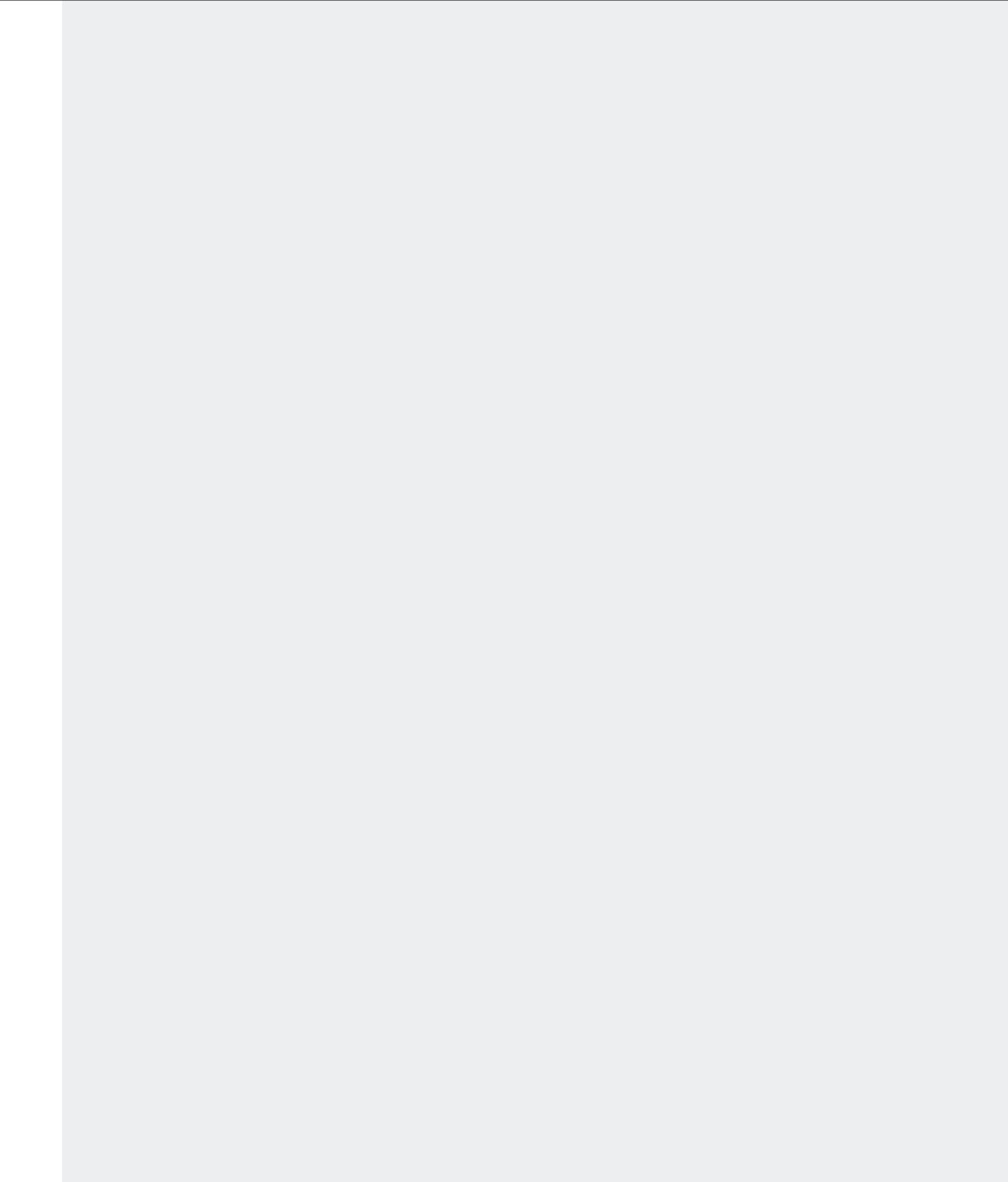


**EB 80** MANUALE D'USO delle funzioni Industry 4.0 Profinet IO  
**EB 80** *USER MANUAL of Industry 4.0 Profinet IO functions*



<b>INTRODUZIONE</b>	PAG. 4
<b>1. CONFIGURAZIONE SISTEMA</b>	PAG. 6
1.1 DESCRIZIONE DELLE FUNZIONI	PAG. 6
1.1.1 Dati di sistema	PAG. 6
1.1.2 Funzione attuatore	PAG. 7
1.1.3 Esempio	PAG. 9
1.1.4 Esempio di come gestire il salvataggio dei dati in un data base PLC	PAG. 10
1.2 LETTURA DEI DATI UTILIZZANDO IL SOFTWARE EB 80 DS	PAG. 12

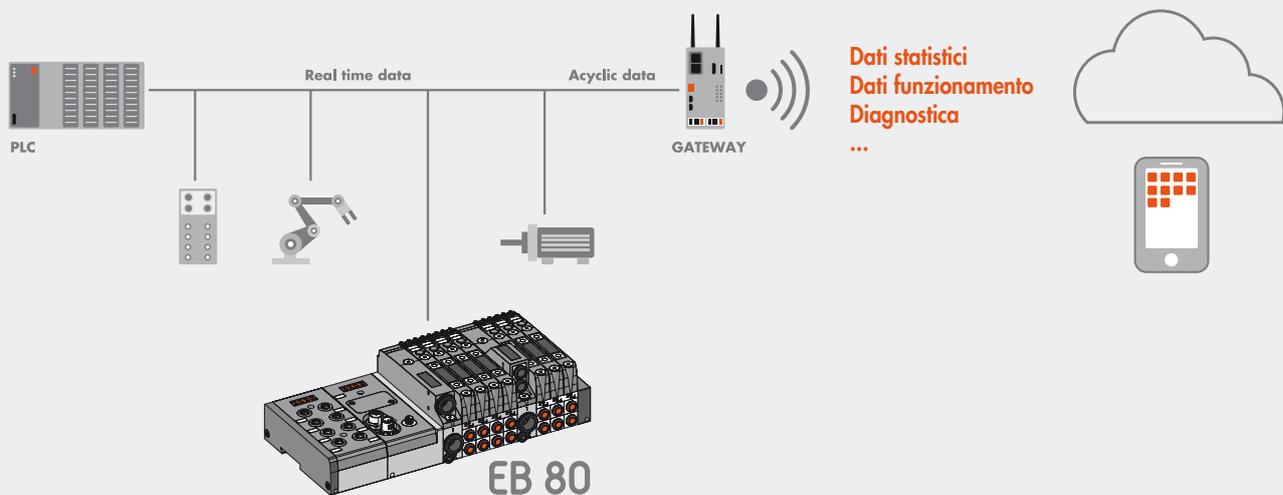


## INTRODUZIONE

I moduli di connessione elettrica permettono l'integrazione di EB 80 con i principali bus di campo presenti sul mercato. In tal modo il sistema di controllo (tipicamente un PLC) può gestire in tempo reale il comportamento dell'isola di elettrovalvole, moduli di segnale compresi.

Con l'introduzione della versione per I4.0 i moduli di connessione per bus di campo inviano alla rete anche i dati storici e diagnostici relativi al comportamento dell'isola (numero di cicli effettuati da ogni elettropilota, tempo totale di attivazione, allarmi, ...) e del circuito pneumatico controllato (tempi di ritardo nella commutazione dei sensori, tempi di movimento degli attuatori, ...).

Anche questi dati sono inviati al sistema di controllo e possono essere gestiti in maniera differente a seconda della situazione: in taluni casi vengono utilizzati in tempo reale, come nel caso degli allarmi di malfunzionamento; in altri possono essere inviati in un'unità di storage, locale o remotata su cloud server, per poi essere analizzati in una fase successiva; in altri casi ancora gli allarmi possono essere inviati ad una stazione di teleassistenza che può monitorare in continuazione lo stato del sistema da remoto.



Le nuove funzioni di diagnostica avanzata di EB 80, denominate EB 80 I4.0, forniscono alla manutenzione tradizionale un potente strumento di analisi, per assicurare un esercizio affidabile, sicuro e duraturo delle unità produttive. Disponibili per tutte le Connessioni Elettriche con Bus di campo e basi identificate da scritta I4.0, integrano una diagnostica avanzata, in completa filosofia Industry 4.0.

Tali funzioni sfruttano la diagnostica nativa di EB 80, integrandola con la capacità di controllo degli IO da parte dell'isola stessa.

Consentono una razionalizzazione ed ottimizzazione della gestione della manutenzione attraverso lo sviluppo della manutenzione predittiva, per:

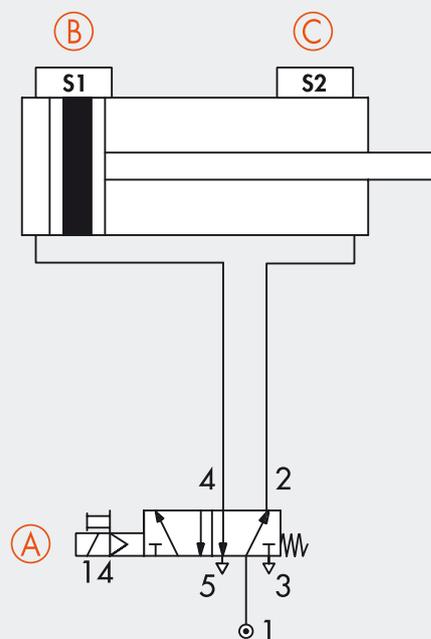
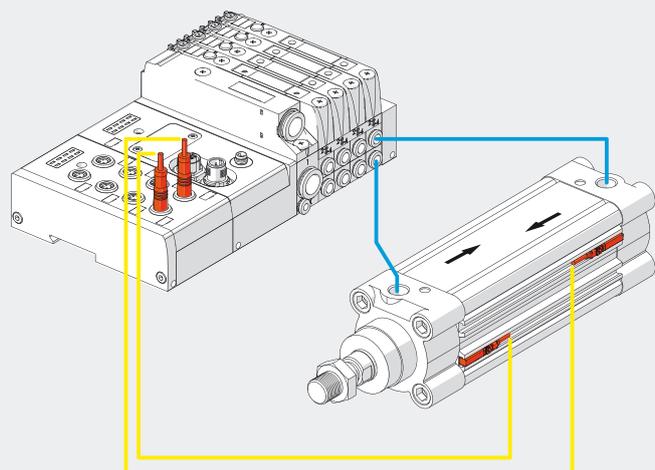
- prevenire i guasti;
- intervenire preventivamente al fine di evitare fermi impianto;
- disporre in tempo reale di tutte le informazioni relative al funzionamento delle apparecchiature;
- monitorare la fine vita dei componenti;
- ottimizzare la gestione dei ricambi in magazzino.

