



SHINYX 2.0 SP1

MANUALE OPERATIVO

Tool di conversione per driver SFSIEETH, SFNETLINK E MODBUS TCP/IP

Dichiarazioni di marchio registrato

Automa, SUPER-FLASH, SFW, WRUNFILE, SHINYX sono marchi registrati di Automa srl.

SIEMENS, S7 sono marchi registrati di SIEMENS.

Modbus, MODBUS-TCP sono marchi registrati di Schneider Automation.

Tutti gli altri marchi registrati non esplicitamente dichiarati, sono di proprietà delle rispettive società.

INDICE

1	Introduzione	1
2	Procedura per la conversione della comunicazione	2
3	Utilizzo e funzionamento di SHINYX	4
4	Tabelle di riferimento per conversione indirizzi	6
4.1	Siemens S7 H1/MPI(Applicom) a SFSIEETH.....	6
4.2	Siemens S7 H1/MPI(Applicom) a SFNETLINK.....	7
4.3	S7- HMI Adapter a SFSIEETH/SFNETLINK	7
4.4	Siemens Progr/P3964 a SFSIEETH/SFNETLINK	8
4.5	Siemens S7 200 a SFSIEETH/SFNETLINK	9
4.6	MODBUS a MODBUS TCP/IP	10
4.7	MODBUSTCP (Driver MicroC) a MODBUS TCP/IP	11
5	Note alle versioni	12
5.1	Versione 2.0 SP1 (Aprile 2017).....	12

1 Introduzione

SHINYX è un tool che facilita la conversione della modalità di comunicazione con alcune tipologie di periferiche, automatizzando alcune delle operazioni necessarie nel caso di migrazione verso l'utilizzo dei driver SFSIEETH, SFNETLINK e MODBUS TCP/IP all'interno di applicazioni già realizzate.

Le conversioni che SHINYX supporta sono elencate nella tabella che segue:

Periferica sorgente	Periferica destinazione
Siemens S7-H1 (Applicom) Siemens S7-MPI (Applicom) Comunicazione tramite schede o software SUPER-FLASH Applicom.	SFNETLINK (Link Driver) SFSIEETH (Link Driver)
S7-MHI ADAPTER (Link Driver) Comunicazione tramite HMI Adapter di Siemens o Helmholz.	
Siemens Progr. Siemens Progr. CP945 Siemens P3964 con RK512 Siemens P3964R con RK512 Comunicazione seriale RS232/RS485.	
Siemens S7-200 Comunicazione seriale RS232/RS485.	
SFNETLINK (Link Driver) Comunicazione tramite Gateway Ethernet NetLINK-MPI di Hilscher.	
MODBUS Comunicazione seriale RS232/RS485.	MODBUS TCP/IP (Link Driver)
Comunicazione ethernet tramite periferiche: MODBUSTCP 1.0 (MicroC Driver) Comunicazione ethernet.	

2 Procedura per la conversione della comunicazione

Per cambiare la modalità di comunicazione passando all'utilizzo di Link Driver come SFSIEETH, SFNETLINK o MODBUS TCP/IP, le operazioni che devono essere effettuate dal programmatore sono:

1. Installazione del driver su PC
2. Installazione del driver nell'applicativo esistente
3. Inserimento del driver nell'archivio Link Driver
4. Conversione degli indirizzi delle variabili
5. Predisposizione del file .INI di configurazione del driver
6. Verifica dell'utilizzo di funzioni MICROC di cambio indirizzo
7. Rimozione dei software/hardware precedentemente utilizzati (opzionale)

SHINYX agisce esclusivamente sui punti 3 e 4.

Le altre operazioni sono a cura del programmatore e devono essere eseguite in rigida sequenza; in breve:

1. Installazione del driver su PC

Il software installatore del driver copia i file in una cartella sul PC.

2. Installazione del driver nell'applicativo esistente

Seguendo le indicazioni contenute nel manuale del driver il programmatore copia i file del driver nella cartella dell'applicativo.

