zenon manuale

Simulazione driver

v.7.20

©2015 Ing. Punzenberger COPA-DATA GmbH

Tutti i diritti riservati.

Tutti i diritti riservati la distribuzione e la copia - indifferentemente dal metodo - può essere consentita esclusivamente dalla dittaCOPA-DATA. I dati tecnici servono solo per la descrizione del prodotto e non rappresentano in alcun modo parti legali. Modifiche - anche sotto aspetti tecnici sono a noi riservate

Indice

1. Benvenuti nell'help COPA-DATA 5

2. Simulazione driver 5

3. Simulazione statica 6

4. Simulazione – conta 6

5. Simulazione - programmata 7

5.1 Editor 8

5.1.1 Imposta progetto 8

5.1.2 Elimina progetto 11

5.1.3 Engineering distribuito 12

5.1.4 Cambia driver 13

5.1.5 XML-Export/Import 13

5.2 Configurazione driver. 14

5.2.1 Stato Variabile driver 14

5.2.2 Sincronizzazione ciclica 16

5.3 zenon Logic Workbench 18

5.4 Runtime 19

5.4.1 Funzionalità 19

5.4.2 Array e simulazione 21

5.4.3 File Runtime 21

5.4.4 Scambio dati 22

5.4.5 Formato dati 24

5.4.6 Ridondanza 24

5.4.7 Stato 25

5.4.8 Start/Stopp 27

5.4.9 Comandi driver 29

5.4.10 Assegnazione variabili 29

5.4.11 Data e ora 30

5.5 Indicazioni sulle variabili in progetti di simulazione 30

5.6 Segnalazioni d'errore 31

1. Benvenuti nell'help COPA-DATA

GUIDA GENERALE

Nel caso in cui non abbiate trovato delle informazioni che cercavate o se avete dei consigli relativi al completamento di questo capitolo dell'help, mandate una Mail a documentation@copadata.com (mailto:documentation@copadata.com).

SUPPORTO ALLA PROGETTAZIONE

Se avete delle domande concernenti progetti concreti, potete rivolgervi per E-Mail al support@copadata.com (mailto:support@copadata.com).

LICENZE E MODULI

Nel caso in cui doveste constatare che avete bisogno di altri moduli o licenze, rivolgetevi ai nostri dipendenti all'indirizzo sales@copadata.com (mailto:sales@copadata.com).

2. Simulazione driver

Se, al momento della progettazione, non è disponibile il processo di base, quest´ultimo può essere simulato e poi testato. A tal scopo, sono a disposizione tre modalità:

- Simulazione - statica (A pagina: 6): valori constanti simulati dal driver.
- Simulazione - conta (A pagina: 6): i valori simulati dal driver vengono calcolati
- Simulazione - programmata (A pagina: 7): i valori vengono calcolati mediante un progetto di simulazione con zenon Logic

Attenzione

Nel caso in cui il driver venga fermato nella modalità ***Simulazione - conta****, si ferma solamente il procedimento di conta. Le variabili non passano a* `disturbato`*. In tutte le altre modalità di funzionamento, invece, il driver viene interrotto per davvero.*

Nota: **Simulazione - programmata** non viene supportata dai driver per:

- Variabili interne
- Variabili matematiche
- Variabili di simulazione
- Variabili di sistema

Informazioni su

Le variabili di zenon che rappresentano , IO variabili zenon Logic, non sono disponibili nel progetto nello stato ***Simulazione driver - programmata****.*

Informazioni sulla licenza

Contenuta nella licenza standard per Editor e Runtime

3. Simulazione statica

Nel caso della simulazione statica, non viene instaurata una comunicazione con il PLC; i valori vengono simulati dal driver. In questa modalità, i valori rimangono costanti. I valori possono essere modificati mediante il runtime oppure dall´utente. Nel caso in cui il runtime venga riavviato in **Simulazione - statica**, tutti questi valori non vengono salvati e vanno dunque perduti.

4. Simulazione – conta

Nel caso di simulazione- conta, non viene instaurata una comunicazione con il PLC; i valori vengono simulati dal driver. In questa modalità il driver calcola i valori automaticamente, cominciando da `0` e nell'ambito della loro area di misurazione. Quando si raggiunge il valore massimo, il procedimento di conta riprende da `0`.

Informazioni su

Nel caso di valori di partenza negativi, il procedimento di conta inizia solamente da 0.

INT usa per contare il valore massimo di USINT.

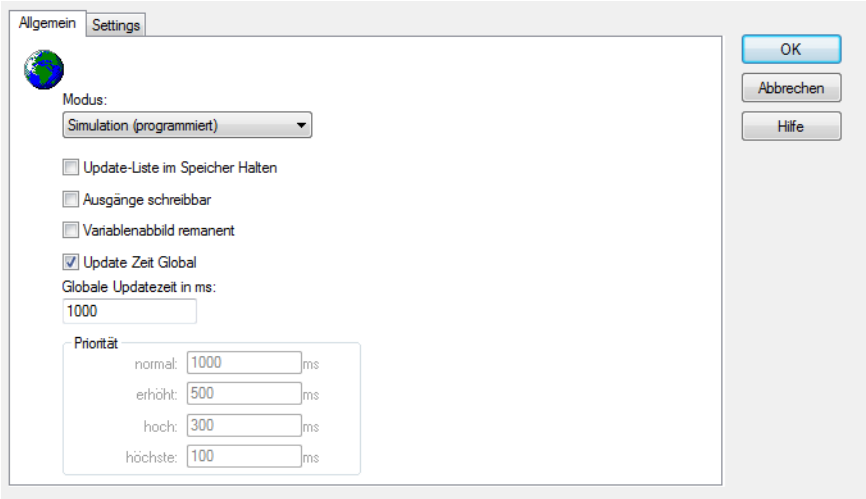
Attenzione

Nel caso in cui il driver venga fermato nella modalità ***`Simulazione - conta`****, si ferma solamente il procedimento di conta. Le variabili non passano a* `disturbato`*. In tutte le altre modalità di funzionamento, invece, il driver viene interrotto per davvero.*

5. Simulazione - programmata

Nel caso di simulazione programmata, non viene instaurata una comunicazione con il PLC; i valori vengono calcolati da un progetto di simulazione liberamente programmabile. Il progetto di simulazione viene realizzato con il workbench di zenon Logic e funziona nel runtime di zenon Logic. Ciò consente di modificare lo stato e il dato tempo di variabili sul driver. Comandi di scrittura a variabili vengono trasmessi alla simulazione tramite zenon; il server ridondante e il server di standby vengono sincronizzati. In tal modo è possibile simulare anche procedimenti complessi.