Tutto ciò consente di trasformare i dati raccolti in azioni concrete, utilizzando isole EB 80 standard, senza la necessità di moduli aggiuntivi.

### Descrizione delle funzioni EB 80 I4.0

- Dati di sistema:
  - contatore accensioni del Sistema EB 80;
  - contatore allarmi di alimentazione.
- Dati valvole. Ogni base valvole, per ogni elettro pilota, memorizza in modo permanente i seguenti dati:
  - contatore cicli;
  - contatore del tempo totale di eccitazione dell'elettro pilota;
  - attivazione di una segnalazione di superamento del 60% della vita media;
  - contatore Allarmi Corto Circuito;
  - contatore Allarmi Circuito Aperto.
- Funzioni di controllo del sistema elettropneumatico (dati aggiornati ad ogni ciclo):
  - misura del ritardo tra l'attivazione dell'elettro pilota "A" e l'inizio del movimento dell'attuatore mediante il segnale del sensore "B", con segnalazione di ritardo fuori soglia;
  - misura del tempo di movimento dell'attuatore, attraverso i due sensori associati "B" e "C", con segnalazione di tempo fuori soglia;
  - misura del ritardo tra la disattivazione dell'elettro pilota "A" (o l'attivazione del secondo pilota) e l'inizio del movimento di ritorno dell'attuatore mediante il segnale del sensore "B", con segnalazione di tempo fuori soglia;

- misura del tempo di movimento di ritorno dell'attuatore, attraverso i due sensori associati "B" e "C", con segnalazione di tempo fuori soglia;
- contatore delle corse dell'attuatore.

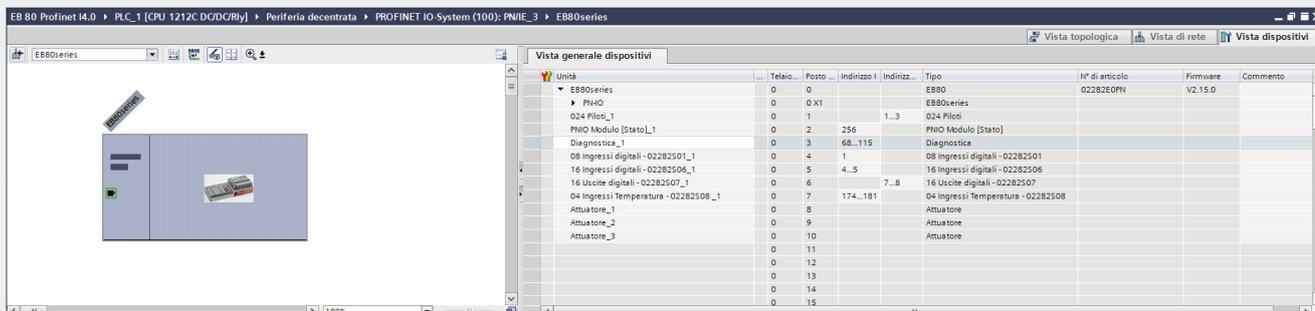


**⚠ ATTENZIONE**

Le funzioni EB 80 I4.0 sono disponibili dalle seguenti versioni software in poi:  
 Profinet 2.16 / Basi valvole 7.03 con logo I4.0 / GSDML-V2.32-Metalwork-EB80-20180115



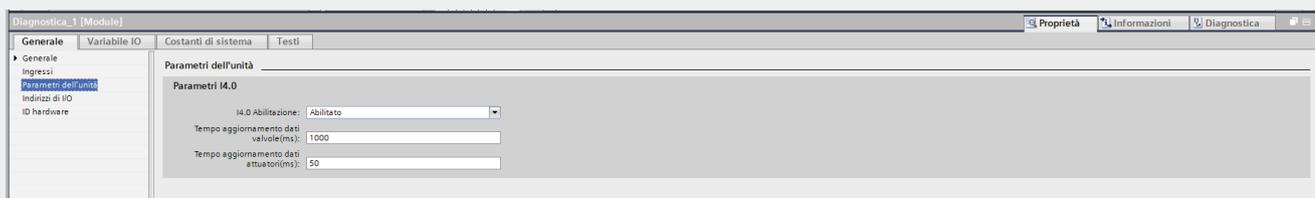
## 1. CONFIGURAZIONE SISTEMA



Configurare il sistema EB 80 con tutti i moduli presenti nel sistema in uso, come descritto nel manuale EB 80 Profinet IO. Successivamente procedere con la configurazione delle funzioni I4.0.

### Abilitazione delle Funzioni

Importare nel sistema dal catalogo hardware – Modulo I4.0, il modulo **Diagnostica**. Nella scheda "Proprietà" selezionare "Parametri dell'unità".



- **I4.0 Abilitazione:** abilita le funzioni diagnostiche
- **Tempo aggiornamento dati valvole:** i dati di sistema e i dati delle valvole memorizzati nelle schede elettroniche delle basi valvole vengono aggiornati con il tempo impostato con questo parametro. Per non sovraccaricare il sistema è consigliabile impostare un tempo maggiore di 1000 ms.
- **Tempo aggiornamento dati attuatori:** i tempi di attuazione e ripristino degli attuatori controllati viene letto con il tempo impostato con questo parametro. Per avere i dati in tempo reale deve essere impostato un tempo breve, per esempio 50 ms.

### 1.1 DESCRIZIONE DELLE FUNZIONI

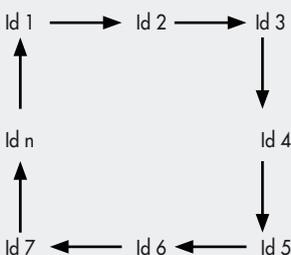
#### 1.1.1 Dati di sistema

**Contatore accensioni:** indica il numero di accensioni del sistema, viene aggiornato ad ogni accensione.

**Contatore allarmi alimentazione:** indica il numero di allarmi dovuti alla tensione di alimentazione fuori dal range ammesso, 10.8 / 31.2 VDC.

#### Dati valvole

Il sistema legge ciclicamente e a rotazione, i dati di ogni elettropilota con il tempo di aggiornamento impostato in "Tempo aggiornamento dati valvole". Il numero della valvola alla quale si riferiscono i dati è indicato con il byte ID valvola. Per non sovraccaricare il sistema è consigliabile impostare un tempo lungo, per esempio in un sistema a 8 valvole con un tempo di aggiornamento di 1000 ms, i dati di ogni valvola vengono aggiornati ogni 8 sec.



**Contatore cicli:** indica il numero di attivazioni effettuato da ognuno degli elettropiloti. Quando il numero di cicli supera il 60% vita media, fissato in 30.000.000 di cicli, si attiva una **segnalazione di superamento della vita media**.

**Contatore del tempo totale di eccitazione dell'elettro pilota:** indica il tempo totale di attivazione di ognuno degli elettropiloti.

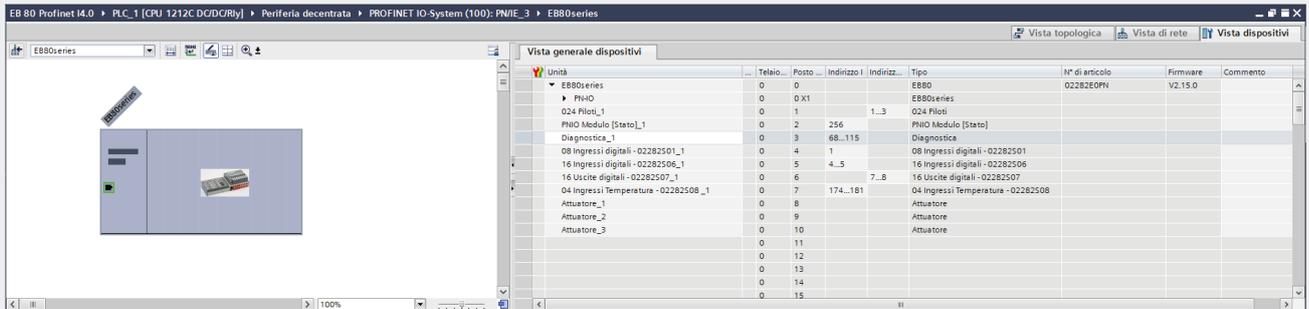
**Contatore Allarmi Corto Circuito:** indica il numero di allarmi per cortocircuito di ogni singolo elettropilota.

**Contatore Allarmi Circuito Aperto:** indica il numero di allarmi per la bobina dell'elettropilota interrotta di ogni singolo elettropilota.

