

## Manuale uso SERVO-BL DRIVER CNC\_ULTIMATE

### Descrizione:

L'azionamento SERVO-BL DRIVER CNC\_ULTIMATE pilota un motore Servo DC oppure un motore Brushless a 3 fasi fino a 36V - 10Ampere cioè motori da 300 Watt continui con correnti di picco fino a 50Ampere.

In questo documento il termine Azionamento e Driver sono equivalenti.

### Caratteristiche:

- Pilotaggio:
  - Ingresso **Analogico 0-10Vdc**(non optoisolato)+**direzione**+enable.
  - Ingresso **Potenzimetro+direzione**+enable.
  - Ingresso **Step+direzione**+enable (optoisolati).
  - Comandi su **porta RS232** (con modulo USB-TTL o con modulo RS232-TTL).
  - Ingresso **Tastierino/display** (con Terminal-TTL).
- **Ingressi:** Ingressi step, dir, enable (optoisolati), ingresso analogico 0-10v con adc a 10bit (non optoisolato), ingresso encoder, ingresso 3 sensori Hall. L'alimentazione degli ingressi optoisolati è a 5volts esterna e gli ingressi sono tutti e 3 di tipo NPN.
- **Uscite:** Segnale di errore(optoisolato), quando è in errore il driver mette in off il fotoaccoppiatore, I 2 pin di uscita del fotoaccoppiatore possono essere collegati in serie con I fotoaccoppiatori di errore degli altri driver e infine collegati agli ingressi Enable dei driver per disabilitare tutti I driver quando almeno uno di questi va in errore, il driver segnala il suo errore tramite l'unico led rosso.
- **Modifica parametri e aggiornamento firmware:** mediante porta seriale rs232 con modulo RS232-TTL settaggio a 38400,n,8,1, programma usato per l'inserimento dei parametri o aggiornamento firmware: hyperterminal o programma simile. La porta rs232 consiste delle linee TX-RX e gnd.
- **Solo Modifica parametri:** Tramite il Tastierino del Terminal-TTL .
- **Tensione di alimentazione logica:** 12-15Vdc – 250mA.
- **Tensione di alimentazione ponte:** 12-36Vdc – 10Ampere.
- **Motore:** 3 morsetti U-V-W per alimentazione motore BL o DC (U-V).
- **Encoder:** Retroazione DC e BL con encoder differenziali o sbilanciati, con alimentazione a 5volts e ppr compreso tra 100 ppr e 2500 ppr con steps di 25ppr. Consumo massimo encoder 200mA.
- **HALL:** Ingresso per 3 sensori hall open collector o push-pull a 5volts per Motore BL.
- **Regolazione:** Sull'anello di regolazione è posto un **PID** che regola ogni 300uSec il valore da dare al ponte trifase( brushless) o Hbridge (Servo).

### Comandi via RS232 38400,n,8,1:

I **Comandi** possono essere inseriti siano in maiuscolo che in minuscolo.

1. **?** Help.
2. **E** Echo mode e0 – e1.
3. **M** Set mode ...(da **m0** a **m15**).
4. **S** Set sub command (esempio: s34 .... s-230.....etc...)
5. **+** Step +1 ...max+32767.
6. **-** Step -1 ...max -32767.
7. **P** Set parametri nella tabella 0(operativa) ...(da P0 a P24).
8. **R** Read tabella parametri ....5 tabelle(5 motori)...(da R0 a R4).
9. **W** Write tabella parametri ....5 tabelle(5 motori)..(da W0 a W4).
10. **@** Trasmette RPM motore misurato con encoder