Per avviare il programma di simulazione dovete selezionare nella Configurazione driver (A pagina: 14) la modalità Simulazione - programmata.



Informazioni su

Windows CE

La ***simulazione-programmata*** *non è disponibile sotto Windows CE.*

5.1 Editor

5.1.1 Imposta progetto

Per poter impostare un progetto per la **simulazione - programmata**, si deve procedere come segue:

- si deve selezionare un solo driver di processo
- il nome progetto deve essere valido.

Attenzione

Simulazione - programmata non viene supportata dai driver per:

- Variabili interne
- Variabili matematiche
- Variabili di simulazione
- Variabili di sistema

Per creare un progetto di simulazione:

- cliccate all´interno del gruppo **Progetto di simulazione del driver**, in corrispondenza della proprietà **Modifica** su **Clicca qui ->**
- viene generato un nuovo progetto zenon Logic
- **Nome** e numero di port per **Porta eventi** e **Porta standard** vengono attribuiti automaticamente; potete modificarli a piacere.
- il workbench di zenon Logic viene aperto

Il workbench del progetto di simulazione viene chiuso automaticamente se:

- l´editor di zenon viene chiuso
- viene cancellato un driver il cui progetto di simulazione viene editato al momento nel workbench
- il progetto di simulazione viene chiuso nel driver facendo uso della proprietà **Elimina**

- il progetto di simulazione viene rinominato

Informazioni su

Numeri di porta: Quando viene generato il progetto di simulazione, gli viene assegnato automaticamente un numero di porta univoco. un numero di porta assegnato automaticamente è univoco per il relativo progetto. Nei casi di progetti multigerachici, ci si deve assicurare che un numero di porta sia usato solamente una sola volta in tutti i progetti attivi nel Runtime. Nel caso in cui un numero di porta venga usato più di una volta, infatti, si possono verificare degli errori di comunicazione con il Workbench/Runtime di zenon Logic.

I numeri di porta possono essere modificati e adeguati manualmente. Un numero di porta deve soddisfare le seguenti condizioni:

- Deve essere libero nel progetto.
- Deve essere univoco per tutti i progetti che sono attivi nel Runtime.
- Deve essere disponibile sul calcolatore su cui è attivo il driver.

Numeri di porta consentiti:

- Minimo: `6000` (in caso di assegnazione automatica, manuale: consigliato)
- Massimo: `6999` (in caso di assegnazione automatica, manuale: consigliato)

LUOGO IN CUI SALVARE IL PROGETTO

Un progetto zenon Logic contiene molti file e cartelle. Per facilitare il lavoro con essi, soprattutto nel caso di engineering diviso, tutti i file vengono salvati in forma compatta nel file **`Simul_<Treiber-ID>.zip`** che si trova nella cartella `<Sql Projekt Pfad>\FILES\zenon\custom\drivers`. Nell´editor, questo file si trova con quelli del driver. Al momento dell´avvio del workbench di zenon Logic, i file vengono "decompressi" automaticamente e, al momento della chiusura del workbench, vengono di nuovo compressi.

IMPOSTA VARIABILI

Impostando variabili in un progetto di simulazione, è importante procedere rispettando la seguente successione di passaggi:

1. prima di tutto configurate la variabile
2. selezionate poi il tipo di dato valido nel workbench di zenon Logic
3. solo dopo attivate la proprietà **embed symbol**

Sfondo: le variabili vengono impostate di default con il tipo di oggetto driver `SPS Merker`. Questo tipo di oggetto non supporta con ogni driver tutti i tipi di dato. Se una variabile viene copiata nel workbench

di zenon Logic con la proprietà **embed symbol** attiva, quest´ultima dovrà essere disattivata per poter poi procedere al cambiamento del tipo di dato. In tal modo, la variabile viene cancellata in zenon.

Informazioni su

Segnalazione di errore "Can not create variable"

Non è possibile impostare una variabile nell´oggetto di simulazione se il driver non ha nessun tipo di oggetto driver (escluso

***Variabile driver**) che supporta questo tipo di dato.*

*Quando si attiva **embed symbol**, verrà poi visualizzata la segnalazione di errore "can not create variable".*

STRUTTURA DI TIPI DI DATO

Quando si impostano delle variabili con un tipo dato strutturato, avviene quanto segue:

- le variabili vengono impostate, se possibile, con il tipo di oggetto driver `SPS Merker`.
- il sistema prova tutti i tipi di oggetto driver (fatta eccezione per **`Variabile driver`**) finché non ne trova uno utilizzabile per tutti gli elementi.
- se non è disponibile nessun tipo di oggetto driver adeguato, avviene quanto segue:
  - non viene creata nessuna variabile nel caso di non-variabili struttura
  - nel caso delle variabili struttura, si usa il tipo di oggetto driver `SPS Merker`.

ESEMPIO:

un tipo dato strutturato contiene elementi del tipo UINT e STRING. Viene creata una variabile per il driver S7TCP e **`embed symbol`** viene attivata.

- La variabile non viene impostata in `SPS Merker`, ma come `Blocco dati esteso`, in cui sono presenti tutti gli elementi struttura.
- Si aggiunge un nuovo tipo LINT; esso non viene supportato da `Blocco dati esteso`.
- Quando si crea una nuova variabile, essa viene impostata come tipo `SPS Merker`. Può essere attivata solamente la prima variabile complessa (UINT). Nel caso della variabile strutturata esistente, rimane il tipo di oggetto `Blocco dati esteso` e l´ultimo elemento di struttura (LINT) non può essere attivato.

COMPILAZIONE DEI FILE

Se un progetto di simulazione viene compilato nel workbench di zenon Logic, ciò comporta in zenon la creazione delle variabili e del file di programma nel runtime.
Nel compilare, nella cartella runtime \RT\FILES\zenon\custom\drivers viene generata una sottocartella

con il nome del progetto di simulazione. In questa cartella viene poi depositato il file con il codice della simulazione in SIMULRT.COD.