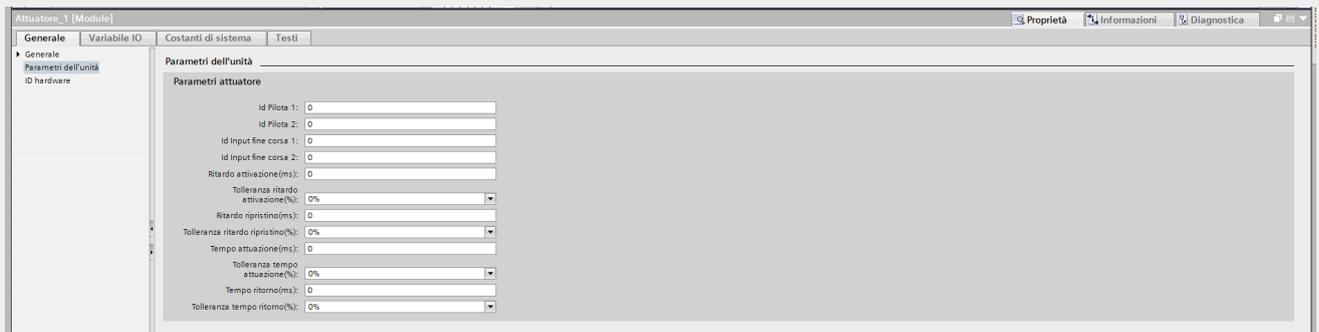
**ATTENZIONE:** il comando di posizioni vuote, come per esempio le false valvole, i bypass, o il secondo pilota di valvole ad un solo pilota, genera un allarme di circuito aperto.

### 1.1.2 Funzione attuatore

Abilita le funzioni di controllo del sistema elettropneumatico. Per l'utilizzo è necessario che nel sistema sia inserito il modulo di segnali Input Digitali. Il sistema consente di inserire fino a 10 moduli Attuatore. Importare nel sistema dal catalogo hardware – Modulo I4.0, il modulo **Attuatore**.



Nella scheda "Proprietà" selezionare "Parametri dell'unità".



#### Parametri attuatore

**Id Pilota 1:** inserire il numero del pilota corrispondente alla valvola associata all'attuatore da controllare.

**Id Pilota 2:** inserire il numero del secondo pilota corrispondente alla valvola associata all'attuatore da controllare, se si utilizza una valvola a due piloti. Nel caso di utilizzo di valvole ad un pilota inserire 0.

Esempio di assegnazione dell'Id piloti

Base per valvole a 8 comandi – è possibile montare valvole a uno o due elettropiloti

Tipo di valvola	Valvola a 2 elettropiloti	Valvola a 1 elettropilota	Falsa valvola o Bypass	Valvola a 1 elettropilota
Elettro pilota 1	14	14	-	14
Elettro pilota 2	12	-	-	-
Uscita	Out 1 Out 2	Out 3 Out 4	Out 5 Out 6	Out 7 Out 8
Id pilota 1	1	3	-	7
Id pilota 2	2	-	-	8

**Id Input fine corsa 1:** inserire il numero di ingresso del modulo di input digitali dove è collegato il primo sensore di fine corsa dell'attuatore.

**Id Input fine corsa 2:** inserire il numero di ingresso del modulo di input digitali dove è collegato il secondo sensore di fine corsa dell'attuatore.

**Ritardo attivazione:** è il tempo di ritardo tra l'attivazione elettrica del pilota 1 e il movimento dell'attuatore, rilevato verificando la disattivazione dell'Input fine corsa 1.

**Tolleranza ritardo attivazione %:** impostare la tolleranza percentuale ammessa per il tempo di attivazione. Il superamento in più o in meno della tolleranza attiva un bit di segnalazione.

**Ritardo ripristino:** è il tempo di ritardo tra la disattivazione elettrica del pilota 1 oppure l'attivazione elettrica del pilota 2 nel caso di valvola a due piloti e il movimento dell'attuatore, rilevato verificando la disattivazione dell'Input fine corsa 2.

**Tolleranza ritardo ripristino %:** impostare la tolleranza percentuale ammessa per il tempo di ripristino. Il superamento in più o in meno della tolleranza attiva un bit di segnalazione.

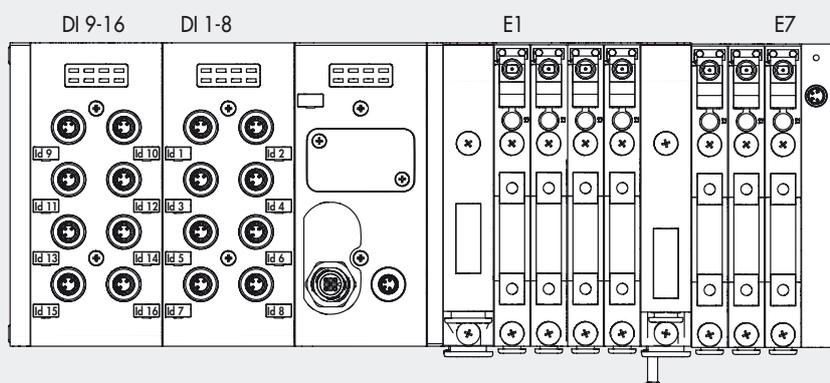
**Tempo attuazione:** misura del tempo di movimento dell'attuatore, attraverso i due sensori associati.

**Tolleranza Tempo attuazione %:** impostare la tolleranza percentuale ammessa per il tempo di attuazione. Il superamento in più o in meno della tolleranza attiva un bit di segnalazione.

**Tempo ritorno:** misura del tempo di movimento di ritorno dell'attuatore, attraverso i due sensori associati.

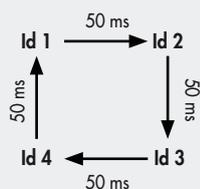
**Tolleranza Tempo ritorno %:** impostare la tolleranza percentuale ammessa per il tempo di ritorno dell'attuatore. Il superamento in più o in meno della tolleranza attiva un bit di segnalazione.

Esempio di assegnazione dell'Id input fine corsa



Il sistema legge ciclicamente e a rotazione, i dati di ogni attuatore con il tempo di aggiornamento impostato in **Tempo aggiornamento dati attuatori**. Il numero dell'attuatore al quale si riferiscono i dati è indicato con il byte **Id attuatore**. Per avere i dati in tempo reale deve essere impostato un tempo breve, per esempio 50 ms.

**Id attuatore:** indica a quale attuatore sono riferiti i dati letti a rotazione ciclicamente. Nel sistema è possibile inserire fino a 10 moduli attuatore. Se per esempio sono inseriti 4 moduli attuatore, con un **Tempo aggiornamento dati attuatori** di 50 ms, l'Id attuatore ruota ciclicamente a intervalli di 50 ms, quindi i dati di ogni attuatore sono aggiornati ogni 200 ms.

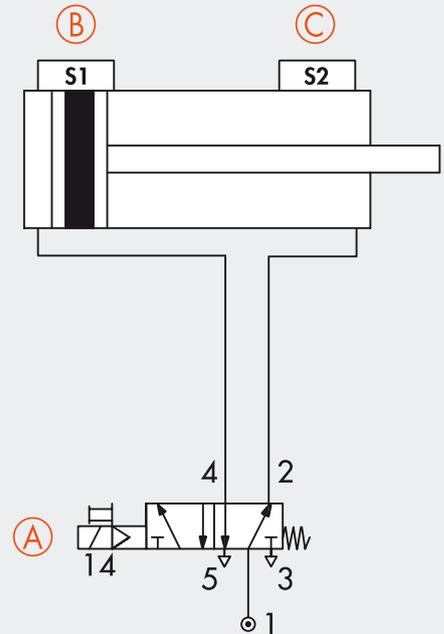
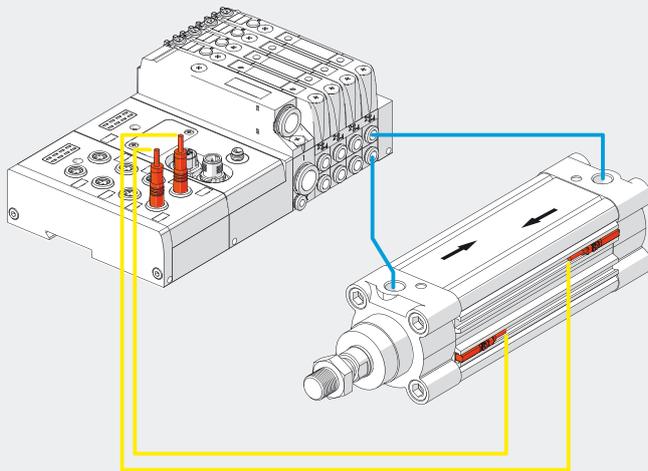


**Stato:** se i tempi rilevati si discostano da quelli impostati, oltre la tolleranza impostata nei campi "Tolleranza", si attivano i rispettivi bit di segnalazione. I bit di segnalazione si resettano al successivo rilevamento all'interno della tolleranza.