3. Inserimento del driver nell'archivio Link Driver

4. Conversione degli indirizzi delle variabili

Le due operazioni sono effettuate da SHINYX secondo la modalità operativa descritta nel paragrafo [Utilizzo e funzionamento di SHINYX](#) a pag. 4.

Nel caso di variabili con indirizzi non compatibili o non supportati dalla periferica di destinazione la conversione non verrà effettuata e SHINYX, in questa situazione, si limiterà a riportare quante e quali sono le variabili non convertite; il programmatore dovrà quindi modificarle manualmente nel sistema di sviluppo.

In alcuni casi la conversione potrebbe avere soluzioni diverse in base a fattori che non è possibile determinare in modo automatico; in questa situazione SHINYX effettua la conversione adottando la soluzione più standard ma segnalerà al programmatore la necessità di una verifica.

Se l'esito di SHINYX è stato positivo è necessario aprire il progetto con il Sistema di Sviluppo, salvarlo e procedere con le restanti operazioni di conversione.

5. Predisposizione del file .INI

Deve essere predisposto il file .INI (SFSIEETH.INI, NETLINK.INI o MODBUS.INI) così come documentato nel manuale operativo del driver; ogni periferica va definita come equipaggiamento identificato da un numero. In fase di conversione SHINYX assegna in automatico un numero di equipaggiamento utilizzando il numero di nodo (dove presente) o del canale utilizzato nella periferica iniziale. Nel caso di configurazioni multi-nodo su più canali e con numero di nodo coincidenti sarà necessario riassegnare il numero di equipaggiamento.

Per informazioni aggiuntive fare riferimento al paragrafo [Tabelle di riferimento per conversione indirizzi](#) a pag. 6.

6. Verifica dell'utilizzo di funzioni MicroC di cambio indirizzo

Verificare l'uso di funzioni `vmc_set_address()` e `set_address_var()` nei programmi MICROC e modificare la composizione dell'indirizzo.

Fare riferimento al paragrafo [Tabelle di riferimento per conversione indirizzi](#) a pag. 6

7. Rimozione dei software/hardware precedentemente utilizzati

L'operazione di rimozione varia in funzione del dispositivo precedentemente utilizzato; si tratta di un'operazione opzionale non necessaria per il funzionamento dell'applicativo con il nuovo driver:

- schede o software Applicom: se presenti, devono essere disinstallati dal PC il software Applicom, la scheda Applicom e l'eventuale driver Sentinel.
- HMI Adapter di Siemens o Helmholtz: deve essere rimosso, nella configurazione dell'applicativo, il driver S7-HMI ADAPTER e devono essere cancellati i file S7HMI.DLL e S7HMI.PAR.
- gateway Ethernet NetLINK-MPI di Hilscher: deve essere rimosso, nella configurazione dell'applicativo, il driver SFNETLINK e devono essere cancellati i file SFNETLNK.DLL e NETLINK.INI.
- MODBUSTCP (Driver MicroC): deve essere rimosso, nella configurazione dell'applicativo, il/i driver MODBUSTCP a 16 bit e MODBUSTCP a 32 bit

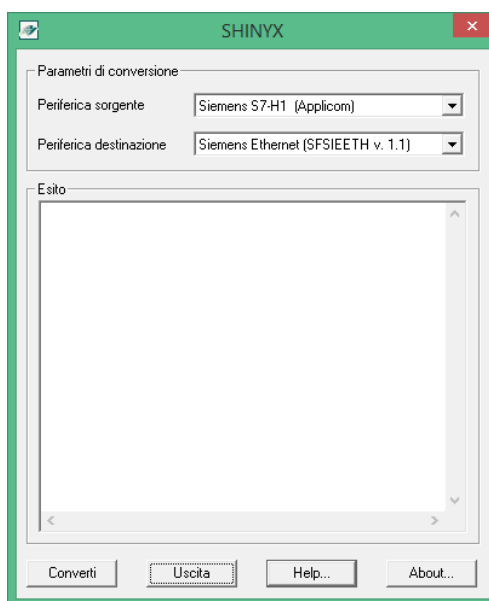
3 Utilizzo e funzionamento di SHINYX

Per utilizzare SHINYX occorre:

1. copiare SHINYX.exe nella directory dell'applicativo
2. chiudere il progetto se è aperto nel Sistema di Sviluppo

Il tool può essere utilizzato con applicativi SUPER-FLASH sviluppati con la versione 3.2 e successive; l'esecuzione di SHINYX non implica l'aggiornamento dell'applicativo, che mantiene quindi la medesima versione con cui è stato originariamente sviluppato.