Durante la compilazione, nella finestra di emissione viene visualizzata la seguente segnalazione:

SIMULRT.COD

E se si compila con C-Compiler, apparirà inoltre:

T5APP.DLL

> **Attenzione**
>
> *Nel caso di* **`Progetti multiutente`** *(A pagina: 12), non si può creare nessun nuovo progetto di simulazione offline. Esso non può essere più cancellato.*

5.1.2 Elimina progetto

Per cancellare un progetto di simulazione:

1. cliccate all´interno del gruppo- **Progetto di simulazione del driver** in corrispondenza della proprietà **Elimina** sul pulsante su **`Clicca qui >`**
2. cliccate su ok quando il sistema Vi richiede di confermare a fine di sicurezza
3. il file ZIP contenente i file di progetto viene cancellato subito
4. il workbench di zenon Logic viene chiuso

Tenete presente:

- la cartella runtime per il progetto di simulazione, invece, viene conservata
- la funzione **`Annulla`** non è disponibile per questa azione,
- cliccando su **Modifica** si crea un nuovo progetto di simulazione
- nel caso di **`Progetti multiutente (A pagina: 12):`**
  - si deve configurare il driver su **`Consenti modifiche`** per consentire la cancellazione del programma di simulazione
  - il progetto non viene visualizzato su altri client come cancellato finché non è stato sincronizzato;

    il tentativo di aprire un progetto cancellato prima della sincronizzazione (clic su **Modifica**) causa la creazione di un nuovo progetto di simulazione.
  - come nel caso di progetti Standalone, la cancellazione mediante il comando **`cancellare modifiche`** è irreversibile.

> **Attenzione**
>
> *Il progetto di simulazione viene rimosso già al momento della cancellazione dalla banca dati locale e da quella del server. Non sarà possibile annullare questa operazione!*

Nota: Anche la cancellazione del file ZIP con i file del driver porta all´eliminazione del progetto di simulazione. Condizione: il workbench di zenon Logic per questo driver non è aperto. Si sconsiglia di adottare questa procedura!

5.1.3 Engineering distribuito

Nel caso di **`progetti multiutente`**, tutte le funzionanti workbenches zenon Logic appartenenti al progetto vengono chiuse per la simulazione driver in corrispondenza di:

- `Riprendi modifiche` per moduli, che per **`Permetti modifiche`** vengono completamente bloccati per altri utenti, come variabili, driver e tipi di dato.
- `Cancella modifiche`:
  - non vengono creati nuovi progetti sul server
  - le modifiche vanno perse nel programma di simulazione
  - i file driver non presenti nella banca dati del server vengono cancellati anche nelle banche dati locali
- Sincronizza:
  le modifiche realizzate localmente nel programma di simulazione vanno perdute.
- Attualizza la versione locale:
  le modifiche realizzate localmente nel programma di simulazione vanno perdute.

> **Attenzione**
>
> *Lo stato del file ZIP del progetto di simulazione non deve essere in alcun modo modificato extra (, `Permetti modifiche`, `Rifiuta modifiche` Stato Multiuser), per salvare operazioni corrette `Riprendi modifiche` e `Permetti modifiche`del driver!*

ELIMINA PROGETTO

Nel caso di **`Progetti multiutente`**

- si deve configurare il driver su **`Consenti modifiche`** per consentire la cancellazione del programma di simulazione
- il progetto non viene visualizzato su altri client come cancellato finché non è stato sincronizzato;
  - il tentativo di aprire un progetto cancellato prima della sincronizzazione (clic su **Modifica**) causa la creazione di un nuovo progetto di simulazione.

- come nel caso di progetti Standalone, la cancellazione mediante il comando **`cancellare modifiche`** è irreversibile.

> **Attenzione**
>
> Un progetto di simulazione con lo stato `rendere possibili le modifiche`, nella modalità `Offline` non può essere cancellato (A pagina: 11).
>
> per questa ragione un progetto di simulazione generato in un **`progetto multiutente`** `offline` non può essere più cancellato.

5.1.4 Cambia driver

Nel caso di cambiamento di driver in zenon vale quanto segue:

- Il progetto di simulazione rimane.
- Tuttavia, tutte le porte vengono settate su `0`. Per ottenere numeri di porta funzionanti, aprite il progetto nell´Editor. Il sistema provvederà automaticamente a inserire nuovi numeri di porta. Questi possono essere anche assegnati manualmente.

> **Attenzione**
>
> *I driver che sono collegati con un progetto zenon Logic tramite la soluzione integrata, non possono essere cambiati.*

5.1.5 XML-Export/Import

*Un progetto in modalità **`simulazione - programmata`** può essere esportato via XML. Non può essere importato però come nuovo progetto per un altro driver.*

Workaround:

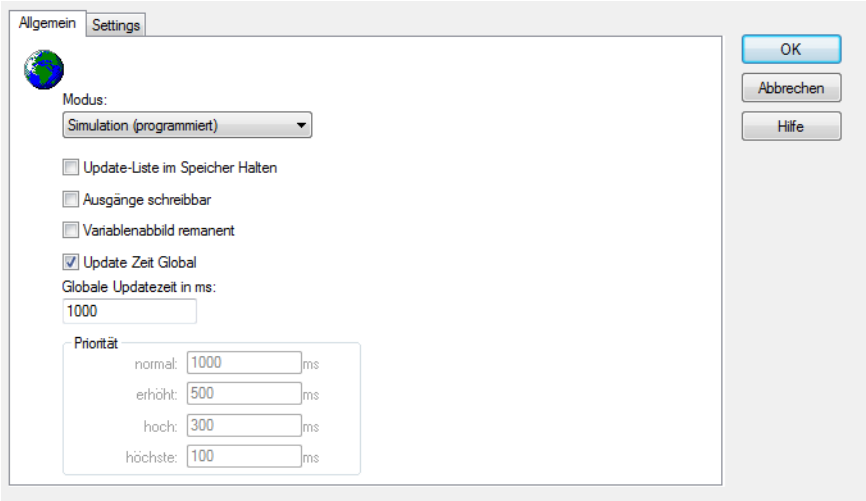
1. Esportate i programmi zenon Logic.
2. Importateli nel nuovo progetto per il nuovo driver.
3. Sostituite le variabili mediante Cerca e Sostituisci.

5.2 Configurazione driver.

Quando procedete alla configurazione del driver, sono a Vostra disposizione quattro modalità:

- **`Hardware`**
- **`Simulazione – conta`**
- **`Simulazione statica`**
- **`Simulazione - programmata`**

Selezionate la modalità **`Simulazione - programmata`**.