- bit 0: segnalazione ritardo attivazione fuori tolleranza
- bit 1: segnalazione ritardo ripristino fuori tolleranza
- bit 2: segnalazione tempo di attuazione fuori tolleranza
- bit 3: segnalazione tempo di ritorno fuori tolleranza

**Contatore corse attuatore:** indica il numero di movimenti di attuazione e ripristino dell'attuatore.

### 1.1.3 Esempio



A = Valvola in uso 5/2 – valvola ad un pilota, posizione pilota su base per due piloti = 7

B = Sensore ritorno attuatore – collegato all'ingresso N° 8

C = Sensore estensione attuatore – collegato all'ingresso N° 7

Id pilota 1 = 7

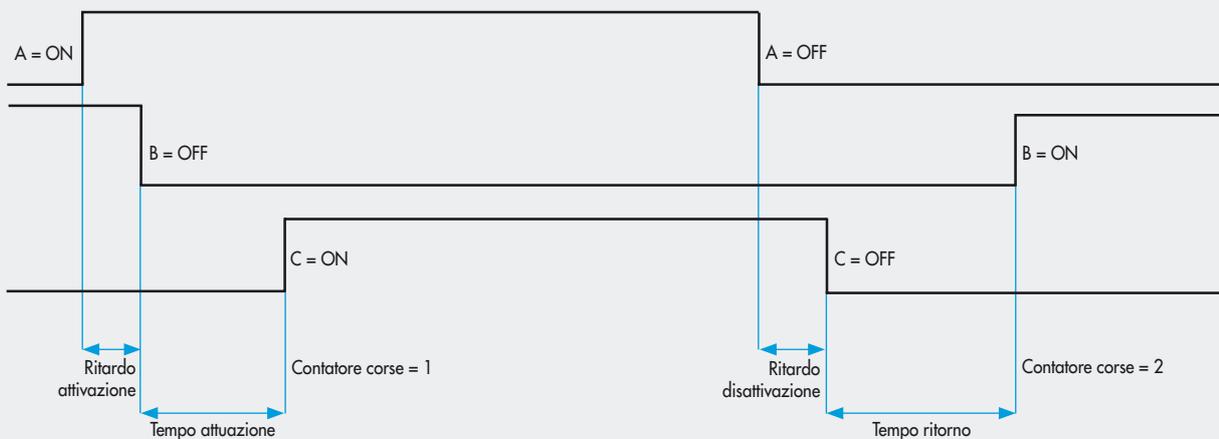
Id pilota 2 = 0

Id Input fine corsa 1 = 8

Id Input fine corsa 2 = 7

Estensione attuatore: A - ON → B - OFF = Ritardo attivazione  
B - OFF → C - ON = Tempo attuazione

Ritorno attuatore: A - OFF → C - OFF = Ritardo disattivazione  
C - OFF → B - ON = Tempo ritorno



Tutti i dati vengono aggiornati, quando l'Id attuatore = 7



Funzione	Impostazioni		Tempo Rilevato [ms]	Bit di segnalazione
	Tempo [ms]	Tolleranza		
Ritardo attivazione	50	± 50 % (25 ÷ 75 ms)	54	Off
			77	On
Tempo attuazione	600	± 20 % (480 ÷ 720 ms)	610	Off
			470	On
Ritardo disattivazione	100	± 25 % (75 ÷ 125 ms)	120	Off
			130	On
Tempo ritorno	700	± 10 % (630 ÷ 770 ms)	650	Off
			610	On

Assegnazione degli indirizzi di diagnostica

		N° byte	Dimensione [byte]
<b>Dati di sistema</b>			
Contatore accensioni		da 1 a 4	4
Contatore allarmi di alimentazione		5	1
Riservato		6	1
<b>Dati delle valvole</b>			
Id valvola		7	1
Riservato		8	1
Pilota 1	Segnalazione di superamento vita media - bit 0	9	1
	Contatore allarmi Corto Circuito	10	1
	Contatore allarmi Circuito Aperto	11	1
	Contatore cicli	da 12 a 15	4
	Contatore del tempo totale di eccitazione pilota 1 [sec]	da 16 a 19	4
Pilota 2	Segnalazione di superamento vita media - bit 0	20	1
	Contatore allarmi Corto Circuito	21	1
	Contatore allarmi Circuito Aperto	22	1
	Contatore cicli	da 23 a 26	4
	Contatore del tempo totale di eccitazione pilota 2 [sec]	da 27 a 30	4
<b>Dati degli attuatori</b>			
Id attuatore		31	1
Stato		32	1
Ritardo attivazione [ms]		33; 34	2
Ritardo ripristino [ms]		35; 36	2
Tempo di attuazione [ms]		da 37 a 40	4
Tempo di ritorno [ms]		da 41 a 44	4
Contatore corse attuatore		da 45 a 48	4
<b>Totale</b>			<b>48</b>

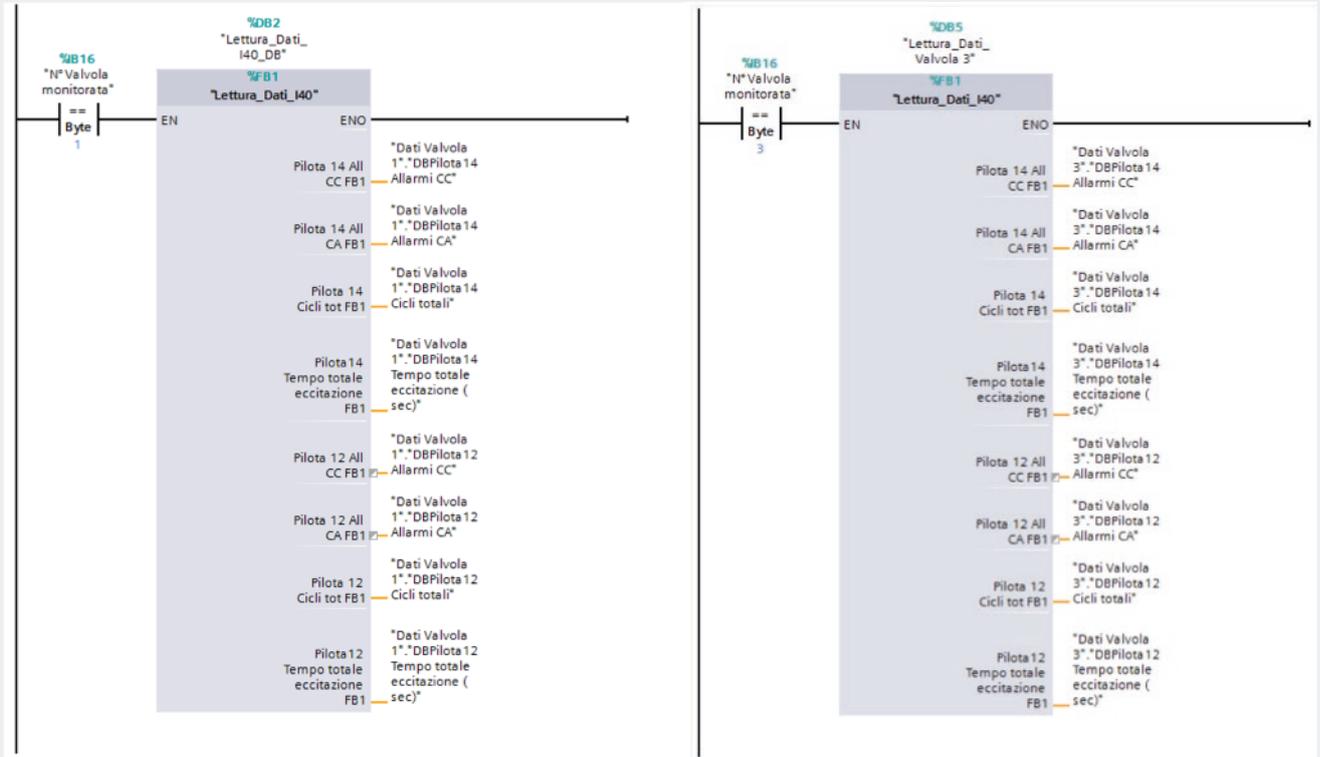
### 1.1.4 Esempio di come gestire il salvataggio dei dati in un data base PLC

Dati valvole

Creare un Data Base dove salvare i dati delle valvole

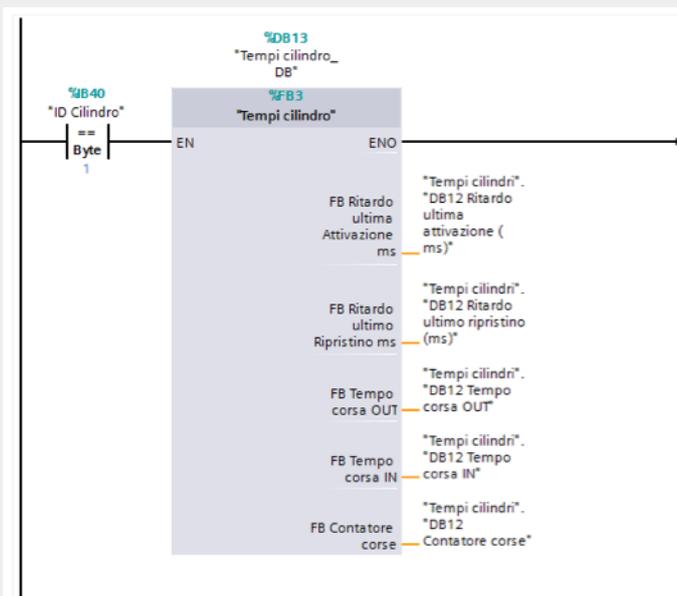
Dati Valvola 1				
	Nome	Tipo di dati	Valore di svvio	Valore di controllo
1	Static			
2	DBPilota14 Flag superamento vita media	Bool	false	FALSE
3	DBPilota14 Allarmi CC	Int	0	0
4	DBPilota14 Allarmi CA	Int	0	0
5	DBPilota14 Cicli totali	DInt	0	52
6	DBPilota14 Tempo totale eccitazione (sec)	DInt	0	52
7	DBPilota12 Flag superamento vita media	Bool	false	FALSE
8	DBPilota12 Allarmi CC	Int	0	0
9	DBPilota12 Allarmi CA	Int	0	0
10	DBPilota12 Cicli totali	DInt	0	87
11	DBPilota12 Tempo totale eccitazione (sec)	DInt	0	130

Creare una funzione che, abilitata dall'Id valvola, salvi i dati nei campi corrispondenti, del data base creato in precedenza.