Lanciando il tool, si presenta una semplice finestra di interfaccia in cui occorre scegliere la "Periferica sorgente" e la "Periferica destinazione" quindi bisogna avviare la conversione con il pulsante "Converti".



Avviando la conversione, le operazioni che SHINYX compie sono:

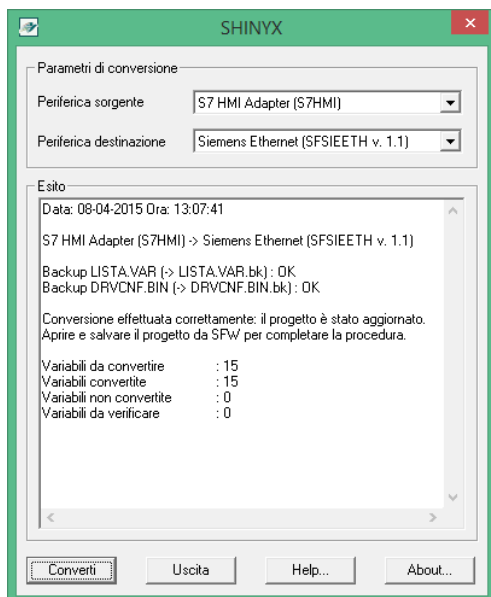
1. crea un backup dei file dell'archivio variabili e dell'archivio Link Driver (rispettivamente LISTA.VAR.bk e DRVCNF.BIN.bk). In caso di necessità è possibile quindi ripristinare, rinominando manualmente i file, la situazione originale. Questa operazione avviene solo al primo lancio del tool, prima di effettuare le operazioni di conversione.
2. inserisce il driver nell'archivio Link Driver, verificando che non sia già presente
3. cerca tutte le variabili che hanno come riferimento la periferica sorgente, se l'indirizzo è compatibile sostituisce la periferica e converte l'indirizzo.

Al termine della conversione, è possibile vedere l'esito dell'operazione nell'apposito riquadro in cui sono indicati:

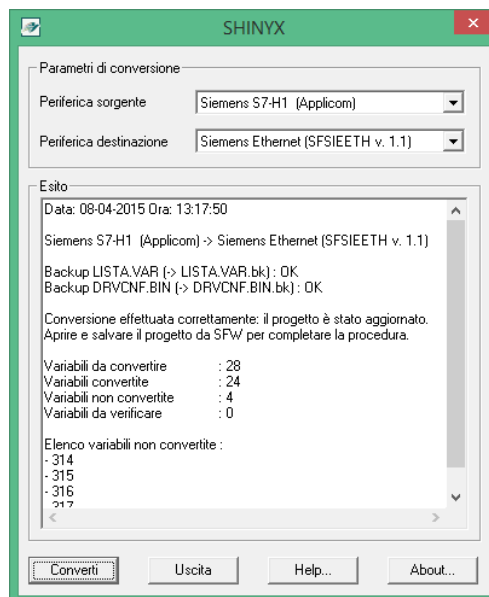
- il numero totale di variabili rilevate da convertire (cioè che utilizzano la periferica sorgente)
- il numero delle variabili convertite correttamente
- il numero di variabili non convertite
- il codice delle variabili non convertite, qualora ce ne fossero: il programmatore dovrà modificare manualmente le variabili
- il numero di variabili convertite da verificare
- il codice delle variabili da verificare, qualora ce ne fossero: il programmatore dovrà verificare e decidere se mantenere o modificare la conversione effettuata
- un messaggio che segnala l'avvenuto aggiornamento o meno del progetto

Nei casi di variabili non convertite o da verificare il programmatore può fare riferimento alle indicazioni contenute nella tabella di riferimento per la conversione dei protocolli interessati.

