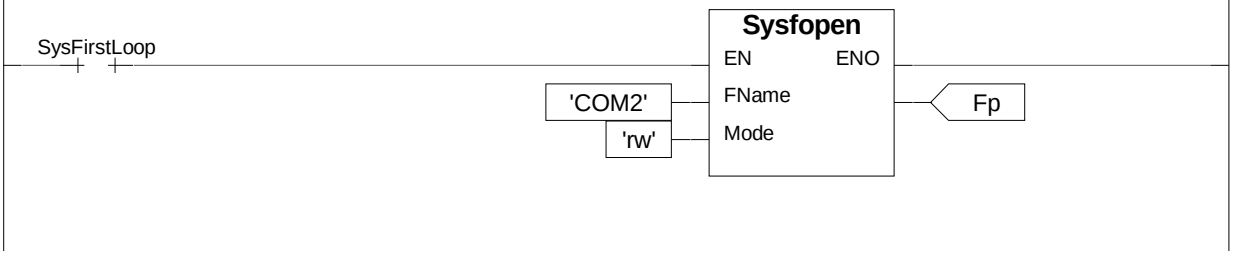
```
VAR_GLOBAL
VIn AT %MX100.16 : ARRAY[ 0..7 ] OF UINT; (* Valori acquisiti da TRP-C68 *)
MdbDone : BOOL; (* Modbus command done *)
END_VAR
```

PROGRAM TRPComm

```
VAR
Error : BOOL; (* Modbus command error *)
VInAddress : DB100AddOffset; (* DB 100 address calculation *)
Fp : FILEP; (* File pointer *)
Mdb : ModbusRTUMaster; (* Modbus master *)
SMode : SetSMode; (* Set serial mode *)
END_VAR
```