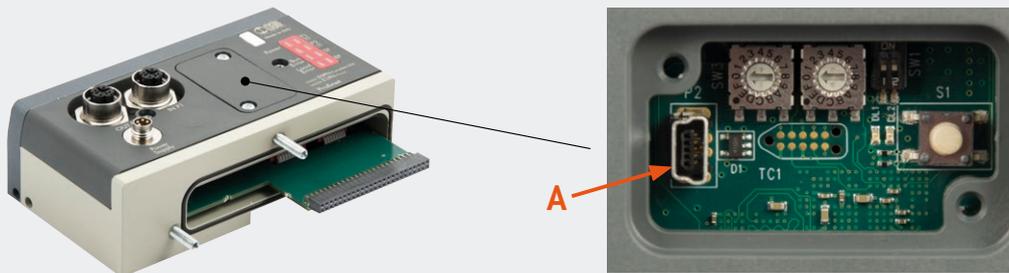
### Dati attuatori

Creare un Data Base dove salvare i dati degli attuatori

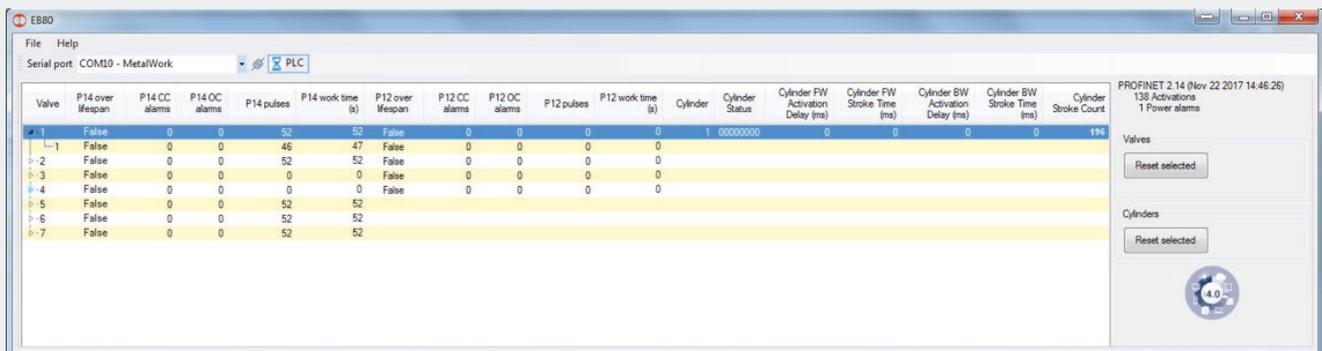


## 1.2 LETTURA DEI DATI UTILIZZANDO IL SOFTWARE EB 80 DS

Il software EB 80 DS consente la lettura dei dati direttamente dalla Connessione Elettrica EB 80 con bus di campo, attraverso la porta USB (A) situata sotto il Coperchio della connessione Elettrica.



Connettere EB 80 al PC. Aprire il software EB 80 DS.



Selezionare la serial port: COMx – MetalWork e connettersi al sistema cliccando sull'icona di connessione. Se il sistema non è connesso ad un PLC, prima di attivare la connessione, cliccare sull'icona PLC. Verranno visualizzati i dati di tutte le valvole e di tutti gli attuatori configurati.

### Reset dei dati delle valvole

Nel caso venga sostituita una valvola, è utile che il numero di cicli venga azzerato. Per fare ciò selezionare la valvola da resettare e cliccare sul pulsante "Valves - Reset Selected". I dati della prima riga verranno azzerati e salvati in un'area di memoria diversa e non azzerabile, visibile cliccando sulla freccia del menù ad albero.

In questo modo si dispone di dati "relativi" alla valvola in uso e di dati "assoluti" del sistema. Ad ogni reset i dati relativi vengono sommati a quelli assoluti.

### Reset dei dati degli attuatori

Nel caso venga sostituito un attuatore, è utile che il numero di corse venga azzerato. Per fare ciò selezionare l'attuatore da resettare e cliccare sul pulsante "Actuators - Reset Selected". I dati dell'attuatore verranno azzerati.

Per ripristinare il sistema dopo la disconnessione, spegnere e riaccendere il sistema.

NOTE

Lined area for notes.





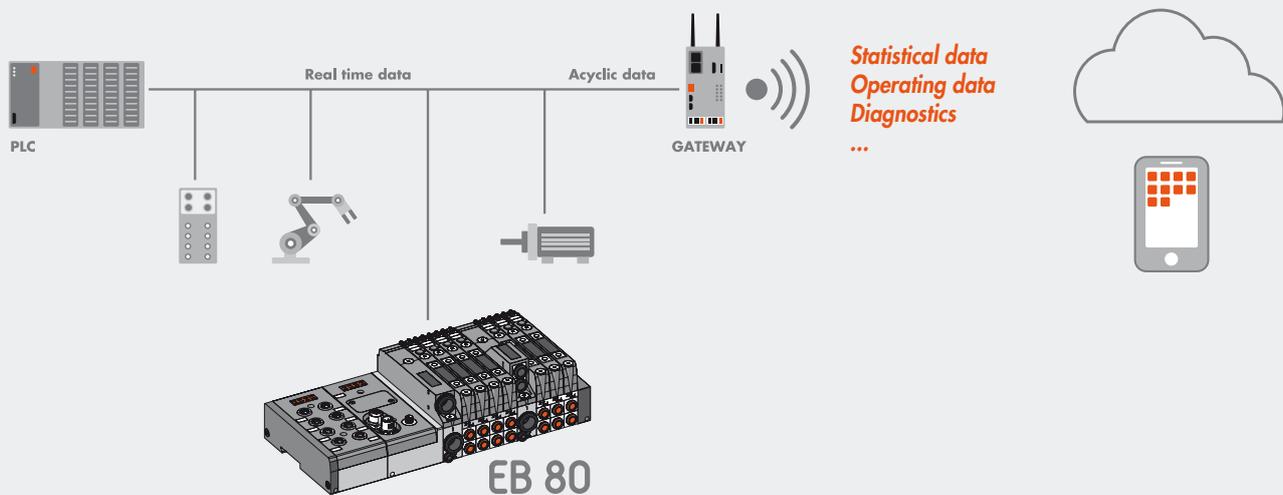
<b>INTRODUCTION</b>	PAGE 16
<b>1. SYSTEM CONFIGURATION</b>	PAGE 18
1.1 DESCRIPTION OF FUNCTIONS	PAGE 18
1.1.1 System data	PAGE 18
1.1.2 Actuation function	PAGE 19
1.1.3 Example	PAGE 21
1.1.4 Example of how to save data in a PLC database	PAGE 22
1.2 DATA READING USING THE EB 80 DS SOFTWARE	PAGE 24

## INTRODUCTION

Electrical connection modules can be used to complement the EB 80 with the main field buses available in the market. In this way, the control system (generally a PLC) can handle in real time the behaviour of the solenoid valve island, including signal modules.

With the introduction of the I4.0 version, the field bus connection modules also send to the network the historical and diagnostic data relating to the behaviour of the island (such as the number of cycles for each solenoid pilot, total activation time and alarms) and the controlled pneumatic circuit (such as the delay times in sensor switching and actuator activation times).

This data is also sent to the control system and can be handled differently depending on the situation: in some cases, it can be used in real time, like in the case of fault alarms; in other cases, it can be sent to a storage local unit or one remotely controlled on a cloud server, and is analysed in a subsequent stage; in other cases, the alarms can be sent to a teleservice station that can monitor the state of the system remotely.



The new advanced EB 80 diagnostic functions, known as EB 80 I4.0, provide a powerful analysis tool for traditional maintenance operations, ensuring the safe, reliable and lasting operation of production units.

They are available for all electrical connections with fieldbuses and bases marked I4.0, with advanced diagnostics integrated in accordance with Industry 4.0 philosophy.

These functions use the original EB 80 diagnostics, integrating them with the ability of the station itself to control IOs.

They re-organise and optimise maintenance management by developing predictive maintenance in order to:

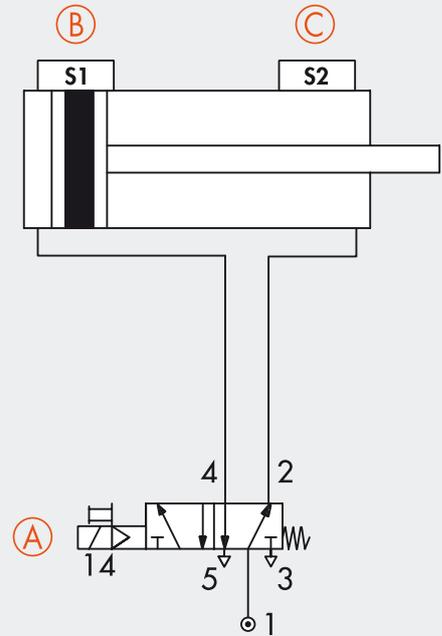
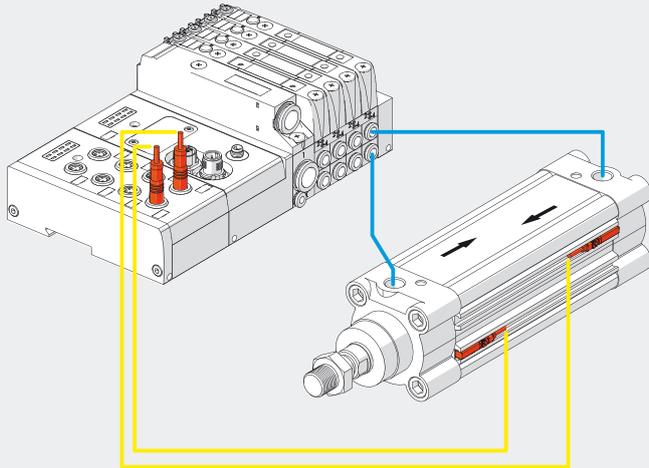
- predict faults;
- intervene early to avoid system downtime;
- have all information on equipment operation available in real time;
- monitor component end-of-lifetime;
- optimise warehouse spare parts management.