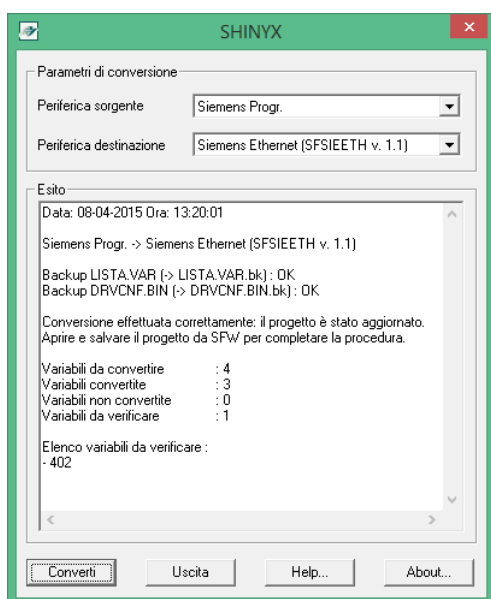
Le immagini che seguono indicano i possibili esiti dell'operazione di conversione:



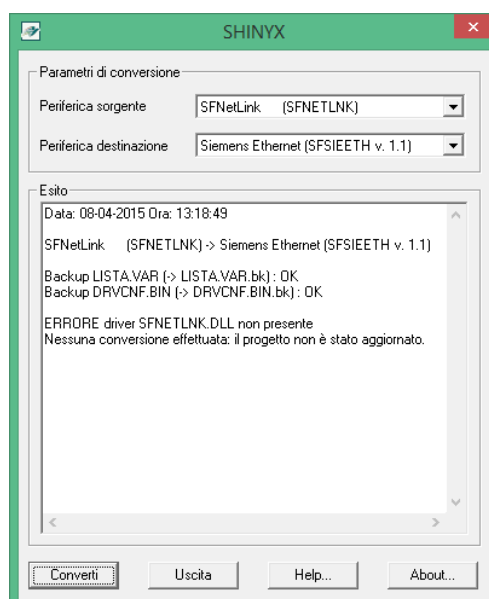
Conversione completa con esito positivo: tutte le variabili sono state convertite



Conversione parziale con esito positivo: 24 variabili su 28 sono state convertite



Conversione con esito positivo con una variabile da verificare.



Conversione con esito negativo: non è stata trovata alcuna variabile da convertire

L'esito dell'ultima conversione viene salvato in un file testo denominato SHINYX.log; il file di logger SHINYX_HISTORY.log memorizza tutte le conversioni eventualmente effettuate nell'applicativo.

Se l'esito è stato positivo, ovvero se il progetto è stato aggiornato, è necessario aprire il progetto con il Sistema di Sviluppo, salvarlo e procedere con le restanti operazioni di conversione elencate nel precedente paragrafo.

4 Tabelle di riferimento per conversione indirizzi

Per comporre il nuovo indirizzo fare riferimento alle tabelle di conversione che seguono.
In alcuni casi la conversione dell'indirizzo è influenzata anche dalla dimensione della variabile;
dove indicata, per brevità, si utilizzano i seguenti identificativi:

DIM16 : Singola word (16 bit), Data, Messaggio
DIM32 : Doppia word (32 bit), ora
BOOL : Booleana
ALFA : Alfanumerica
BLOCCO : Blocco

