



**LSI LASTEM S.r.l.**

Via Ex S.P. 161 Dosso, n.9 - 20090 Settala Premenugo (MI) - Italia

**Tel.:** (+39) 02 95 41 41

**Fax:** (+39) 02 95 77 05 94

**e-mail:** info@lsi-lastem.it

**WEB:** <http://www.lsi-lastem.it>

**CF./P. Iva:** (VAT) IT-04407090150

**REA:**1009921 **Reg.Imprese:** 04407090150



# *Data logger*



## **Protocolli di comunicazione**

**Aggiornamento 28/05/2018**

Copyright 2010-2018 LSI LASTEM. Tutti i diritti riservati.

Le informazioni contenute nel presente manuale sono soggette a modifiche senza preavviso.  
Nessuna parte di questo manuale può essere riprodotta in qualsiasi forma o mezzo elettronico o meccanico, per alcun uso, senza il permesso scritto di LSI LASTEM.

LSI LASTEM si riserva il diritto di intervenire sul prodotto, senza l'obbligo di aggiornare tempestivamente questo documento.

# Sommario

1	Introduzione.....	4
2	Protocollo TTY.....	4
2.1	Descrizione dei messaggi.....	4
2.2	Formato dei messaggi di richiesta.....	4
2.2.1	Aggiornamento dell’ora dello strumento .....	5
2.3	Formato dei messaggi di risposta.....	5
2.3.1	Calcolo del campo <i>checksum</i> .....	5
2.3.2	Risposta di conferma semplice .....	6
2.3.3	Valori istantanei .....	6
2.3.4	Dati anagrafici.....	6
2.3.5	Dati diagnostici e statistici .....	7
3	Protocollo Modbus.....	8
3.1	Formato dei messaggi .....	8
3.1.1	Indirizzo .....	8
3.1.2	Codice funzione .....	9
3.1.3	CRC16.....	9
3.2	Funzioni supportate.....	9
3.2.1	Read Coils .....	9
3.2.2	Read Holding Registers .....	12
3.2.3	Read Input Register.....	12
3.2.4	Write Single Coil .....	17
3.2.5	Write Multiple Coils .....	19
3.2.6	Write Multiple Registers.....	20
3.2.7	Read Device Identification.....	22
3.3	Codici eccezione .....	24
3.4	Modalità Master .....	25

# 1 Introduzione

Gli strumenti E-Log e M-Log supportano diversi protocolli di comunicazione utilizzabili dalle due porte di comunicazione seriale. In questo manuale vengono descritti i protocolli *TTY* e *Modbus* disponibili sulla porta di comunicazione nr. 2.

## 2 Protocollo TTY

Tramite il protocollo TTY è possibile interrogare lo strumento per ottenere in modo semplice i dati campionati e calcolati dalle misure (valori istantanei) ed altre informazioni utili. La richiesta dei valori misurati avviene in modalità *polling* (su comando dell'host), oppure in modalità spontanea con rata di trasmissione programmabile.

I dati sono trasmessi in un formato di semplice interpretazione, sia direttamente dall'operatore tramite programma terminale, sia da parte di programmi di decodifica o importazione automatica.

L'interrogazione dello strumento avviene specificando nel comando di richiesta l'indirizzo di rete; ciò consente di eseguire l'interrogazione in sequenza di diversi strumenti connessi mediante un'unica linea di comunicazione.

### 2.1 Descrizione dei messaggi

Nelle descrizioni successive saranno utilizzati i caratteri < e > per delimitare i vari campi che costituiscono i messaggi di richiesta e risposta. Si tenga presente che questi caratteri sono qui utilizzati a scopo descrittivo e non compaiono perciò nei veri messaggi scambiati durante la comunicazione.

### 2.2 Formato dei messaggi di richiesta

L'invio del comando di richiesta allo strumento avviene tramite la seguente sequenza di caratteri:

```
<A><addr><cmd><prm><CR>
```

dove:

<A>: carattere ASCII 'A' (esadecimale 41);

<addr>: indirizzo dello strumento, espresso dal carattere ASCII da '1' (esadecimale 31) a '9' (esadecimale 39); lo strumento risponde solo se esiste corrispondenza tra l'indirizzo programmato tramite *3DOM* (parametro *Indirizzo di rete*) e questo valore;

<cmd>: codice del comando; un carattere ASCII a scelta tra:

- <I>: richiesta dei valori istantanei delle misure;
- <S>: richiesta dei dati anagrafici dello strumento;
- <T>: aggiornamento dell'ora (E-Log supporta questa funzione dalla versione 2.01.04);

- <A>: avvio della modalità di invio spontaneo dei dati istantanei; lo strumento inizia a trasmettere i dati istantanei delle misure, con rata temporale configurabile tramite il programma *3DOM*; se questo parametro è impostato a zero l'invio del comando determina l'invio automatico con rata fissa di 10 secondi; in caso di riavvio dello strumento l'automatismo non è riattivato; per ottenere la trasmissione automatica anche dopo il riavvio dello strumento, programmare con *3DOM* il tempo di invio automatico con un valore diverso da zero;
- <M>: termine della modalità di invio spontaneo dei dati istantanei; il riavvio dello strumento determina nuovamente la trasmissione automatica dei dati solo se è programmato un tempo di trasmissione diverso da zero;
- <D>: richiesta dei dati diagnostici e statistici di funzionamento dello strumento;
- <R>: azzeramento dei dati diagnostici e statistici di funzionamento dello strumento; si noti che l'azzeramento comprende, oltre ai dati inviati come risposta al comando <D>, l'azzeramento di tutte le altre informazioni statistiche (visualizzate dalle maschere diagnostiche; per maggiori informazioni fare riferimento al manuale dello strumento in uso).