5.2.1 Stato Variabile driver

Lo stato del runtime di zenon Logic viene visualizzato nella variabile driver SimulRTState su Offset 60. La variabile è numerica e non può essere scritta. Il valore fornisce informazioni sullo stato del runtime:

Bit	Significato	Descrizione
31	senza applicazione	Runtime non ha caricato nessun programma; il programma è stato bloccato, oppure non è caricato nel runtime.
30	non instanziato	Il runtime non è stato instanziato, per esempio perché non è stato possibile caricare DLL con il runtime, oppure perché non c`è abbastanza spazio di memoria, oppure perché DLL non è disponibile o lo è nella versione sbagliata.
18	p-code available	p-Code è disponibile insieme a codice compilato; è possibile passare dall´uno all´altro.
17	compiled code active	Il runtime è funzionante con code compilato (altrimenti: p-Code interpretato).
16	compiled code available	compiled code di C-Compiler è disponibile.
15	application is loaded	un´applicazione è presente, il runtime funziona.
14	cant start - missing handlers	Mancano alcune funzioni "C".
13	active breakpoints installed	E´stato settato dal debugger almeno un breakpoint.
12	CT segment exists	L´applicazione è stata compilata con l´opzione "complex variables in separate segment".
11	Riservato	(Interno, solo per il runtime.)
10	sysinfo request is available	(Interno, solo per il runtime.)
9	freeze event production	Binding non trasmette nessun evento.
8	single cycle mode	(Interno, solo per il runtime e debugger)
7	Riservato	(Interno, solo per il runtime.)
6	locked variables	Perlomeno una variabile è bloccata.
5	trigo functions are in degrees	Le funzioni trigonometriche vengono indicate in gradi.
4	log message(s) in stack	(Interno, solo per il runtime.)
3	application stopped between 2 progs	Il runtime é stato bloccato nell´effettuazione dell´operazione di debugging nel mezzo di due passaggi.
2	application stopped on SFC breakpt	(Interno, solo per il runtime.)
1	application stopped on error	Il runtime viene bloccato per errore grave. Necessario un nuovo start.
0	application is running	TRUE = l´applicazione è nella modalità "run". FALSE = l´applicazione fa una pausa nella modalità "cycle to cycle" .

Informazioni su

La variabile driver SimulRTState con Offset 60 indica lo stato del runtime sul server. Ciò vale anche se viene visualizzata la variabile sul server in standby.

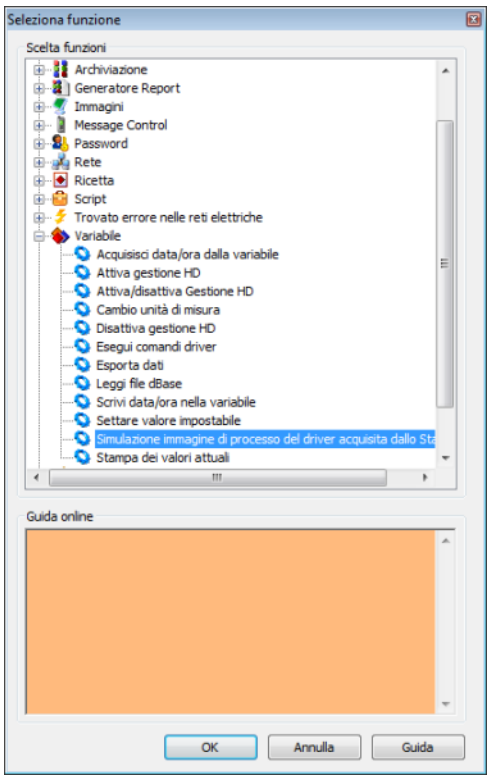
5.2.2 Sincronizzazione ciclica

Le simulazioni funzionano in rete sul server e sul Server-Standby in modo autonomo. Al momento dello start del Server-Standby si verifica una sincronizzazione con il server, ma:

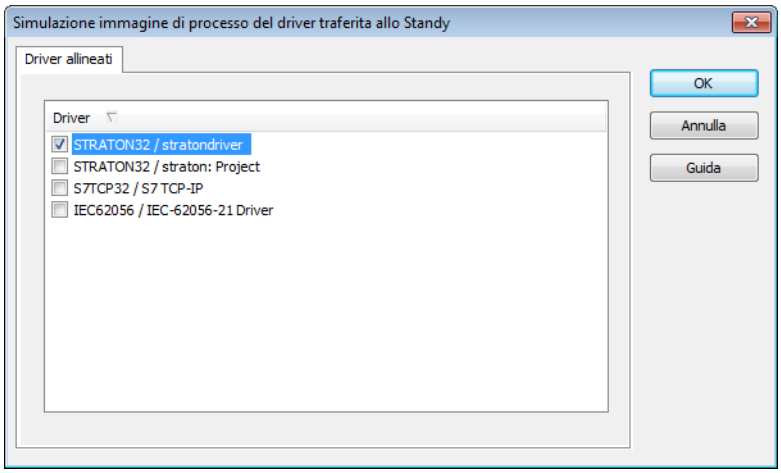
- se questa prima sincronizzazione viene rimandata in modo corrispondente alla scadenza pacchetti
- delle piccole differenze portano a crescenti divergenze nell´esecuzione.

Anche lo stato delle variabili simulate varia. Per limitare l´entità di queste differenze, sfruttate la funzione **`Trasmetti Immagine di simulazione driver allo standby`** per la sincronizzazione dell´immagine di simulazione:

- create una nuova funzione.
- selezionate **`Trasmetti driver-immagine di simulazione allo standby`**



- Si apre il dialogo per la selezione dei driver.



- nella lista sono indicati tutti i driver collegati al processo (escluso simulatore) disponibili (cliccate sulla cima della colonne per cambiare il criterio secondo il quale i driver sono ordinati nella lista).
- selezionate adesso quel driver che devono essere sincronizzati
- per ogni richiamo della funzione sul Server-Standby viene richiesta un´immagine dal server per il driver selezionato.

Mediante attivazione ciclica, possono essere ridotte al minimo le differenze nello stato di simulazione attuale. Adattate la griglia dell´esecuzione alla quantità dei dati da sincronizzare, visto che questa operazione significa un carico per rete e computer.

La **Esecuzione:** della funzione è impostata in modo fisso a **Server Standby** e non può essere modificata. Essa si verifica solamente se:

- il calcolatore è il Server-Standby
- il server online è
- il progetto è un progetto di rete
- sono stati selezionati driver per la sincronizzazione

PER MODIFICARE LA SELEZIONE DEL DRIVER

- cliccate nelle proprietà del driver sul gruppo **Generale** e sulla proprietà **Parametri**
- Si apre il dialogo per la selezione dei driver.

5.3 zenon Logic Workbench

La **Simulazione - programmata** viene programmata nel workbench zenon Logic, che collabora in modo stretto con l´Editor di zenon. Per la simulazione (a differenza della soluzione integrata) vengono scambiate fra gli editor solamente le variabili di un driver.