4.1 Siemens S7 H1/MPI(Applicom) a SFSIEETH

Siemens S7-H1 (Applicom) Siemens S7-MPI (Applicom)	SFSIEETH
Vecchio indirizzo	Nuovo Indirizzo
cc\eee\MM\y.z	M\0\eee\0\My.z (1)
cc\eee\DM\x\y.z	M\0\eee\0\DBx.y.z (1)
cc\eee\IM\y.z	M\0\eee\0\Iy.z (1)
cc\eee\OM\y.z	M\0\eee\0\Oy.z (1)
cc\eee\MB\y	B\0\eee\0\My
cc\eee\DB\x\y	B\0\eee\0\DBx.y
cc\eee\IB\y	B\0\eee\0\Iy
cc\eee\OB\y	B\0\eee\0\Oy
cc\eee\MW\y	W\0\eee\0\My
cc\eee\DW\x\y	W\0\eee\0\DBx.y
cc\eee\IW\y	W\0\eee\0\Iy
cc\eee\OW\y	W\0\eee\0\Oy
cc\eee\MD\y	D\0\eee\0\My
cc\eee\DD\x\y	D\0\eee\0\DBx.y
(1) Il driver SFSIEETH v. 1.1 supporta solo i bit singoli; la conversione quindi non viene effettuata per le variabili con dimensione Blocco o Alfanumerica.	

4.2 Siemens S7 H1/MPI(Applicom) a SFNETLINK

Siemens S7-H1 (Applicom) Siemens S7-MPI (Applicom)	SFNETLINK
Vecchio indirizzo	Nuovo Indirizzo (2)
cc\eee\MM\y.z	Non convertibile (1)
cc\eee\DM\x\y.z	Non convertibile (1)
cc\eee\IM\y.z	Non convertibile (1)
cc\eee\OM\y.z	Non convertibile (1)
cc\eee\MB\y	B\0\eee\0\My
cc\eee\DB\x\y	B\0\eee\0\DBx.y
cc\eee\IB\y	B\0\eee\0\Iy
cc\eee\OB\y	B\0\eee\0\Oy
cc\eee\MW\y	W\0\eee\0\My
cc\eee\DW\x\y	W\0\eee\0\DBx.y
cc\eee\IW\y	W\0\eee\0\Iy
cc\eee\OW\y	W\0\eee\0\Oy
cc\eee\MD\y	D\0\eee\0\My
cc\eee\DD\x\y	D\0\eee\0\DBx.y
<p>(1) Il driver SFNETLINK non supporta la gestione di risorse a bit.</p> <p>(2) Il numero di equipaggiamento eee viene assegnato con lo stesso numero utilizzato in Applicom tranne nel caso dello 0 (zero), in quanto il driver SFNETLINK non lo supporta; in quel caso verrà assegnato con il primo numero libero.</p>	

4.3 S7- HMI Adapter a SFSIEETH/SFNETLINK

S7- HMI Adapter	SFSIEETH SFNETLINK
Vecchio indirizzo	Nuovo Indirizzo (1)(2)
W\n\y\x\D	W\0\n\0\DBx.y
W\n\y\0\M	W\0\n\0\My
W\n\y\0\O	W\0\n\0\Oy
W\n\y\0\I	W\0\n\0\Iy
D\n\y\x\D	D\0\n\0\DBx.y
D\n\y\0\M	D\0\n\0\My
D\n\y\0\O	D\0\n\0\Oy
D\n\y\0\I	D\0\n\0\Iy
<p>(1) Il numero di equipaggiamento viene assegnato come il numero di nodo n del vecchio indirizzo tranne nel caso dello 0 (zero), in quanto il driver SFNETLINK non lo supporta; in quel caso verrà assegnato con il primo numero libero.</p> <p>(2) I driver SFSIEETH e SFNETLINK gestiscono l'indirizzamento a byte: se nel file S7HMI.PAR la macro ADDBYTE non è presente (ovvero era utilizzato l'indirizzamento a word) il nuovo indirizzo deve essere calcolato moltiplicando il precedente per due (Nuovo y = Vecchio y*2) ES. W\1\6\10\D W\0\1\0\DB10.12</p>	