1

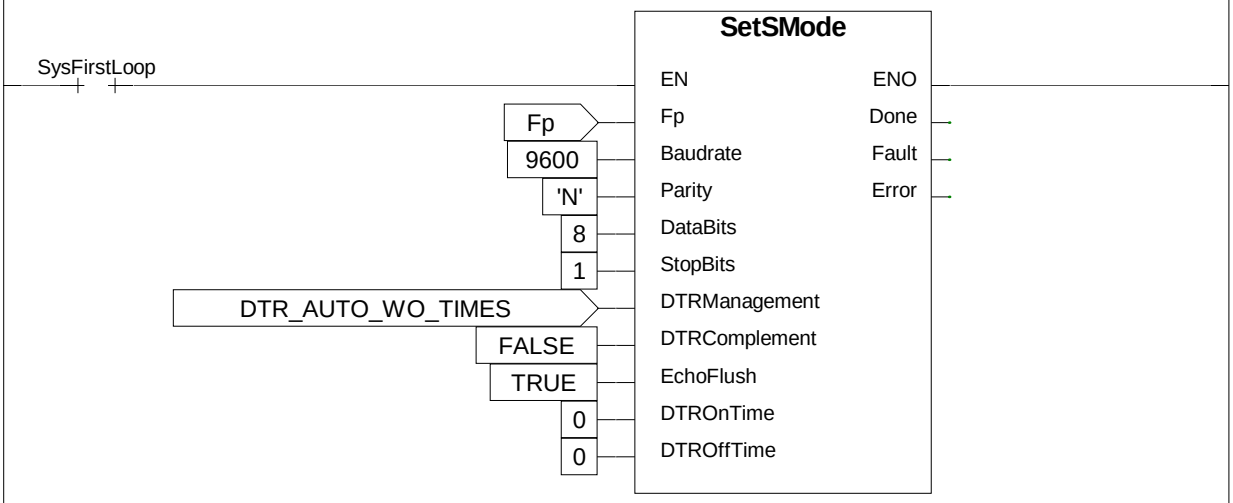
Eseguo set porta seriale da utilizzare



2

Eseguo set parametri di comunicazione

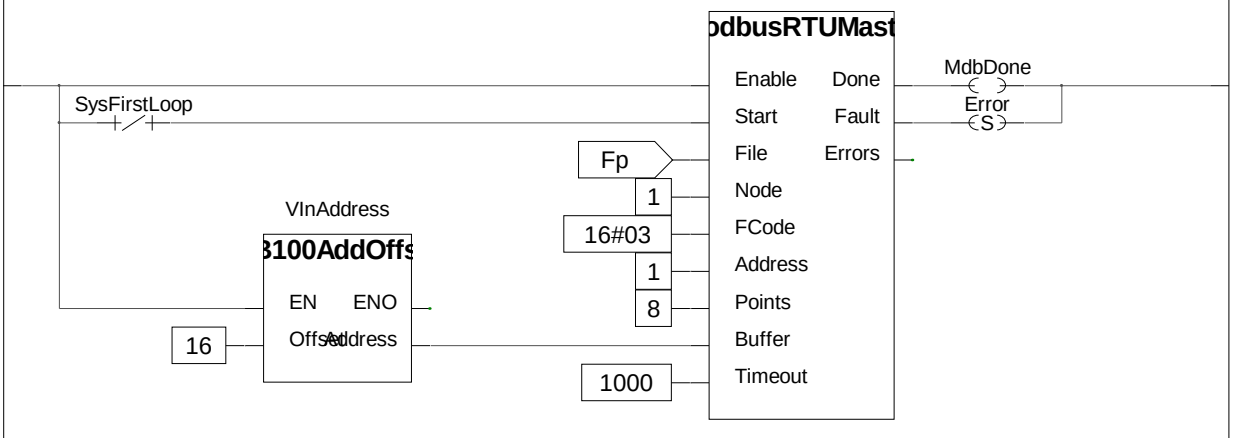
SMode



3

Lettura ingressi da modulo TRP-C68 ed appoggio in memoria MX100.16

Mdb



	Project : TRPAInput	
	PROGRAM : TRPComm	
	Release :	Ver :1.00
	Author :	Date:03/05/2011
	Note :	Page:1 of 1

PROGRAM Conversion

```
VAR
AValue : ARRAY[ 0..7 ] OF INT; (* Valore acquisito ofsettato *)
Volt : ARRAY[ 0..7 ] OF REAL; (* Valore acquisito in volt *)
i : USINT; (* Auxiliary counter *)
END_VAR
```

```
 1 (* ************************************************************************** *)
 2 (* ESEGUO CONVERSIONE VALORI                                                  *)
 3 (* ************************************************************************** *)
 4 (* Il valore analogico letto dal convertitore TRP-C68 è in complemento a 2,   *)
 5 (* -10 Volt:0x0000, 0 Volt:0x8000, +10 Volt:0xFFFF, lo converto in REAL.      *)
 6 (* -------------------------------------------------------------------------- *)
 7
 8     (* ---------------------------------------------------------------------- *)
 9     (* CONVERSIONE VALORI ACQUISITI                                           *)
10     (* ---------------------------------------------------------------------- *)
11     (* Abilitazione, converto solo al termine della acquisizione modbus. *)
12
13     IF NOT(MdbDone) THEN RETURN; END_IF;
14
15     (* Eseguo loop di conversione su tutti i canali acquisiti. *)
16
17     FOR i:=0 TO 7 DO
18         AValue[i]:=VIn[i]-16#8000; (* Valore acquisito ofsettato *)
19         Volt[i]:=TO_REAL(AValue[i])/3276.8; (* Valore acquisito in volt *)
20     END_FOR;
21
22 (* [End of file] *)
23
24
```

	Project : TRPAInput	
	PROGRAM : Conversion	
	Release :	Ver :1.00
	Author :	Date:03/05/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK DB100AddOffset

Returns the DB100 address offset
ENCRYPTED CODE

```
VAR_INPUT
Offset : UINT; (* Address offset *)
END_VAR

VAR_OUTPUT
Address : UDINT; (* Address value *)
END_VAR
```

1

	Project : TRPAInput	
	FUNCTION BLOCK : DB100AddOffset	
	Release :	Ver :1.00
	Author :	Date:03/05/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK SetSMode

Set the serial communication mode
ENCRYPTED CODE

```
VAR_INPUT
Fp : FILEP;
Baudrate : UDINT; (* Baudrate *)
Parity : STRING[ 1 ]; (* Parity type *)
DataBits : USINT; (* Nr of data bits *)
StopBits : USINT; (* Nr of stop bits *)
DTRManagement : USINT; (* DTR management type *)
DTRComplement : BOOL; (* Complement the DTR signal *)
EchoFlush : BOOL; (* Flush the echo *)
DTROnTime : UINT; (* DTR On wait time *)
DTROffTime : UINT; (* DTR Off wait time *)
END_VAR

VAR_OUTPUT
Done : BOOL; (* Execution done *)
Fault : BOOL; (* Execution fault *)
END_VAR
```

1

	Project : TRPAInput	
	FUNCTION BLOCK : SetSMode	
	Release :	Ver :1.00
	Author :	Date:03/05/2011
	Note :	Page:1 of 1

FUNCTION_BLOCK ModbusRTUMaster

Manages the modbus RTU master communication
ENCRYPTED CODE

```
VAR_INPUT
Enable : BOOL;
Start : BOOL; (* Start command *)
File : FILEP; (* Terminal I/O pointer *)
Node : USINT; (* Node number *)
FCode : USINT; (* Function code *)
Address : UINT; (* Start address *)
Points : USINT; (* Number of points *)
Buffer : @USINT; (* Address of data buffer *)
Timeout : UINT; (* Timeout time (mS) *)
END_VAR

VAR_OUTPUT
Done : BOOL; (* Command done *)
Fault : BOOL; (* Command fault *)
Errors : UDINT; (* Error counter *)
END_VAR
```

1

	Project : TRPAInput	
	FUNCTION BLOCK : ModbusRTUMaster	
	Release :	Ver :1.00
	Author :	Date:03/05/2011
	Note :	Page:1 of 1