



**OPEN DRIVE**

---

**OPEN DRIVE**

*Allegato CAN BUS*

## INDICE

|         |  |    |
|---------|--|----|
| 1.      | Configurazione applicazione .....  | 2  |
| 1.1.    | Configurazione del nodo.....   | 2  |
| 1.2.    | Configurazione oggetti di comunicazione .....  | 3  |
| 2.      | Servizi gestiti .....  | 3  |
| 2.1.    | Service Data Object (SDO) .....  | 3  |
| 2.2.    | Process Data Object (PDO) .....  | 4  |
| 2.2.1.  | Transmit PDO .....   | 4  |
| 2.2.2.  | Receive PDO.....   | 5  |
| 2.3.    | Emergency Object (EMCY) .....  | 5  |
| 2.4.    | Network Management Objects (NMT).....  | 6  |
| 2.5.    | Dizionario degli oggetti: communication profile area .....   | 7  |
| 2.6.    | Dizionario degli oggetti: manufacturer specific profile area .....                                 | 8  |
| 2.6.1.  | Tabella formati parametri (Tab_formati 2001h).....   | 9  |
| 2.6.2.  | Tabella formati connessioni (Tab_con_formati 2002h).....   | 10 |
| 2.6.3.  | Tabella formati parametri extra (Tab_formati 2026h).....   | 11 |
| 2.6.4.  | Tabella formati grandezze interne (Tab_exp_int 2003h).....   | 12 |
| 2.6.5.  | Tabella formati grandezze monitor (Tab_exp_osc 2004h).....   | 12 |
| 2.6.6.  | Gestione sensore di velocità ( hw_software 2007h e hw_sensore 2008h).....                          | 13 |
| 2.6.7.  | Gestione monitor (oggetti da 2009h a 200Ch +2012h).....  | 13 |
| 2.6.8.  | Funzioni logiche di ingresso (oggetti 2010h, 2013h, 2014h, 2016h, 201Fh, 2020h, 2021h, 2022h)..... | 13 |
| 2.6.9.  | Funzioni logiche di uscita (oggetti 2011h, 2015h, 2023h).....                                      | 14 |
| 2.6.10. | Variabili di stato (oggetti 2017h, 2018h e 2019h).....   | 15 |
| 2.6.11. | Riferimenti via CAN BUS (oggetti 201Ah, 201Bh, 201Ch e 201Dh).....                                 | 15 |

I prodotti della linea OPEN drive sono compatibili al CAN open Communication Profile DS301 della CiA rev 4.02. In questo documento sono descritte le funzionalità obbligatorie più quelle opzionali che completano l'implementazione effettuata.

## 1. Configurazione applicazione

### 1.1. Configurazione del nodo

La configurazione del convertitore come nodo CAN prevede l'utilizzo dei seguenti parametri utente di uso generale:

| Nome | Descrizione                      | Range  | Default      |
|------|----------------------------------|--|--------------|
| P162 | ID nodo CAN BUS                  | 1÷127  | 1            |
| C48  | Configurazione baud rate CAN BUS | 0 ÷ 7<br>0 = 1 Mbit/s<br>1 = 800 Kbit/s<br>2 = 500 Kbit/s<br>3 = 250 Kbit/s<br>4 = 125 Kbit/s<br>5 = 50 Kbit/s<br>6 = 20 Kbit/s<br>7 = 10 Kbit/s | 0 = 1 Mbit/s |

Questi parametri andranno opportunamente configurati e salvati nella memoria permanente del convertitore (C63=1). In fase di inizializzazione del convertitore (e solo allora) questi dati verranno considerati e saranno resi operativi.

## 1.2. Configurazione oggetti di comunicazione

La configurazione degli oggetti di comunicazione CAN OPEN DS301 potrà avvenire unicamente via CAN. Alla prima accensione il convertitore è un nodo non configurato che soddisfa al “*Pre-Defined Connection Set*” per quanto riguarda l’allocazione degli identificatori, pertanto sono disponibili i seguenti oggetti:

- un rx SDO con COB-ID = 600h + ID nodo CAN (parametro P162)
- un tx SDO con COB-ID = 580h + ID nodo CAN
- un oggetto emergency con COB-ID = 80h + ID nodo CAN
- gli oggetti NMT (Network Management) : in broadcast (COB-ID = 0) per quanto riguarda i servizi di *Module Control* e per quanto riguarda l’*Error Control* il COB-ID = 700h + ID nodo CAN
- l’oggetto SYNC in broadcast con COB-ID = 80h

Mediante l’SDO disponibile si potrà completamente parametrizzare il convertitore come nodo CAN e poi gli oggetti di comunicazione potranno essere salvati nella memoria permanente con il comando preposto attraverso l’oggetto “**Store parameters** (1010h)” facendo riferimento al Sub-Index 2.

E’ gestito anche l’oggetto “**Restore default parameters** (1011h)” Sub-Index 2 per caricare tutti gli oggetti di default di comunicazione e salvarli automaticamente nella memoria permanente ( in questo caso si dovrà poi spegnere e riaccendere il convertitore per renderli operativi).