This makes it possible to turn the data collected into concrete actions using standard EB 80 stations without needing additional modules.

### Description of EB 80 I4.0 functions

- System data:
  - EB 80 system startup counter;
  - supply alert counter.
- Valve data. Each valve base for each solenoid valve permanently stores the following information:
  - cycle counter;
  - counter for total solenoid valve excitation time;
  - activation of an indication of exceeding 60% of the average life;
  - short circuit alert counter;
  - open circuit alert counter.
- Electropneumatic system control functions (data updated with each cycle):
  - measurement of the delay between activating the solenoid valve "A" and actuator movement commencing via the signal of sensor "B", with delays that exceed the limit flagged;
  - measurement of actuator movement time using two linked sensors "B" and "C", with exceeded time limits flagged;
  - measurement of the delay between deactivating the solenoid valve "A" (or activating a second valve) and actuator return commencing via the signal of sensor "B", with exceeded time limits flagged;

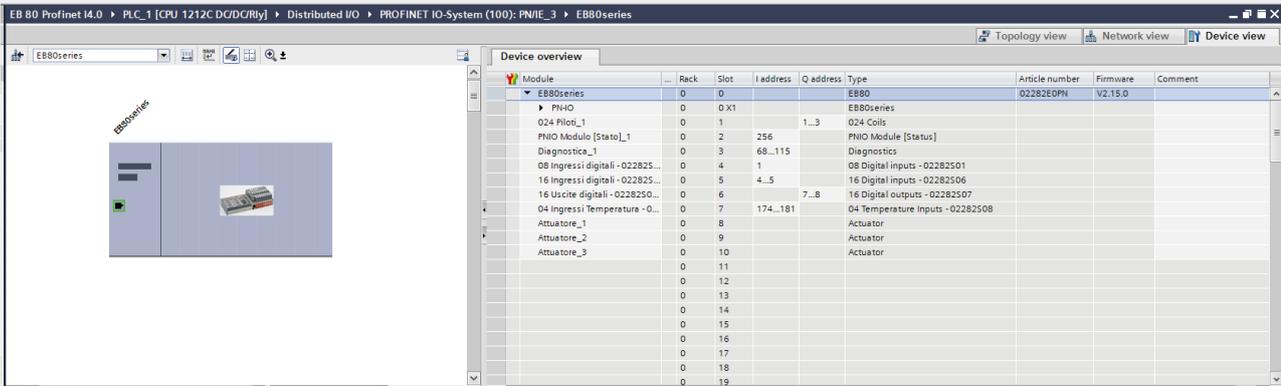
- measurement of actuator return time using two linked sensors "B" and "C", with exceeded time limits flagged;
- counter for actuator range of motion.



**⚠ WARNING**

EB 80 14.0 functions are available for software versions installed after:  
 Profinet 2.16 / Valve bases 7.03 with 14.0 logo / GSDML-V2.32-Metalwork-EB80-20180115

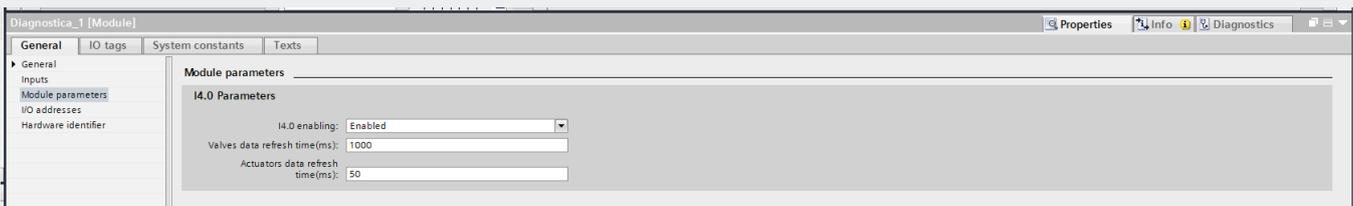
## 1. SYSTEM CONFIGURATION



Configure the EB 80 system with all the modules installed in the system in use, as described in the EB 80 Profinet IO user manual. Then complete the configuration of I4.0 functions.

### Enabling the functions

The **Diagnostics** module must be uploaded to the system from the Module I4.0 hardware catalogue. Select "Unit Parameters" in the "Property" page.



- **I4.0 Enabling:** to enable diagnostics functions
- **Valve data update time:** the system and valve data stored in the electronic circuit boards of the valve bases are updated with the time set for this parameter. It is advisable to set a time greater than 1000 ms to avoid overloading the system.
- **Actuator data update time:** the time relating to the actuation and reset of the controlled actuators. A short time, e.g. 50ms, must be set to obtain data in real time.

## 1.1 DESCRIPTION OF FUNCTIONS

### 1.1.1 System data

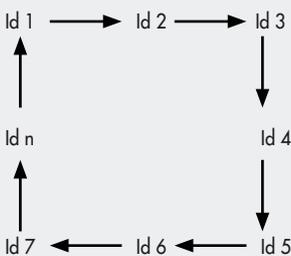
**Actuation counter:** showing the number of system actuations; it is updated every time it is switched on.

**Power alarm counter:** showing it indicates the number of alarms caused by power supply values outside the 10.8V-31.2V DC permitted range.

#### Valve data

The system reads the data relating to each solenoid pilot cyclically and sequentially at the time set under "Valve data update time".

The valve number associated with the data is shown with the valve ID byte. In order to avoid overloading the system, it is advisable to set a long time, e.g. in a system with 8 valves and a set update time of 1000 ms, the data of each valve is updated every 8 seconds.



**Cycle counter:** showing the number of actuations of each solenoid pilot. When the number of cycles exceeds 60% average life, set at 30,000,000 cycles, an alert comes on to **indicate that the average operating lifetime has been exceeded**.

**Total solenoid valve energising time counter:** it indicates the total actuation time of the solenoid pilots.

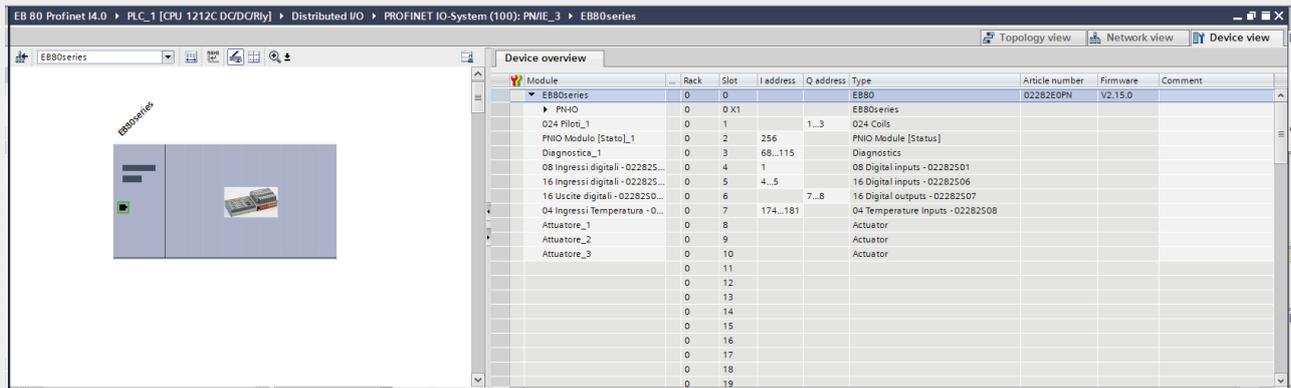
**Short circuit alarm counter:** it indicates the number of alarms due to the short circuiting of each solenoid valve.

**Open-circuit alarm counter:** it indicates the number of alarms due to the interruption of the solenoid valve reel of each solenoid pilot.

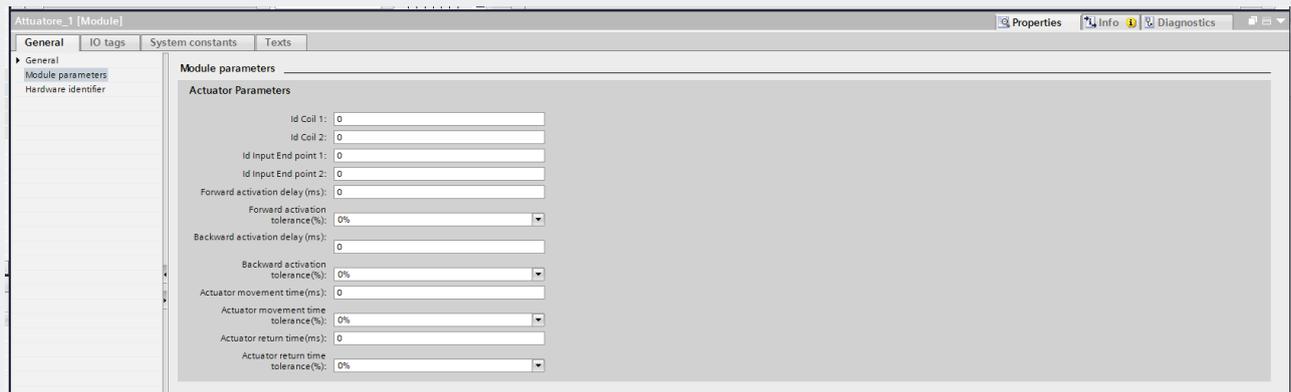
**IMPORTANT:** a command relating to free positions, e.g. dummy valves, bypass valves or the second pilot of one-pilot valves, generates an open-circuit alarm.

### 1.1.2 Actuation function

This function enables the control of the electro-pneumatic system. Its operation requires the installation of an input digital signal module in the system. Up to 10 actuator modules can be installed. Upload the **Actuator** module to the system from the I4.0 Module hardware catalogue.



Select "Module parameters" in the "Property" page.