<prm>: parametri opzionali alla richiesta.

<CR>: carattere ASCII *carriage return*.

Si noti che il riavvio dello strumento determina automaticamente l'avvio della trasmissione spontanea dei dati istantanei solo se la rata di trasmissione è impostata con valore diverso da zero, anche senza l'invio del comando <A>.

Tutti i comandi sopra elencati devono essere inviati in modo completo, in un periodo temporale in cui lo strumento non esegua l'invio automatico dei dati; in caso contrario lo strumento non è in grado di interpretare il comando inviatogli.

## 2.2.1 Aggiornamento dell'ora dello strumento

La richiesta di aggiornamento dell'ora dello strumento avviene tramite una richiesta così costituita:

<addr><T><P|I><yy/mm/dd hh:mm:ss>

dove:

<P|I>: da utilizzare in alternativa; "P" indica che la regolazione dell'ora da parte dello strumento avviene in modo progressivo; "I" determina invece la correzione in modo immediato;

<yy/mm/dd hh:mm:ss> sono la data e l'ora da impostare; la sequenza numerica va rispettata, mentre i separatori possono essere anche sostituiti da altri caratteri non numerici.

## 2.3 Formato dei messaggi di risposta

### 2.3.1 Calcolo del campo *checksum*

Ogni messaggio di risposta contenente dati approntati dallo strumento include, in coda, un campo di controllo della trasmissione (*checksum*), utile per validare i dati ricevuti rispetto a quelli effettivamente trasmessi.

Il campo *checksum* è contenuto fra due parentesi quadre ed è espresso in formato esadecimale ASCII, corrispondente alla somma ad 8 bit in OR esclusivo di tutti i caratteri compresi fra il primo carattere ed il carattere immediatamente precedente la parentesi quadra aperta.

### **2.3.2 Risposta di conferma semplice**

Nel caso di accettazione positiva del comando a cui non debbano seguire dati (come avviene per le risposte ai comandi <M> e <R>), lo strumento invia un messaggio di risposta contenente il testo <Ok>; in caso di rifiuto o non riconoscimento del comando ricevuto, o per qualsiasi altro errore, il messaggio di risposta contiene il testo <Err>.

### **2.3.3 Valori istantanei**

La richiesta dei valori istantanei delle misure determina l'invio di una risposta così costituita:

```
<addr><I><nnn.nn>;<nnn.nn>;...<[CS]><CR><LF>
```

dove:

<addr>: indirizzo dello strumento;

<I>: codice del comando;

<nnn.nn>: valore istantaneo (ultimo campionato o calcolato) della *n*-esima misura, dove *n* è la posizione del valore rispetto agli altri contenuti nello stesso messaggio; il numero di caratteri della parte intera è variabile in base al valore assunto da ogni specifica misura; il numero di caratteri decimali dipende dalla precisione programmata per la misura; il massimo numero di misure contenute in un unico messaggio è 25; nel caso in cui lo strumento sia programmato con più di 25 misure, alla primo messaggio seguono ulteriori messaggi preceduti dal suffisso <M<sub>xx</sub>>, dove *xx* corrisponde all'indice (base 0) della prima misura appartenente al gruppo di misure nel messaggio;

<[CS]>: campo *checksum*;

<CR>: carattere ASCII *carriage return*;

<LF>: carattere ASCII *line feed*.

### **2.3.4 Dati anagrafici**

La richiesta dei dati anagrafici dello strumento determina l'invio di una risposta così costituita:

```
<addr><S><M:ccc/sss>;<V:MM.mm.bb>;<S:fff/uuu><[CS]><CR><LF>
```

dove:

<addr>: indirizzo dello strumento;

<S>: codice del comando;

<M:ccc/sss>: modello dello strumento (ccc=codice, sss=sottocodice);

<V:MM.mm.bb>: versione del firmware (MM=versione maggiore, mm=versione minore, bb=versione interna);

<S:fff/uuu>: numero di serie (fff=di fabbrica, uuu=valore definito dall'utente).

<[CS]>: campo *checksum*;  
<CR>: carattere ASCII *carriage return*;  
<LF>: carattere ASCII *line feed*.

### 2.3.5 Dati diagnostici e statistici

La richiesta dei dati diagnostici e statistici determina l'invio di una risposta così costituita:

<addr><D><L:l1l1>;<S:sss>;<C:ccc>;<E:eee><[CS]><CR><LF>

dove:

<addr>: indirizzo dello strumento;  
<D>: codice del comando;  
<L:l1l1>: tempo di funzionamento ininterrotto (di vita) dello strumento, espresso nel formato *ddd hh:mm:ss*;  
<S:sss>: data/ora dell'ultimo azzeramento dei dati statistici, espresso nel formato *yy/mm/dd hh:mm:ss*;  
<C:ccc>: data/ora di sistema (corrente) dello strumento, espresso nel formato *yy/mm/dd hh:mm:ss*;  
<E:eee>: codice esadecimale degli errori di funzionamento riscontrato dallo strumento (per maggiori informazioni fare riferimento al manuale dello strumento in uso);  
<[CS]>: campo *checksum*;  
<CR>: carattere ASCII *carriage return*;  
<LF>: carattere ASCII *line feed*.