DEFINIZIONE DELLE VARIABILI DA SCAMBIARE

Al momento dell´avvio del workbench di zenon Logic, tutte le variabili del driver che non sono ancora esistenti in zenon Logic vengono impostate come variabili globali. Per queste ultime è sempre attiva automaticamente l´impostazione **Rendere simbolo indipendente**.

VISIBILITÀ

Se si definisce nel workbench una nuova variabile nell´area **globale** oppure **ritentiva**, essa è visibile solamente in zenon Logic. Per renderla visibile anche in zenon, deve essere attivata l´opzione **Rendi indipendenti simboli**. A partire da questo momento, la variabile viene attualizzata da entrambi gli editor.

Nel caso in cui si disattiva per la variabile l´opzione **Rendere simbolo indipendente**, la variabile viene cancellata nell´editor di zenon. Questo comportamento del sistema permette di definire in zenon Logic delle variabili **globali** oppure **ritentive**, che non sono visibili per zenon.

Per una variabile creata con il workbench di zenon Logic in zenon, il sistema prova l´assegnazione all´area **SPSMERKER**. Se però il zenon Driver (A pagina: 8)non supporta il tipo di dato configurato in essa, il sistema cerca la prima area dati che supporta questo tipo di dato. Se il sistema non trova nessun´area, la variabile non può essere impostata.

Una variabile è bloccata solamente se si ha un **Progetto multiutente** e quest´ultimo non possiede ancora lo stato `Abilitare le modifiche`.

IMPOSTAZIONI DI PROGETTO

Alcune impostazioni sotto Progetto -> Parametri-progetto.... -> Altre opzioni -> Estese in zenon Logic hanno influenza sulla funzionalità della simulazione.

Impostazioni Estese	Descrizione
Runtime: Stato di esecuzione	Raccomandazione: **Modalità**: con trigger **Tempo di ciclo**: da 1/4 a 1/3 del tempo necessario all´attualizzazione del driver Delle modifiche di valore troppo frequenti di variabili simulate portano a sottoporre il computer ad un carico elevato
Compiler/Opzioni: Rendere indipendenti simboli di tutte le variabili	Raccomandazione: disattivare. `Attiva:` i dati per una variabile vengono scambiati visto che l´assegnazione avviene nel runtime mediate i nomi simbolo. Ma: le modifiche alla progettazione delle variabili non vengono attualizzate fra zenon e zenon Logic. non è più possibile l´online change.
Compiler/Opzioni: mantenere la scrittura in lettere maiuscole/minuscole dei simboli.	Senza conseguenza: nel runtime, i nomi variabili vengono sempre convertiti nella scrittura in lettere maiuscole.
Compiler/Opzioni: impostare bit di stato per variabili con profilo.	`Attivo` Nel programma di simulazione lo stato delle variabili viene simulato o il sistema reagisce ad esso.
„C" Compiler	Nativer Code viene supportato. Nome del DLL che deve contenere il codice compilato: **`t5app.dll`**.

5.4 Runtime

5.4.1 Funzionalità

Il runtime di zenon Logic mette a disposizione le seguenti funzionalità.

- Trasferimento file
- Tipi di dato:
  - BOOL
  - BYTE
  - DINT
  - DWORD
  - INT

- LINT
- LREAL
- REAL
- SINT
- STRING
- TIME
- UDINT
- UINT
- USINT
- WORD

- Array di zenon (vedi anche Array e simulazione (A pagina: 21))
- Dato relativo alla data/ dato ora
- Registrazione valori digitale
- Blocchi funzione collegati in modo dinamico, reperibili in `t5block*.dll`
- Hot restart
- Logging:
  - Segnalazioni Trace del programma **`(printf)`**
  - Segnalazioni errore del runtime
  - tutte su workbench e server di diagnosi
- Codice macchina: Native compiled code con C-Postcompiler (non disponibile per Windows CE)
- Online change
- Linguaggi di programmazione: AWL, AS, FUP, KOP, ST
- Dati Retain (dati ritentivi)
- Blocchi funzioni e features speciali:
  - serializzazione dati
  - Occupazione di memoria dinamica
  - Ricette indipendenti
  - Lista variabile indipendente
  - File Management
  - Files
  - Mathe
  - Random

- Segnali
- Tabelle stringe
- Trigonometria
- Ora

▶ Comunicazione spontanea:

- Cambio valori come evento
- 16 collegamenti
- Isteresi

▶ UDP & TCP/IP per linguaggi IEC

▶ Variabili - Interblocco

5.4.2 Array e simulazione

Per simulare array multidimensionali nel contesto di una simulazione driver, dovete attivare nel workbench di zenon Logic la proprietà "Salva variabili complesse in segmento proprio".

ARRAY CON INDICE DI PARTENZA 1

Se, nell´ambito di una simulazione, ci sono degli array con indice di partenza 1, essi vengono impostati nel workbench di zenon Logic con un indice in più.

Esempio

Un array dispone in zenon degli indici da 1 a 4. Questo tipo di array ha, invece, nel workbench di zenon Logic gli indizi da 0 a 4.

In tal modo si assicura che in zenon e in zenon Logic vengano usati gli stessi indici. L´indice 0 non viene trasmesso al driver.

Se cambiate degli array nel workbench di zenon Logic, dovete tener presente questo ulteriore indice.

5.4.3 File Runtime

Al momento della creazione dei file runtime, viene generata nel **Cartella Runtime** del progetto nel percorso `\RT\FILES\zenon\custom\drivers` una sottocartella recante il nome del progetto di simulazione. In questa cartella viene depositato il file `simulrt.cod` con il codice della simulazione.

Nella caso che la compilazione nel workbench e la trasmissione nella cartella runtime abbiano avuto successo, nella finestra di emissione dell´editor apparirà la seguente segnalazione:

SIMULRT.COD

E se si compila con C-Compiler, apparirà inoltre:

T5APP.DLL

TRASPORTO REMOTO

I file runtime per la simulazione vengono generati come sottocartella dei file del driver. Nel caso del Trasporto Remoto, la sottocartella viene trasmessa con tutti i file in essa contenuti al computer di destinazione.

5.4.4 Scambio dati

Lo scambio dati provenienti dal Runtime di zenon Logic verso il driver inizia con il secondo ciclo del Runtime di zenon Logic. In tal modo, il primo ciclo è a disposizione per inizializzare tutte le variabili nel runtime.

Uno scambio ha luogo solamente se:

- la variabile viene richiesta e:
  - non ha ancora ricevuto nessun valore.
  - nel driver non ha lo stato `Spento` (OFF/_VSB_N_UPD) oppure `Valore di riserva` (ALT_VAL/_VSB_AVALUE)
  - è ancora usato dal driver.
  - sono cambiati valore o stato.