#### Actuator parameters

**Pilot 1 Id:** enter the pilot number corresponding to the valve associated with the actuator to be controlled.

**Pilot 2 Id:** enter the second pilot number corresponding to the valve associated with the actuator to be controlled, when a two-pilot valve is used. Enter 0 when using one-pilot valves.

#### Example of pilot Id assignment

Base for 8-control valves. Valves with one or two solenoid pilots can be mounted.

Type of valve	2 solenoid pilots valve	1 solenoid pilots valve	Dummy valve or Bypass	1 solenoid pilots valve
Solenoid pilot 1	14	14	-	14
Solenoid pilot 2	12	-	-	-
Output	Out 1 Out 2	Out 3 Out 4	Out 5 Out 6	Out 7 Out 8
Pilot 1 Id	1	3	-	7
Pilot 2 Id	2	-	-	8

**Limit switch 1 Input ID:** enter the input number of the digital input module to which the first sensor of the actuator limit switch is connected.

**Limit switch 2 Input ID:** enter the input number of the digital input module to which the first sensor of the actuator limit switch is connected.

**Actuation delay:** the time elapsing between the enabling of electrical pilot 1 and the actuator movement, which is read by verifying that the limit switch 1 input has been disabled.

**Actuation delay % tolerance:** set the percentage tolerance allowed for the enabling time. Any plus or minus deviation from the set tolerance causes a signalling bit to come on.

**Reset delay:** the time elapsing between the disabling of electrical pilot 1 or the enabling of electrical pilot 2 in the case of a two-pilot valve and the actuator movement, which is read by verifying that the limit switch 2 input has been disabled.

**Reset delay % tolerance:** set the percentage tolerance allowed for the reset time. Any plus or minus deviation from the set tolerance causes a signalling bit to come on.

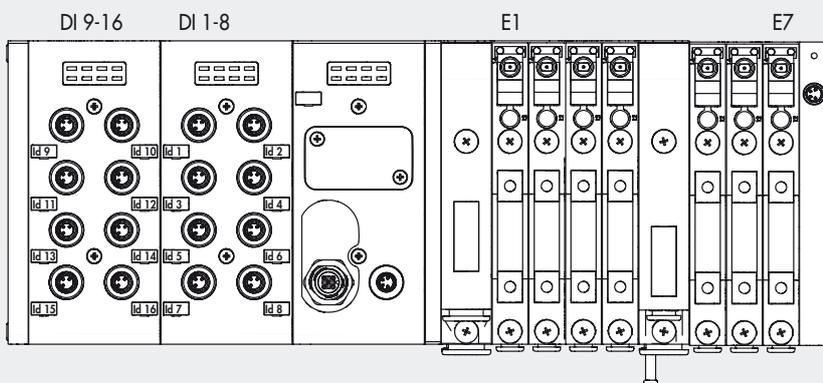
**Actuation time:** the time it takes the actuator to move via the two associated sensors.

**Actuation time % tolerance:** enter the percentage tolerance allowed for the actuation time. Any plus or minus deviation from the set tolerance causes a signalling bit to come on.

**Return time:** the time it takes for the actuator to return via the two associated sensors.

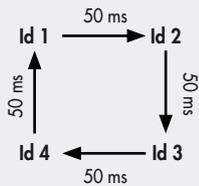
**Return time % tolerance:** enter the percentage tolerance allowed for the actuator return time. Any plus or minus deviation from the set tolerance causes a signalling bit to come on.

Example of how to assign the limit switch input ID



The system reads the data of each actuator cyclically and sequentially according to the update time set under **Actuator Data Update Time**. The actuator number to which the values refer is indicated by the **actuator ID** byte. A short time, e.g. 50 ms, must be set to get readings in real time.

**Actuator Id:** indicating the actuator which the data read cyclically and sequentially refer to. Up to 10 actuator modules can be installed in the system. If, for instance, 4 actuator modules are installed, **with an update time of 50 ms**, the actuator ID rotates cyclically at 50 ms intervals, which means that the data of each actuator is updated every 200 ms.

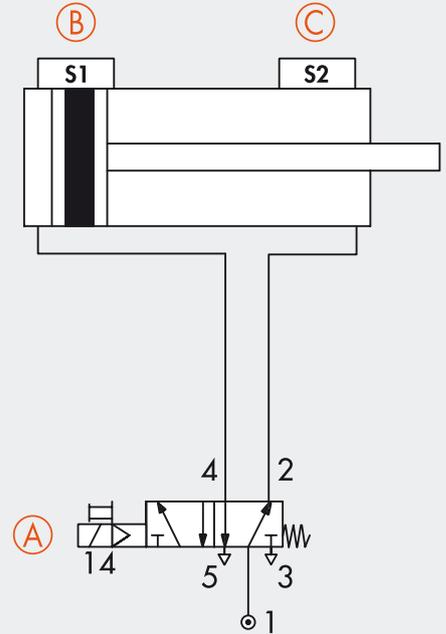
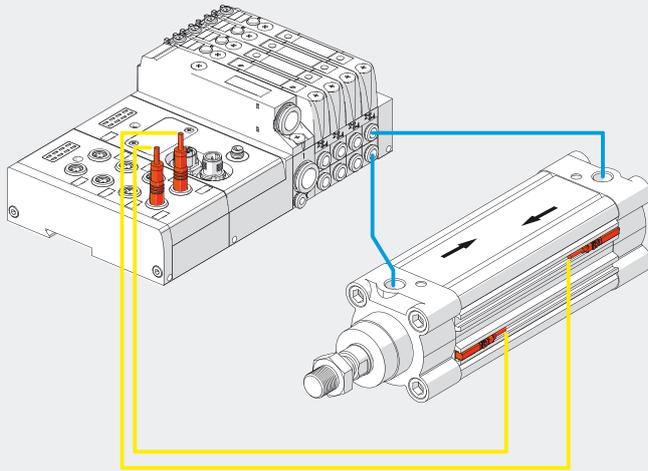


**State:** if the values read exceed the times set in the "Tolerance" fields, the corresponding signalling bits activate and reset at the next reading with values within the tolerance.

- bit 0: actuation delay signal due to out of tolerance
- bit 1: reset delay signal due to out of tolerance
- bit 2: actuation time signal due to out of tolerance
- bit 3: return time signal due to out of tolerance

**Actuator stroke counter:** showing the number of actuations and reset movements of the actuator.

### 1.1.3 Example



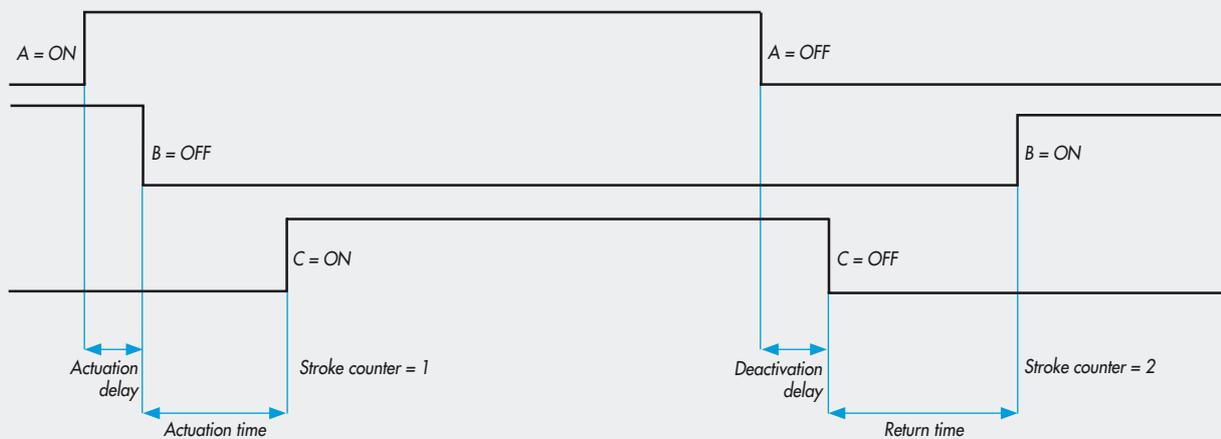
A = Valve in use, 5/2 one-pilot valve, pilot position on a base for two pilots = 7  
 B = Actuator return sensor – connected to input no. 8  
 C = Actuator extension sensor – connected to input no. 7

Pilot Id 1 = 7  
 Pilot Id 2 = 0

Limit switch input Id 1 = 8  
 Limit switch Input Id 2 = 7

Actuator extension: A - ON → B - OFF = Actuation delay  
 B - OFF → C - ON = Actuation time

Actuator return: A - OFF → C - OFF = Deactivation delay  
 C - OFF → B - ON = Return time



All data is updated when actuator ID is equal to 7.