## 3 Protocollo Modbus

Modbus è un protocollo di comunicazione seriale molto utilizzato in ambito industriale per consentire la comunicazione tra un *master* (solitamente un PC o un sistema SCADA) ed uno o più *slave* (strumentazioni di misura, di controllo o PLC), connessi alla stessa rete. Modbus definisce come il *master* e gli *slave* instaurano ed interrompono la comunicazione, come vengono scambiati i messaggi e come vengono rilevati gli errori. Solo il *master* può iniziare la comunicazione.

A ogni dispositivo della rete è assegnato un indirizzo univoco. Un comando Modbus contiene l'indirizzo del dispositivo destinatario verso il quale è diretto il messaggio. Solo il dispositivo indirizzato risponderà al comando, sebbene anche gli altri strumenti ricevano il medesimo messaggio. Tutti i comandi Modbus contengono informazioni di controllo, che assicurano che il comando arrivato sia corretto.

Dal punto di vista di Modbus il data logger si pone nel sistema come apparato sia *master* che *slave* ed implementa parte del protocollo in versione RTU.

Il data logger è in grado di comunicare via seriale con altri dispositivi che supportano la parità impostata a *Nessuno* (None).

### 3.1 Formato dei messaggi

Il formato dei messaggi utilizzato dai dispositivi master/slave è il seguente:

Nome campo	Dimensione	Descrizione
Indirizzo	1 byte	L'indirizzo dello slave con cui il master deve comunicare (indirizzo di rete dello strumento); non è supportato il messaggio <i>broadcast</i> (ID = 0)
Codice funzione	1 byte	Comando da eseguire (o eseguito)
Dati	n byte	Dati che devono essere trasmessi
CRC16	2 byte	Controllo dell'errore secondo l'algoritmo CRC16

Se un dispositivo slave individua un errore nel messaggio ricevuto (errore di formato o nel CRC16) il messaggio viene considerato non valido e scartato, di conseguenza non sarà eseguita l'azione e non sarà data risposta al comando, così come se l'indirizzo non corrisponde al dispositivo interrogato.

#### 3.1.1 Indirizzo

L'indirizzo è utilizzato per identificare il destinatario del messaggio: esso contiene l'indirizzo numerico dello slave selezionato. Può assumere valori da 1 a 200 se utilizzato in una rete RS-232. Lo strumento confronta il valore indirizzo ricevuto con l'indirizzo di rete programmato e risponde solo se corrispondenti. Utilizzare il programma *3DOM* è modificare il parametro *Indirizzo di rete* in base a necessità.

I messaggi di tipo *broadcast* (indirizzo uguale a 0) non sono supportati.



### 3.1.2 Codice funzione

Il codice funzione identifica il comando che deve essere eseguito o che è appena stato gestito dallo slave.

I codici funzione supportati dallo strumento sono i seguenti:

<i>Codice funzione</i>	<i>Nome Funzione</i>	<i>Descrizione</i>
01	Read Coils	Legge lo stato degli attuatori e lo stato di funzionamento dello strumento
03	Read Holding Registers	Come <i>Read Input Registers</i>
04	Read Input Registers	Legge gli ultimi valori acquisiti dallo strumento (sia misure analogiche che digitali) e la data ora di sistema
05	Write Single Coil	Imposta lo stato di un attuatore dello strumento
15	Write Multiple Coils	Azzera gli errori di funzionamento dello strumento
10	Write Multiple Registers	Imposta la data ora di sistema e alcuni parametri di configurazione delle misure
43/14	Read Device Identification	Legge le informazioni dello strumento

Qualsiasi altro codice funzione non inserito in tabella viene ignorato dallo strumento, il quale a fronte della ricezione di un comando non supportato non genera alcuna risposta di errore.

### 3.1.3 CRC16

Il CRC16 contiene il codice di ridondanza ciclica (Cyclic Redundancy Check) calcolato utilizzando l'algoritmo CRC16 (polinomio  $X^{16} + X^{15} + X^2 + 1$ ). Per il calcolo di questi due caratteri il messaggio (indirizzo, codice funzione e dati scartando i bit di start, stop e l'eventuale parità) viene considerato come un unico numero binario continuo di cui il bit più significativo (MSB) viene trasmesso prima.

## 3.2 Funzioni supportate

### 3.2.1 Read Coils

Utilizzare la funzione *Read Coils* per leggere lo stato degli attuatori dello strumento ed eventuali errori di funzionamento.

Il significato dei valori assunti dagli attuatori dipende dalla logica di funzionamento degli attuatori impostata sullo strumento (per maggiori informazioni fare riferimento al manuale dello strumento in uso). La tabella seguente ne riassume il significato.

<i>Tipo di logica di funzionamento</i>	<i>Valore</i>	<i>Stato</i>	<i>Significato</i>
A basso consumo	0	OFF	Uscita attuazione spenta
	1	ON	Uscita attuazione accesa
In sicurezza	0	ON	Uscita attuazione accesa
	1	OFF	Uscita attuazione spenta

Per quanto riguarda gli errori di funzionamento, invece, il valore 0 indica *non in errore* mentre il valore 1 indica *errore*.