Lo scambio dati in direzione del driver avviene in modo asincrono rispetto al ciclo del runtime. Il driver passa in rassegna le modifiche dei valori una volta per ciclo di update. Tutti i valori modificati - fatta eccezione per quelli con lo stato `Spento` (OFF/_VSB_N_UPD) oppure `valore di riserva` (ALT_VAL/_VSB_AVALUE) - vengono inseriti nella lista delle variabili da attualizzare nel driver e vengono registrati durante il runtime.

FUNZIONE `TRASMETTI DRIVER-IMMAGINE DI SIMULAZIONE ALLO STANDBY`

Questa funzione viene eseguita solo sul server di standby (per la configurazione vedi capitolo Allineamento ciclico (A pagina: 16)). Sul server che mette insieme l´immagine, i dati vengono presi in modo sincrono dal driver. Il driver ha 5 secondi a disposizione. Visto che i dati per l´immagine sono creati nello stesso Thread che è attivo nella simulazione, ci si deve assicurare che la simulazione stessa venga espletata nel giro di 5 secondi. Nel caso in cui si superi questo intervallo di tempo, non viene trasmessa nessuna immagine per questo driver.

SIMULAZIONE VALORE

Le modifiche di valore non vengono trasmesse subito, ma sono depositati un un buffer. La grandezza del buffer è pari a 8192 modifiche valore, oppure al quintuplo del numero delle variabili, a seconda di quale valore è più grande. Il ritardo della trasmissione a causa del ciclo del driver può essere di al massimo 100 ms. Se cambia di nuovo il valore di una variabile mentre essa attende di essere trasmessa, tutti i valori non ancora trasmessi durante il runtime vengono depositati subito; solo dopo si prende nota dei nuovi valori. Ciò può provocare un carico più elevato per computer e rete, ma, nel contempo, è garanzia del fatto che tutti i valori vengano trasmessi durante il runtime.

DRIVER ORDINE DI SCRITTURA

Se la **`Simuzione - programmata`** è progettata e attiva, i comandi di scrittura sullo standby vengono passati fino al runtime.

Operazioni di scrittura vengono eseguite dal driver in modo asincrono rispetto al runtime. Lo scambio dati ha luogo tramite un buffer. La grandezza del buffer è pari a 8192 modifiche valore, oppure al quintuplo del numero delle variabili, a seconda di quale valore è più grande.

Comandi scrittura vengono passati attraverso il runtime dopo le modifiche di valore. Se un comando scrittura non può essere eseguito in modo corretto, viene visualizzata la segnalazione di errore **`Write queue full! Write command for <DP-Name> lost!`**.

Se i bit di stato sono attivi, il settaggio di `Scrittura avvenuta con successo` indica che il driver ha scritto sul valore delle variabili. Il programma Runtime dovrebbe cancellare questo stato dopo aver trattato il comando scrittura. Un bit di stato già attivo ritarda una nuova operazione di scrittura del driver. I comandi di scrittura vengono eseguiti anche quando il programma Runtime è stato interrotto tramite un breakpoint.

Nota: non tutti i comandi scrittura del driver devono essere visibili nel runtime. Se per una variabile ci sono nel buffer più comandi scrittura, è visibile nel programma runtime solamente l´ultimo.

DISATTIVA VARIABILE, PASSA A VALORE SOSTITUTIVO.

Se la variabile viene disattivata oppure se si passa al valore sostitutivo, ciò non è riconoscibile per il runtime. Delle modifiche di valore che vengono eseguite mediante il runtime, non vengono assegnate alle variabili nel driver. Se la variabile viene portata di nuovo al valore spontaneo, oppure se la si riattiva, anche il valore attuale proveniente dal runtime viene di nuovo trasmesso al driver. Allo spegnimento del valore spontaneo viene settato il bit di stato OFF; quando si ritorna al valore spontaneo, viene resettato il bit OFF.

5.4.5 Formato dati

DATI RITENTIVI

Dati ritentivi vengono depositati nel file **`SimulRt<TreiberID>.ret`**. Per ogni simulazione viene generato un file del genere, sempre che siano presenti delle variabili ritentive. Si tiene conto di questo file al momento del confronto in rete.

DRIVER RITENTIVO

L´impostazione **`Simulazione - programmata`** all´avvio del driver non è un criterio per restaurare l´immagine variabile ritentivamanente.

5.4.6 Ridondanza

START SUL SERVER-STANDBY

Se il driver è già attivo sul Server-Standby, dovrebbe essere disponibile anche l´immagine per la simulazione. In questo caso, la simulazione viene avviata "hot" con quest´immagine. In caso di mancanza dell´immagine si verifica un warm start.

FUNZIONE `TRASMETTI DRIVER-IMMAGINE DI SIMULAZIONE ALLO STANDBY`

Questa funzione viene eseguita solo sul server di standby (per la configurazione vedi capitolo Allineamento ciclico (A pagina: 16)). Sul server che mette insieme l´immagine, i dati vengono presi in modo sincrono dal driver. Il driver ha 5 secondi a disposizione. Visto che i dati per l´immagine sono creati nello stesso Thread che è attivo nella simulazione, ci si deve assicurare che la simulazione stessa venga espletata nel giro di 5 secondi. Nel caso in cui si superi questo intervallo di tempo, non viene trasmessa nessuna immagine per questo driver.

ALLINEAMENTO

Oltre ai file di programma della simulazione, vengono sincronizzati anche:

- dati ritentivi (*.ret)
- tutti i file *.simul
  Questa parte finale del nome file viene generata dal programma di simulazione per file specifici della simulazione stessa e collocata nella cartella con la denominazione computer sotto a **Cartella Runtime**.

5.4.7 Stato

SIMULAZIONE STATO

Per quello che concerne la simulazione di stato, in zenon Logic alla voceProgetto -> Parametro di progetto -> Estese -> Compiler -> Opzioni -> deve essere attivo Imposta bit di stato per variabil con profilo. Nel caso in cui lo stato di una variabile venga modificato, si verifica la trasmissione della variabile al driver.