4.4 Siemens Progr/P3964 a SFSIEETH/SFNETLINK

Siemens Progr. Siemens Progr. CP945 Siemens P3964 con RK512 Siemens P3964R con RK512	SFSIEETH SFNETLINK
Vecchio indirizzo	Nuovo Indirizzo (1)(2)
D\x:y (BOOL, DIM16)	W\0\n\0\DBx.y
D\x:y (DIM32)	D\0\n\0\DBx.y (3)
D\x:y (BLOCCO)	W\0\n\0\DBx.y (4)
D\x:y (ALFA)	W\0\n\0\DBx.y o B\0\n\0\DBx.y (5)
<p>(1) Il numero di equipaggiamento n viene assegnato come il numero di canale configurato nella variabile.</p> <p>(2) I driver SFSIEETH e SFNETLINK gestiscono l'indirizzamento a byte: il nuovo indirizzo deve essere calcolato moltiplicando il vecchio per due (Nuovo y = Vecchio y *2). Es. D\10:6 W\0\1\0\DB10.12</p> <p>(3) L'indirizzo viene convertito se il parametro di "Swap 32 bit" è impostato a NO.</p> <p>(4) L'indirizzo viene convertito con il tipo W: se la variabile viene però trattata per derivare delle doppie word l'indirizzo va modificato con il tipo D.</p> <p>(5) Se il parametro "Swap alfanumerica" è impostato a SI va usato il tipo di risorsa W altrimenti B.</p>	

4.5 Siemens S7 200 a SFSIEETH/SFNETLINK

Siemens S7 200	SFSIEETH SFNETLINK
Vecchio indirizzo	Nuovo Indirizzo (1)(2)
n\V\y (DIM16,BOOL)	W\0\n\0\DB1.y
n\V\y (DIM32)	D\0\n\0\DBx.y (3)
n\V\y (BLOCCO)	W\0\n\0\DBx.y (4)
n\V\y (ALFA)	W\0\n\0\DBx.y o B\0\n\0\DBx.y (5)
n\M\y (DIM16,BOOL)	W\0\n\0\M.y
n\M\y (DIM32)	D\0\n\0\My (3)
n\M\y (BLOCCO)	W\0\n\0\My (4)
n\M\y (ALFA)	W\0\n\0\My o B\0\n\0\My (5)
n\I\y (DIM16,BOOL)	W\0\n\0\I.y
n\I\y (DIM32)	D\0\n\0\Iy (3)
n\I\y (BLOCCO)	W\0\n\0\Iy (4)
n\I\y (ALFA)	W\0\n\0\Iy o B\0\n\0\Iy (5)
n\O\y (DIM16,BOOL)	W\0\n\0\O.y
n\O\y (DIM32)	D\0\n\0\Oy (3)
n\O\y (BLOCCO)	W\0\n\0\Oy (4)
n\O\y (ALFA)	W\0\n\0\Oy o B\0\n\0\Oy (5)

(1) Il numero di equipaggiamento n viene assegnato come il numero di nodo del vecchio indirizzo tranne nel caso dello 0 (zero), in quanto il driver SFNETLINK non lo supporta; in quel caso verrà assegnato con il primo numero libero. Nel caso di situazione multi-periferiche su più canali con numeri di nodo condivisi il parametro andrà riassegnato in base alla configurazione fatta nel file di configurazione del driver.

(2) I driver SFSIEETH e SFNETLINK gestiscono l'indirizzamento a byte: il nuovo indirizzo deve essere calcolato moltiplicando il vecchio per due (Nuovo y = Vecchio y *2).
Es. D\10:6 W\0\1\0\DB10.12

(3) L'indirizzo viene convertito se il parametro di "Swap 32 bit" è impostato a NO

(4) L'indirizzo viene convertito con il tipo W: se la variabile viene però trattata per derivare delle doppie word l'indirizzo va modificato con il tipo D.

(5) Se il parametro "Swap alfanumerica" è impostato a SI va usato il tipo di risorsa W altrimenti B.

