 4 del numero intero da presentare sull'uscita OutDe.
Inb5	BOO	Valore del Bit 5 del numero intero da presentare sull'uscita OutDe.
Inb6	BOO	Valore del Bit 6 del numero intero da presentare sull'uscita OutDe.
Inb7	BOO	Valore del Bit 7 del numero intero da presentare sull'uscita OutDe. Bit più significativo.
OutDe	INT	Numero intero a 8 bit rappresentante la combinazione dei singoli bit presenti nei rispettivi ingressi booleani Inbnn.

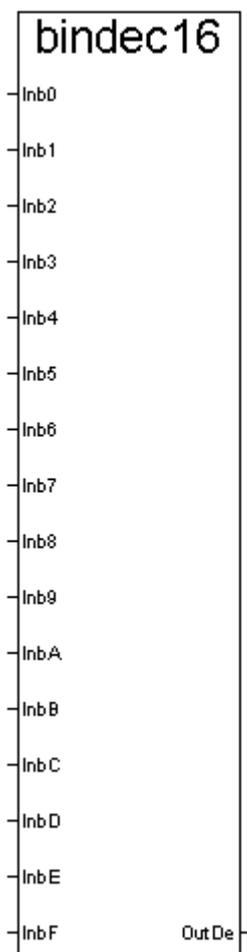
### Descrizione:

Questo blocco consente di convertire otto variabili booleane presenti agli ingressi da Inb0 a Inb7 in 1 variabile intera rappresentante la combinazione degli otto ingressi booleani ciascuno rappresentante un bit.

Se la variabile booleana è FALSE il bit sarà "zero", viceversa se è TRUE il bit sarà "uno".

Il blocco non prevede abilitazione. Ad ogni chiamata converte gli 8 dati in ingresso nell'unico dato in uscita.

## BinDec16



### Argomenti:

Inb0	BOO	Valore del Bit 0 del numero intero da presentare sull'uscita OutDe. Bit meno significativo.
Inb1	BOO	Valore del Bit 1 del numero intero da presentare sull'uscita OutDe.
Inb2	BOO	Valore del Bit 2 del numero intero da presentare sull'uscita OutDe.
Inb3	BOO	Valore del Bit 3 del numero intero da presentare sull'uscita OutDe.
Inb4	BOO	Valore del Bit 4 del numero intero da presentare sull'uscita OutDe.
Inb5	BOO	Valore del Bit 5 del numero intero da presentare sull'uscita OutDe.
Inb6	BOO	Valore del Bit 6 del numero intero da presentare sull'uscita OutDe.
Inb7	BOO	Valore del Bit 7 del numero intero da presentare sull'uscita OutDe.
Inb8	BOO	Valore del Bit 8 del numero intero da presentare sull'uscita OutDe.
Inb9	BOO	Valore del Bit 9 del numero intero da presentare sull'uscita OutDe.
InbA	BOO	Valore del Bit 10 del numero intero da presentare sull'uscita OutDe.
InbB	BOO	Valore del Bit 11 del numero intero da presentare sull'uscita OutDe.
InbC	BOO	Valore del Bit 12 del numero intero da presentare sull'uscita OutDe.
InbD	BOO	Valore del Bit 13 del numero intero da presentare sull'uscita OutDe.

InbE	BOO	Valore del Bit 14 del numero intero da presentare sull'uscita OutDe.
InbF	BOO	Valore del Bit 15 del numero intero da presentare sull'uscita OutDe. Bit più significativo.
OutDe	INT	Numero intero a 16 bit rappresentante la combinazione dei singoli bit presenti nei rispettivi ingressi booleani Inbnn.

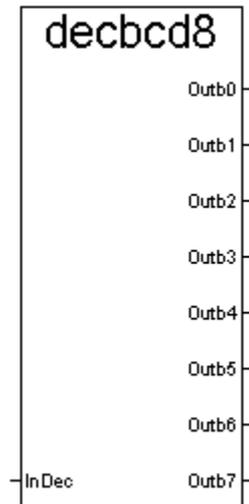
## Descrizione:

Questo blocco consente di convertire sedici variabili booleane presenti agli ingressi da Inb0 a InbF in una variabile intera rappresentante la combinazione dei sedici ingressi booleani ciascuno rappresentante un bit.

Se la variabile booleana è FALSE il bit sarà "zero" , viceversa se è TRUE il bit sarà "uno".

Il blocco non prevede abilitazione. Ad ogni chiamata converte i 16 dati in ingresso nell'unico dato in uscita.

## DecBCD8



### Argomenti:

InDec	INT	Numero intero decimale a 8 bit che verrà convertito in bit (Outbn) corrispondenti al formato BCD.
Outb0	BOO	Valore del Bit 0 in formato BCD del numero intero in ingresso InDec. Bit meno significativo.
Outb1	BOO	Valore del Bit 1 in formato BCD del numero intero in ingresso InDec.
Outb2	BOO	Valore del Bit 2 in formato BCD del numero intero in ingresso InDec.
Outb3	BOO	Valore del Bit 3 in formato BCD del numero intero in ingresso InDec.
Outb4	BOO	Valore del Bit 4 in formato BCD del numero intero in ingresso InDec.
Outb5	BOO	Valore del Bit 5 in formato BCD del numero intero in ingresso InDec.
Outb6	BOO	Valore del Bit 6 in formato BCD del numero intero in ingresso InDec.
Outb7	BOO	Valore del Bit 7 in formato BCD del numero intero in ingresso InDec. Bit più significativo.

### Descrizione:

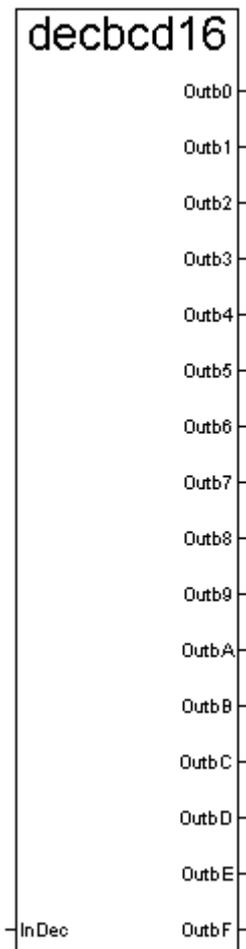
Questo blocco consente di convertire una variabile intera, presente in ingresso (InDec), in otto variabili booleane (da Outb0 a Outb7) che rappresentano il formato BCD del numero decimale in ingresso.

Il formato BCD prevede che ogni singola cifra del numero decimale da convertire sia rappresentata da 4 bit (nibble). Quindi un numero decimale di tre cifre è convertito in un numero binario di 12 bit (3 nibble). Es. 99 decimale => 10011001 binario. Da quanto appena descritto ogni nibble può variare da 0 (0000b) a 9 (1001b).

Numeri con la virgola non possono essere rappresentati. Il numero massimo rappresentabile è 99.

Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso negli otto bit in uscita.

## DecBCD16



### Argomenti:

InDec	INT	Numero intero decimale a 16 bit che verrà convertito in bit (Outbn) corrispondenti al formato BCD.
Outb0	BOO	Valore del Bit 0 in formato BCD del numero intero in ingresso InDec. Bit meno significativo.
Outb1	BOO	Valore del Bit 1 in formato BCD del numero intero in ingresso InDec.
Outb2	BOO	Valore del Bit 2 in formato BCD del numero intero in ingresso InDec.
Outb3	BOO	Valore del Bit 3 in formato BCD del numero intero in ingresso InDec.
Outb4	BOO	Valore del Bit 4 in formato BCD del numero intero in ingresso InDec.
Outb5	BOO	Valore del Bit 5 in formato BCD del numero intero in ingresso InDec.
Outb6	BOO	Valore del Bit 6 in formato BCD del numero intero in ingresso InDec.
Outb7	BOO	Valore del Bit 7 in formato BCD del numero intero in ingresso InDec.
Outb8	BOO	Valore del Bit 8 in formato BCD del numero intero in ingresso InDec.

Outb9	BOO	Valore del Bit 9 in formato BCD del numero intero in ingresso InDec.
OutbA	BOO	Valore del Bit 10 in formato BCD del numero intero in ingresso InDec.
OutbB	BOO	Valore del Bit 11 in formato BCD del numero intero in ingresso InDec.
OutbC	BOO	Valore del Bit 12 in formato BCD del numero intero in ingresso InDec.
OutbD	BOO	Valore del Bit 13 in formato BCD del numero intero in ingresso InDec.
OutbE	BOO	Valore del Bit 14 in formato BCD del numero intero in ingresso InDec.
OutbF	BOO	Valore del Bit 15 in formato BCD del numero intero in ingresso InDec. Bit più significativo.

## Descrizione:

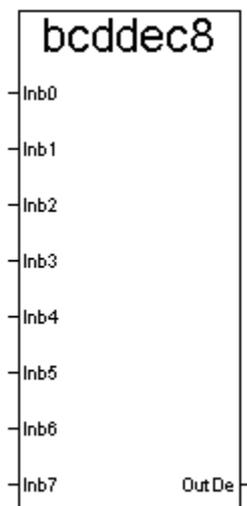
Questo blocco consente di convertire una variabile intera, presente in ingresso (InDec), in sedici variabili booleane (da Outb0 a OutbF) che rappresentano il formato BCD del numero decimale in ingresso.

Il formato BCD prevede che ogni singola cifra del numero decimale da convertire sia rappresentata da 4 bit (nibble). Quindi un numero decimale di tre cifre è convertito in un numero binario di 12 bit (3 nibble). Es. 99 decimale => 10011001 binario. Da quanto appena descritto ogni nibble può variare da 0 (0000b) a 9 (1001b).

Numeri con la virgola non possono essere rappresentati. Il numero massimo rappresentabile è' 9999.

Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso nei sedici bit in uscita.

## BCDDec8



### Argomenti:

Inb0	BOO	Bit 0 del numero intero BCD in ingresso da presentare sull'uscita OutDe. Bit meno significativo.
Inb1	BOO	Valore del Bit 1 del numero intero da presentare sull'uscita OutDe.
Inb2	BOO	Valore del Bit 2 del numero intero da presentare sull'uscita OutDe.
Inb3	BOO	Valore del Bit 3 del numero intero da presentare sull'uscita OutDe.
Inb4	BOO	Valore del Bit 4 del numero intero da presentare sull'uscita OutDe.
Inb5	BOO	Valore del Bit 5 del numero intero da presentare sull'uscita OutDe.
Inb6	BOO	Valore del Bit 6 del numero intero da presentare sull'uscita OutDe.
Inb7	BOO	Valore del Bit 7 del numero intero da presentare sull'uscita OutDe. Bit più significativo.
OutDe	INT	Numero intero a 8 bit rappresentante la combinazione dei singoli bit, presenti nei rispettivi ingressi booleani Inbnn, in formato BCD.