<b>Indirizzo (Hex)</b>	<b>Coil</b>	<b>Significato</b>
<b>Stato Attuatori</b>		
0x00	01	Stato attuatore nr. 1
	02	Stato attuatore nr. 2
	03	Stato attuatore nr. 3
	04	Stato attuatore nr. 4
	05	Stato attuatore nr. 5
	06	Stato attuatore nr. 6
	07	Stato attuatore nr. 7
	08	<i>Non utilizzato</i>
<b>Stato funzionamento (errori)</b>		
0x01	09	Errore Read After Write
	10	Configurazione memorizzata non valida
	11	Errore di ricerca della pagina di memoria
	12	Nr. massimo di loop superato durante ricerca dati in memoria
	13	Overrun coda di accettazione richieste di acquisizione
	14	Overrun coda di memorizzazione risultati di acquisizione
	15	<i>Non utilizzato</i>
16	Scrittura di pagina in memoria fallita	
0x02	17	Una o più pagine di dati di elaborazione perse
	18	Cancellazione settore di memoria fallita
	19	Timeout su attesa termine operazione in memoria
	20	Dispositivo di memoria non supportato
	21	Messaggio ricevuto errato (codice di controllo non valido)
	22	Messaggio ripetuto fino ad un massimo di 3 volte
	23	A seguito della ripetizione massima del messaggio questo è stato perso
	24	Errore riscontrato durante la scrittura/lettura della EEPROM interna allo slave
0x03	25	Lo slave non è in grado di operare in quanto la sua config. non è programmata
	26	C'è stato un overflow sulla ricezione di un singolo messaggio
	27	Coda dei messaggi piena nello slave
	28	Raggruppa tutti gli errori CISS relativi alla sintassi
	29	Errore CISS non specificato (Unspecified)
	30	Errore CISS (BadCommandCode)
	31	Errore CISS (BadParameter)
	32	Errore CISS (ParameterOutOfRange)
0x04	33	Errore CISS (UnrecognizedCDV)
	34	Errore CISS (BeyondMaxClassLevel)
	35	Errore CISS (ParameterIndexOutOfRange)
	36	Errore CISS (ClassIndexOutOfRange)
	37	Errore CISS (RequestNotPermitted)
	38	<i>Non utilizzato</i>
	39	<i>Non utilizzato</i>
40	<i>Non utilizzato</i>	

**Richiesta**

<i>Codice funzione</i>	1 byte	0x01
<i>Indirizzo inizio</i>	2 byte	da 0x00 a 0x27
<i>Numero di coils</i>	2 byte	da 1 a 40

**Risposta**

<i>Codice funzione</i>	1 byte	0x01
<i>Numero di byte</i>	1 byte	1
<i>Stato attuatore/errore</i>	1 byte	1=On, 0=Off

**Errore**

<i>Codice funzione</i>	1 byte	0x81
<i>Codice eccezione</i>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Esempio:** richiesta di lettura dello stato degli attuatori dello strumento con ID uguale a 01:

<i>Richiesta</i>	
<i>Nome campo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	01
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	00
Numero di attuatori (Hi)	00
Numero di attuatori (Lo)	08
CRC16 (Hi)	3D
CRC16 (Lo)	CC

Risposta alla richiesta con i seguenti valori: 0, 0, 1, 0, 0, 0, 0, 0:

<i>Risposta</i>	
<i>Nome campo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	01
Numero di byte	01
Valore	04
CRC16 (Hi)	50
CRC16 (Lo)	4B

### 3.2.2 Read Holding Registers

*Read Holding Register* e *Read Input Register*, dal punto di vista dell'implementazione Modbus nel data logger, si equivalgono e sono perciò trattati nello stesso modo. Nella richiesta indicare il codice funzione 0x03(*Read Holding Register*) oppure 0x04 (*Read Input Register*).

Per maggiori informazioni in merito fare riferimento alla funzione *Read Input Register*.

### 3.2.3 Read Input Register

La funzione *Read Input Registers* permette di leggere gli ultimi valori acquisiti dallo strumento (sia misure analogiche che digitali) e la data ora di sistema. I valori acquisiti possono essere letti in formato floating point richiedendoli dall'indirizzo 0 (0x0000) ed in formato binario intero dall'indirizzo 1000 (0x03E8), mentre la data ora è disponibile all'indirizzo 2000 (0x07D0).

L'indirizzo da impostare per ciascuna richiesta è il seguente:

<i>Indirizzo (Hex)</i>	<i>Numero di registri</i>	<i>Significato</i>
<b><i>Valori float IEEE754</i></b>		
0x0000	2	Valore della misura 1
0x0002	2	Valore della misura 2
...	...	...
0x00C4	2	Valore della misura 99
<b><i>Valori interi (WORD)</i></b>		
0x03E8	1	Valore della misura 1
0x03E9	1	Valore della misura 2
...	...	...
0x044B	1	Valore della misura 99
<b><i>Data ora di sistema (yy MM dd hh mm ss)</i></b>		
0x07D0	1	yy MM
0x07D1	1	dd hh
0x07D2	1	mm ss

#### **Valori espressi in formato a virgola mobile**

Ogni misura trasmessa in virgola mobile è composta da 4 byte ovvero da 2 registri Modbus. Essi sono espressi in base al formato floating point indicato dalla norma IEEE754.

Tramite il programma 3DOM è possibile scegliere il formato dei dati fra *Big Endian* (impostazione di default) e *Little Endian* (opzione *Inversione valore a virgola mobile* disponibile nella maschera di impostazione dei parametri di comunicazione della porta seriale 2). La memorizzazione del dato nel primo caso, inizia dal byte più significativo (MSB – Most Significant Byte) mentre nel secondo inizia dal byte meno significativo (LSB – Least Significant Byte). Il valore 11,0 viene quindi memorizzato, a partire dall'indirizzo 0x20 ad esempio, come segue:

			<b>Indirizzo di memoria (Hex)</b>	<b>Valore in formato Big Endian (Hex)</b>	<b>Valore in formato Little Endian (Hex)</b>
			...		
Valore 11,0	Registro 1	Byte 1	0x20 (Hi)	00	30
		Byte 2	0x20 (Lo)	00	41
	Registro 2	Byte 1	0x21 (Hi)	41	00
		Byte 2	0x21 (Lo)	30	00
			...		

