

Porte aperte

Holger Zeltwanger

Il collegamento del sistema di controllo della macchina alla rete di automazione aziendale è un'operazione espressamente richiesta ai costruttori di macchina.

Se il controllore interno della macchina è basato su una rete 'embedded', occorre un gateway che colleghi la rete aziendale al bus di campo utilizzato sulla macchina. Il gruppo internazionale di utenti e costruttori CAN in Automation (CiA) si è impegnato, pertanto, a standardizzare i servizi e i protocolli di comunicazione, rendendo possibile la comunicazione tra reti eterogenee e omogenee.

Reti gerarchiche

Il settore dell'automazione industriale utilizza da anni diversi sistemi di comunicazione. A livello più basso vengono utilizzati semplici sistemi a bus sensore/attuatore e backplane (ad esempio in unità di I/O modulari). A livello intermedio la macchina è controllata da reti 'embedded': esse collegano controllori a logica programmabile, complessi azionamenti elettrici e idraulici, moduli di I/O e altri dispositivi comandati da microcontrollori, tra cui interfacce uomo-macchina e dispositivi di acquisizione dati.

Reti di automazione estese a tutto un complesso industriale rappresentano il livello più alto e sono responsabili della comunicazione tra le macchine.

Sistemi e reti a bus sono collegati tramite dispositivi in grado di interpretare i segnali provenienti dai diversi sistemi di comunicazione. Essi sono denominati ripetitori, se le reti non sono logicamente separate. Se la comunicazione avviene al livello collegamento dati (livello 2 del modello di riferimento OSI), occorre un bridge. Tale dispositivo può, ad esempio, tradurre messaggi CAN in messaggi Ethernet.

Tuttavia, in questo caso, i protocolli di livello superiore (applicativo) devono essere identici. In genere non è così, per cui è necessario un gateway.

Un gateway è in grado di tradurre i servizi di un livello applicativo in un altro livello applicativo.

È così che un gateway collega una rete CANopen a una rete EtherCAT, Modbus TCP o Powerlink Ethernet.

Teoricamente dovrebbe funzionare così com'è. In pratica, richiede dei comandi che fanno sì che il gateway avvii specifiche operazioni dall'altro lato.

Per esempio, un controllore collegato in rete tramite TCP con un gateway CANopen richiede informazioni su un sensore della rete CANopen. Finora il costruttore della macchina doveva utilizzare un comando proprietario del costruttore del gateway per attivare un servizio SDO (Service Data Object) rendendo i costruttori di macchine dipendenti dal costruttore del gateway, motivo per cui i costruttori di macchine hanno richiesto la standardizzazione dei protocolli di comunicazione.

Gateway standardizzati semplificano le connessioni di rete

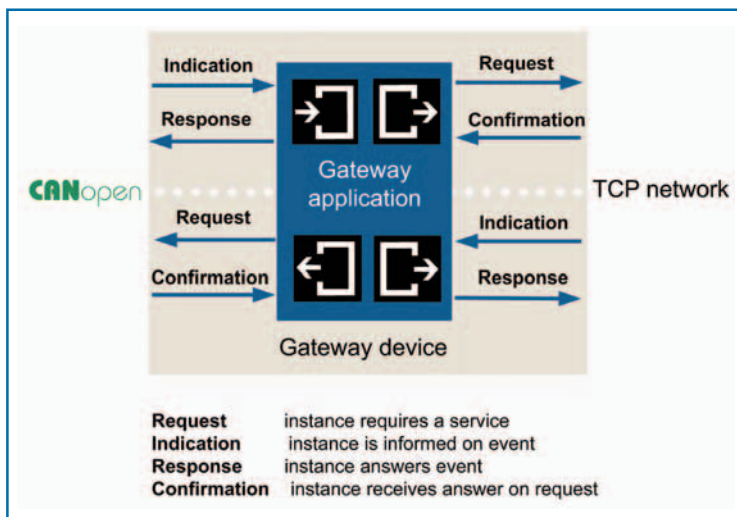


Fig. 1 - Tutti i servizi di comunicazione si basano sui quattro servizi di base che sono disponibili in entrambi i sensi (da Ethernet a CANopen e viceversa)

