Il protocollo Modbus

Massimo Giussani

Breve introduzione a uno dei più antichi e diffusi protocolli di comunicazione industriale

Modbus è un protocollo di comunicazione di alto livello (si posiziona al settimo livello della pila ISO-OSI) basato sullo scambio di messaggi tra dispositivi in modalità master/slave e client/server. Ha conosciuto un'ampia diffusione a livello mondiale, per via della semplicità di implementazione e della libera disponibilità delle specifiche. Lo sviluppo di una variante con incapsulamento TCP/IP e la successiva cessione delle specifiche da parte di Schneider a un'organizzazione no profit (Modbus-IDA) gli ha permesso di prosperare tra le emergenti reti di comunicazione basate su Ethernet Industriale. La varietà e la quantità di dispositivi che implementano Modbus come mezzo di comunicazione è la cartina di tornasole dello stato di salute di questo protocollo: tra sensori intelligenti, attuatori, motori, azionamenti, PLC, interfacce uomo macchina e computer industriali, sono letteralmente milioni le apparecchiature realizzate da produttori indipendenti che adottano questo standard. In questo articolo saranno delineate le caratteristiche essenziali di questo protocollo, con particolare riferimento al modello di comunicazione di tipo client/server.

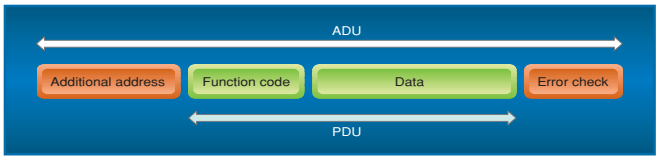

Fig. 1 - Frame Modbus: l'unità dati del protocollo (PDU) si trova nell'unità dati dell'applicazione (ADU) che contiene ulteriori informazioni per l'indirizzamento in rete e per il controllo degli errori

Master/Slave o Client/Server

Modbus consente il funzionamento in modalità half duplex e full duplex su reti seriali RS-485 e RS-232 e si integra senza problemi, nella sua incarnazione Modbus TCP/IP, nelle moderne reti Ethernet. Un qualunque computer connesso in rete può agire da client o server Modbus scambiando messaggi tramite la porta riservata 502 dello stack TCP/IP.
Nella modalità di comunicazione seriale l'interazione tra i dispositivi della rete Modbus è di tipo master/slave e sono previsti due tipi di trasmissione: Ascii e RTU. Nella modalità Ascii (American standard code for interchange of information) i singoli byte da otto bit che costituiscono il messaggio sono trasmessi sotto forma di due caratteri Ascii; durante il funzionamento in modalità RTU (Remote Terminal Unit) ogni byte viene invece suddiviso in due caratteri esadecimali da 4 bit. Chiaramente in questo modo, a parità di velocità di trasmissione, è possibile trasmettere una maggior quantità di informazioni rispetto al caso precedente. Il vantaggio della modalità Ascii sta nel fatto che sono ammesse pause fino a un secondo tra un carattere e l'altro senza che questo determini un errore di comunicazione.
La scelta del tipo di codifica dati da utilizzare viene tipicamente fatta dall'utente durante la configurazione dei dispositivi che compongono la rete, unitamente all'impostazione dei parametri di comunicazione della porta seriale.
La variante TCP aggiunge l'incapsulamento dei dati Modbus in un frame conforme alle specifiche TCP/IP, aprendo in questo modo la strada a tutti i dispositivi dotati di connettività Ethernet. Il principale vantaggio di questo approccio sta nella modalità di interazione tra i vari nodi della rete: essendo di tipo client/server, ogni dispositivo server è in grado di scambiare dati in maniera simultanea con più dispositivi client.

Il protocollo in pillole

Una tipica comunicazione via Modbus comprende essenzialmente tre stadi: la formulazione di una richiesta da parte di un dispositivo a un altro, l'esecuzione delle azioni necessarie a soddisfare la richiesta, e la restituzione al dispositivo iniziale delle informazioni risultanti, siano esse l'effettivo risultato dell'elaborazione o un codice di errore derivante dall'impossibilità di portare a termine il compito. Le informazioni vengono scambiate sotto forma di unità dati indipendenti dai livelli sottostanti nella pila ISO-OSI. Il mattone portante delle comunicazioni è Protocol Data Unit (PDU) costituita da un campo che contiene il codice funzione (codificato con un solo byte) e un campo dati di lunghezza variabile, eventualmente nulla, che contiene il corpo del messaggio. L'integrazione all'interno di altre reti può richiedere dei campi aggiuntivi che sono raccolti in quella che viene definita Application Data Unit (ADU); tipicamente è presente un campo indirizzi che consente di tracciare il dispositivo che ha effettuato la richiesta e un campo con i codici per la correzione degli errori. La figura 1 illustra il tipico frame Modbus, nelle sue due varianti.

Codici funzione e gestione delle eccezioni

Il codice funzione è un numero compreso tra 1 e 255, che specifica il tipo di azione che deve essere o è stata eseguita. I codici si possono riferire a funzioni già incluse nello standard (funzioni pubbliche) o a estensioni programmate dall'utente. Gli intervalli di valori relativi sono riassunti nella tabella 'Codici funzione'. Le funzioni pubbliche sono state convalidate dalla comunità di sviluppatori e utilizzatori Modbus, sono documentate nella RFC Ietf relativa allo standard Modbus e rappresentano una base condivisa da tutti i dispositivi che si conformano ad esso. Alcuni produttori possono decidere di implementare funzioni particolari utilizzando uno dei codici riservati agli utenti, cosa che può essere fatta senza dover richiedere l'approvazione dell'ente di riferimento: occore tuttavia tenere presente che questa libertà espone il prodotto a potenziali incompatibilità con applicazioni di terze parti che potrebbero utilizzare il medesimo codice per altri compiti.