Essendo il frame composto da 255 caratteri, è possibile richiedere al massimo 60 misure per ogni messaggio di richiesta. Per richiedere tutte le 99 misure potenzialmente disponibili nel data logger si rende quindi necessario effettuare due richieste distinte.

Il valore -999999 (0xF02374C9), se non diversamente specificato (cap. 3.2.6), corrisponde a *misura in errore*.

### Richiesta

<b>Codice funzione</b>	1 byte	0x04
<b>Indirizzo inizio</b>	2 byte	da 0x0000 a 0x00C4
<b>Numero di registri</b>	2 byte	da 2 a 198 (max 120)

### Risposta

<b>Codice funzione</b>	1 byte	0x04
<b>Numero di byte</b>	1 byte	2 x N*
<b>Valore</b>	2 byte x N*	

\*N = numero di misure.

### Errore

<b>Codice funzione</b>	1 byte	0x84
<b>Codice eccezione</b>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

### Area dati

<b>Nr. Misura</b>	1	2	3	4	5	6	...	99
<b>Indirizzo (hex)</b>	0x00	0x02	0x04	0x06	0x08	0x0A	...	0x0C4

**Esempio:** richiesta di lettura dei valori, nel formato float, delle misure 3 e 4 dello strumento con ID uguale a 01:

<b>Richiesta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	04
Numero di uscite (Hi)	00
Numero di uscite (Lo)	04
CRC16 (Hi)	B0
CRC16 (Lo)	08

La prima misura da leggere è la numero 3 quindi, come indicato nell'*Area dati*, l'indirizzo di inizio da impostare è 0x04 mentre il numero di uscite (registri) è 0x04 (2 misure per 2 registri).

Risposta alla richiesta con i valori 99.0 per la misura 3 (valore uscita 1) e 101.0 per la misura 4 (valore uscita 2):

<b>Risposta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Numero di byte	08
Valore uscita 1 (byte 1)	00
Valore uscita 1 (byte 2)	00
Valore uscita 1 (byte 3)	42
Valore uscita 1 (byte 4)	C6
Valore uscita 2 (byte 1)	00
Valore uscita 2 (byte 2)	00
Valore uscita 2 (byte 3)	42
Valore uscita 2 (byte 4)	C4
CRC16 (Hi)	13
CRC16 (Lo)	C9

**Valori espressi in formato binario intero**

Nel formato binario ogni misura è di tipo intero, quindi composta da 2 byte ovvero da un registro Modbus. Il formato dei dati è *Little Endian*. La memorizzazione del dato inizia dal byte meno significativo (LSB – Least Significant Byte). Il valore 1149 viene quindi memorizzato, a partire dall'indirizzo 0x03E8 ad esempio, come segue:

			<b>Indirizzo di memoria (Hex)</b>	<b>Valore in formato Little Endian (Hex)</b>
			...	
Valore 1149	Registro 1	Byte 1	0x03E8 (Hi)	05
		Byte 2	0x03E8 (Lo)	45

Fino alla V. 2.36.00 di E-Log e alla V. 2.15.00 di M-Log per non perdere la parte decimale della misura era possibile applicare a ciascun valore letto, un offset ed un guadagno, impostando

rispettivamente *Costante matematica 1* e *Costante matematica 2* delle grandezze standard con il programma 3DOM. In tal modo, avendo ad esempio impostato un offset pari a 0 e un guadagno pari a 100, il valore letto 11,49 diventerebbe  $(11,49 + 0) * 100 = 1149$ . Per ottenere il valore corretto, la misura ricevuta va divisa per il guadagno ed eventualmente sottratta all'offset. Dalle versioni di programma successive questa operazione viene fatta basandosi sulla precisione della misura. Il valore letto 23.45 della misura con due valori decimali diventerebbe 2345 mentre il valore 68.4 con un valore decimale diventerebbe 684.

Al contrario del formato float, per il tipo intero è possibile ottenere tutte le 99 misure potenzialmente gestibili dal data logger tramite una singola richiesta.

Il valore -1 (0xFFFF), se non diversamente specificato (cap. 3.2.6), corrisponde a *misura in errore*.

**Richiesta**

<i>Codice funzione</i>	1 byte	0x04
<i>Indirizzo inizio</i>	2 byte	da 0x03E8 a 0x044B
<i>Numero di registri</i>	2 byte	da 1 a 99

**Risposta**

<i>Codice funzione</i>	1 byte	0x04
<i>Numero di byte</i>	1 byte	2 x N*
<i>Valore</i>	2 byte x N*	

\*N = numero di misure.

**Errore**

<i>Codice funzione</i>	1 byte	0x84
<i>Codice eccezione</i>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Area dati**

<i>Nr. Misura</i>	1	2	3	4	...	99
<i>Indirizzo (hex)</i>	0x03E8	0x03E9	0x03EA	0x03EB	...	0x044B

**Esempio:** richiesta di lettura dei valori, nel formato intero, della misura 3 dello strumento con ID uguale a 01:

<b>Richiesta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Indirizzo inizio (Hi)	03
Indirizzo inizio (Lo)	EA
Numero di uscite (Hi)	00
Numero di uscite (Lo)	01
CRC16 (Hi)	A5
CRC16 (Lo)	BA

La misura da leggere è la numero 3 quindi, come indicato nell'*Area dati*, l'indirizzo di inizio da impostare è 0x03EA mentre il numero di uscite (registri) è 0x01 (1 misura per 1 registro).