4.6 MODBUS a MODBUS TCP/IP

MODBUS	MODBUS TCP/IP
Vecchio indirizzo	Nuovo Indirizzo ⁽¹⁾
n\W:y o n\R:y (DIM16,ALFA)	W\0\n\0\Oy
n\W:y o n\R:y (BLOCCO)	W\0\n\0\Oy ⁽³⁾
n\W:y o n\R:y (DIM32)	D\0\n\0\Oy
n\A:y (DIM16,BLOCCO,ALFA)	W\0\n\0\Iy
n\A:y (DIM32)	D\0\n\0\Iy
n\B:y o n\F:y (DIM16,BOOL)	M\0\n\0\Oy
n\D:y (DIM16, BOOL)	M\0\n\0\Iy
n\B:y o n\F:y (BLOCCO)	Non convertibile ⁽²⁾
n\D:y (BLOCCO)	Non convertibile ⁽²⁾

(1) Il numero di equipaggiamento n viene assegnato come il numero di nodo del vecchio indirizzo. Nel caso di situazione multi-periferiche su più canali con numeri di nodo condivisi il parametro andrà riassegnato in base alla configurazione fatta nel file di configurazione del driver.

(2) I blocchi di bit non possono essere convertiti. Il driver MODBUS mette ogni bit in word separate mentre il driver MODBUS TCP/IP usa la modalità packed che prevede l'accorpamento dei bit.

(3) L'indirizzo viene convertito con il tipo W con l'ipotesi che dal blocco vengano derivate esclusivamente delle word 16 bit. Se dal blocco sono derivate delle doppie word l'indirizzo va modificato con il tipo D; se infine il blocco è usato in modo misto per derivare sia word che doppie word sono necessarie ulteriori verifiche.

4.7 MODBUSTCP (Driver MicroC) a MODBUS TCP/IP

MODBUSTCP (Driver MicroC)	MODBUS TCP/IP
Vecchio indirizzo	Nuovo Indirizzo ⁽¹⁾
c\n\OM\0\y (BOOL, DIM16, DIM32)	M\0\n\0\Oy
c\n\SM\0\y (BOOL, DIM16, DIM32)	M\0\n\0\Oy (3)
c\n\IM\0\y (BOOL, DIM16, DIM32)	M\0\n\0\Iy
c\n\OM\0\y (BLOCCO)	Non convertibile (2)
c\n\SM\0\y (BLOCCO)	Non convertibile (2)
c\n\IM\0\y (BLOCCO)	Non convertibile (2)
c\n\OW\0\y	W\0\n\0\Oy
c\n\SW\0\y (DIM16, BLOCCO, ALFA)	W\0\n\0\Oy (3)
c\n\IW\0\y (DIM16, BLOCCO, ALFA)	W\0\n\0\Iy
c\n\OD\0\y (DIM32, BLOCCO)	D\0\n\0\Oy (4)
c\n\SD\0\y (DIM32, BLOCCO)	D\0\n\0\Oy (3)(4)
c\n\ID\0\y (DIM32, BLOCCO)	D\0\n\0\Iy (4)

(1) Il numero di equipaggiamento si assegna come il numero di nodo. Nel caso di situazione multi-periferiche su più canali con numeri di nodo condivisi il parametro andrà riassegnato in base alla configurazione fatta nel file di configurazione del driver.

(2) I blocchi di bit non possono essere convertiti. Il driver MODBUSTCP mette ogni bit in word separate mentre il driver MODBUS TCP/IP usa la modalità packed che prevede l'accorpamento dei bit nei byte.

(3) Se il tipo di risorsa del vecchio indirizzo è di tipo SM/SW/SD, nella configurazione dell'equipaggiamento potrebbe essere necessario specificare il parametro SingleWrite per impostare la modalità di scrittura singola.

(4) Se il parametro ADDRESS32 del profilo di configurazione del driver MicroC è impostato a SI il nuovo indirizzo deve essere calcolato moltiplicando il vecchio per due (Nuovo y = Vecchio y *2).

5 Note alle versioni

5.1 Versione 2.0 SP1 (Aprile 2017)

Problemi risolti:

- Nelle conversioni al driver SFNETLINK corretta l'assegnazione del numero di equipaggiamento per evitare l'utilizzo del numero 0 (zero) (non gestito dal driver).
- In presenza di variabili tutte di dimensione "Blocco" alcune conversioni (es. MODBUS a MODBUS TCP/IP) non vengono eseguite e l'esito è "Nessuna conversione effettuata".