### Parametri (comando P) via RS232 38400,n,8,1:

<u>parametro</u>	<u>tipo variabile</u>	<u>range</u>	<u>MOTORE RS</u>
<b>P0</b> Velocity limit	Integer	1-65535	<b>1500</b>
<b>P1</b> Velocity feedback gain	8.8 fixed point	1-65535	<b>6550</b>
<b>P2</b> Proportional gain	8.8 fixed point	1-65535	<b>300</b>
<b>P3</b> Integral gain	8.8 fixed point	1-65535	<b>30</b>
<b>P4</b> Derived gain	8.8 fixed point	0-65535	<b>500</b>
<b>P5</b> EG feedback gain	8.8 fixed point	1-65535	<b>400</b>
<b>P6</b> Gear ratio for pulsed input	8.8 fixed point	1-65535	<b>1280</b>
<b>P7</b> Torque Limit	Integer	1-65535	<b>500</b>
<b>P8</b> Position Error	Integer	1-32767	<b>2000</b>
<b>P9</b> Torque Time Error	Integer	1-65535	<b>500</b>
<b>P10</b> Modalità funzionamento	Integer	0-15	<b>11</b>
<b>P11</b> RampaUP	Integer	1-100	<b>50</b>
<b>P12</b> RampaDW	Integer	1-100	<b>14</b>
<b>P13</b> Bias	Integer	-120+120	<b>0</b>
<b>P14</b> ppr encoder	Integer	100-2500	<b>500</b>
<b>P15</b> Dip-switch eeprom	Binary	0-65535	<b>0</b>
<b>P16</b> nu	..	..	..
<b>P17</b> nu	..	..	..
<b>P18</b> nu	..	..	..
<b>P19</b> nu	..	..	..
<b>P20</b> nu	..	..	..
<b>P21</b> nu	..	..	..
<b>P22</b> nu	..	..	..
<b>P23</b> nu	..	..	..
<b>P24</b> nu	..	..	..

## **MEMORIA MOTORI:**

La memoria motori contiene 5 motori, per default I valori dei parametri memorizzati sono I seguenti:

Per gli esempi in questo documento si farà riferimento ad un motore commerciale BL disponibile su RS al costo di circa 100 euro + iva, nelle prove viene chiamato: **MOTORE RS codice 536-6052 inserito nella memoria R4.**

**Memoria R0(operativa):** Lo stesso della Memoria R4.

**Memoria R1:** Motore BL MAE **BM10/1A** + encoder 1000ppr **TAMAGAWA OIH48-1000P8-L6-5V TS5207.**

Dati targa: 1,6 Nm e 3000 rpm a 220Vrms(310Vdc), **500watt**, coppia di picco 6Nm, corrente di picco 10Adc, 3Adc a rotore bloccato, peso 3.9kg, bemf=100V/Krpm, costante di coppia 0.66Nm/A, diametro perno frontale= 14mm, diametro perno per encoder=8mm. Fornitore:SERVOTRADE.

Essendo alimentato a 1/10 della tensione nominale 30V invece di 300V la velocità massima è 300rpm, aumentarla almeno a 150Vdc per far andare a 1500rpm. Bisogna sostituire I mosfet con modelli almeno a 250V e alimentare con un trafo 230->115V/300VA.

**Memoria R2:** Motore DC BUHLER **1.13.021.605** diametro 31mm (lunghezza cassa ridotta da 75 a 51mm) + encoder **Olivetti** 100ppr, di recupero da stampante MPS1220.

Dati targa: 0.032Nm e 3000rpm a 12Vdc, **10watt**, coppia di picco 0.12Nm, corrente di picco 4Adc, 1.2Adc a rotore bloccato, peso 0.24kg, bemf=4V/Krpm, costante di coppia 0.027Nm/A, diametro perno frontale= 3mm , diametro perno per encoder=3mm.

**Memoria R3:** Motore DC MAE **M540-0741-0606** + encoder 1250ppr TEKEL **TK561.**

Dati targa: 0.20Nm e 6000rpm a 45Vdc, **100watt**, coppia di picco 1.05Nm, corrente di picco 14Adc, 3Adc a rotore bloccato, peso 1kg, bemf=7.41V/Krpm, costante di coppia 0.071Nm/A, diametro perno frontale= 6mm , diametro perno per encoder=6mm. 3000Rpm a 24Vdc, 4000rpm a 32Vdc.