La modifica al seguente stato non causa un aggiornamento del valore nel driver e non viene neanche ripresa dalla simulazione:

- `External Real-Time` (T_EXTERN/_VSB_RT_E)
- `Internal Real-Time` (T_INTERN/_VSB_RT_I)
- `Ora solare` (T_STD/_VSB_WINTER)
- `Conferma scrittura` (WR_ACK/_VSB_WR_ACK)
- `Scrittura avvenuta con successo` (WR_SUC/_VSB_WR_SUC)
- `Stato normale` (NORM/_VSB_NORM)
- `Normal deviation` (N_NORM/_VSB_ABNORM)
- `Seleziona nella rete` (NET_SEL/_VSB_SELEC)
- `Runtime exceeded` (TIMEOUT/_VSB_RTE)
- `In costruzione` (PROGRESS/_VSB_DIREC)
- `Spento` (OFF/_VSB_N_UPD)
- `Valore di riserva` (ALT_VAL/_VSB_AVALUE)

> **Informazioni su**
>
> *Il bit di stato `Scrittura avvenuta con successo` (WR_SUC/_VSB_WR_SUC) è disponibile nel programma Runtime può essere resettato per visualizzare la scrittura avvenuta con successo.*
>
> *Tuttavia, il resettaggio non causa di per sé la trasmissione dal Runtime di zenon. Solo i cambiamenti di valore vengono inviati dal driver al Runtime.*

BIT DI STATO TRATTATI IN MODO PARTICOLARE

I seguenti bit di stato si escludono a vicenda:

Bit	Descrizione
17: Spontaneo- (SPONT/_VSB_SPONT)	Viene settato lo stato SPONTAN. Lo stato settato è lo stesso anche se il trattamento di stato è disattivato, oppure non è stato definito alcuno stato. Lo stato GI + INVALID viene cancellato.
18: Non valido- (INVALID/_VSB_I_BIT)	Viene settato lo stato INVALID. Lo stato SPONTAN + GI viene cancellato.
16: General Interrogation (GI/_VSB_GR)	Viene settato lo stato General Interrogation . Lo stato SPONTAN + INVALID viene cancellato.
8: Seleziona nella rete (NET_SEL/_VSB_SELEC)	Se questo bit di stato è attivo (BSO attivazione/disattivazione), per il riconoscimento di una modifica i seguenti bit di stato devono essere pari a 0 e devono essere trasmessi come 0 al driver: ▸ Seleziona nella rete (NET_SEL) ▸ Causa di trasmissione (COTx) ▸ Select (SE_870) ▸ N_CONF (P/N-BIT) ▸ Test-BIT (TEST) ▸ Scrittura avvenuta con successo (WR_SUC)

SELECT BEFORE OPERATE (SBO)

Perché il sistema reagisca a **Seleziona prima di operare**, nel programma runtime deve essere disponibile la simulazione di stato. La procedura per **Seleziona prima di operare** viene fissata nel programma runtime.

ATTIVAZIONE

Se viene inviato al driver un **SBO select**, ciò provoca per il runtime una scrittura di valore con lo stato Seleziona nella rete (NET_SEL) + Causa di trasferimento (COT_act).

Nel programma Runtime deve essere implementata una corrispondente procedura SBO che verrà avviata.

Il bit di stato Seleziona nella rete (NET_SEL) deve essere resettato nel programma Runtime.

DISATTIVAZIONE

La disattivazione causa una scrittura di valore con lo stato Seleziona nella rete (NET_SEL) + Causa di trasferimento (COT_deact).

Nel programma runtime deve essere terminata la procedura SBO.

Il bit di stato Seleziona nella rete (NET_SEL) deve essere resettato nel programma Runtime.

5.4.8 Start/Stopp

PARTENZA DRIVER

L´aggiornamento dei valori inizia sul Server-Standby se il driver ha ricevuto dal suo server l'immagine di processo. L´immagine deve giungere prima della scadenza del timeout del modulo, altrimenti il driver si avvia senza di essa. Se il Server-Standby passa ad essere server prima della scadenza del tempo di attesa, l´attesa stessa termina. I driver che si avviano in un progetto server oppure senza rete, non hanno tempo di attesa. Questo comportamento si verifica sia nel caso di una simulazione che nella modalità hardware.

LANCIARE RUNTIME

Se il driver lancia la **`Simulazione - programmata`**, in un primo passo il sistema carica la DLL con il runtime di simulazione. In seguito viene parametrizzato il runtime e il programma runtime si avvia, sempre che sia stato possibile caricarlo con successo.

Se è disponibile un´immagine del runtime, quest´ultimo viene avviato con essa `Hot`. Altrimenti il sistema esegue un `Warm start`. Se anche quest´ultimo non risulta possibile, si cerca di lanciare il runtime in modalità `Cold Start`. Nel caso in cui non sia disponibile alcun programma valido, il runtime si avvia "fermato". Lo stato di tutte le variabili viene settato su INVALID.

Il programma di simulazione funziona in un proprio thread e, perciò, è completamente indipendente dal ciclo driver.

Informazioni su

Variabili ritentive

Dati ritentivi ricevono solamente il valore delle variabili di zenon Logic, non del loro stato. Per lo start ciò significa quanto segue:

- Warm start Lo stato settato per una variabile viene ripristinato - indipendentemente dal fatto se si tratta di una variabile ritentiva o no.
- Cold start con variabili ritentive: solo il valore delle variabili ritentive in zenon Logic viene ripristinato, non lo stato.

ARRESTA RUNTIME

Se il runtime viene fermato, viene settato per tutte le variabili lo stato INVALID. Le variabili che vengono richieste mentre il runtime è bloccato, ricevono come inizializzazione lo stato INVALID. Il runtime salva tutti i dati per il warm start e l´online change:

- `SIMULRT.HOT`: contiene i dati per il riavvio hot.
- `SIMULRT.UPD`: contiene i dati per l´online change.

Questi due file vengono generati nella cartella del programma runtime. Il runtime deve possedere un´autorizzazione di scrittura relativa a questa cartella.

AGGIORNA PROGETTO RUNTIME

- Nuova compilazione di un progetto di simulazione:
  fa sì che il driver corrispondente venga caricato di nuovo.
- Modifica del timeout di modulo:
  il runtime non la riconosce come modifica e non determina il fatto che il driver venga caricato un´altra volta.

L´implementazione standard nel kit driver provoca un hot start del runtime nel momento in cui il progetto di simulazione viene aggiornato. Se ciò non risulta possibile (per esempio perché il programma è diverso), viene eseguito un cold start con `retain`.

SCARICO DEL RUNTIME

Se il runtime è aperto, esso viene arrestato. Prima di scaricare il runtime, il sistema trasmette tutti gli aggiornamenti di valori in attesa al driver. Solo in seguito il runtime viene reso disponibile e si scarica **`SimulRuntime.dll`** sul PC.

5.4.9 Comandi driver

I comandi driver servono a influenzare i driver tramite zenon, per esempio avviarli e chiuderli.
La progettazione avviene attraverso la funzione **Comandi driver**.