## 2. Servizi gestiti

### 2.1. Service Data Object (SDO)

Gli SDO sono utilizzati per accedere al dizionario degli oggetti. Nella nostra implementazione si potranno avere fino ad un massimo di **4 Server SDO** che potranno essere configurati con i seguenti oggetti:

- 1200h 1<sup>st</sup> Server SDO parameter
- 1201h 2<sup>nd</sup> Server SDO parameter
- 1202h 3<sup>rd</sup> Server SDO parameter
- 1203h 4<sup>rd</sup> Server SDO parameter

La tipologia di trasferimento dipende dalla lunghezza dei dati da trasferire: fino a 4 byte si utilizza la modalità *expedited* semplice ed immediata, mentre per oggetti di dimensione superiore sono supportate sia la modalità *segmented* che quella *block*. Si rimanda alle specifiche del Communication Profile DS301 per i dettagli sulle varie modalità di trasmissione, di seguito vengono solo indicate alcune peculiarità della nostra implementazione:

- Un SDO in scrittura dovrà obbligatoriamente indicare il numero di byte significativi (data set size)
- La scrittura di dati attraverso gli SDO è soggetta alle medesime regole (stato convertitore, chiavi, range ammessi ..) previste per la altre modalità di modifica parametri (seriale e tastierino).
- Nel caso di SDO strutturati in più segmenti il convertitore inizierà dal primo segmento a scrivere i dati all’indirizzo indicato senza prevedere alcun buffer temporaneo.
- E’ previsto un meccanismo di controllo per impedire che 2 SDO accedano contemporaneamente in scrittura al medesimo oggetto
- Nel caso di trasmissione in modalità block non è supportato il calcolo del CRC ed il “Protocol Switch Thresold”
- E’ possibile impostare il block size dell’SDO Block Download service all’indirizzo 2000h del dizionario degli oggetti, nella sezione manufacturer specific

## 2.2. Process Data Object (PDO)

I PDO sono utilizzati per lo scambio di dati real-time negli oggetti del Dizionario che supportano questa funzionalità.

### 2.2.1. Transmit PDO

Nella nostra implementazione si potranno avere fino ad un massimo di **4 TPDO** che potranno essere configurati con i seguenti oggetti:

- 1800h 1<sup>st</sup> Transmit PDO Communication parameter
- 1801h 2<sup>nd</sup> Transmit PDO Communication parameter
- 1802h 3<sup>rd</sup> Transmit PDO Communication parameter
- 1803h 4<sup>rd</sup> Transmit PDO Communication parameter

Sono gestiti tutti e 5 i Sub-Index relativi ad ogni TPDO, sarà quindi possibile impostare la transmission type (vedi tabella seguente), l'inhibit time con risoluzione di 100µs ed il periodo dell'event timer con risoluzione di 1ms.

| transmission type | Trasmissione PDO   |
|-------------------|--|
| 0                 | <b>Sincrona</b> : ad ogni oggetto <b>SYNC</b> ricevuto vengono aggiornati i dati e trasmessi               |
| 1-240             | <b>Sincrona e ciclica</b> : il numero indica ogni quanti <b>SYNC</b> avverrà la trasmissione               |
| 241-251           | ----- riservati -----  |
| 252               | Alla ricezione del <b>SYNC</b> i dati sono aggiornati per poi essere trasmessi sull' <b>RTR</b> successivo |
| 253               | I dati sono aggiornati e trasmessi alla ricezione dell' <b>RTR</b> (remote transmission request)           |
| 254               | <b>Event timer</b> : trasmissione ciclica con periodo impostabile in ms nel Sub-Index 5                    |
| 255               | <b>Manufacturer specific</b> : è possibile concordarlo di volta in volta                                   |

Nota: nella transmission type 255 è possibile scegliere su quale evento avvenga la trasmissione del TPDO. La scelta dell'evento potrà essere effettuata unicamente in fase di compilazione del codice.

Il mapping dei TPDO potrà essere effettuato dinamicamente configurando opportunamente i seguenti oggetti di comunicazione:

- 1A00h 1<sup>st</sup> Transmit PDO Mapping parameter
- 1A01h 2<sup>nd</sup> Transmit PDO Mapping parameter
- 1A02h 3<sup>rd</sup> Transmit PDO Mapping parameter
- 1A03h 4<sup>rd</sup> Transmit PDO Mapping parameter

Il mapping dei PDO andrà eseguito seguendo i punti indicati:

1. Porre a zero il numero di oggetti mappati nel Sub-Index 0
2. Configurare gli indirizzi dei vari oggetti mappati
3. Indicare il corretto numero di oggetti mappati nel Sub-Index 0

## 2.2.2. Receive PDO

Nella nostra implementazione si potranno avere fino ad un massimo di **4 RPDO** che potranno essere configurati con i seguenti oggetti:

- 1400h 1<sup>st</sup> Receive PDO Communication parameter
- 1401h 2<sup>nd</sup> Receive PDO Communication parameter
- 1402h 3<sup>rd</sup> Receive PDO Communication parameter
- 1403h 4<sup>rd</sup> Receive PDO Communication parameter

Sono gestiti i primi 2 Sub-Index relativi ad ogni RPDO, sarà quindi possibile impostare la transmission type:

| transmission type | Ricezione PDO   |
|-------------------|---|
| 0-240             | <b>Sincrona:</b> i valori ricevuti negli RPDO verranno attuati alla ricezione del SYNC successivo |
| 241-253           | ----- riservati -----   |
| 254               | <b>Asincrona:</b> i valori ricevuti negli RPDO sono attuati immediatamente                        |

Il mapping degli RPDO potrà essere effettuato dinamicamente configurando opportunamente i seguenti oggetti di comunicazione:

- 1600h 1<sup>st</sup> Receive PDO Mapping parameter
- 1601h 2<sup>nd</sup> Receive PDO Mapping parameter
- 1602h 3<sup>rd</sup> Receive PDO Mapping parameter
- 1603h 4<sup>rd</sup> Receive PDO Mapping parameter

Anche il mapping degli RPDO andrà eseguito seguendo i punti indicati:

4. Porre a zero il numero di oggetti mappati nel Sub-Index 0
5. Configurare gli indirizzi dei vari oggetti mappati
6. Indicare il corretto numero di oggetti mappati nel Sub-Index 0

## 2.3. Emergency Object (EMCY)

L'oggetto emergency viene trasmesso dal convertitore quando interviene un nuovo allarme abilitato oppure quando uno o più allarmi vengono resettati. L'Emergency telegram è costituito da 8 byte come si vede nella tabella seguente:

| Byte        | 0                           | 1 | 2                     | 3  | 4 | 5 | 6 | 7 |
|-------------|-----------------------------|---|-----------------------|--|---|---|---|---|
| Significato | <b>Emergency Error Code</b> |   | <b>Error register</b> | <b>Manufacturer specific</b><br>Allarmi LSB –MSB |   |   |   |   |

Nella nostra implementazione sono gestiti solamente 2 codici dell'Error Code :

- 00xx = Error Reset or No Error
- 10xx = Generic Error

Per quanto riguarda l'**Error register** (oggetto 1001h) vengono gestiti i seguenti bit in corrispondenza degli allarmi indicati:

| Bit | Significato     | Allarmi corrispondenti |
|-----|-----------------|------------------------|
| 0   | Errore generico | tutti                  |
| 1   | Corrente        | A3                     |
| 2   | Tensione        | A10 - A11 -A13         |
| 3   | Temperatura     | A4 - A5 - A6           |

Nei byte **Manufacturer specific** sono stati assegnati i soli byte 3 e 4 che contengono lo stato dei vari allarmi del convertitore. Sono disponibili gli ulteriori 3 byte per la trasmissione di eventuali altri dati utente.

E' prevista la gestione dell'oggetto **1003h "Pre-defined Error Field"** che memorizza la cronologia degli eventi di allarme (dall'accensione del convertitore) fino ad un massimo di 32 elementi.

Ad ogni nuovo evento di allarme vengono memorizzati 4 byte , 2 sono obbligatori e corrispondono all'Error code , gli altri 2 che sono Manufacturer specific e nel nostro caso corrispondono allo stato di tutti gli allarmi del convertitore.

| MSB                    |             | LSB            |                |
|------------------------|-------------|----------------|----------------|
| Additional information |             | Error code     |                |
| Allarmi MSB            | Allarmi LSB | Error code MSB | Error code LSB |

## 2.4. Network Management Objects (NMT)

Questa funzione dà la possibilità all'NMT master di controllare ed imporre lo stato a tutti gli NMT slave. Sono stati implementati tutti i servizi del Module Control ed in più il Node Guarding Protocol che utilizza il COB-ID = 700h + ID nodo CAN, attraverso cui lo slave comunica che dopo il bootup è entrato in modalità pre-operational ed il master può interrogare i vari slave con un RTR.

E' stata implementata anche la funzione di **Life guarding**: il convertitore (NMT slave) può essere parametrizzato attraverso gli oggetti:

100Ch **Guard time** in ms

100Dh **Life time factor** (fattore moltiplicativo)

} il loro prodotto dà il **Node life time**

Nota: il Node life time è internamente saturato ad un tempo pari a 32767/fpwm secondi

Il Life guarding è abilitato solo se il Node life time è diverso da zero, in quel caso il controllo inizia dopo aver ricevuto il primo RTR dall'NMT master.

Il Communication profile DS301 non stabilisce che azione intraprendere qualora non fosse rispettato il vincolo temporale del life guarding. E' possibile decidere cosa fare in fase di compilazione del firmware. Di default non viene intrapresa alcuna azione.

## 2.5. Dizionario degli oggetti: communication profile area

Sono gestiti i seguenti oggetti del communication profile:

| Index (hex)                                | Oggetto | Nome                                   | Tipo           | Accesso           | Par.  |
|--|---------|--|----------------|-------------------|-------|
| 1000                                       | VAR     | Device type                            | UNSIGNED32     | lettura           |       |
| 1001                                       | VAR     | Error register                         | UNSIGNED8      | lettura           | 2.3   |
| 1002                                       | VAR     | Manufacturer status register           | UNSIGNED32     | lettura           |       |
| 1003                                       | ARRAY   | Pre-defined error field                | UNSIGNED32     | lettura           | 2.3   |
| 1005                                       | VAR     | COB-ID SYNC                            | UNSIGNED32     | lettura/scrittura | 2.2   |
| 1006                                       | VAR     | Communication cycle period             | UNSIGNED32     | lettura/scrittura | 2.2   |
| 1008                                       | VAR     | Manufacturer device name               | Vis-String     | costante          |       |
| 1009                                       | VAR     | Manufacturer hardware version          | Vis-String     | costante          |       |
| 100A                                       | VAR     | Manufacturer software version          | Vis-String     | costante          |       |
| 100C                                       | VAR     | Guard time                             | UNSIGNED16     | lettura/scrittura | 2.4   |
| 100D                                       | VAR     | Life time factor                       | UNSIGNED8      | lettura/scrittura | 2.4   |
| 1010                                       | ARRAY   | Store parameters                       | UNSIGNED32     | lettura/scrittura | 1.2   |
| 1011                                       | ARRAY   | Restore default parameters             | UNSIGNED32     | lettura/scrittura | 1.2   |
| 1014                                       | VAR     | COB-ID EMCY                            | UNSIGNED32     | lettura/scrittura | 2.3   |
| 1015                                       | VAR     | Inhibit Time EMCY                      | UNSIGNED16     | lettura/scrittura | 2.3   |
| 1018                                       | RECORD  | Identity Object                        | Identity (23h) | lettura           |       |
| <b>Server SDO Parameter</b>                |         |  |                |                   |       |
| 1200                                       | RECORD  | 1 <sup>st</sup> Server SDO parameter   | SDO parameter  | lettura/scrittura | 2.1   |
| 1201                                       | RECORD  | 2 <sup>nd</sup> Server SDO parameter   | SDO parameter  | lettura/scrittura | 2.1   |
| 1202                                       | RECORD  | 3 <sup>rd</sup> Server SDO parameter   | SDO parameter  | lettura/scrittura | 2.1   |
| 1203                                       | RECORD  | 4 <sup>rd</sup> Server SDO parameter   | SDO parameter  | lettura/scrittura | 2.1   |
| <b>Receive PDO Communication Parameter</b> |         |  |                |                   |       |
| 1400                                       | RECORD  | 1 <sup>st</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.2 |
| 1401                                       | RECORD  | 2 <sup>nd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.2 |
| 1402                                       | RECORD  | 3 <sup>rd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.2 |
| 1403                                       | RECORD  | 4 <sup>rd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.2 |
| <b>Receive PDO Mapping Parameter</b>       |         |  |                |                   |       |
| 1600                                       | RECORD  | 1 <sup>st</sup> receive PDO mapping    | PDO Mapping    | lettura/scrittura | 2.2.2 |
| 1601                                       | RECORD  | 2 <sup>nd</sup> receive PDO mapping    | PDO Mapping    | lettura/scrittura | 2.2.2 |
| 1602                                       | RECORD  | 3 <sup>rd</sup> receive PDO mapping    | PDO Mapping    | lettura/scrittura | 2.2.2 |
| 1603                                       | RECORD  | 4 <sup>rd</sup> receive PDO mapping    | PDO Mapping    | lettura/scrittura | 2.2.2 |
| <b>Transmit PDO Mapping Parameter</b>      |         |  |                |                   |       |
| 1800                                       | RECORD  | 1 <sup>st</sup> transmit PDO parameter | PDO CommPar    | lettura/scrittura | 2.2.1 |
| 1801                                       | RECORD  | 2 <sup>nd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.1 |
| 1802                                       | RECORD  | 3 <sup>rd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.1 |
| 1803                                       | RECORD  | 4 <sup>rd</sup> receive PDO parameter  | PDO CommPar    | lettura/scrittura | 2.2.1 |
| <b>Transmit PDO Mapping Parameter</b>      |         |  |                |                   |       |
| 1A00                                       | RECORD  | 1 <sup>st</sup> transmit PDO mapping   | PDO Mapping    | lettura/scrittura | 2.2.1 |
| 1A01                                       | RECORD  | 2 <sup>nd</sup> transmit PDO mapping   | PDO Mapping    | lettura/scrittura | 2.2.1 |
| 1A02                                       | RECORD  | 3 <sup>rd</sup> transmit PDO mapping   | PDO Mapping    | lettura/scrittura | 2.2.1 |
| 1A03                                       | RECORD  | 4 <sup>rd</sup> transmit PDO mapping   | PDO Mapping    | lettura/scrittura | 2.2.1 |

## 2.6. Dizionario degli oggetti: manufacturer specific profile area

Le variabili indicate in **grossetto** sono mappabili nei PDO.