Gateway TCP/CANopen

Dalla collaborazione tra Modbus-IDA e CAN in Automation è nata una specifica per gateway, che consente di sviluppare e fabbricare gateway standardizzati. Ciò è vantaggioso in quanto controllore e strumenti di configurazione possono accedere a qualsiasi dispositivo CANopen attraverso un set

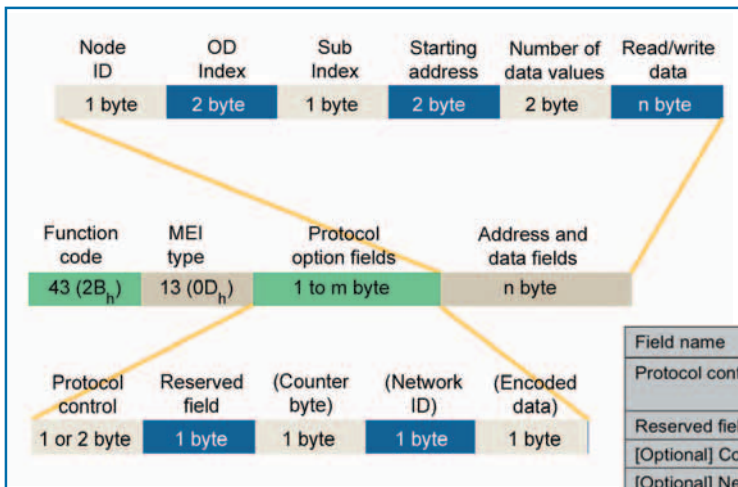


Fig. 2 - Il protocollo Modbus TCP per l'inizializzazione di funzioni CANopen

di comandi e risposte predefiniti. Inoltre, CiA ha definito un protocollo basato su comandi Ascii in grado di accedere a un dispositivo CANopen tramite un dispositivo qualsiasi di una rete TCP attraverso il gateway standardizzato.

La specifica per gateway (CiA 309) supporta, tra i vari servizi, anche i seguenti compiti CANopen: servizi di accesso per SDO (Service Data Object), per esempio lettura e scrittura di voci del dizionario degli oggetti in un dispositivo CANopen; servizi di accesso per PDO (Process Data Object), per esempio, configurazione di PDO nel gateway; servizi NMT (Network Management) per reti CANopen, per esempio, avviamento e arresto di un dispositivo CANopen, servizi di gestione degli errori, per esempio, lettura di un errore del gateway; servizi di configurazione interfacce CANopen per esempio, inizializzazione del gateway; servizi di gestione del gateway, per esempio, impostazione dell'ID di nodo CANopen del gateway; servizi di gestione del controllore, per esempio, avviamento e arresto di programmi nel gateway. I servizi di base includono (figura 1): request, richiesta di un servizio; indication, accettazione di una richiesta; response, risposta a una richiesta; confirmation, conferma di un servizio. E' stato definito uno speciale formato per i comandi Modbus TCP (figura 2) e per la risposta. I protocolli della rete CANopen sono equivalenti a quelli dei protocolli specificati al livello applicativo CANopen (CiA 301/302 o EN 50325-4).

Un esempio è il comando di avviamento NMT-Modbus TCP per un dispositivo

CANopen in figura 3. La figura 4 illustra la tipica struttura del protocollo di generici comandi Ascii mentre la figura 5 mostra il comando Ascii per l'avviamento NMT.