Risposta alla richiesta con il valore 1343:

<b>Risposta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Numero di byte	02
Valore uscita 1 (byte 1)	05
Valore uscita 1 (byte 2)	3F
CRC16 (Hi)	FB
CRC16 (Lo)	04

### **Data ora dello strumento**

#### **Richiesta**

<b>Codice funzione</b>	1 byte	0x04
<b>Indirizzo inizio</b>	2 byte	da 0x07D0 a 0x07D2
<b>Numero di registri</b>	2 byte	da 1 a 3

#### **Risposta**

<b>Codice funzione</b>	1 byte	0x04
<b>Numero di byte</b>	1 byte	2 x N*
<b>Valore</b>	2 byte x N*	

\*N = numero di registri.

La sequenza dei campi data/ora a partire dall'indirizzo 0x07D0 è: yy MM dd hh mm ss.



**Errore**

<b>Codice funzione</b>	1 byte	0x84
<b>Codice eccezione</b>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Esempio:** richiesta di lettura della data ora dello strumento con ID uguale a 01:

<b>Richiesta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Indirizzo inizio (Hi)	07
Indirizzo inizio (Lo)	D0
Numero di uscite (Hi)	00
Numero di uscite (Lo)	03
CRC16 (Hi)	B0
CRC16 (Lo)	86

Risposta alla richiesta con data ora uguale a 10/06/08 10:40:03:

<b>Risposta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	04
Numero di byte	06
Valore uscita 1 (byte 1)	0A
Valore uscita 1 (byte 2)	06
Valore uscita 2 (byte 1)	08
Valore uscita 2 (byte 2)	0A
Valore uscita 3 (byte 1)	28
Valore uscita 3 (byte 2)	03
CRC16 (Hi)	94
CRC16 (Lo)	5A

### **3.2.4 Write Single Coil**

Utilizzare la funzione *Write Single Coil* per impostare lo stato degli attuatori dello strumento (uscite digitali).

Il valore *0x0000* imposta l'uscita dell'attuatore a *0* mentre *0xFF00* la imposta ad *1*. Normalmente lo stato *0* indica attuatore spento ed *1* attuatore acceso. Se però è impostata sullo strumento la logica di funzionamento degli attuatori in sicurezza, gli stati sono invertiti e cioè *0* indica attuatore acceso ed *1* attuatore spento (per maggiori informazioni fare riferimento al manuale dello strumento in uso).

**Richiesta**

<i>Codice funzione</i>	1 byte	0x05
<i>Indirizzo inizio</i>	2 byte	da 0x0000 a 0x0007
<i>Valore</i>	2 byte	0x0000 o 0xFF00

**Risposta**

<i>Codice funzione</i>	1 byte	0x01
<i>Indirizzo inizio</i>	2 byte	da 0x0000 a 0x0007
<i>Valore</i>	2 byte	0x0000 o 0xFF00

**Errore**

<i>Codice funzione</i>	1 byte	0x85
<i>Codice eccezione</i>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Esempio:** richiesta di impostare a 0 lo stato dell'attuatore numero 3 dello strumento con ID uguale a 01:

<b>Richiesta</b>	
<i>Nome campo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	05
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	02
Valore (Hi)	00
Valore (Lo)	00
CRC16 (Hi)	6C
CRC16 (Lo)	0A

<b>Risposta</b>	
<i>Indirizzo dispositivo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	05
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	02
Valore (Hi)	00
Valore (Lo)	00
CRC16 (Hi)	6C
CRC16 (Lo)	0A

### 3.2.5 Write Multiple Coils

Utilizzare la funzione *Write Multiple Coils* per azzerare eventuali errori di funzionamento riscontrati con la funzione *Read Coils*. Il comando agisce su tutti i bit insieme; non è possibile azzerare uno o una sola parte di errori. Il comando, come descritto di seguito, azzerà tutti gli errori verificatisi.

#### Richiesta

<i>Codice funzione</i>	1 byte	0x0F
<i>Indirizzo inizio</i>	2 byte	0x0000
<i>Numero di coil</i>	2 byte	32
<i>Numero di byte</i>	1 byte	4
<i>Valore</i>	4 byte	0x0000

#### Risposta

<i>Codice funzione</i>	1 byte	0x0F
<i>Indirizzo inizio</i>	2 byte	0x0000
<i>Numero di coil</i>	2 byte	32

#### Errore

<i>Codice funzione</i>	1 byte	0x8F
<i>Codice eccezione</i>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Esempio:** richiesta di azzeramento degli errori di funzionamento dello strumento con ID uguale a 01:

<i>Richiesta</i>	
<i>Nome campo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	0F
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	00
Numero di coil (Hi)	00
Numero di coil (Lo)	20
Numero di byte	04
Valore 1 (Hi)	00
Valore 1 (Lo)	00
Valore 2 (Hi)	00
Valore 2 (Lo)	00
CRC16 (Hi)	C4
CRC16 (Lo)	88

<i>Risposta</i>	
<i>Nome campo</i>	<i>(Hex)</i>
Indirizzo dispositivo	01
Codice funzione	0F
Indirizzo inizio (Hi)	00
Indirizzo inizio (Lo)	00
Numero di coil (Hi)	00
Numero di coil (Lo)	20
CRC16 (Hi)	21
CRC16 (Lo)	79

### 3.2.6 Write Multiple Registers

Utilizzare la funzione *Write Multiple Registers* per impostare la data ora dello strumento e alcuni parametri di configurazione delle misure.