| Index (hex) | Oggetto      | Tipo              | Nome                               | Descrizione   | Accesso           |
|-------------|--------------|-------------------|------------------------------------|---|-------------------|
| 2000        | VAR          | INTEGER16         | Block size                         | Block size dell'SDO Block Download                                | lettura/scrittura |
| 2001        | VAR          | DOMAIN            | Tab_formati                        | Formati dei parametri   | lettura           |
| 2002        | VAR          | DOMAIN            | Tab_con_formati                    | Formati delle connessioni   | lettura           |
| 2003        | VAR          | DOMAIN            | Tab_exp_int                        | Formati delle grandezze interne                                   | lettura           |
| 2004        | VAR          | DOMAIN            | Tab_exp_osc                        | Formati delle grandezze del monitor                               | lettura           |
| 2005        | VAR          | DOMAIN            | Tab_par_def                        | Valori di default dei parametri                                   | lettura           |
| 2006        | VAR          | DOMAIN            | Tab_con_def                        | Valori di default delle connessioni                               | lettura           |
| <b>2007</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>hw_software</b>                 | Sensore gestito dal firmware                                      | lettura           |
| <b>2008</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>hw_sensore</b>                  | Sensore gestito dall'hardware                                     | lettura           |
| 2009        | VAR          | INTEGER16         | K_zz                               | Contatore del monitor   | lettura           |
| 200A        | VAR          | INTEGER16         | Via_alla_conta                     | Trigger del monitor   | lettura           |
| 200B        | VAR          | DOMAIN            | Tab_monitor_A                      | Buffer canale A del monitor                                       | lettura           |
| 200C        | VAR          | DOMAIN            | Tab_monitor_B                      | Buffer canale B del monitor                                       | lettura           |
| 200D        | ARRAY        | INTEGER16         | Tab_par [200]                      | Valori attuali dei parametri                                      | lettura/scrittura |
| 200E        | ARRAY        | INTEGER16         | Tab_con [100]                      | Valori attuali delle connessioni                                  | lettura/scrittura |
| <b>200F</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_int [64]</b>                | Valori attuali delle grandezze interne                            | lettura           |
| <b>2010</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_inp_dig [32]</b>            | Valori attuali delle funzioni logiche d'ingresso standard         | lettura           |
| <b>2011</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_out_dig [32]</b>            | Valori attuali delle funzioni logiche d'uscita                    | lettura           |
| <b>2012</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_osc [64]</b>                | Valori attuali grandezze monitorabili                             | lettura           |
| <b>2013</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>ingressi</b>                    | Stato logico 8 ingressi a morsettiera                             | lettura           |
| <b>2014</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>ingressi_hw</b>                 | Stato logico 3 ingressi dalla potenza                             | lettura           |
| <b>2015</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>uscite_hw</b>                   | Stato logico 4 uscite digitali                                    | lettura           |
| <b>2016</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_inp_dig_field [32]</b>      | Valori imposti via CAN delle funzioni logiche d'ingresso standard | lettura/scrittura |
| <b>2017</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>stato</b>                       | Variabile di stato del convertitore                               | lettura           |
| <b>2018</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>allarmi</b>                     | Stato degli allarmi del convertitore                              | lettura           |
| <b>2019</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>abilitazione_allarmi</b>        | Parola di abilitazione allarmi convertitore                       | lettura           |
| <b>201A</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>f_fieldbus</b>                  | Riferimento di velocità in % di $n_{MAX}$ in 16384                | lettura/scrittura |
| <b>201B</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>limit_fieldbus</b>              | Limite di coppia in % di $T_{nom}$ in 4095                        | lettura/scrittura |
| <b>201C</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>trif_fieldbus</b>               | Riferimento di coppia in % di $T_{nom}$ in 4095                   | lettura/scrittura |
| <b>201D</b> | <b>VAR</b>   | <b>INTEGER16</b>  | <b>theta_fieldbus</b>              | Riferimento di velocità in impulsi elettr x $T_{pwm}$             | lettura/scrittura |
| <b>201E</b> | <b>ARRAY</b> | <b>INTEGER16</b>  | <b>Tab_dati_applicazione [100]</b> | Area dati disponibile per l'applicazione                          | lettura/scrittura |
| <b>201F</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>Ingressi_standard_wr</b>        | Scrittura ingressi standard via CAN                               | lettura/scrittura |
| <b>2020</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>Ingressi_appl_wr</b>            | Scrittura ingressi applicaz. via CAN                              | lettura/scrittura |
| <b>2021</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>Ingressi_standard_rd</b>        | Letture ingressi standard   | lettura           |
| <b>2022</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>Ingressi_appl_rd</b>            | Letture ingressi applicazione                                     | lettura           |
| <b>2023</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>Uscite_logiche_rd</b>           | Letture uscite logiche  | lettura           |
| <b>2024</b> | <b>VAR</b>   | <b>UNSIGNED16</b> | <b>word_vuota</b>                  | Word inutilizzata   | lettura/scrittura |
| <b>2025</b> | <b>VAR</b>   | <b>UNSIGNED32</b> | <b>double_vuota</b>                | Double word inutilizzata  | lettura/scrittura |
| 2026        | VAR          | DOMAIN            | Tab_formati_extra                  | Formati dei parametri extra                                       | lettura           |

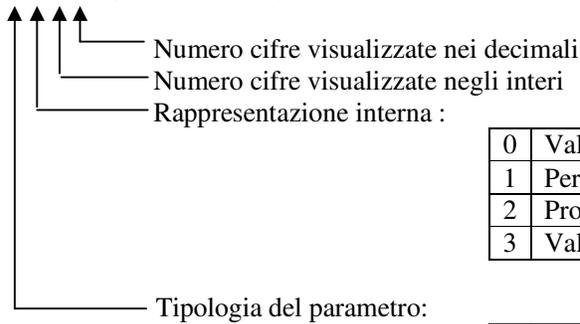
Di seguito sono riportate le descrizioni di alcuni degli oggetti Manufacturer specific.

## 2.6.1. Tabella formati parametri (Tab\_formati 2001h)

Questa tabella è composta da 800 word (200 x 4) , in pratica ci sono 4 word per ogni parametro:

1<sup>a</sup> word : definisce la tipologia del parametro, la sua rappresentazione interna ed il numero di cifre intere e decimali che verranno visualizzate nel display. Ogni nibble ha il seguente significato:

0x 0 0 0 0 (in esadecimale)



|   |                                   |
|---|-----------------------------------|
| 0 | Valore diretto                    |
| 1 | Percentuale della base (100/base) |
| 2 | Proporzionale alla base (1/base)  |
| 3 | Valore diretto unsigned           |

|   |  |
|---|--|
| 0 | Non gestito                                |
| 1 | Libero (modificabile on-line)              |
| 2 | Riservato (modifica off-line + chiave P60) |
| 4 | TDE (modifica off-line + chiave P99)       |

Ad esempio:

0x1231 → parametro libero proporzionale alla base, quindi il valore reale è dato dalla rappresentazione interna diviso la base (4<sup>a</sup> word).