Il protocollo prevede tre diversi tipi di PDU: richiesta (mb_req_pdu), risposta (mb_rsp_pdu) e risposta con eccezione (mb_excep_rsp_pdu). Un dispositivo client inoltra al server la propria richiesta di esecuzione di una determinata azione, sostanziandola, se il caso, con delle informazioni aggiuntive nel campo dati (un esempio potrebbe essere la richiesta di leggere la temperatura di un particolare sensore connesso alla sottorete gestita dal dispositivo server). Il dispositivo server, ricevuta la richiesta e verificata la validità del codice funzione, dell'informazione ad esso eventualmente allegata e dell'indirizzo del mittente, esegue l'operazione richiesta.

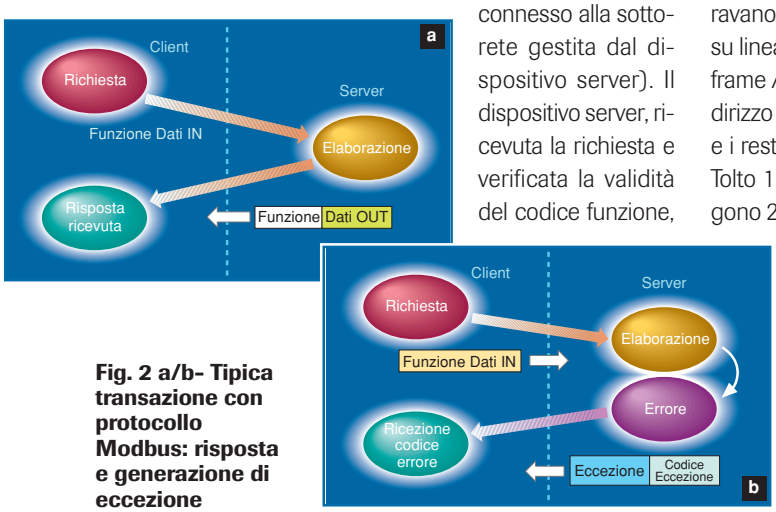

Fig. 2 a/b- Tipica transazione con protocollo Modbus: risposta e generazione di eccezione

Se l'operazione va a buon fine, il server crea un frame contenente le informazioni risultanti (il valore di temperatura memorizzato in un registro, ad esempio) e lo invia, assieme al relativo codice funzione, a chi ne aveva fatto richiesta (la figura 2 a); se la verifica dei dati o l'operazione hanno generato errori o non sono state portate a termine, viene invece restituito nel campo del codice funzione un codice di errore (pari al codice funzione della chiamata più 127) e nel campo dati un codice dell'eccezione che ha causato l'impossibilità a rispondere (figura 2 b).

TABELLA CODICI FUNZIONE DEL PROTOCOLLO MODBUS

Codice funzione	Tipo
0	*Non valido*
1 - 64	*Codici funzione pubblici*
65 - 72	*Codici funzione definiti dall'utente*
73 - 99	*Codici funzione pubblici*
100 - 110	*Codici funzione definiti dall'utente*
111 – 127	*Codici funzione pubblici*
128 -255	*Codici errore (codice funzione + 127)*

I dati: pochi ma buoni

Il campo dati passato dal client al server nella fase di richiesta può essere vuoto (nel caso in cui non siano necessarie ulteriori informazioni per portare a termine il compito indicato dal codice funzione) o può contenere una serie di informazioni che dettagliano il tipo di operazione da svolgere con un eventuale codice di sottofunzione, le variabili coinvolte, i registri da utilizzare, eventuali dati da trasferire nel dispositivo ricevente e via di seguito. Analogamente il campo dati restituito dal server riporterà tutte le informazioni pertinenti al compito svolto, con il tipo e numero di variabili restituite. La lunghezza del campo dati è limitata a un valore ereditato dalle prime versioni del protocollo che, lo ricordiamo, è stato sviluppato nel lontano 1979, quando ancora le memorie si misuravano in centinaia di byte e non in GB. La versione di Modbus su linea seriale può contare su una dimensione massima del frame ADU di 256 byte; di questi, 1 byte è utilizzato per l'indirizzo del server, 2 per il controllo ciclico di ridondanza (CRC) e i restanti 253 vanno a costituire l'unità dati del protocollo. Tolto 1 byte per il codice funzione/eccezione, ai dati rimangono 252 byte da occupare. La versione per TCP/IP richiede 7 byte aggiuntivi alla PDU, andando a occupare un totale di 260 byte. La codifica utilizzata per la trasmissione di numeri che richiedono più byte è di tipo big-endian, ossia i byte più significativi sono inviati per primi.

I dati passati nel messaggio possono essere di diverso tipo e sono organizzati in tabelle. Le quattro tabelle principali sono: ingressi discreti e registri di ingresso (rispettivamente da 1 e 16 bit, in sola lettura), registri per la memorizzazione (16 bit, in lettura e scrittura) e avvolgimenti (un bit in lettura e scrittura).

L'indirizzamento dei dati è limitato a 16 bit, cosa che comporta un range di valori per gli indirizzi compreso tra 0 e 65535 per ciascun blocco. Ogni produttore di dispositivi Modbus deve dotare il proprio prodotto di un mezzo per tradurre (rimappare) i valori impiegati nelle effettive locazioni di memoria utilizzate per memorizzare i dati di funzionamento. In questo modo si crea un livello di astrazione che permette agli utilizzatori Modbus di accedere ai parametri di funzionamento del dispositivo in maniera trasparente e indipendente dai dettagli della particolare implementazione. ■