L'indirizzo da utilizzare per ciascuna richiesta è il seguente:

<i>Indirizzo (Hex)</i>	<i>Numero di registri</i>	<i>Significato</i>
<i>Data ora di sistema (yy mm dd hh mm ss)</i>		
0x07D0	3	yy mm dd hh mm ss
<i>Configurazione misure</i>		
0x07DA	5	Valore assunto per misura in errore per tipo decimale binario, floating point e attivazione misura (fino max 32)

I parametri di configurazione delle misure che possono essere modificati sono:

- Valore da attribuire alla misura in errore per misure trasmesse in formato WORD (2 byte = 1 registro);
- Valore da attribuire alla misura in errore per misure trasmesse in formato float (4 byte = 2 registri); ATTENZIONE: utilizzare il formato *Little Endian*;
- Attivazione della misura (un bit per ogni stato di attivazione della misura per un massimo di 32 misure; 4 byte = 2 registri); la misura non attivata verrà considerata in errore.

Questi parametri sono memorizzati in modo definitivo fino alla loro prossima reimpostazione o fino alla riconfigurazione dello strumento da parte di 3DOM. In questo caso essi sono impostati con valori di default: -1 per i valori decimale binario, -999999 per i valori floating point, tutte le misure abilitate.

La scrittura deve essere eseguita su tutti i 5 registri per intero.

**Esempio 1:** richiesta di modifica della data ora dello strumento con il valore 09/06/10 16:03:05 (formato yy/mm/dd hh:mm:ss):

<b>Richiesta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	10
Indirizzo inizio (Hi)	07
Indirizzo inizio (Lo)	D0
Numero di registri (Hi)	00
Numero di registri (Lo)	03
Numero di byte	06
Valore 1 (Hi)	0A
Valore 1 (Lo)	06
Valore 2 (Hi)	09
Valore 2 (Lo)	10
Valore 3 (Hi)	03
Valore 3 (Lo)	05
CRC16 (Hi)	B2
CRC16 (Lo)	5D

<b>Risposta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	10
Indirizzo inizio (Hi)	07
Indirizzo inizio (Lo)	D0
Numero di coil (Hi)	00
Numero di coil (Lo)	03
CRC16 (Hi)	80
CRC16 (Lo)	85

**Esempio 2:** richiesta di modifica dei parametri delle misure con i seguenti valori: -12345 (0xCFC7) per valore in errore per il tipo decimale binario, -12345678 (0x4E613CCB) per valore in errore per il tipo float e disattivazione di tutte le misure ad eccezione delle prime 3:

<b>Richiesta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	10
Indirizzo inizio (Hi)	07
Indirizzo inizio (Lo)	DA
Numero di registri (Hi)	00
Numero di registri (Lo)	05
Numero di byte	0A
Valore 1 (Hi)	C7
Valore 1 (Lo)	CF
Valore 2 (Hi)	4E
Valore 2 (Lo)	61
Valore 3 (Hi)	3C
Valore 3 (Lo)	CB
Valore 4 (Hi)	07
Valore 4 (Lo)	00
Valore 5 (Hi)	00
Valore 5 (Lo)	00
CRC16 (Hi)	6C
CRC16 (Lo)	11

<b>Risposta</b>	
<b>Nome campo</b>	<b>(Hex)</b>
Indirizzo dispositivo	01
Codice funzione	10
Indirizzo inizio (Hi)	07
Indirizzo inizio (Lo)	DA
Numero di coil (Hi)	00
Numero di coil (Lo)	05
CRC16 (Hi)	20
CRC16 (Lo)	85

### **3.2.7 Read Device Identification**

Utilizzare la funzione *Read Device Identification* per ottenere informazioni sullo strumento quali: nome dell'azienda produttrice, tipo, codice, numero di serie e versione dello strumento.

#### **Richiesta**

<b>Codice funzione</b>	1 byte	0x2B
<b>Tipo MEI</b>	1 byte	0x0E
<b>Codice Id lettura dispositivo</b>	1 byte	01
<b>Id oggetto</b>	1 byte	0x00

**Risposta**

<i>Codice funzione</i>	1 byte	0x2B
<i>Tipo MEI</i>	1 byte	0x0E
<i>Codice Id lettura dispositivo</i>	1 byte	01
<i>Livello di conformità</i>	1 byte	0x01
<i>Segue</i>	1 byte	0
<i>Id oggetto successivo</i>	1 byte	Numero Id dell'oggetto
<i>Numero di oggetti</i>	1 byte	3
<i>Id oggetto 1</i>	1 byte	0
<i>Lunghezza oggetto 1</i>	1 byte	26
<i>Valore oggetto 1</i>	Lunghezza oggetto 1	“LSI-Lastem - Milan (Italy)”
<i>Id oggetto 2</i>	1 byte	1
<i>Lunghezza oggetto 2</i>	1 byte	31
<i>Valore oggetto 2</i>	Lunghezza oggetto 2	Tipo, codice, numero di serie di fabbrica e nome definito dall'utente
<i>Id oggetto 3</i>	1 byte	2
<i>Lunghezza oggetto 3</i>	1 byte	8
<i>Valore oggetto 3</i>	Lunghezza oggetto 3	Versione dello strumento

**Errore**

<i>Codice funzione</i>	1 byte	Codice funzione + 0x80
<i>Codice eccezione</i>	1 byte	01 o 02 o 03 o 04

Per maggiori informazioni su *Codice eccezione* fare riferimento al cap. 3.3.