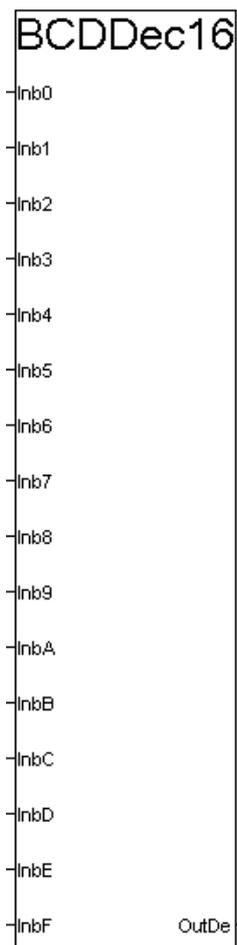
### Descrizione:

Questo blocco consente di convertire otto variabili booleane, rappresentanti un numero in formato BCD e presenti agli ingressi da Inb0 a Inb7, in una variabile intera.

Il formato BCD prevede che ogni singola cifra sia rappresentata da 4 bit (nibble). Quindi un numero decimale di tre cifre è convertito in un numero binario di 12 bit (3 nibble). Es. 99 decimale => 10011001 binario. Da quanto appena descritto ogni nibble può variare da 0 (0000b) a 9 (1001b).

Numeri con la virgola non possono essere rappresentati. Il numero massimo rappresentabile è 9999.

Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso nei sedici bit in uscita.

**BCDDec16****Argomenti:**

Inb0	BOO	Bit 0 del numero BCD in ingresso da presentare sull'uscita OutDe. Bit meno significativo.
Inb1	BOO	Valore del Bit 1 del numero intero BCD da presentare sull'uscita OutDe.
Inb2	BOO	Valore del Bit 2 del numero intero BCD da presentare sull'uscita OutDe.
Inb3	BOO	Valore del Bit 3 del numero intero BCD da presentare sull'uscita OutDe.
Inb4	BOO	Valore del Bit 4 del numero intero BCD da presentare sull'uscita OutDe.
Inb5	BOO	Valore del Bit 5 del numero intero BCD da presentare sull'uscita OutDe.
Inb6	BOO	Valore del Bit 6 del numero intero BCD da presentare sull'uscita OutDe.
Inb7	BOO	Valore del Bit 7 del numero intero BCD da presentare sull'uscita OutDe.

Inb8	BOO	Valore del Bit 8 del numero intero BCD da presentare sull'uscita OutDe.
Inb9	BOO	Valore del Bit 9 del numero intero BCD da presentare sull'uscita OutDe.
InbA	BOO	Valore del Bit 10 del numero intero BCD da presentare sull'uscita OutDe.
InbB	BOO	Valore del Bit 11 del numero intero BCD da presentare sull'uscita OutDe.
InbC	BOO	Valore del Bit 12 del numero intero BCD da presentare sull'uscita OutDe.
InbD	BOO	Valore del Bit 13 del numero intero BCD da presentare sull'uscita OutDe.
InbE	BOO	Valore del Bit 14 del numero intero BCD da presentare sull'uscita OutDe.
InbF	BOO	Valore del Bit 15 del numero intero BCD da presentare sull'uscita OutDe. Bit più significativo.
OutDe	INT	Numero intero a 16 bit in formato BCD rappresentante la combinazione dei singoli bit, presenti nei rispettivi ingressi booleani Inbnn, in formato BCD.

### Descrizione:

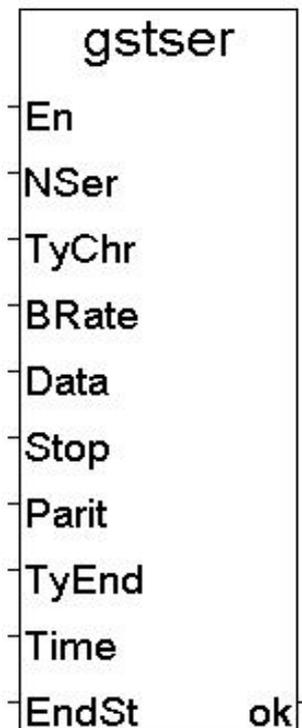
Questo blocco consente di convertire sedici variabili booleane, rappresentanti un numero in formato BCD e presenti agli ingressi da Inb0 a InbF, in una variabile intera.

Il formato BCD prevede che ogni singola cifra sia rappresentata da 4 bit (nibble). Quindi un numero decimale di tre cifre è convertito in un numero binario di 12 bit (3 nibble). Es. 99 decimale => 10011001 binario. Da quanto appena descritto ogni nibble può variare da 0 (0000b) a 9 (1001b).

Numeri con la virgola non possono essere rappresentati. Il numero massimo rappresentabile è 9999.

Il blocco non prevede abilitazione. Ad ogni chiamata converte il dato in ingresso nei sedici bit in uscita.

## GstSer



### Argomenti:

En	BOO	Sul fronte positivo abilita o disabilita la gestione della seriale da programma applicativo.
NSer	INT	Numero della seriale da utilizzare.
TyChr	INT	0 = i caratteri sono gestiti in codice binario 1 = i caratteri sono gestiti in codice ASCII
Brate	INT	Indica il valore desiderato per il BaudRate della seriale. I codici da inserire sono rispettivamente: 0 → Disattiva la gestione seriale da applicativo 1 → BaudRate=57600 bit/sec (non disponibile) 2 → BaudRate=38400 bit/sec 3 → BaudRate=19200 bit/sec 4 → BaudRate=9600 bit/sec 5 → BaudRate=4800 bit/sec (non disponibile)
Data	INT	Numero dei bit di dati (7 o 8)
Stop	INT	Numero dei bit di stop (1 o 2)
Parit	INT	Tipo di parità 0. Nessuna parità 1. Parità pari 2. Parità dispari

TyEnd	INT	Tipo di terminazione per i messaggi in ricezione 0 => Nessuna 1 => Terminazione a tempo 2 => Terminazione a stringa
Time	INT	Usato nel caso di terminazione a tempo indica il tempo in ms che deve intercorrere dall'ultimo bit ricevuto per considerare il messaggio terminato. Tale valore può essere arrotondato ad 1 ms in eccesso o in difetto dal controllo.
EndSt	MSG	Usato nel caso di terminazione a stringa indica una stringa di uno o più caratteri che in ricezione indica che il messaggio è terminato.
Ok	INT	Se l'attivazione o la disattivazione della gestione seriale si è conclusa senza errori viene posto a 1.

## Descrizione:

Questo blocco, insieme a ReadSer e WriteSer descritti di seguito, permette all'utente la gestione di una linea seriale mediante un applicativo che gestisce un protocollo personalizzato di invio e ricezione di stringhe con caratteristiche di comunicazione selezionate dall'utente.

Il controllo CND51 (o SU21X) non gestisce il protocollo, ma si limita ad inviare stringhe preparate dall'utente mediante applicativo ed a restituire i messaggi ricevuti su seriale e terminati nel modo richiesto dall'utente.

La gestione della seriale viene attivata sul fronte positivo dell'ingresso En.

Per disattivarla occorre porre Brate = 0 e generare un fronte positivo sull'ingresso En.

Il numero della seriale da utilizzare deve essere compatibile con gli altri protocolli gestiti dalla scheda. In particolare non è possibile attivare la gestione della seriale da applicativo per una seriale configurata per un altro protocollo (es: per il WorkBench di ISaGRAF). Per maggiori informazioni sulle possibili configurazioni delle seriali si rimanda al manuale delle personalizzazioni - Blocco ConfSer.

L'ingresso TyChr permette di selezionare se i caratteri da inviare o ricevuti devono essere codificati in binario o in codice ASCII.

Gli ingressi Brate, Data, Stop e Parit indicano i parametri di configurazione della seriale.

**Attenzione:** nel caso si sia selezionata la COM0, non è possibile impostare 7 bit di dati e nessuna parità. Questa è l'unica combinazione che non viene gestita. Il blocco segnala errore.

L'ingresso TyEnd permette di selezionare il tipo di terminazione dei messaggi in fase di ricezione.

Nel caso di terminazione a stringa, il messaggio viene considerato terminato quando vengono ricevuti caratteri della stringa di terminazione impostata in EndSt.

Nel caso di terminazione a tempo, il messaggio viene considerato terminato per un tempo fissato dal parametro Time.

Nel caso di nessuna terminazione, è l'utente che segnala la terminazione del messaggio agendo sul blocco ReadSer (v. descrizione Blocco ReadSer).

## ReadSer



### Argomenti:

En	BOO	Sul livello positivo abilita la visualizzazione dei messaggi ricevuti dalla seriale. Tali messaggi sono comunque registrati anche se En è FALSE.
NSer	INT	Numero della seriale da cui leggere i messaggi. Deve essere stata inizializzata con il blocco GstSer.
res	BOO	Resetta l'indicazione di messaggio terminato e nel caso di protocollo senza terminazione permette di azzerare i buffer di uscita dei messaggi (v. descrizione dettagliata).
Ok	BOO	Indica la terminazione del messaggio. Nel caso di protocollo senza terminatore viene posto a TRUE alla ricezione del primo carattere dopo il reset.
Nchr	INT	Numero dei caratteri ricevuti (se la codifica è ASCII) o dei byte ricevuti (se la codifica è binaria)
Mess	MSG	Messaggio ricevuto

### Descrizione:

Questo blocco visualizza i messaggi ricevuti sulla seriale. La seriale deve essere stata inizializzata con il blocco GstSer.

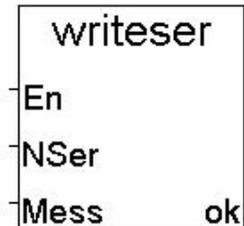
L'ingresso di abilitazione En abilita la visualizzazione dei messaggi ricevuti. Se En è FALSE, la scheda continua a registrare internamente i caratteri ricevuti ma non avviene l'aggiornamento dei buffer di uscita.

Vediamo in dettaglio la fase di ricezione messaggi.

- ◆ **Terminazione a stringa (TyEnd=2).** Quando sulla seriale sono ricevuti i caratteri che definiscono la fine del messaggio, l'uscita OK viene posta a TRUE, Nchr indica il numero di caratteri ricevuti (senza contare quelli di terminazione) e il messaggio (senza la terminazione) viene visualizzato nell'uscita Mess. Alla ricezione di una nuova terminazione, i precedenti valori vengono sovrascritti con i nuovi. Per resettare l'uscita OK occorre porre RES a TRUE.  
Nel caso di codifica binaria Nchr indica il numero di byte ricevuti e nell'uscita Mess viene visualizzata la codifica esadecimale dei byte, cioè per ogni byte si ottengono 2 caratteri.
  
- ◆ **Terminazione a tempo (TyEnd=1).** Quando sulla seriale è passato un tempo pari a quello impostato nel blocco GstSer (par.Time) dall'ultimo carattere ricevuto, il messaggio è considerato terminato. L'uscita OK viene posta a TRUE, Nchr indica il numero di caratteri ricevuti e il messaggio viene visualizzato nell'uscita Mess. Al termine del messaggio successivo i precedenti valori vengono sovrascritti con i nuovi. Per resettare l'uscita OK occorre porre RES a TRUE.  
Nel caso di codifica binaria Nchr indica il numero di byte ricevuti e nell'uscita Mess viene visualizzata la codifica esadecimale dei byte, cioè per ogni byte si ottengono 2 caratteri.
  