Per alcune applicazioni, CANopen è sorprendentemente il sistema di comunicazione di livello più elevato nella scala gerarchica delle reti. In tali applicazioni, le reti basate su Ethernet collegano azionamenti estrema-

Field name	Byte size and order	Example/range
Protocol control	1 byte	05 _n : specified network 01 _n : default network
Reserved field	1 byte	00 _n
[Optional] Counter byte	1 byte	not applicable
[Optional] Network ID	1 byte	1 to 255 if protocol control is 05 _n
[Optional] Encoded data	1 byte	not applicable

Field name	Byte size and order	Example/range
Node ID	1 byte	7F _n
Index	1 byte, high 1 byte, low	FFFF _n
Sub-index	1 byte	04 _n : start all nodes 12 _n : start one node
Starting address	1 byte, high 1 byte, low	0000 _n
Number of data values	1 byte, high 1 byte, low	0000 _n : if sub-index is 04 _n 0001 _n : if sub-index is 12 _n
Read/write data	1 byte	not applicable (if number of data values is 0000 _n) 01 _n to 7F _n : target node-ID (if number of data values is 0001 _n)

Fig. 3 - Comandi Modbus TCP per avviare un dispositivo CANopen

mente rapidi, che sono a un livello gerarchico più basso rispetto alla rete CANopen. Nello scenario attuale, non esistono comandi standard dal lato CANopen. Tuttavia, tali protocolli avrebbero senso per Powerlink-Ethernet ed EtherCAT poiché questi due sistemi di comunicazione supportano i pro-

```

<command-request> ::= "["<sequence>"]" [[<net>] <node>] <command>
<sequence> ::= UNSIGNED32
<net> ::= UNSIGNED8
<node> ::= UNSIGNED8
<command> ::= <command-specifier> | <compound-command>
<compound-command> ::= <command-specifier> <parameter>
<parameter> ::= <value> | <compound-parameter>
<compound-parameter> ::= <value> <parameter>

<command-response> ::= "["<sequence>"]" <response>
<response> ::= <value> | <error-string> | <emcy-list> | "OK"
<error-string> ::= "Error:" <error code>
<error-code> ::= <internal-error-code> | <sdo-abort-code>
<emcy-list> ::= [<emcy1> " " ..<emcy254>]
    
```

Fig. 4 - Struttura di un generico comando TCP-Ascii per avviare una funzione CANopen

ASCII command

```
net] node] start
8 start
```

ASCII answer

```
[[net] node] start ok
1 8 start ok
```

Fig. 5 - Comandi Ascii per avviare un dispositivo CANopen

filii procedurali CANopen. Essi sono persino compatibili, in larga misura, con il livello applicativo CANopen.

L'impiego degli stessi profili procedurali semplifica di gran lunga la realizzazione dei gateway. In questi casi, il gateway non deve riformattare i dati applicativi. La riformattazione può richiedere molte risorse al processore, in particolare per quanto riguarda i dati bit-oriented. Inoltre, per i valori analogici utilizzati in unità fisiche diverse, il microcontrollore potrebbe dover sprecare una consistente percentuale della sua potenza di calcolo.

collocati tra un punto e l'altro devono far sì che i messaggi penetrino o comunque avanzino tra di essi. Il gruppo CiA sta già lavorando sulla standardizzazione di queste reti multilivello. Sono inoltre previste architetture di rete omogenee basate su CANopen.

Grazie all'uso di profili applicativi, vengono già realizzate reti multiple CANopen omogenee. Ad esempio, con un profilo applicativo CANopen è possibile realizzare controllori per ascensori con otto cabine ciascuno fino a 254 piani. Poiché tutte le comunicazioni PDO sono predefinite, si tratta di una rete logica. Questa rete logica può essere distribuita su più reti fisicamente separate, collegate tra di loro tramite gateway trasparenti (figura 6).