2<sup>a</sup> word: definisce il minimo valore ammesso nella rappresentazione interna del parametro

3<sup>a</sup> word: definisce il massimo valore ammesso nella rappresentazione interna del parametro

4<sup>a</sup> word: definisce la base di rappresentazione del parametro

Esempio 1 (in esadecimale se preceduto da 0x):

1<sup>a</sup> word = 0x1131

2<sup>a</sup> word = 0000

3<sup>a</sup> word = 8190

4<sup>a</sup> word = 4095

parametro libero percentuale della base, quindi il valore reale è dato dalla rappresentazione interna diviso la base per 100

Se il valore corrente è 1000 →  $(1000/4095) * 100 = 24,4\%$

Il range di variazione va da 0 al 200%

Esempio 2 (in esadecimale se preceduto da 0x):

1<sup>a</sup> word = 0x2231

2<sup>a</sup> word = 5

3<sup>a</sup> word = 1000

4<sup>a</sup> word = 10

parametro riservato proporzionale alla base, quindi il valore reale è dato dalla rappresentazione interna diviso la base

Se il valore corrente è 400 →  $(400/10) = 40,0\%$

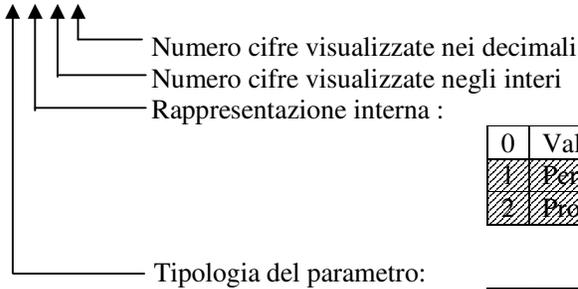
Il range di variazione va da 0,5 al 100%

## 2.6.2. Tabella formati connessioni (Tab\_con\_formati 2002h)

Questa tabella è composta da 400 word (100 x 4) , in pratica ci sono 4 word per ogni connessione:

1<sup>a</sup> word : definisce la tipologia della connessione, la sua rappresentazione interna ed il numero di cifre intere e decimali che verranno visualizzate nel display. Ogni nibble ha il seguente significato:

0x 0 0 0 0 (in esadecimale)



|   |                                   |
|---|-----------------------------------|
| 0 | Valore diretto                    |
| 1 | Percentuale della base (100/base) |
| 2 | Proporzionale alla base (1/base)  |

|   |  |
|---|--|
| 0 | Non gestito                                |
| 1 | Libero (modificabile on-line)              |
| 2 | Riservato (modifica off-line + chiave P60) |
| 4 | TDE (modifica off-line + chiave P99)       |

2<sup>a</sup> word: definisce il minimo valore ammesso nella rappresentazione interna della connessione

3<sup>a</sup> word: definisce il massimo valore ammesso nella rappresentazione interna della connessione

4<sup>a</sup> word: definisce la base di rappresentazione della connessione (sempre 1)

La rappresentazione interna è sempre il valore diretto.

Esempio (in esadecimale se preceduto da 0x):

1<sup>a</sup> word = 0x2020

2<sup>a</sup> word = 0                      connessione riservata il cui valore può andare da 0 a 18

3<sup>a</sup> word = 18

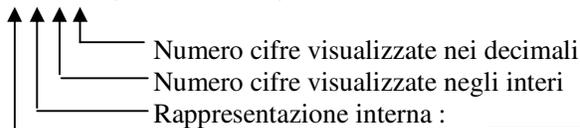
4<sup>a</sup> word = 1

## 2.6.3. Tabella formati parametri extra (Tab\_formati 2026h)

Questa tabella è composta da 500 word (100 x 5), in pratica ci sono 5 word per ogni parametro:

1<sup>a</sup> word : definisce la tipologia del parametro, la sua rappresentazione interna ed il numero di cifre intere e decimali che verranno visualizzate nel display. Ogni nibble ha il seguente significato:

0x 0 0 0 0 (in esadecimale)



|   |                                   |
|---|-----------------------------------|
| 0 | Valore diretto                    |
| 1 | Percentuale della base (100/base) |
| 2 | Proporzionale alla base (1/base)  |
| 3 | Valore diretto unsigned           |

Tipologia del parametro:

|   |  |
|---|--|
| 0 | Non gestito                                |
| 1 | Libero (modificabile on-line)              |
| 2 | Riservato (modifica off-line + chiave P60) |
| 4 | TDE (modifica off-line + chiave P99)       |

Ad esempio:

0x1231 → parametro libero proporzionale alla base, quindi il valore reale è dato dalla rappresentazione interna diviso la base (4<sup>a</sup> word).

2<sup>a</sup> word: definisce il minimo valore ammesso nella rappresentazione interna del parametro

3<sup>a</sup> word: definisce il massimo valore ammesso nella rappresentazione interna del parametro

4<sup>a</sup> word: definisce la base di rappresentazione del parametro

5<sup>a</sup> word: definisce il valore di default del parametro

Esempio 1 (in esadecimale se preceduto da 0x):

1<sup>a</sup> word = 0x1131

2<sup>a</sup> word = 0000

3<sup>a</sup> word = 8190

4<sup>a</sup> word = 4095

5<sup>a</sup> word = 4095

parametro libero percentuale della base, quindi il valore reale è dato dalla rappresentazione interna diviso la base per 100