- ◆ **Protocollo senza terminazione (TyEnd=0).** Alla ricezione del primo carattere l'uscita OK viene posta a TRUE, tutti i caratteri ricevuti in seguito vengono accodati e visualizzati nell'uscita Mess e Nchr ne indica il numero. Il messaggio è considerato terminato quando si attiva l'ingresso res, in corrispondenza viene anche resettata l'uscita OK. Alla ricezione del primo carattere successivo al res il buffer viene svuotato.  
Nel caso di codifica binaria Nchr indica il numero di byte ricevuti e nell'uscita Mess viene visualizzata la codifica esadecimale dei byte, cioè per ogni byte si ottengono 2 caratteri.

Se i caratteri devono essere codificati in binario (TyChr=0), il messaggio sarà costituito da una stringa che rappresenta il valore esadecimale dei dati ricevuti.  
(es: Se si riceve un byte "0100 0001" , viene visualizzato "41" che è la codifica esadecimale).

## WriteSer



### Argomenti:

En	BOO	Sul fronte positivo trasferisce il messaggio sulla linea seriale.
NSer	INT	Numero della seriale da cui leggere i messaggi. Deve essere stata inizializzata con il blocco GstSer.
Mess	MSG	Messaggio da inviare.
Ok	INT	Indica che il messaggio è stato inviato.

### Descrizione:

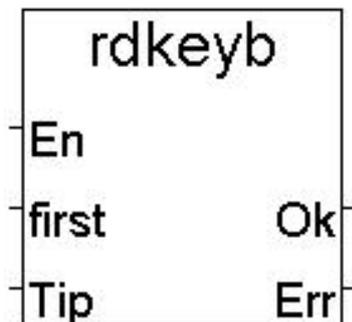
Questo blocco permette di inviare sulla seriale una stringa di caratteri o di byte. La seriale deve essere stata inizializzata con il blocco GstSer.

Nel caso di codifica binaria la stringa deve rappresentare la codifica esadecimale dei byte che sono da spedire; poiché per ogni byte sono da impostare 2 caratteri, la stringa deve essere costituita da un numero pari di elementi.

(es: se si desidera inviare il byte "0100 0001", si deve impostare la stringa "41" che rappresenta la codifica esadecimale del byte)

Il controllo non esegue alcuna verifica sui messaggi inviati, ogni azione o elaborazione è demandata al programma applicativo. Anche nel caso di Protocollo con terminazione a stringa, qualora si desideri che ogni messaggio inviato abbia la corretta terminazione, sarà cura dell'utente accodare al messaggio gli opportuni caratteri di terminazione.

## RdKeyb



### Argomenti:

En	BOO	Sul fronte positivo attiva la lettura dei tasti dalla keyboard e attua l'accensione dei led.
first	INT	Indirizzo della prima variabile ISaGRAF contenente lo stato dei tasti e dei led della keyboard
Tip	INT	Non utilizzato
Ok	BOO	Tastiera attivata
Err	INT	Segnalazione di errore: 0 -> Nessun errore 1 -> Parametri errati 2 -> Variabili non dichiarate 3 -> Variabili non booleane

### Descrizione:

Questo blocco permette la gestione della keyboard KD per SU210, SU110 e SU112.

### Scheda SU210

Nell'applicativo devono essere presenti 46 variabili booleane ISaGRAF con indirizzi consecutivi, che vengono utilizzate per tale gestione e l'ingresso first del blocco deve contenere l'indirizzo della prima.

La keyboard prevede 35 tasti e 11 led. Le prime 35 variabili rappresentano lo stato dei tasti: alla pressione di un tasto, la corrispondente variabile assume il valore true.

Le successive 11 variabili rappresentano il comando per i led: ponendo a true una variabile, si attiva il led corrispondente.

Di seguito è riportata la mappa di corrispondenza variabile/tasto (o led) per la SU210.

var. 1) - SW1	Tasto Utente 1
var. 2) - SW2	F1
var. 3) - SW3	AUT
var. 4) - SW4	Tasto 1
var. 5) - SW5	Tasto 2
var. 6) - SW6	Tasto 3
var. 7) - SW7	Tasto 4
var. 8) - SW8	Tasto 5
var. 9) - SW9	Tasto utente 2
var. 10) - SW10	F2
var. 11) - SW11	Tasto +
var. 12) - SW12	Tasto 6
var. 13) - SW13	Tasto 7
var. 14) - SW14	Tasto 8
var. 15) - SW15	Tasto 9
var. 16) - SW16	Tasto 0
var. 17) - SW17	Tasto Utente 3
var. 18) - SW18	F3
var. 19) - SW19	Freccia giu
var. 20) - SW20	MAN
var. 21) - SW21	DEL
var. 22) - SW22	Freccia su
var. 23) - SW23	BSP
var. 24) - SW24	START
var. 25) - SW25	Tasto Utente 4
var. 26) - SW26	F4
var. 27) - SW27	Tasto -
var. 28) - SW28	PROG
var. 29) - SW29	Freccia Sx
var. 30) - SW30	SHIFT
var. 31) - SW31	Freccia Dx
var. 32) - SW32	STOP
var. 33) - SW33	Tasto Utente 5
var. 34) - SW34	F5
var. 35) - SW35	ENTER
var. 36) - DL1	led WD
var. 37) - DL2	led RUN
var. 38) - DL3	led SIM
var. 39) - DL4	led EMG
var. 40) - DL5	led LOW BATT
var. 41) - DL6	led CAN
var. 42) - DL7	led User 1
var. 43) - DL8	led User 2
var. 44) - DL9	led User 3
var. 45) - DL10	led User 4
var. 46) - DL11	led User 5

## Scheda SU110

Nell'applicativo devono essere presenti 37 variabili booleane ISaGRAF con indirizzi consecutivi, che vengono utilizzate per tale gestione e l'ingresso first del blocco deve contenere l'indirizzo della prima.

La keyboard prevede 37 tasti, non ci sono led. Le 37 variabili rappresentano lo stato dei tasti: alla pressione di un tasto, la corrispondente variabile assume il valore true.

## Scheda SU112

Nell'applicativo devono essere presenti 48 variabili booleane ISaGRAF con indirizzi consecutivi, che vengono utilizzate per tale gestione e l'ingresso first del blocco deve contenere l'indirizzo della prima.

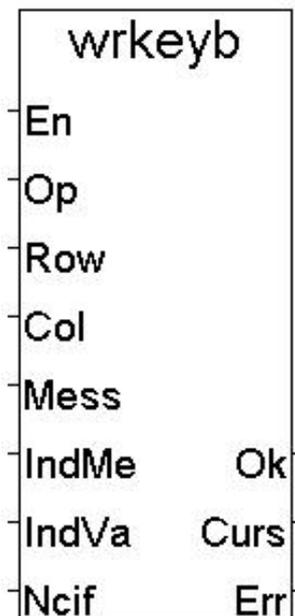
La keyboard prevede 40 tasti e 8 led. Le prime 40 variabili rappresentano lo stato dei tasti: alla pressione di un tasto, la corrispondente variabile assume il valore true.

Le successive 8 variabili rappresentano il comando per i led: ponendo a true una variabile, si attiva il led corrispondente.

Di seguito è riportata la mappa di corrispondenza variabile/tasto (o led) per SU110 e SU112:

var. 1) - SW1	Punto
var. 2) - SW2	Tasto 0
var. 3) - SW3	Tasto -
var. 4) - SW4	STOP
var. 5) - SW5	AUT
var. 6) - SW6	ENTER
var. 7) - SW7	Tasto utente g (solo per SU112)
var. 8) - SW8	Tasto utente h (solo per SU112)
var. 9) - SW9	Tasto 7
var. 10) - SW10	Tasto 8
var. 11) - SW11	Tasto 9
var. 12) - SW12	START
var. 13) - SW13	Freccia giu
var. 14) - SW14	ESC
var. 15) - SW15	Tasto utente e (solo per SU112)
var. 16) - SW16	Tasto utente f (solo per SU112)
var. 17) - SW17	Tasto 4
var. 18) - SW18	Tasto 5
var. 19) - SW19	Tasto 6
var. 20) - SW20	Freccia Sx
var. 21) - SW21	SHIFT
var. 22) - SW22	Freccia Dx
var. 23) - SW23	Tasto utente c (solo per SU112)
var. 24) - SW24	Tasto utente d (solo per SU112)

var. 25) - SW25	Tasto 1	
var. 26) - SW26	Tasto 2	
var. 27) - SW27	Tasto 3	
var. 28) - SW28	DEL	
var. 29) - SW29	Freccia su	
var. 30) - SW30	PROG	
var. 31) - SW31	Tasto utente a	(solo per SU112)
var. 32) - SW32	Tasto utente b	(solo per SU112)
var. 33) - SW33	F1	
var. 34) - SW34	F2	
var. 35) - SW35	F3	
var. 36) - SW36	F4	
var. 37) - SW37	F5	
var. 38) - SW38	HELP	(solo per SU112)
var. 39) - SW39	F6	(solo per SU112)
var. 40) - SW40	F7	(solo per SU112)
var. 41) - DL1	led User	(solo per SU112)
var. 42) - DL2	led User	(solo per SU112)
var. 43) - DL3	led User	(solo per SU112)
var. 44) - DL4	led User	(solo per SU112)
var. 45) - DL5	led User	(solo per SU112)
var. 46) - DL6	led User	(solo per SU112)
var. 47) - DL7	led User	(solo per SU112)
var. 48) - DL8	led User	(solo per SU112)

**WrKeyb****Argomenti:**

En	BOO	Sul fronte positivo attiva l'operazione richiesta relativa al display della tastiera.
Row	INT	Numero della riga (per SU210 vale 0,1 o 2, per SU110 vale 0,1,...,4 e per SU112 vale 0,1,...,8). Se vale zero viene considerata la riga corrente.
Col	INT	Numero della colonna (per SU210 0,1,2,...40, per SU110 0,1,2,...20, per SU112 0,1,.....,21). Se vale zero viene considerata la colonna corrente
Mess	MSG	Stringa da scrivere sul display
IndMes	INT	Indirizzo della variabile (ISaGRAF o virtuale) contenente la stringa di caratteri da stampare, o il pattern del carattere da definire, o il pattern della colonna grafica da stampare
IndVar	INT	Indirizzo della variabile ISaGRAF che memorizza il valore letto
Ncif	INT	Numero massimo caratteri da inserire