ARRESTA/AVVIA DRIVER

Se il driver viene terminato nella modalità **Simulazione - programmata**, ne consegue la chiusura del runtime. utte le variabili ricevono lo stato **INVALID**. Se so riavvia il driver, la simulazione viene ricaricata e il runtime viene avviato.

DRIVER PASSAGGIO HARDWARE/SIMULAZIONE

Nel cambio fra le modalità Hardware e Simulazione, il driver si comporta secondo il seguente schema:

1. se il driver è stato configurato nell´Editor per la modalità hardware:
   - dopo l´esecuzione della funzione **comando driver** con il parametro Driver in modalità simulazione il driver viene portato nella modalità **simulazione - statica**.
2. Se il driver è stato configurato nell´Editor per una delle modalità di simulazione (statica, conta, programmata):
   - dopo l´esecuzione della funzione **comando driver** con il parametro Driver in modalità hardware il driver viene portato nella modalità hardware e comunica con il controllo.
   - il driver ritorna dopo l´esecuzione della funzione **comando driver** con il parametro Driver in modalità simulazione alla modalità di simulazione configurata.

5.4.10 Assegnazione variabili

L´assegnazione del driver di zenon alle variabili di zenon Logic avviene mediante il nome convertito in lettere maiuscole. Se il runtime di zenon richiede una variabile dal driver, quest´ultimo la trasmette alla simulazione. Se non è caricata nessuna simulazione, la variabile riceve a driver interrotto lo stato spento (OFF/VSB_N_UPD) , altrimenti quellonon valido (INVALID/_VSB_I_BIT) . Le variabili del tipo oggetto-driver **Variabile driver** non vengono mai richieste dalla simulazione.

5.4.11 Data e ora

Per quello che concerne l´uso del dato ora, deve essere attivo in zenon Logic l´inserimento Progetto -> Parametro di progetto -> Esteso -> Compiler -> Opzioni -> Imposta bit di stato per variabil con profilo

Se nel programma runtime viene settata la data per una variabile, il valore di data e tempo viene usato come time stamp. Ogni volta che cambia la data o l´ora di una variabile per l´ultimo ciclo, si verifica una trasmissione dei valori al driver. Ciò vale anche nel caso in cui non siano cambiati né valore né stato.

La maggior parte dei driver trasmettono un nuovo time stamp al runtime solo quando cambiano contemporaneamente anche valore e/o stato. Un´eccezione, però, è rappresentata, per esempio, dai driver IEC870, che trasmettono nuovi time stamps al runtime dall´hardware (oppure **`Simulazione - programmata`**) con lo stesso valore o lo stesso stato.

Informazioni su

Si deve far in modo che sia garantito il fatto che data e ora progrediscano. Se ciò non si verifica, si avranno dei problemi di archiviazione. La data deve avere un valore maggiore/minore a `0`.

Alle variabili che hanno ricevuto uno stamp dal programma runtime viene attribuito lo stato `Tempo reale esterno` (T_EXTERN/_VSB_RT_E).

Tutte le variabili trasmesse al driver che non hanno ricevuto uno stamp dal runtime, ottengono durante il ciclo un time stamp comune. In tal modo si assicura che il tempo corrisponda a quello della modifica. Questi driver ottengono lo stato `Tempo reale interno` (T_INTERN/_VSB_RT_I).

DATA & TEMPO

Per tutte le informazioni concernenti data e tempo vale quanto segue:

- i tempi corrispondono a UTC.
- Tutti i tempo devono essere compresi fra 02. 01. 1970 e 2038.
- La conversione della data in una stringa avviene nel formato YYYY/MM/DD.
- Il tempo viene convertito nel formato HH:MM:SS.

5.5 Indicazioni sulle variabili in progetti di simulazione

Se si vogliono creare e modificare delle variabili in progetti di simulazione, vale quanto segue:

1. zenon Logic

   Se le variabili per un progetto di simulazione vengono create nel workbench di zenon Logic, esse non dispongono ancora di un indirizzamento valido per la comunicazione. Se si passa alla

comunicazione hardware, essa può essere interrotta se le nuove variabili vengono usate in immagini senza un indirizzamento adeguato in modo corretto.

2. Rinomina

   Se una variabile di un progetto di simulazione viene rinominata, cambia in modo corrispondente anche il nome della variabile in zenon. Così, ad esempio, i nomi delle variabili che vengono usate in un progetto VBA, devono essere adeguate anche nel codice VBA.

3. Soluzioni integrata

   Variabili di soluzioni integrate come zenon Logic, IEC 870 oppure IEC 850, non possono essere rinominate nel progetto di simulazione. In questo caso, la variabili perdono le loro informazioni di progetto.
   Ad esempio: se nel progetto di simulazione **`Project0/Global/NewVar`** viene rinominata in **`myNeVariablenwVar`**, essa diviene in zenon **`myNewVar`**. Dopo il cambiamento di nome, la comunicazione con il runtime di zenon Logic è impossibile.

4. La comunicazione si basa sul nome variabile

   Le variabili di un driver che comunica sulla base di nomi variabili, non possono essere rinominate nel progetto di simulazione.

5. Cancella o importa variabili

   Quando si cancella o importa una variabile mentre in zenon Logic è attivo **`online change`**, in zenon Logic vengono eliminate e poi inserite di nuovo tutte le variabili.

6. Variabili con simbolo "`embed`" attivo

   Le variabili di un programma il cui simbolo "`embed`" è stato attivato, non sono disponibili tramite l´interfaccia Shared Memory in zenon. Quando una variabile con simbolo"`embed`" attivo viene spostata dall´area globale ad un programma, essa viene cancellata in zenon.

5.6 Segnalazioni d'errore

Le segnalazioni di errore nell´Editor di zenon:

Messaggio di errore	Causa e soluzione
Il nome selezionato è già utilizzato da un altro driver o contiene caratteri non validi. Selezionare un altro nome.	Il nuovo nome del progetto di simulazione viene usato già nel progetto oppure non è valido. ▶ Assegnate un nome valido.
Il progetto di simulazione del driver **<Nome file>/<Denominazione>**non è compilato e perciò non può essere usato! Compilare il progetto nel Workbench.	Per il driver è a disposizione un progetto di simulazione. Esso, però, non è stato compilato. ▸ avviate il workbench di zenon Logic con il progetto di simulazione di questo driver. ▸ compilate il progetto Suggerimento: se la simulazione è impostata nelle impostazioni driver su **Simulazione - programmata**, non ha alcun effetto
Write queue full! Write command for <Name Variable> lost!	Indica la perdita di un comando scrittura.