Se il valore corrente è 1000 →  $(1000/4095) * 100 = 24,4\%$

Il range di variazione va da 0 al 200%

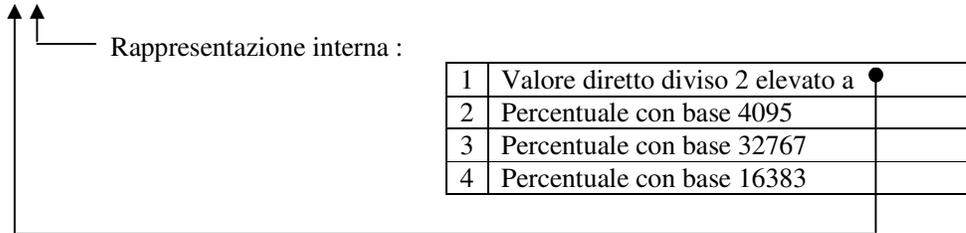
Il valore di default è il 100%

## 2.6.4. Tabella formati grandezze interne (Tab\_exp\_int 2003h)

Questa tabella è composta da 64 word, in pratica c'è una word ogni grandezza interna:

1<sup>a</sup> word : definisce la rappresentazione delle grandezze interne:

0x 0 0 0 0 (in esadecimale)



Esempio 1 (in esadecimale se preceduto da 0x):

0x0002 rappresentazione interna della grandezza: percentuale di 4095.  
Per esempio se vale 2040  $\rightarrow (2040/4095)*100 = 49,8\%$

Esempio 2 (in esadecimale se preceduto da 0x):

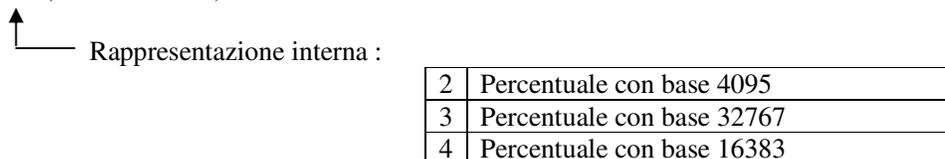
0x0041 rappresentazione interna della grandezza: valore diretto diviso  $2^4$   
Per esempio se vale 120  $\rightarrow (120/2^4) = 7,5$

## 2.6.5. Tabella formati grandezze monitor (Tab\_exp\_osc 2004h)

Questa tabella è composta da 64 word, in pratica c'è una word ogni grandezza del monitor:

1<sup>a</sup> word : definisce la rappresentazione delle grandezze interne:

0x 0 0 0 0 (in esadecimale)



Esempio 1 (in esadecimale se preceduto da 0x):

0x0003 rappresentazione interna della grandezza: percentuale di 32767.  
Per esempio se vale 5000  $\rightarrow (5000/32767)*100 = 15,2\%$

### 2.6.6. Gestione sensore di velocità ( hw\_software 2007h e hw\_sensore 2008h)

Le due variabili hw\_software e hw\_sensore possono assumere i seguenti valori:

| Valore | Sensore corrispondente                         |
|--------|--|
| 0      | --- nessuno ---                                |
| 1      | Encoder incrementale                           |
| 2      | Encoder incrementale + sonde di Hall           |
| 4      | Resolver                                       |
| 8      | Encoder sinusoidale Sin/Cos analogico          |
| 9      | Encoder sinusoidale Sin/Cos assoluto analogico |
| 10     | Endat  |

**hw\_software** rappresenta il sensore gestito dalla versione del firmware presente nel convertitore  
**hw\_sensore** rappresenta il sensore gestito dalla scheda retroazione presente all'interno del convertitore.

### 2.6.7. Gestione monitor (oggetti da 2009h a 200Ch +2012h)

Questi oggetti sono relativi al monitor delle grandezze del convertitore.

**K\_zz (2009h)** è il contatore interno del buffer circolare di 2000 punti.

**Via\_alla\_conta (200Ah)** se  $\neq 0$  indica che è avvenuto l'evento di trigger selezionato con la connessione C14.

**Tab\_monitor\_A (200Bh)** e **Tab\_monitor\_B (200Ch)** sono i buffer circolari dove vengono memorizzati i campioni delle grandezze selezionate con le connessioni C15 e C16.

Sono inoltre coinvolti i parametri P54 che setta il periodo di campionamento del monitor (in unità pari al periodo della PWM), P55 che setta i punti post-trigger e P56 che setta il livello del trigger se questo è effettuato sulle grandezze monitorabili.

Si rimanda alla documentazione del prodotto per la particolarizzazione della grandezze monitorabili.

L'oggetto **Tab\_osc (2012h)** è un array di 64 grandezze con i valori più recenti di tutte le grandezze monitorabili. I singoli oggetti potranno così essere mappati nei PDO per tenere sotto controllo le varie grandezze del convertitore.

### 2.6.8. Funzioni logiche di ingresso (oggetti 2010h, 2013h, 2014h, 2016h, 201Fh, 2020h, 2021h, 2022h)

La gestione delle funzioni logiche di ingresso è completamente controllabile via CAN.

Nella variabile **ingressi (2013h)** è possibile leggere lo stato degli 8 ingressi disponibili in morsettiera negli 8 bit meno significativi. Gli 8 ingressi logici sono configurati attraverso le connessioni C1 ÷ C8 a controllare ognuno una particolare funzione logica di ingresso.

#### Funzioni logiche di ingresso standard (I00 ÷ I28)

Lo stato delle 32 funzioni logiche di ingresso standard è disponibile in due diversi oggetti del dizionario: l'array **Tab\_inp\_dig (2010h)** nel quale si accede funzione per funzione utilizzando il sub-index (stato logico 0 = low ; 32767 = high) e la variabile a 32 bit **Ingressi\_standard\_rd (2021h)** nella quale ad ogni bit è associato lo stato della funzione corrispondente.