Op	INT	<p>Operazione richiesta:</p> <p>0 -&gt; Nessuna operazione</p> <p>1 -&gt; Scrive una stringa sul display prelevandola dall'ingresso MESS</p> <p>2 -&gt; Scrive una stringa sul display prelevandola da una variabile di cui si indica l'indirizzo</p> <p>3 -&gt; Cancella il display</p> <p>4 -&gt; Cancella una riga del display</p> <p>5 -&gt; Visualizza il cursore sul display e attiva il blink</p> <p>6 -&gt; Visualizza il cursore sul display e disattiva il blink</p> <p>7 -&gt; Nasconde il cursore sul display e attiva il blink</p> <p>8 -&gt; Nasconde il cursore sul display e disattiva il blink</p> <p>9 -&gt; Posiziona il cursore nella posizione richiesta del display</p> <p>10 -&gt; Stampa una stringa prendendola da Mess e attende l'immissione da tastiera di un valore da assegnare ad una variabile</p> <p>11 -&gt; Stampa una stringa prendendola dall'indirizzo IndMes e attende l'immissione da tastiera di un valore da assegnare ad una variabile</p> <p>12 -&gt; Shifta il cursore a destra di tante posizioni quante sono indicate in Ncif</p> <p>13 -&gt; Shifta il cursore a sinistra di tante posizioni quante sono indicate in Ncif</p> <p>14 -&gt; Inserisce tanti caratteri bianchi quanti indicati in Ncif</p> <p>15 -&gt; Termina lo stato di introduzione valori da tastiera, senza assegnare alcun valore alla variabile</p> <p>16 -&gt; Definisce un carattere utente prelevando il pattern dall'ingresso MESS</p> <p>17 -&gt; Definisce un carattere utente prelevando il pattern da una variabile di cui si indica l'indirizzo</p> <p>18 -&gt; Visualizza su display il set di caratteri</p> <p>19 -&gt; Scrive su display le singole colonne grafiche; il pattern è prelevato direttamente dall'ingresso</p> <p>20 -&gt; Scrive su display le singole colonne grafiche; il pattern è prelevato da variabile isagraf tipo messaggio di cui si indica l'indirizzo</p> <p>21 -&gt; Scrive su display le singole colonne grafiche; il pattern è prelevato da una o più variabili isagraf di tipo analogico, di cui si indica l'indirizzo della prima</p> <p>22 -&gt; Scrive su display le singole colonne grafiche; il pattern è ottenuto da una o più variabili virtuali di cui si indica l'indirizzo della prima</p>
----	-----	--

Ok	BOO	TRUE = operazione terminata correttamente
Curs	INT	Posizione del cursore (per SU210 e SU110 da 1 a 80, per SU112 con lcd grafico da 1 a 168)
Err	INT	Segnalazione di errore: 0 -> Nessun errore 1 -> Parametri errati 2 -> Stringa troppo lunga 3 -> La variabile indicata è di un tipo diverso da quello richiesto 4 -> Errore nella gestione del display 5 -> La variabile indicata non esiste 6 -> Ncif incompatibile con la posizione del cursore 7 -> Operazione non possibile perché il display è in modalità inserimento dati 8 -> Operazione non prevista

### Descrizione:

Questo blocco permette di gestire il display della tastiera KD da applicativo ISaGRAF.

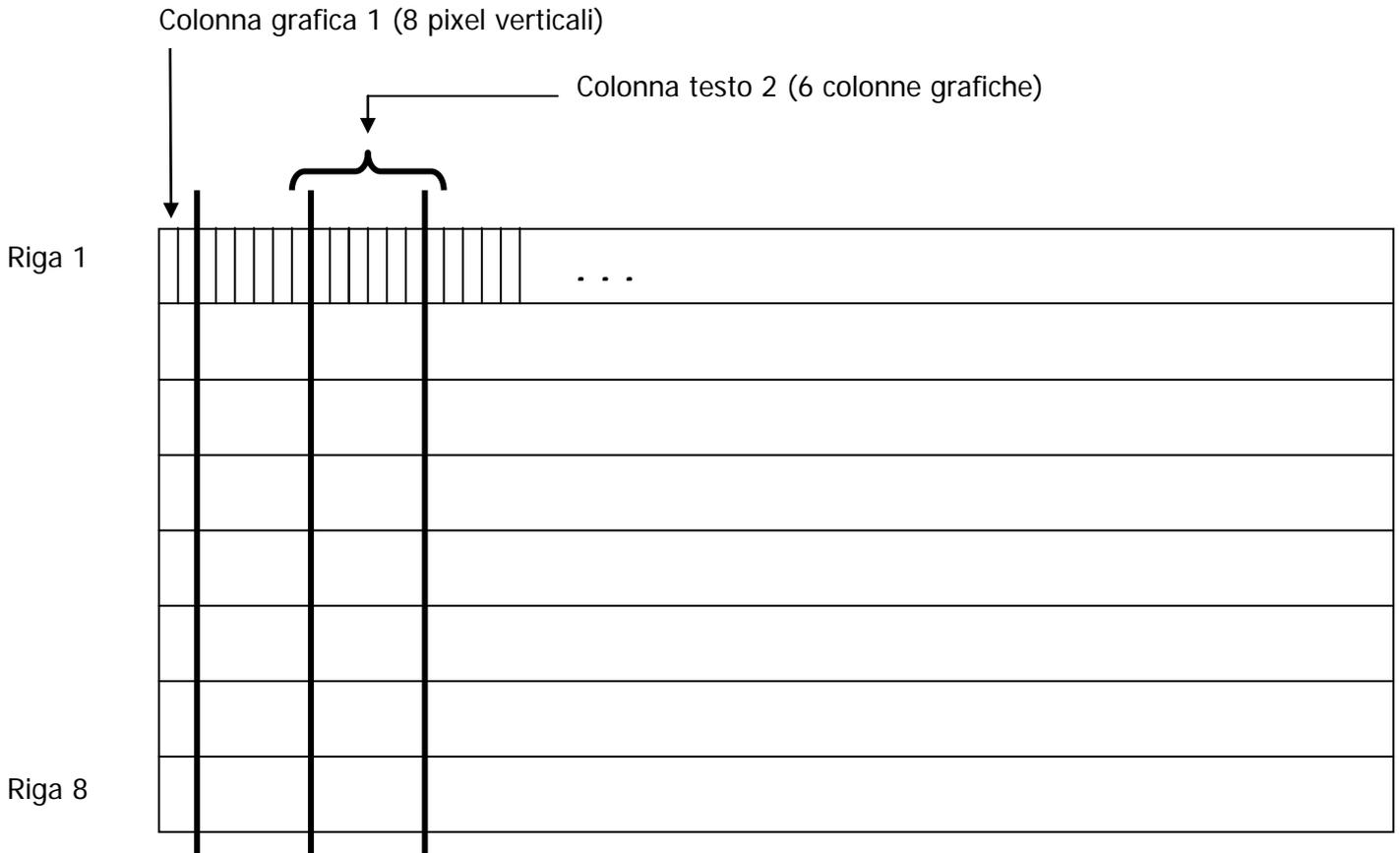
Per la scheda SU210: il display è suddiviso in 2 righe di 40 colonne (2x40)

Per la scheda SU110: il display è suddiviso in 4 righe di 20 colonne (4x20)

Per la scheda SU112: il display può essere di tipo alfanumerico (in questo caso è lo stesso della scheda SU110) oppure di tipo grafico (64x128 pixel).

Quest'ultimo è formato da un insieme di 8 righe di 128 colonne (che chiameremo **colonne grafiche**, ognuna formata da 8 pixel verticali). Il display di tipo grafico ha anche la gestione di tipo testo: in tal caso, il carattere è dato da un insieme di 6 colonne grafiche adiacenti (ottenendo un display 8x21, in cui 21 sono quelle che chiameremo **colonne testo**); si noti che  $128/6=21$  con resto 2, quindi per ogni riga si rende necessario saltare le prime due colonne grafiche. (vedi figura seguente)

Infine, per il display di tipo grafico è possibile la definizione da utente dei caratteri aventi codice identificativo compreso fra 221 e 255; chiameremo **codice ASCII** del carattere, l' identificativo (che va da 0 a 255) del carattere in questione, senza porci il problema della standardizzazione del pattern di tutti i caratteri.



L'ingresso Op permette di scegliere l'operazione da effettuare:

- 0 - Nessuna operazione
- 1 - Scrive nel display la stringa contenuta nell'ingresso Mess. La stringa viene scritta a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna). Se tali ingressi valgono zero viene considerata la riga o la colonna corrente.
- 2 - Scrive nel display la stringa contenuta nella variabile di cui è indicato l'indirizzo nell'ingresso Ind. La stringa viene scritta a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna). Se tali ingressi valgono zero viene considerata la riga o la colonna corrente.
- 3 - Cancella l'intero display e riposiziona il cursore all'inizio
- 4 - Cancella la riga indicata da Row (quella corrente se row=0) e posiziona il cursore all'inizio di tale riga
- 5 - Visualizza il cursore e attiva il blink (solo per i displays tipo testo)
- 6 - Visualizza il cursore e disattiva il blink (solo per i displays tipo testo)

- 7 - Nasconde il cursore e attiva il blink (solo per i displays tipo testo)
- 8 - Nasconde il cursore e disattiva il blink (solo per i displays tipo testo)
- 9 - Posiziona il cursore alla riga e colonna richieste; se valgono zero vengono considerate la riga o la colonna correnti)
- 10 - Stampa una stringa prelevandola dall'ingresso Mess e poi attende l'immissione da tastiera di un valore numerico da assegnare alla variabile ad indirizzo indicato nell'ingresso IndVar
- 11 - Stampa una stringa prelevandola dalla variabile di indirizzo IndMes e poi attende l'immissione da tastiera di un valore numerico da assegnare alla variabile ad indirizzo indicato nell'ingresso IndVar
- 12 - Shifta a destra il cursore di tante posizioni quante ne indica l'ingresso Ncif
- 13 - Shifta a sinistra il cursore di tante posizioni quante ne indica l'ingresso Ncif
- 14 - Inserisce tanti caratteri bianchi quanti ne indica l'ingresso Ncif
- 15 - Esce dalla modalità di immissione valori da tastiera senza assegnare alcun valore alla variabile
- 16 - Definisce la forma di un carattere tramite una stringa esadecimale fornita direttamente nell'ingresso Mess. In questa modalità gli ingressi Row ed Ncif non sono considerati, mentre Col contiene il codice ASCII del carattere da definire; sono definibili da utente i soli caratteri con codice ASCII compreso fra 221 e 255 (solo per display grafico)
- 17 - Definisce la forma di un carattere tramite una stringa esadecimale fornita in una variabile Isagraf®, di cui se ne indica l'indirizzo nell'ingresso IndVar. In questa modalità gli ingressi Row ed Ncif non sono considerati, mentre Col contiene il codice ASCII del carattere da definire; sono definibili da utente i soli caratteri con codice ASCII compreso fra 221 e 255 (solo per display grafico)
- 18 - Scrive su display il set dei caratteri disponibili (nel display grafico sono compresi anche i caratteri definiti da utente). Siccome il set completo di caratteri è costituito da 256 elementi, non è possibile visualizzarlo per intero, e si rende necessario dividere il set in pagine; la particolare pagina sarà selezionata fornendo il corrispettivo valore nell'ingresso Col (la prima pagina è ottenuta fornendo 1, mentre l'ultima dipende dal tipo di display); se il valore di Col è superiore al valore dell'ultima pagina, viene fornito il codice di errore 1 (parametro errato).  
I valori di pagina validi sono:  
per la scheda SU210 -> da 1 a 4  
per la scheda SU110 e SU112 con display alfanumerico -> da 1 a 4  
per la scheda SU112 con display grafico -> da 1 a 3  
In questa modalità gli ingressi Row ed Ncif non sono considerati.