Settings			Time read [ms]	Signalling bit
Function	Time [ms]	Tolerance		
Actuation delay	50	± 50 % (25 to 75 ms)	54	Off
			77	On
Actuation time	600	± 20 % (480 to 720 ms)	610	Off
			470	On
Deactivation delay	100	± 25 % (75 to 125 ms)	120	Off
			130	On
Return time	700	± 10 % (630 to 770 ms)	650	Off
			610	On

#### Assignment of diagnosis addresses

		N° byte	Dimensions [byte]
<b>System data</b>			
Counter switching		from 1 to 4	4
Power alarm counter		5	1
Reserved		6	1
<b>Valve data</b>			
Valve ID		7	1
Reserved		8	1
Pilot 1	Average life excess signal - bit 0	9	1
	Short-circuit alarm counter	10	1
	Circuit open alarm counter	11	1
	Cycle counter	from 12 to 15	4
	Counter of total pilot 1 energising time [sec]	from 16 to 19	4
Pilot 2	Average life excess signal - bit 0	20	1
	Short-circuit alarm counter	21	1
	Circuit open alarm counter	22	1
	Cycle counter	from 23 to 26	4
	Counter of total pilot 2 energising time [sec]	from 27 to 30	4
<b>Actuator data</b>			
Actuator Id		31	1
State		32	1
Actuation delay [ms]		33; 34	2
Reset delay [ms]		35; 36	2
Actuation time [ms]		from 37 to 40	4
Return time [ms]		from 41 to 44	4
Actuator stroke counter		from 45 to 48	4
<b>Total</b>			<b>48</b>

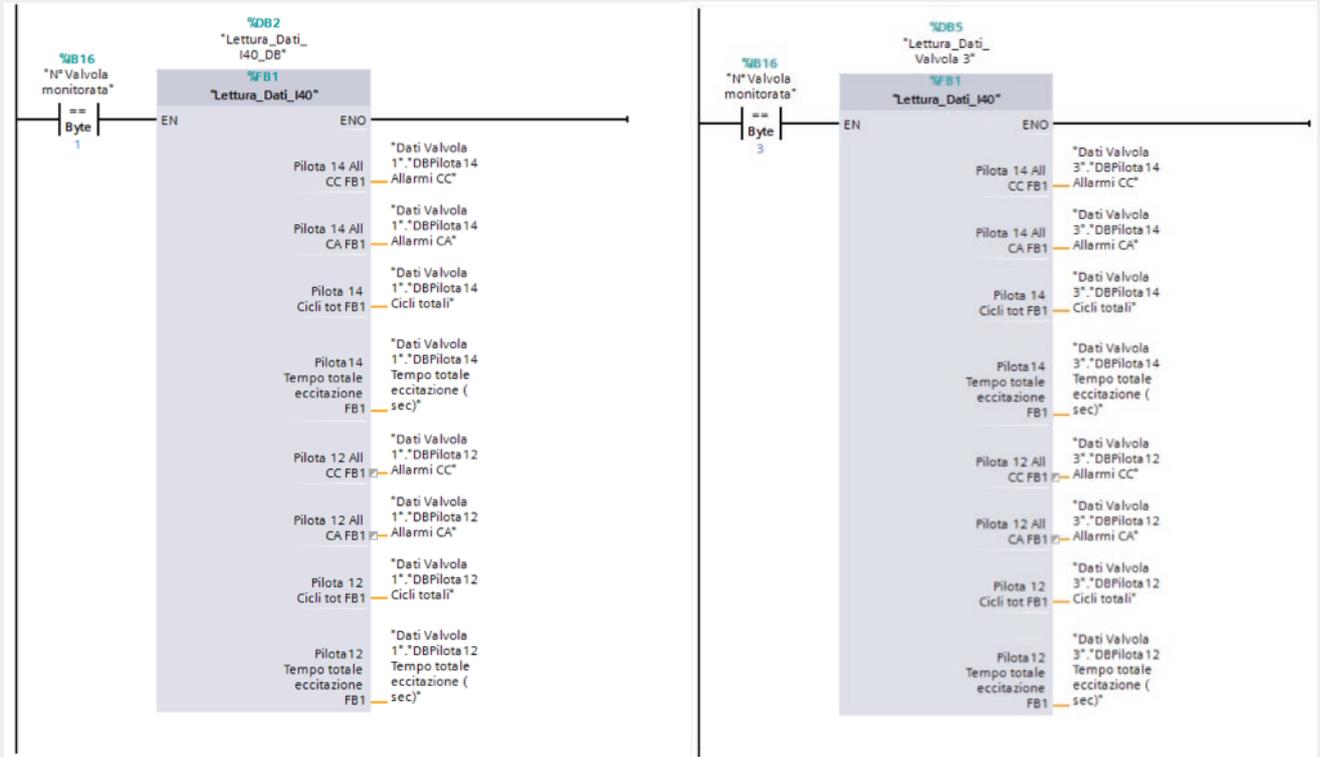
#### 1.1.4 Example of how to save data in a PLC database

##### Valve data

Creating a database where to save valve data

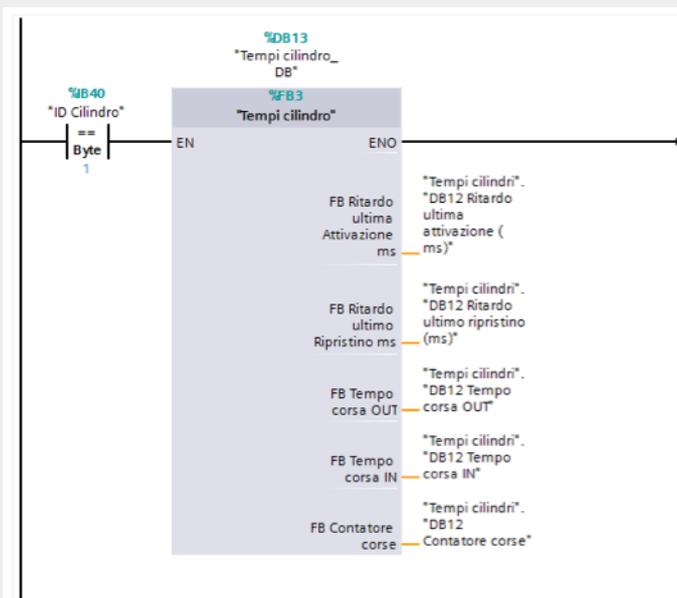
Dati Valvola 1				
	Nome	Tipo di dati	Valore di svvio	Valore di controllo
1	Static			
2	DBPilota14 Flag superamento vita media	Bool	false	FALSE
3	DBPilota14 Allarmi CC	Int	0	0
4	DBPilota14 Allarmi CA	Int	0	0
5	DBPilota14 Cicli totali	DInt	0	52
6	DBPilota14 Tempo totale eccitazione (sec)	DInt	0	52
7	DBPilota12 Flag superamento vita media	Bool	false	FALSE
8	DBPilota12 Allarmi CC	Int	0	0
9	DBPilota12 Allarmi CA	Int	0	0
10	DBPilota12 Cicli totali	DInt	0	87
11	DBPilota12 Tempo totale eccitazione (sec)	DInt	0	130

Creating a function which, when enabled by valve ID, saves data in the corresponding fields of the previously created database.



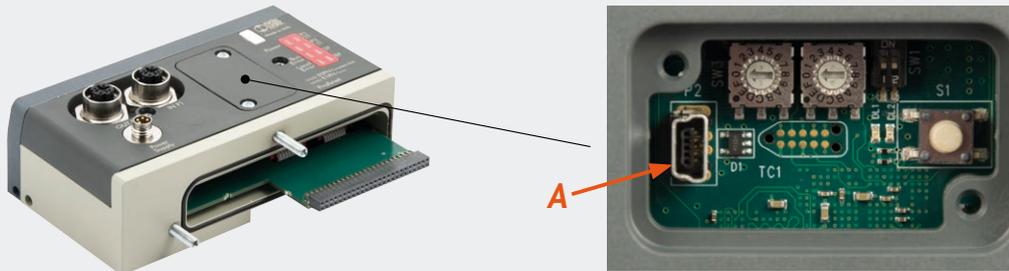
### Actuator data

Creating a database where to save actuator data

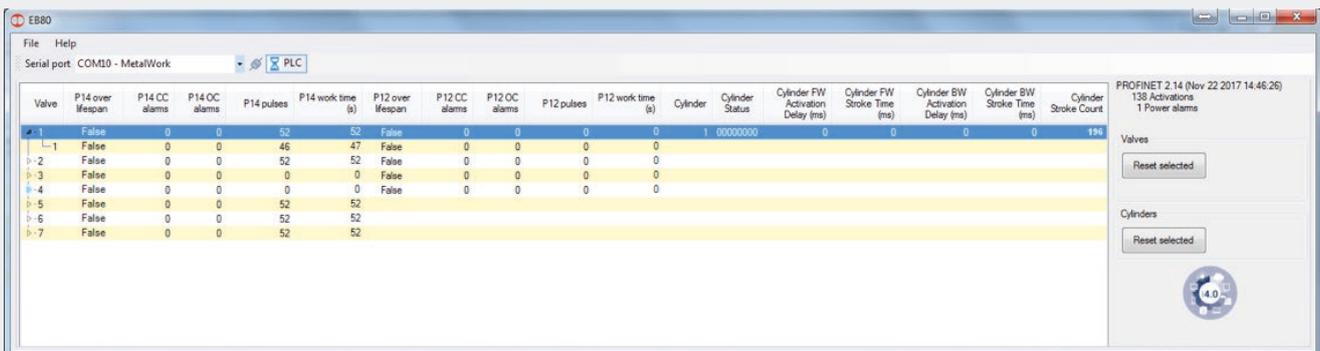


## 1.2 DATA READING USING THE EB 80 DS SOFTWARE

The EB 80 DS software is used to read data directly from the EB 80 power connection with fieldbus, via the USB port (A) located under the power connection cover.



Connecting the EB 80 to the PC. Open the EB 80 DS software.



Select the serial port: COMx-MetalWork and connect up to the system by clicking on the connection icon. If the system is not connected to a PLC, click on the PLC icon before activating the connection. The data of all the valves and configured actuators will be displayed.

### Data reset of valves

When a valve is replaced, it is advisable to reset the number of cycles. To do this, select the valve to be reset and click on the "Valves - Reset Selected" button. The data of the first row will be reset and saved in another non-resettable memory area, which can be viewed by clicking on the arrow of the tree menu. In this way, the "relative" data of the valve being used and the "absolute" data of the system are available. At each reset, the relative data is added to the absolute data.

### Data reset of actuators

When an actuator is replaced, it is advisable to reset the number of strokes. To do this, select the actuator to be reset and click on the "Actuators - Reset Selected" button. The actuator data will be reset. To restore the system after disconnection, turn the system off and then on again.