E' possibile via CAN imporre anche lo stato delle funzioni logiche di ingresso standard, si potrà agire funzione per funzione scrivendo il valore desiderato (0=low, 32767=high) negli oggetti dell'array

**Tab\_inp\_dig\_field (2016h)**, oppure imponendo lo stato di tutte e 32 le funzioni logiche standard scrivendo la variabile a 32 bit **Ingressi\_standard\_wr (201Fh)**.

La logica implementata prevede che:

- La funzione d'ingresso logico standard 0, ovvero la marcia, è data dall'AND dei vari canali di ingresso: morsettiera, field-bus e seriale.
- Tutte le altre funzioni logiche standard potranno essere portate high dall'OR dei vari canali.

Nella fase di inizializzazione **Tab\_inp\_dig\_field[0]=high** pertanto se questo valore non viene mai sovrascritto il convertitore può essere controllato via morsettiera.

## **Funzioni logiche di ingresso applicazione (I29÷I63)**

Lo stato delle prime 32 funzioni logiche di ingresso applicazione (I29÷I60) è disponibile nella variabile a 32 bit **Ingressi\_appl\_rd (2022h)** nella quale ad ogni bit è associato lo stato della funzione corrispondente. E' possibile via CAN imporre anche lo stato delle funzioni logiche di ingresso applicazione, si potrà agire imponendo lo stato di tutte e 32 le funzioni scrivendo la variabile a 32 bit **Ingressi\_appl\_wr (2020h)**.

La logica implementata prevede che:

- Le 32 funzioni logiche applicazione potranno essere forzate via CAN.
- Qualora qualcuna delle funzioni sia associata anche ad un ingresso fisico in morsettiera, lo stato logico fisico imporrà lo stato della funzione corrispondente

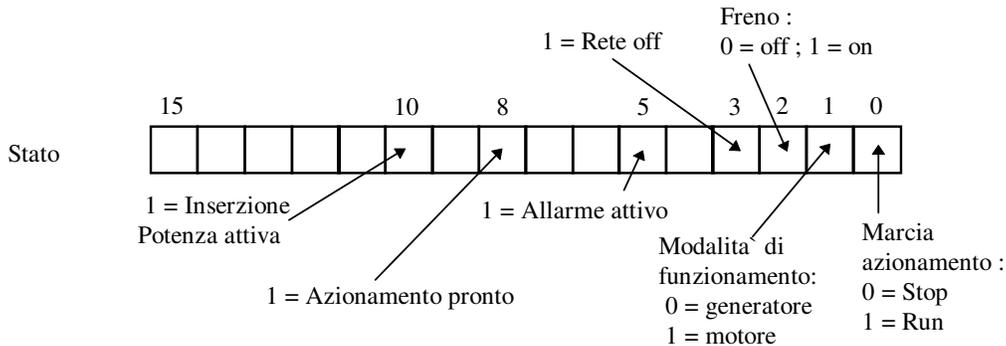
## **2.6.9. Funzioni logiche di uscita (oggetti 2011h, 2015h, 2023h)**

Via CAN bus è possibile monitorare lo stato:

- delle 4 uscite logiche in morsettiera nei 4 bit meno significativi della variabile **uscite (2015h)**
- lo stato delle 32 funzioni logiche di uscita nell'array **Tab\_out\_dig (2011h)** utilizzando il sub-index. Come per gli ingressi si intende che 0 = low e 32767 = high
- lo stato di tutte e 32 le funzioni logiche di uscita nella variabile a 32 bit **Uscite\_logiche\_rd (2023h)** nella quale ad ogni bit è associato lo stato della funzione corrispondente.

## 2.6.10. Variabili di stato (oggetti 2017h, 2018h e 2019h)

Nell'oggetto **2017h** è disponibile una variabili di **stato** del convertitore con il seguente significato:



Nell'oggetto **2018h** c'è invece lo stato dei vari **allarmi** del convertitore bit per bit, ovvero all'allarme A8 è associato il bit 8.

Nell'oggetto **2019h** c'è la maschera di **abilitazione degli allarmi**, anche qui il significato è bit per bit. Questa variabile è disponibile in sola lettura, far riferimento al parametro P163 per la scrittura.

## 2.6.11. Riferimenti via CAN BUS (oggetti 201Ah, 201Bh, 201Ch e 201Dh)

Per poter utilizzare questi oggetti per dare dei riferimenti di velocità, coppia, limite di coppia al convertitore è necessario abilitare la loro gestione ponendo **C52=1**.

**f\_fieldbus (201A)** = riferimento di velocità in percentuale della velocità massima impostata. Base di rappresentazione pari a 16384: pertanto a 16384 corrisponde il 100%

**theta\_fieldbus (201D)** = riferimento di velocità in impulsi elettrici per periodo di PWM, considerando che si hanno 65536 impulsi sul giro e che il termine "elettrici" significa che dovranno essere moltiplicati per il numero di coppie polari del motore.

**trif\_fieldbus (201C)** = riferimento di coppia in percentuale della coppia nominale del motore. Base di rappresentazione pari a 4095: pertanto a 4095 corrisponde il 100%

**limit\_fieldbus (201A)** = limite di coppia in percentuale della coppia nominale del motore (che andrà in alternativa agli altri limiti esistenti, vale il più restrittivo). Base di rappresentazione pari a 4095: pertanto a 4095 corrisponde il 100%