- 19 - Scrive su display una serie di colonne grafiche consecutive, la cui definizione è fornita direttamente nell'ingresso Mess tramite stringa esadecimale; Il display è scritto a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna): vedi esempi più avanti. Se tali ingressi valgono zero, si considerano la riga e la colonna corrente. Ncif contiene il numero di colonne grafiche che vengono scritte, e deve corrispondere alla metà della lunghezza della stringa fornita a Mess. (solo per display grafico)
- 20 - Scrive su display una serie di colonne grafiche consecutive, la cui definizione è fornita dalla stringa esadecimale contenuta in una variabile Isagraf® di tipo messaggio, di cui se ne indica l'indirizzo nell'ingresso IndVar. Il display è scritto a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna) : vedi esempi più avanti. Se tali ingressi valgono zero, si considerano la riga e la colonna corrente. Ncif contiene il numero di colonne grafiche che sono scritte, e deve corrispondere alla metà della lunghezza della stringa fornita a Mess. (solo per display grafico)
- 21 - Scrive su display una serie di colonne grafiche consecutive, la cui definizione è prelevata da una o più variabili Isagraf® di tipo analogico, di cui si indica l'indirizzo della prima nell'ingresso IndVar; ogni variabile contiene la definizione di 4 colonne, ed Ncif ne determina il numero, rappresentando il numero di colonne da stampare a video. Il display è scritto a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna) : vedi esempi più avanti. Se tali ingressi valgono zero, si considerano la riga e la colonna corrente. (solo per display grafico)
- 22 - Scrive su display una serie di colonne grafiche consecutive, la cui definizione è prelevata da una o più variabili Virtuali, di cui si indica l'indirizzo della prima nell'ingresso IndVar. Ogni variabile contiene la definizione di 4 colonne; Ncif, rappresentando il numero di colonne da stampare a video, determina il numero delle variabili utilizzate. Il display è scritto a partire dalla posizione indicata dagli ingressi Row e Col (rispettivamente riga e colonna) : vedi esempi più avanti. Se tali ingressi valgono zero, si considerano la riga e la colonna corrente. (solo per display grafico)

Le operazioni 1 e 2 consentono la scrittura di un carattere tramite il suo codice ASCII a 3 cifre in base dieci. Prima di utilizzare tale codice, si deve immettere una '\ (barra inversa) nella stringa. Per ottenere la scrittura della barra inversa, occorre immettere nella stringa '\\.

Esempio: La stringa 'abc\033ABC\\abc' genera su display 'abc!ABC\abc' essendo 033 il codice ASCII del carattere punto esclamativo.

NOTA:

Vari display non seguono esattamente la codifica ASCII standard per tutti i caratteri, quindi è normale che il display alfanumerico stampi un carattere diverso dalla barra inversa; sono comunque garantiti sempre i numeri e le lettere sia maiuscole che minuscole.

Le operazioni 10 e 11 pongono il display nella modalità di inserimento valori. In questa modalità il cursore lampeggia e l'utente può inserire da tastiera un valore numerico di un numero massimo di caratteri indicato dall'ingresso Ncif.

Se Ncif è tale per cui il valore oltrepassa la fine della riga e proseguirebbe nell'altra, viene generato errore.

Sono accettati come valori, i numeri da 0 a 9 e il segno meno (-) per indicare un numero negativo. Per cancellare l'ultima cifra immessa nella SU210 premere BSP (Backspace), nella SU110 e SU112 premere DEL.

Per confermare il valore immesso premere ENTER. Per uscire dalla modalità di inserimento valori senza assegnare alcun valore nella SU210 premere DEL, nella SU110 e SU112 premere ESC, oppure inviare il comando 15.

Le operazioni 16, 17, 19 e 20 richiedono la definizione del carattere o della conca grafica a mezzo di stringa esadecimale, in cui il primo valore rappresenta la parte bassa della prima colonna, il secondo la parte alta della prima colonna, il terzo la parte bassa della seconda colonna... e via di seguito.

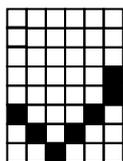
Ogni carattere è definito da 6 colonne di 8 punti e ogni colonna da due cifre esadecimali, per un totale di 12 cifre esadecimali per carattere; con le operazioni 16 e 17, l'immissione di stringhe di lunghezza diversa da 12 è segnalata come errore, mentre le operazioni 19 e 20 ammettono stringhe di lunghezza arbitraria, purché pari, volendo ottenere la stampa di più righe.

Le operazioni 21 e 22 definiscono un numero arbitrario di righe, utilizzando una variabile ogni 4 colonne. La prima colonna grafica è descritta dal byte più significativo (8 bit più alti) della variabile a 32 bit, e ogni byte rappresenta il codice binario della colonna tradotta in decimale.



**Esempio1:** si supponga di voler scrivere la colonna a lato (con i pixel neri si intendono i pixel illuminati) :

Avere una sola colonna implica Ncif=1, e quindi basterà una sola variabile di cui andremo a scrivere il solo byte più alto. Partendo dal pixel più in basso, si scrive la rappresentazione binaria della colonna, sostituendo un '1' ai pixel accesi e uno '0' per gli altri; si ottiene '00010011'. Gli altri 3 byte della variabile non sono utilizzati, quindi la rappresentazione binaria della variabile risulta essere compresa fra i valori '00010011,00000000,00000000,00000000' e '00010011,11111111,11111111,11111111', corrispondenti ai valori decimali 318767104 e 335544319 (per la conversione si può utilizzare la calcolatrice scientifica di Windows)



**Esempio2:** si supponga di voler disegnare il simbolo a lato. Le colonne sono 6, quindi occorre porre Ncif=6, e le variabili occupate risultano 2. La rappresentazione binaria della prima variabile risulterà data dalle 4 colonne più a sinistra: '00100000,01000000,10000000,01000000', equivalente a 541098048. La seconda variabile conterrà le definizioni delle colonne 5 e 6: '00100000,00011000,xxxxxxx,xxxxxxx', dove x indica 1 o 0. Considerando tutti gli x come 0, si ottiene il valore decimale 538443776.

Con le scritture a display su posizione corrente (Row e/o Col posti a zero) si possono scrivere consecutivamente sia colonne grafiche che caratteri; occorre però ricordare che, a differenza delle colonne grafiche, i caratteri sono scritti in posizioni ben definite del display.

L'uscita Curs mostra in ogni momento la posizione del cursore.

Per display alfanumerico scheda SU210:

- da 1 a 40 -> caratteri della prima riga
- da 41 a 80 -> caratteri della seconda riga

Per display alfanumerico schede SU110-SU112:

- da 1 a 20 -> caratteri della prima riga
- da 21 a 40 -> caratteri della seconda riga
- da 41 a 60 -> caratteri della terza riga
- da 61 a 80 -> caratteri della quarta riga

Per display grafico scheda SU112:

- da 1 a 21 -> caratteri della prima riga
- da 22 a 42 -> caratteri della seconda riga
- da 43 a 63 -> caratteri della terza riga
- da 64 a 84 -> caratteri della quarta riga
- da 85 a 105 -> caratteri della quinta riga
- da 106 a 126 -> caratteri della sesta riga
- da 127 a 147 -> caratteri della settima riga
- da 148 a 168 -> caratteri della ottava riga

## Tabella dei caratteri per display grafico (solo per SU112)

CODICI ASCII 0 ~ 31: CORRISPONDENTI AI CARATTERI DI CONTROLLO (VUOTI)

CODICI ASCII 221 ~ 255: DISPONIBILI PER DEFINIZIONI UTENTE

32

46

SPAZIO	!	"	#	\$	%	&	'	(	)	*	+	,	-	.
--------	---	---	---	----	---	---	---	---	---	---	---	---	---	---

47

61

/	0	1	2	3	4	5	6	7	8	9	:	;	<	=
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

62

76

>	?	@	A	B	C	D	E	F	G	H	I	J	K	L
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

77

91

M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	[
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

92

106

\	]	^	_	`	a	b	c	d	e	f	g	h	i	j
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

107

121

k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

122

136

z	{		}	~	□	Ç	ü	é	â	ä	à	á	ç	ê
---	---	--	---	---	---	---	---	---	---	---	---	---	---	---

137

151

ë	è	ï	í	ì	Ä	Å	É	æ	Æ	ò	ö	ò	ú	ù
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

152

166

ÿ	Ö	Ü	¢	£	¥	Pt	f	á	í	ó	ú	ñ	Ñ	ª
---	---	---	---	---	---	----	---	---	---	---	---	---	---	---

167

181

°	¿	®	¬	½	¼	¡	«	»	€	♯	♯			
---	---	---	---	---	---	---	---	---	---	---	---	--	--	--

182

196

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

197

211

SIGMA				ALFA	BETA	GAMMA	DELTA	EPSILON	FI	LAMBDA	MU	OMEGA	PI	RO
-------	--	--	--	------	------	-------	-------	---------	----	--------	----	-------	----	----

212

220

TAU	TETA	PSI	DELTA MAIUSC	FI MAIUSC	LAMBDA MAIUSC	OMEGA MAIUSC	PI MAIUSC	SIGMA MAIUSC
-----	------	-----	-----------------	--------------	------------------	-----------------	--------------	-----------------

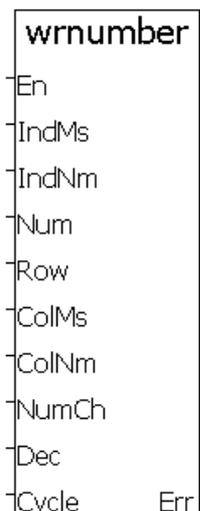
## Tabella dei caratteri per displays alfanumerici

La seguente tabella è stata fornita dal costruttore dei displays alfanumerici, ed il codice ASCII che individua il carattere è qui fornito in codice esadecimale.

ESEMPIO DI LETTURA: la 'R' è individuata dalla colonna 5 e riga 2, ottenendo così il codice ASCII esadecimale 50, equivalente in decimale a:  $5 \cdot 16 + 2 = 82$ .

		Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)																		
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F			
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0 CG RAM (1)			0	Q	P	`	P						-	タ	≡	α	p		
	1 CG RAM (2)		!	1	A	Q	a	a						。	ア	チ	△	ä	q	
	2 CG RAM (3)		"	2	B	R	b	r							「	イ	ツ	×	φ	θ
	3 CG RAM (4)		#	3	C	S	c	s							」	ウ	テ	ε	ε	ω
	4 CG RAM (5)		\$	4	D	T	d	t							、	エ	ト	μ	Ω	
	5 CG RAM (6)		%	5	E	U	e	u							・	オ	ナ	1	ü	Ü
	6 CG RAM (7)		&	6	F	V	f	v							ヲ	カ	ニ	ヨ	ρ	Σ
	7 CG RAM (8)		'	7	G	W	g	w							ア	キ	ヌ	ラ	g	π
	8 CG RAM (1)		(	8	H	X	h	x							イ	ク	ネ	リ	5	⊗
	9 CG RAM (2)		)	9	I	Y	i	y							ウ	ケ	ル	リ	γ	γ
	A CG RAM (3)		*	:	J	Z	j	z							エ	コ	ハ	レ	J	〒
	B CG RAM (4)		+	:	K	[	k	[							オ	サ	ヒ	ロ	*	〒
	C CG RAM (5)		,	<	L	¥	l	l							ハ	シ	フ	ワ	φ	円
	D CG RAM (6)		-	=	M	]	m	]							ユ	ス	ハ	ン	も	÷
	E CG RAM (7)		.	>	N	^	n	^							ヨ	セ	ホ	°	ん	
	F CG RAM (8)		/	?	O	_	o	_							ツ	ッ	マ	°	ö	■

## WrNumber



### Argomenti:

En	BOO	Se attivo mostra il numero sul display in modo continuativo.
IndMs	INT	Indirizzo eventuale stringa messaggio da visualizzare con il numero: in questo caso la stringa e' scritta una sola volta al fronte positivo di En.
IndNm	INT	Indirizzo variabile il cui numero deve essere visualizzato. Se zero considera il numero della variabile dell'ingresso Num.
Num	INT	Valore da mostrare se l'ingresso IndNm vale zero.
Row	INT	Riga su cui scrivere il numero e/o la stringa.
ColMs	INT	Colonna di inizio del messaggio di cui all'ingresso IndMs.
ColNm	INT	Colonna di inizio del numero di cui agli ingressi IndNm o Num.
NumCh	INT	Numero di caratteri, virgola compresa, per la visualizzazione del numero.
Dec	INT	Numero di decimali del numero da visualizzare.
Cycle	INT	Tempo di rinfresco del solo numero, espresso in msec. Tale tempo risulta limitato dal tempo di loop del plc e dalla frequenza di chiamata del blocco.
Err	INT	Errore occorso. Se zero l'operazione non ha riscontrato problemi.

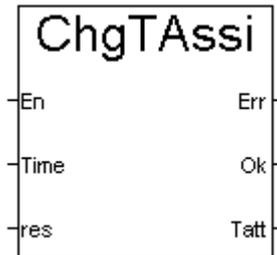
## Descrizione:

Il blocco consente di visualizzare su display numeri, anche con decimali, sempre allineati a destra e di dimensione predefinita, con visualizzazione di numero troppo grande per essere visualizzato mediante stampa di una serie di punti interrogativi (?). Il numero da visualizzare puo' essere fornito o come variabile diretta o inserendo il virtual address della variabile stessa. Il numero di decimali determinerà quante cifre mettere dopo il simbolo ".", se zero il numero sarà visualizzato come intero. Il parametro Cycle consente di decidere ogni quanti msec rieseguire il rinfresco del numero, fermo restando che non potrà avvenire più volte per ogni ciclo di PLC, quindi il tempo di rinfresco è limitato dal tempo di ciclo. L'ingresso Row determina la riga del display su cui scrivere il numero e/o la stringa di testo, ColMs indica la colonna da cui partire con la scrittura del messaggio testo (IndMs) e ColNm indica la colonna da cui partire per scrivere il numero. Il testo viene scritto una sola volta al fronte positivo dell'ingresso En, e può essere sovrascritto dal numero da visualizzare. Ciò permette di impostare una stringa con il nome della variabile da scrivere, seguito da una serie di spazi che saranno riempiti dal numero, seguito dall'unità di misura del numero.

Es: messaggio = "posizione            mm"  
     numero    =            "200.22"

In questo modo con un unico blocco è possibile predisporre la visualizzazione di un dato sul display con un aggiornamento e allineamento automatico.

## ChgTAssi



### Argomenti:

En	BOO	Sul fronte positivo attiva il blocco
Time	INT	Assegna il tempo corrispondente all'intervallo che intercorre fra due chiamate consecutive della routine di gestione dell'asse, in ms
Res	BOO	Resetta l'uscita Ok
Err	INT	Emette il codice d'errore: 0= nessun errore 1= valore di Time non valido 2= asse non fermo 3= versione hardware che non supporta il blocco
Ok	BOO	Si setta quando TIME è assegnato effettivamente
Tatt	INT	Rappresenta il valore di TIME attualmente utilizzato

### Descrizione:

Questo blocco permette di assegnare il tempo che si vuole che intercorra fra due chiamate consecutive della routine di gestione dell'asse (ingresso Time). Durante il suo utilizzo, l'asse deve essere fermo.

Nella regolazione del suddetto tempo, si deve ricordare che valori troppo alti degradano le prestazioni dell'asse, mentre valori troppo piccoli comportano il togliere tempo alle altre attività del PLC. Attualmente i valori accettati sono compresi nel range 2 ~ 13 (ms), e tramite l'uscita Tatt se ne vede il valore attualmente utilizzato.

L'uscita Ok si setta quando il valore impostato è accettato; Res permette di azzerarla.

Utilizzabile solo su SU210.

Con SU212, ha effetto per i soli assi sulla scheda base.

## Set\_Ip



## Argomenti:

En	BOO	Sul fronte positivo attiva il blocco
IP	MSG	Assegna l'indirizzo IP del dispositivo Ethernet all'interno della rete lan. Il numero deve essere composto da 4 cifre suddivise dal carattere <i>punto</i> . Es: 192.168.10.1
mask	MSG	Assegna la maschera per la valutazione dell'indirizzo IP all'interno della rete lan. Il numero deve essere composto da 4 cifre suddivise dal carattere <i>punto</i> . Es: 255.255.255.0
gatew	MSG	Indica al controllo Ethernet l'indirizzo del Gateway della rete lan. Il numero deve essere composto da 4 cifre suddivise dal carattere <i>punto</i> . Es: 192.168.10.100
Ok	BOO	Lo stato true indica l'avvenuta inizializzazione della scheda Ethernet.
Err	INT	Emette il codice d'errore: 0 = nessun errore ≠0 = errore. Contattare Arteco spa

## Descrizione:

Questo blocco permette di assegnare l'indirizzo IP, la maschera e il gateway alla scheda Ethernet che contiene il server web per la visualizzazione di pagine Html statiche o con codifica particolare per mostrare lo stato di variabili interne.

Per la codificazione delle istruzioni che consentono di visualizzare valori di variabili intere contattare ARTECO SpA.

## Appendice A – Variabili Virtuali

Con il termine **variabili virtuali** si intende un'area di memoria disponibile per memorizzare variabili gestibili esclusivamente mediante blocchi Arteco senza la necessità di dichiararle precedentemente.

Sono disponibili 8000 variabili virtuali con indici compresi tra 1 e 8000 compresi. Tali variabili sono di tipo intero e possono assumere valori compresi tra +2.147.483.647 e -2.147.483.648.

I blocchi Arteco destinati alla gestione di tali variabili sono VarIndex, che consente la lettura e la scrittura di una singola variabile e VarCpy che consente la copia di gruppi di variabili da una zona ad un'altra, con la possibilità di copiare i valori delle variabili dalla zona di variabili Virtuali alle variabili standard di ISaGRAF e viceversa.

Le variabili virtuali hanno lo scopo di fornire all'utente un cospicuo numero di variabili senza richiederne la dichiarazione.

Un esempio potrebbe essere un programma i cui dati variano al variare di un pezzo lavorato. Si possono dichiarare N variabili contenenti i dati del pezzo da lavorare e memorizzare nelle variabili virtuali i dati di svariati pezzi.

Ad es.

Se il programma in corso utilizza 20 variabili per eseguire un pezzo e si vogliono memorizzare i 10 pezzi eseguiti più comunemente, con le variabili di ISaGRAF è necessario dichiarare  $20 \times 10 = 200$  variabili.

In alternativa si possono memorizzare i 200 valori nelle variabili virtuali e richiamare le 20 variabili del pezzo da lavorare dalla zona virtuale prima di iniziare la lavorazione, mettendole in 20 variabili standard di ISaGRAF mediante il comando VarCpy.

Non è possibile l'introduzione di valori in formato floating point all'interno dell'area virtuale, mentre è possibile trasferire mediante il comando VarCpy variabili di diverso tipo (vedi descrizione del blocco).

Le variabili virtuali sono tutte di tipo Analogico Intero.

Le variabili virtuali sono retentive per cui dopo lo spegnimento/riaccensione della scheda mantengono il valore.

Dalla versione del firmware v68 in poi, le variabili virtuali mantengono il loro valore anche dopo la rimozione dell'applicativo e lo scaricamento di un nuovo applicativo. Vengono azzerate effettuando il reset della scheda.

## Appendice B – Utility di lettura e scrittura variabili via seriale

Questa utility consente di leggere il valore delle variabili di un applicativo ISaGRAF (sia variabili standard che variabili virtuali) mediante seriale. I valori di tutte le variabili sono restituiti in un file di tipo testo modificabile a piacimento. I files generati possono essere utilizzati per inserire i valori in un controllo diverso o sullo stesso controllo se vengono persi.

L'utility si chiama RWISaGRAFVar e deve essere lanciata da una finestra DOS o da gestione risorse, comunque su un PC con sistema operativo Windows 95 o Windows 98.

La figura 1 mostra la prima richiesta presentata dall'utility dopo l'esecuzione.

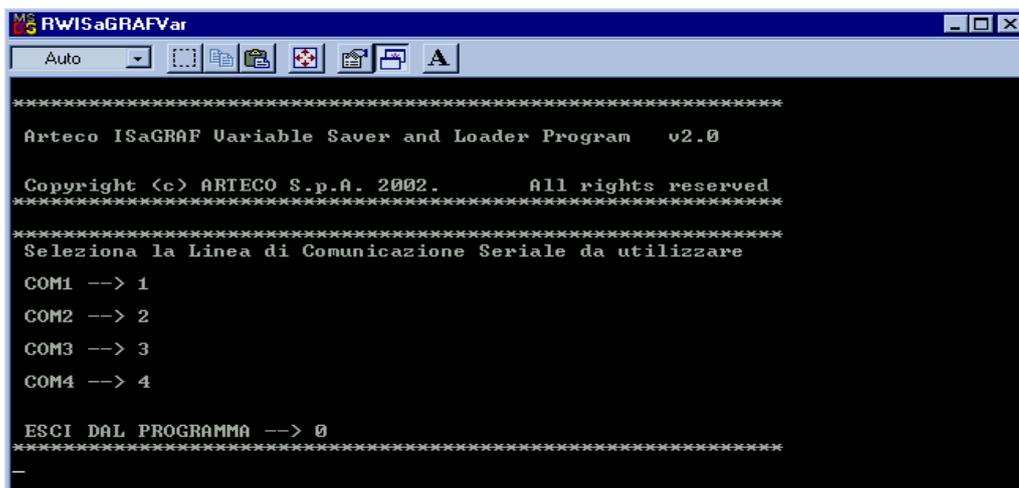
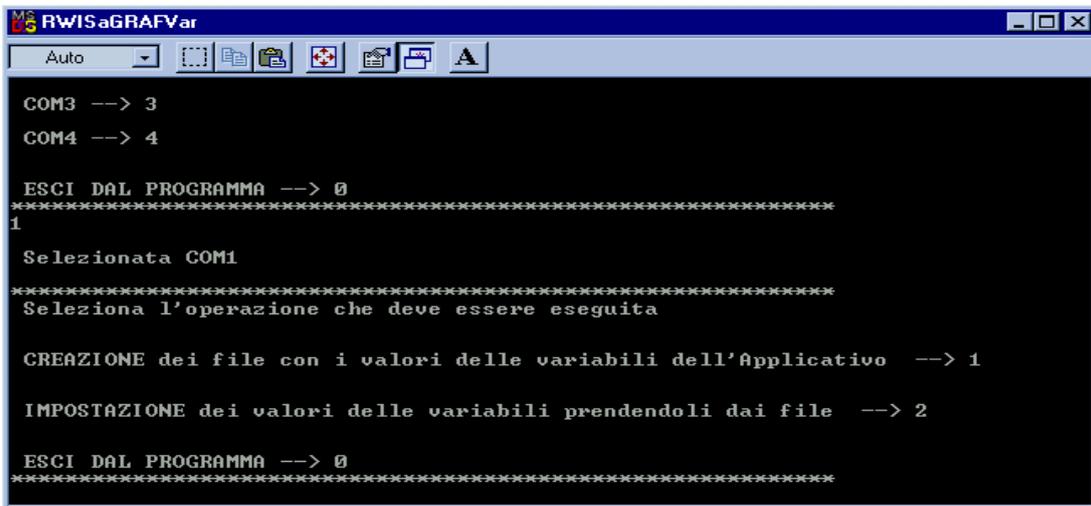