**Esempio:** segue esempio con E-Log tipo 305, numero di serie 08030284/08030284, versione programma 2.08.01 con ID impostato a 01:

<i>Richiesta</i>	
<i>Nome campo</i>	<i>Valore</i>
Indirizzo dispositivo	01
Codice funzione	2B
Tipo MEI	0E
Codice Id lettura dispositivo	01
Id oggetto	00
CRC16 (Hi)	70
CRC16 (Lo)	77

<i>Risposta</i>	
<i>Nome campo</i>	<i>Valore</i>
Indirizzo dispositivo	01
Codice funzione	2B
Tipo MEI	0E
Codice Id lettura dispositivo	01
Livello di conformità	01
Segue	00
Id oggetto successivo	00
Numero di oggetti	03
Id oggetto 1	00
Lunghezza oggetto 1	1A
Valore oggetto 1*	“LSI-Lastem - Milan (Italy)”
Id oggetto 2	01
Lunghezza oggetto 2	21
Valore oggetto 2*	“ELog-305; Serial08030284/08030284”
Id oggetto 3	02
Lunghezza oggetto 3	08
Valore oggetto 3*	“02.08.01”
CRC16 (Hi)	9A
CRC16 (Lo)	6B

\*Valori in esadecimale per:

- Valore oggetto 1: [4C][53][49][2D][4C][61][73][74][65][6D][20][2D][20][4D][69][6C][61][6E][20][28][49][74][61][6C][79][29]
- Valore oggetto 2: [45][4C][6F][67][2D][33][30][35][3B][20][53][65][72][69][61][6C][30][38][30][33][30][32][38][34][2F][30][38][30][33][30][32][38][34]
- Valore oggetto 3: [30][32][2E][30][38][2E][30][31]

### 3.3 Codici eccezione

I codici di eccezione sono trasmessi quando il comando inviato allo slave, sebbene corretto nella forma, non può essere eseguito. I codici di eccezione restituiti sono i seguenti:

<i>Codice</i>	<i>Nome</i>	<i>Significato</i>
01	ILLEGAL FUNCTION	Il codice di funzione non corrisponde ad una funzione supportata dal dispositivo slave
02	ILLEGAL DATA ADDRESS	Il registro specificato ha indirizzo non valido
03	ILLEGAL DATA VALUE	Il valore da assegnare non è valido per l'indirizzo specificato
04	SLAVE DEVICE FAILURE	Si è verificato un errore durante l'esecuzione del comando richiesto



## 3.4 Modalità Master

Il data logger (E-Log dalla versione 2.36.01 e M-Log dalla versione 2.15.00) è in grado di interrogare uno o più apparati slave tramite protocollo Modbus RTU. Questa funzione ha le seguenti caratteristiche:

- L'interrogazione avviene con un ciclo di polling determinato dal tempo di acquisizione programmato in ogni specifica misura.
- Ogni misura programmata nel data logger può selezionare indipendentemente:
  - Apparato slave di indirizzo Modbus specificato.
  - Indirizzo del registro Modbus.
  - Tipologia del dato da leggere all'indirizzo specificato:
    - Valore floating point da 2 registri consecutivi a 16 bit.
    - Valore intero da registro a 16 bit.
  - Comando di interrogazione del dato (da V. 2.39 di E-Log e V. 2.18 di M-Log):
    - Read Holding Registers (hex code 0x03)
    - Read Input Registers (hex code 0x04)
- Dalla versione 2.36 fino alla versione 2.38 di E-Log e dalla versione 2.15 fino alla versione 2.17 di M-Log l'unico comando di interrogazione utilizzato è *Read Holding Registers (0x03)* a cui segue la selezione di uno o due registri (in base al tipo di formato di dato prescelto) ma riferiti alla singola misura. Dalle versioni successive il comando è specificato direttamente nei parametri della misura (vedi punto precedente).
- Tutte le misure lette in virgola mobile subiscono allo stesso modo (non è possibile differenziarle) l'impostazione "Inversione dati floating point" programmata per il protocollo Modbus.
- Fino alla versione 2.37.00 di E-Log e 2.17.00 di M-Log la modalità "Master" del protocollo viene attivata qualora sia impostato il protocollo Modbus per la linea seriale del data logger e, contemporaneamente, siano attive una o più misure di tipo seriale. Se nessuna misura seriale è impostata, il protocollo Modbus si predispose nella modalità slave. Per le versioni successive di entrambi gli strumenti la modalità master è attivabile esplicitamente dal programma 3DOM come selezione di protocollo per la seriale 2.
- La gestione delle misure contempla quanto segue:
  - Identificazione di errore se segnalazione di eccezione da parte dell'unità slave tramite impostazione del bit 7 del codice comando inviato come risposta.
  - Identificazione di errore qualora la misura abbia un valore corrispondente a quanto impostato nei valori di errore floating point (-999999) o interi (-1) nella configurazione del data logger.
  - Identificazione di errore per mancata risposta da parte dello slave; il timeout impostato è fissato a 1 s e il numero di ritentativi per mancata risposta a 3 (valori non modificabili).