Le reti CANopen multiple possono essere realizzate anche con l'impiego di profili procedurali. In questo caso i gateway

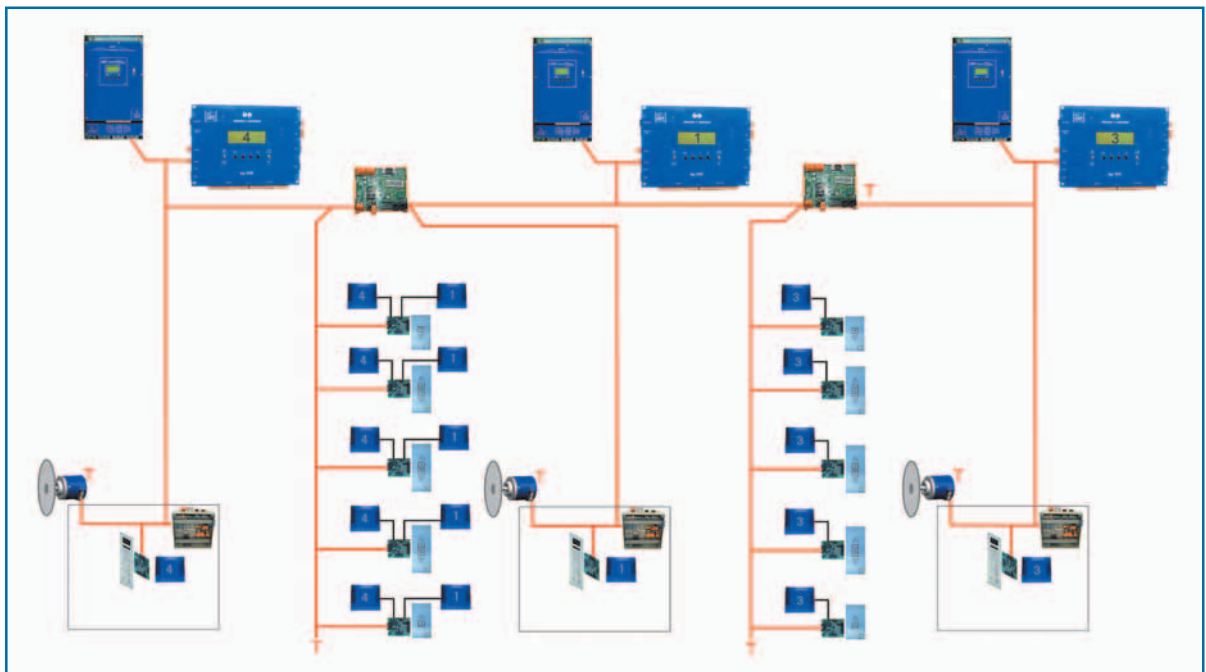


Fig. 6 - Il profilo applicativo CANopen Lift consente di descrivere una rete logica CANopen con gateway trasparenti, anche in cascata

Reti non gerarchiche

Il futuro appartiene alle reti non gerarchiche. Anche se sembra utopistico, i sistemi master/slave saranno via via sostituiti da sistemi decentralizzati. Tuttavia, questi controller moderni richiedono (almeno in parte) un'architettura di rete non gerarchica. In reti non gerarchiche, il gateway è aperto da entrambi i lati, ossia ciascun lato è in grado di dare inizio a comunicazioni verso l'altra rete. Queste reti CAN di tipo non gerarchico sono già utilizzate negli autoveicoli e anche in altri mezzi di trasporto. I gateway possono persino implementare interfacce multiple allo scopo di realizzare architetture di rete con topologia a maglia. Nelle applicazioni essi vengono distribuiti in modo da coprire più reti mentre i sistemi o le reti a bus

devono essere tuttavia standardizzati, dal momento che non sono trasparenti. Essi devono essere in grado di dare inizio alle corrispondenti funzioni di comunicazione sulla seconda rete CANopen tramite comandi predefiniti. Se le reti CANopen devono essere collegate in cascata, occorre ancora definire un protocollo di risoluzione inversa degli indirizzi CANopen (CARP, CANopen Address Resolution Protocol). Questo protocollo è in grado di tradurre gli ID dei nodi CANopen di una rete in ID di nodi CANopen inequivocabili dell'altra rete. Un protocollo simile può anche essere utilizzato per collegare reti Ethernet e CANopen. ■

CAN in Automation readerservice.it n. 48