Fig. 1

Dopo aver digitato il numero corrispondente alla seriale scelta premere <INVIO>.  
In fig.2 è illustrata la successiva richiesta.



```
MS-DOS 5.02 C:\RWISaGRAFVar
Auto
COM3 --> 3
COM4 --> 4

ESCI DAL PROGRAMMA --> 0
*****
1
Selezionata COM1
*****
Seleziona l'operazione che deve essere eseguita

CREAZIONE dei file con i valori delle variabili dell'Applicativo --> 1

IMPOSTAZIONE dei valori delle variabili prendendoli dai file --> 2

ESCI DAL PROGRAMMA --> 0
*****
```

Fig. 2

Selezionare 1 per creare i files con i valori delle variabili. Alla pressione del tasto <INVIO> l'utility richiede se occorre cancellare eventuali file contenenti i valori delle variabili creati in precedenza. È consigliabile cancellarli poiché in caso contrario i nuovi valori vengono accodati. Inserire 1 e premere il tasto <INVIO>. Il programma comincia immediatamente a scaricare le variabili dal controllo e crea quattro files chiamati FileANA.dat, FileBOO.dat, FileTMR.dat e FileVIR.dat, rispettivamente per variabili analogiche, booleane, timer e virtuali.

Durante il download delle variabili dalla scheda viene visualizzato lo stato di avanzamento dell'operazione.

```

RWISaGRAFVar
Auto
Vuoi cancellare i file di variabili creati in precedenza?
SI --> 1
NO --> 2

ESCI dal programma --> 0
*****
<P.S. E' consigliabile cancellarli poiche' in caso contrario i valori
delle variabili saranno aggiunti in fondo ai presenti file di dati>
*****
1
I files sono stati cancellati

ATTENDERE !?...Scansione delle Variabili dell'Applicativo in corso...
.. scandite 65536 Variabili di cui 81 utilizzate nell'Applicativo

ATTENDERE !?...Scansione delle Variabili Virtuali in corso...
.. scandite 1564 Variabili Virtuali
  
```

Fig 3

Ogni file si presenta come indicato di seguito:

```

4094 0
4101 6
4102 1278
...
4210 0
4211 99
4212 16
4213 416
0 0
  
```

Dove per ogni riga il numero indicato a sinistra rappresenta il Virtual Address della variabile e il numero a destra il valore della medesima.

E' possibile modificare i valori delle variabili prima di scaricarli su un controllo.

I files terminano con "0 0".

E' possibile "scaricare" esclusivamente i valori delle variabili che hanno il Virtual Address dichiarato, mentre saranno scaricate tutte le variabili virtuali.

Al termine della generazione dei files sarà chiesto se visualizzarli a schermo.

L'utilizzo dell'opzione di visualizzazione a schermo con un numero di variabili molto alto può comportare problemi. Si consiglia di consultarle aprendo i files con un text editor comune.

In riferimento alla figura 2 selezionare 2 per trasferire il valore delle variabili al controllo.

Inizierà il trasferimento del valore delle variabili, con indicazione in tempo reale delle variabili effettivamente spedite.

```

RWISaGRAFVar
Auto
IMPOSTAZIONE dei valori delle variabili prendendoli dai file --> 2

ESCI DAL PROGRAMMA --> 0
*****
2
ATTENDERE !!...trasferimento delle Variabili dell'Appilacativo in corso
... Trasferite 32 Variabili Booleane
Le variabili Booleane sono state impostate correttamente
... Trasferite 46 Variabili Analogiche
Le variabili Analogiche sono state impostate correttamente
... Trasferite 3 Variabili Timer
Le variabili Timer sono state impostate correttamente

ATTENDERE !!...trasferimento delle Variabili Virtuali in corso
..trasferite 1380 Variabili Virtuali

```

Fig. 4

### Indicazioni per l'utilizzo.

L'utility di lettura e scrittura deve essere utilizzata collegandosi alla scheda SU210/250 su seriale che supporti il protocollo ModBus ISaGRAF sviluppato da Artec.

Occorre assicurarsi che lo standard elettrico delle seriali del PC e della scheda sia lo stesso (non RS232 con RS485).

Per qualsiasi informazione contattare l'Artec spa.

## Appendice C – Libreria I/O

La gestione degli I/O digitali/analogici avviene mediante l'utilizzo di librerie fornite dall'Artec SpA, da utilizzare per abbinare le variabili I/O utilizzate nel programma applicativo agli ingressi ed alle uscite fisiche della scheda SU210.

Di seguito sono spiegate le librerie per la gestione degli I/O Artec, mentre per la procedura di collegamento tra variabili e ingressi/uscite fisiche è necessario fare riferimento al manuale **ISaGRAF User's Guide** al capitolo **A.10 Using I/O connection editor**.

### I/O Digitali

Il sistema SU210 è costituito da un certo numero di I/O di base ed una serie di espansioni. Sono disponibili le librerie per la gestione sia degli I/O di base che per le eventuali espansioni utilizzate.

La libreria per ogni scheda di espansione (o per gli I/O di base) è indicata come *I/O complex equipment*. La libreria per gli I/O di base si chiama nS210\_IO e presenta 16 ingressi più 16 uscite per un totale di 32 I/O.

Le librerie per le espansioni si chiamano S210\_X, seguite da un numero che ne indica la posizione di montaggio rispetto alla scheda base e da una lettera che indica il tipo di espansione utilizzata.

Le schede di espansione si suddividono in UP e DOWN, come indicato in fig.1.

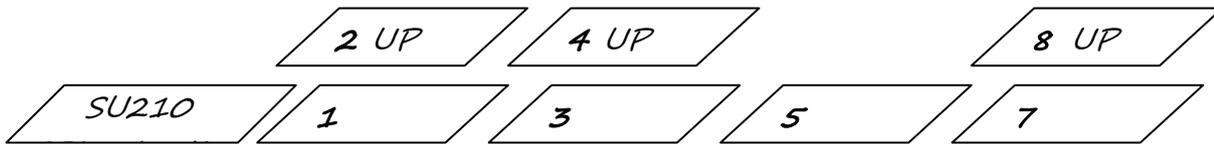


Fig.1

Tutte le espansioni in linea con la scheda CPU SU210 sono denominate DOWN e hanno un numero dispari nella libreria, mentre le espansioni che sono, a loro volta, montate al disopra di altre espansioni sono denominate UP e nella libreria riportano un numero pari (numero scheda inferiore + 1).

Le lettere indicano il tipo di scheda scelto (indipendentemente dalla posizione scelta nella "catena" delle espansioni). La lettera *a* indica una espansione con 32 ingressi, la lettera *b* indica espansione con 32 uscite e la lettera *c* indica espansione con 16 ingressi + 16 uscite.

Non sono presenti I/O complex equipments con numero pari e lettera *c* perché la scheda 16+16 non può essere collegata al di sopra di un'altra espansione ma può andare solo in linea con l'SU210.

Da quanto illustrato fino a qui risulta che sarà necessario selezionare l'I/O complex equipment da utilizzare (tra tutti quelli forniti dall'Arteco SpA) in base al tipo di scheda da utilizzare ed alla sua posizione nella "catena" dei collegamenti.

Ad esempio (vedere fig.2) la prima espansione (ad es. 32 ingressi) collegata dovrà utilizzare la libreria S210\_X1a. La seconda espansione (ad es. 32 uscite) utilizzerà la libreria S210\_X2b se sarà collegata sopra l'espansione precedente, mentre utilizzerà la libreria S210\_X3b se sarà collegata "in coda" all'espansione già presente.

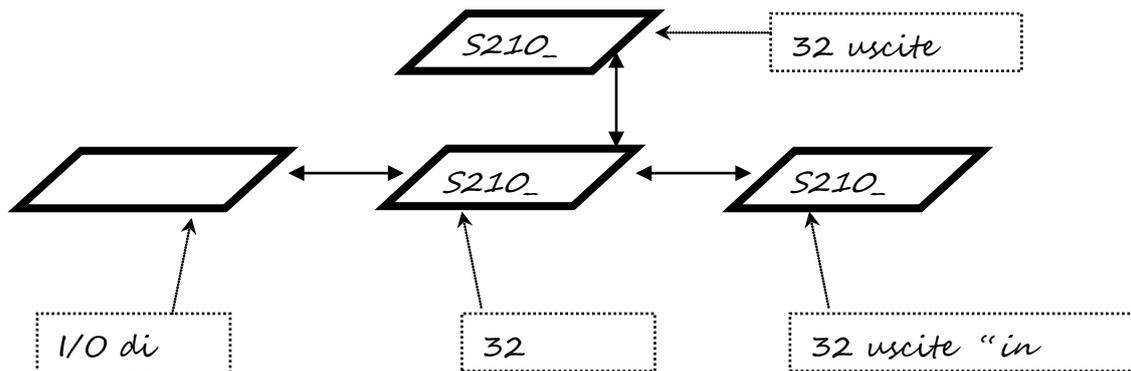


Fig.2

Sarà necessario cambiare I/O complex equipment esclusivamente se si sostituisce una scheda con una di tipo diverso.

Ogni libreria di gestione degli I/O (I/O complex equipment) è ulteriormente divisa in board, contenente ciascuna un dato numero di ingressi e/o uscite, vedi fig.3.

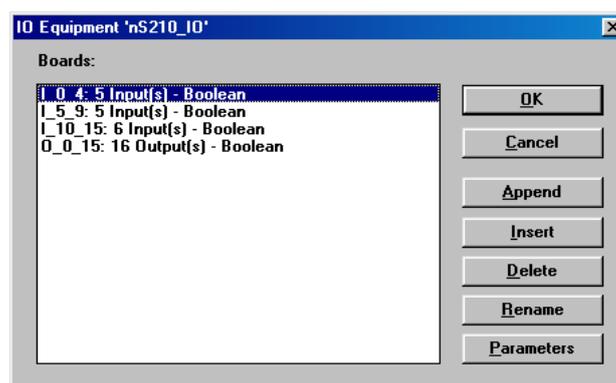


Fig.3

Ad es. nS210\_IO contiene 3 board di input per un totale di 16 ingressi (5+5+6) e 1 board di output di 16 uscite.

Le espansioni (S210\_X) sono suddivise in tre modi:

- S210\_X...a : 4 board di 8 input per un totale di 32 ingressi.
- S210\_X...b : 4 board di 8 output per un totale di 32 uscite.
- S210\_X...c : 2 board di 8 input per un totale di 16 ingressi e 2 board di 8 output per un totale di 16 uscite.

Tale suddivisione non dipende dalla posizione dell'espansione ma esclusivamente dalla sua implementazione hardware.

Qualora il numero di espansioni utilizzato sia superiore al numero dei driver forniti inizialmente, contattare l'Arteco per avere i nuovi driver o la documentazione per crearseli autonomamente.

## I/O Analogici

Il sistema su210 è dotato di cinque ingressi per la lettura di canali analogici gestibili da programma applicativo.

La libreria da utilizzare per la lettura dei suddetti canali è fornita dall'Arteco SpA e si chiama nS210Ana. Tale libreria è presente nella categoria degli I/O complex equipment ed è costituita da un'unica board chiamata *anal*.

Sono disponibili cinque canali analogici che rappresentano il valore di tensione presente all'ingresso dedicato (vedere schema dei collegamenti) espresso in mvolt.

Più precisamente:

- 1) Ingresso utente presente su J2 pin 12.
- 2) Ingresso utente presente su P4 pin 4.
- 3) Ingresso lettura tensione di alimentazione.
- 4) Non disponibile.
- 5) Non disponibile.

## Stato alimentazione

Il sistema su210 consente di leggere da programma applicativo alcune informazioni inerenti lo stato delle alimentazioni della scheda, mediante la libreria nS210Ali.

Le informazioni disponibili sono:

- Prot\_Out : Controllo di protezione delle uscite. Consente di rilevare alcune situazioni anomale sulle uscite digitali quali il corto circuito, il carico aperto (corrente inferiore a 50mA) e la sovratemperatura del dispositivo di pilotaggio delle uscite.
- Fuse\_Out : Indica l'attivazione dei fusibili di protezione delle uscite utente.
- Fuse\_Enc : Indica l'attivazione del fusibile di protezione dell'alimentazione encoder.
- Batt\_NO : Consente di sapere da applicativo se la batteria è carica o scarica. TRUE indica che il livello di carica della batteria è insufficiente a mantenere i dati in memoria. E' necessario sostituire la batteria al più presto senza spegnere il controllo. Vedere manuale tecnico e di manutenzione.

- Liv\_Alim : Consente la lettura della tensione di alimentazione da programma applicativo. Vedere il manuale tecnico e di manutenzione.

### Dati gestione Asse

La libreria nS210Ass consente di avere informazioni inerenti la gestione dell'asse.

Le informazioni disponibili sono:

- Emerg : Ingresso emergenza. TRUE indica tutto ok, FALSE indica stato di emergenza.
- FC\_Min : Ingresso sensore di minima. FALSE indica fine corsa di minima raggiunto.
- FC\_Max : Ingresso sensore di massima. FALSE indica fine corsa di massima raggiunto.
- FC\_Zero : Ingresso sensore di zero. TRUE indica sensore di zero attivo.
- Mrk\_Zero : Ingresso tacca di zero. TRUE indica tacca di zero attiva e presente.
- Hold : Ingresso di hold. Non utilizzato. Disponibile per il programma utente.
- IN\_SP : Ingressi speciali utilizzati per la funzione di cattura quota ad interrupt.
- PosAtt : Lettura della posizione attuale espressa in mm. Sostituisce il blocco ReadPos.
- VelAtt : Lettura della velocità attuale espressa in mm/sec. Sostituisce il blocco ReadPos.
- Vout : Tensione in uscita espressa in millivolt.

Tutte le informazioni sono disponibili esclusivamente in lettura. La funzione lock di uno qualsiasi dei suddetti ingressi non ha alcun effetto sul gestore asse.

La libreria per la lettura dei sensori della scheda di espansione asse EXP\_AX si chiama *expaxe* e presenta le stesse informazioni della libreria nS210Ass, ad eccezione degli ingressi speciali IN\_SP e Hold.

La libreria per la lettura dei sensori della scheda di espansione asse "bus" si chiama *expaxbus* e presenta le stesse informazioni della libreria nS210Ass, ad eccezione degli ingressi speciali IN\_SP e Hold.

## Appendice D – Errori flash card

0	Nessun Errore	Nessun errore occorso dall'accensione
1	Errore lettura/scrittura	Errore segnalato dalle operazioni di basso livello. Durante un accesso alla flash card viene rilevato l'errore.
2	Errore nel primo settore	Il primo settore della flash card contiene dei dati errati: il numero dei file contenuti non corrisponde ai file effettivamente presenti, il file richiesto non e' presente.
3	Trovato firmware	Sulla flash card e' presente un file di firmware e si e' richiesta un'operazione incompatibile con esso.
4	Trovato applicativo	Sulla flash card e' presente un applicativo e si e' richiesta un'operazione incompatibile con esso.
5	File con uguale nome	Sulla flash card e' già presente un file con il nome indicato e si e' richiesta un'operazione incompatibile con esso.
6	Spazio insufficiente	Nella flash card non c'e' spazio sufficiente per salvare il file.
7	Errore dati	I dati trovati sulla flash card sono inconsistenti: lo spazio effettivamente utilizzato da un file non corrisponde al valore memorizzato nella flash card.

## Appendice E – Compilatore Arteco per C167

L'applicativo sviluppato in ambiente ISaGRAF® può essere compilato appositamente per il processore presente all'interno del PLC SU210, SU212, SU250, CND51. Questo consente una notevole riduzione del tempo di ciclo del PLC, in modo dipendente dal tipo di linguaggio scelto e dalla percentuale di blocchi funzione presenti rispetto all'applicativo totale.

Per l'installazione del compilatore fare riferimento ad Arteco SpA. Di seguito sarà illustrato come utilizzarlo, cioè come compilare e caricare l'applicativo generato.

### Fase 1 – Opzione di compilazione

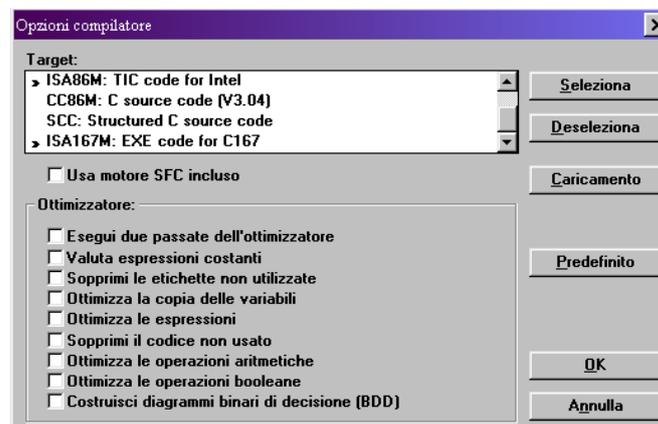


Fig. 1

Nel progetto selezionato scegliere "Compila" e "Opzioni Compilatore". Apparirà la finestra riportata sopra. Per l'utilizzo del compilatore Arteco per C167 selezionare la voce evidenziata: **ISA167M: EXE code for C167**.

Al momento della compilazione, mediante il comando "Compila Applicazione" verrà visualizzato quanto riportato di seguito, ad indicare l'effettiva compilazione mediante compilatore per C167.

```
Verifica  
Code generation : code for SABC167 : Release <0> Version <1> Revision <1>  
Nessun errore rilevato
```

Al termine della compilazione l'applicativo potrà essere scaricato all'interno del target.

## Fase 2 – Download

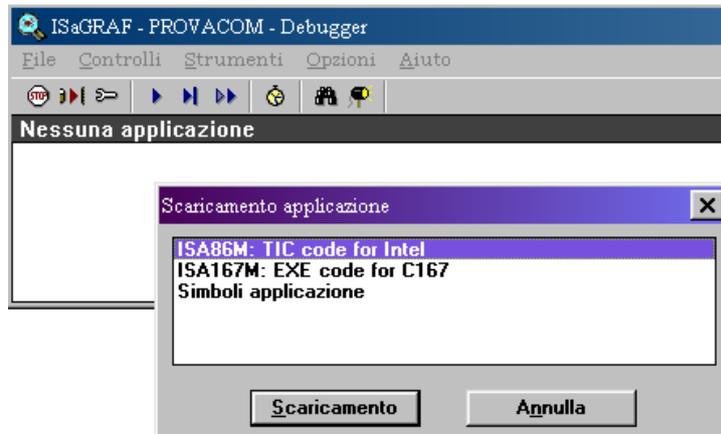


Fig. 2

Dopo il collegamento con il target mediante "Debug" si procede al Download (o scaricamento). Premendo l'apposito comando si aprirà una finestra in cui compaiono le voci "**ISA86M: TIC code for Intel**" e "**ISA167M: EXE code for C167**". La prima invia al target l'applicativo compilato con formato tradizionale (codice interpretato), mentre la seconda invia al target l'applicativo compilato per scheda SU.

Tutte le funzionalità di debug mediante Workbench continuano ad essere utilizzabili anche con applicativo compilato, ad eccezione del breakpoint in ambiente SFC che è disabilitato.

La dimensione del programma applicativo compilato è notevolmente superiore al programma interpretato (TIC Code), quindi il tempo di scaricamento (download) dell'applicativo mediante workbench risulta sensibilmente superiore.

E' possibile salvare su SIM l'applicativo compilato "EXE Code" e ripristinarlo con le stesse modalità dell'applicativo "TIC Code". Il ripristino da SIM ha una durata invariata, indipendente dall'applicativo e/o dal tipo di eseguibile (TIC o EXE).

Il formato "EXE Code for C167" è ammesso esclusivamente su schede che lo prevedono. Le versioni che consentono questo tipo di applicativo sono la 49 del 12/03/2001 e le versioni successive alla 59, compresa.

Al momento del download (scaricamento) mediante workbench di un applicativo EXE viene controllata la versione del target e, se non prevede il formato in oggetto, l'opzione non sarà mostrata nella maschera di scelta (vedi Fig.2).

Il caricamento di un applicativo EXE mediante SIM su un target che non lo consente può provocare malfunzionamenti della scheda SU.



motion control technologies

**ARTECO MOTION TECH SpA**

Via Gentili, 22 – 48018 Faenza (RA) Italy

Phone: +39 0546 645777 – Fax +39 0546 645750

[info@arteco-cnc.com](mailto:info@arteco-cnc.com) – [www.arteco-cnc.com](http://www.arteco-cnc.com)