TESTS:

Con cavo encoder sbilanciato in m2 e m3 arriva a 900rpm.

Con cavo encoder bilanciato ma non twisted/schermato in m2 e m3 arriva a 1600rpm, probabilmente il limite è dovuto al campionamento encoder, 1600rpm corrisponde a 26.6rps x 5000 cpr\_encoder = 133.4khz.

**Memoria R4:** Motore **RS 536-6052** della fulling motor modello **FL57BL04**, A questo motore viene aggiunto un encoder RENCO **RM21D-500-1/4-5-CA18-LD** da 500ppr (2000cpr in quadratura).

Dati targa (rating): 0.43Nm e 4000rpm a 36Vdc, **180watt**, coppia di picco 1.27Nm, corrente di picco 20Adc, resistenza line-to-line 0,35ohm, induttanza line-to-line 1mH, peso 1.2kg, bemf=6.6V/Krpm, costante di coppia 0.063Nm/A, diametro perno frontale= 8mm, diametro perno per encoder=6mm.

Le memorie possono essere utilizzate per motori diversi ma anche e soprattutto per lo stesso motore con configurazioni differenti.

### **PARTENZA MOTORE:**

Per far partire l'azionamento quando si fornisce alimentazione 12v impostare il parametro **P10** con la modalità che vi necessita, con I valori mostrati l'azionamento **MOTORE RS** parte in step-dir (m11) come emulazione a mezzo passo di uno stepper (400 spr passi per giro) (**P6**=1280) pur avendo un encoder da 500ppr. Il motore si può far ripartire dalla condizione di errore, o modificare la modalità....tramite rs232 inviando mxx dove per xx è la modalità scelta. Comunque togliendo e didando l'alimentazione il driver parte sempre con il parametro P10.

### **RAMPE:**

Il driver può effettuare rampe tramite quando è comandato da **ingresso analogico** o **ingresso RS232** (non con ingresso step/dir) I valori delle rampe vengono introdotti nei parametri **P11** per la rampa di accelerazione e **P12** per la rampa di decelerazione, quando il valore=0 non effettua la rampa, una rampa seppur piccola v'è sempre fatta...è accettabile un valore da 1 a 100 circa.

Nello **step/dir** le rampe verranno effettuate dal controller.

### **AGGIORNAMENTO FIRMWARE:**

L'aggiornamento del firmware si effettua tramite la porta rs232:

1. Avviare Hyperterminal e dare alimentazione 12V, entro 1 secondo premere il tasto 'd'
2. Cliccare su invia file e selezionare il firmware con il nome xxx\_enc.bin e inviarlo tramite protocollo xmodem. Dopo pochi secondi il firmware è caricato e parte in esecuzione.

Se il led della scheda lampeggia e la scheda non funziona significa che il software è corrotto o mancante quindi caricarlo dal punto2.

### **Trasmissione RPM:**

Ogni volta che viene inviato il comando @ in rs232 il driver risponde con gli RPM mandrino, l'errore è quantificabile in 0.1% sulla misura, l'acquisizione della velocità viene fatta tramite l'encoder in maniera molto veloce (75millisec).

## **Comandi + / -**

Quando si usano I **comandi +/-** tramite porta rs232 il parametro Position error **P8** deve essere settato a 32767 cioè al massimo valore che corrisponde al valore massimo di **+/-** senza generare errore. I comandi +/- funzionano solo in modalità m3/m11. Il comando inviato per essere accettato deve trovare l'enable attivo, altrimenti viene scartato.

Ad ogni unità del comando **+/-** corrisponde un incremento di un'unità dell'encoder clock pulse per revolution (cpr in quadratura) quindi in massima risoluzione.

La velocità si imposta in Velocity limit **P0**. Se P0 viene variato durante un movimento la velocità varia istantaneamente, se il valore viene impostato superiore la velocità è limitata al valore massimo. Il valore massimo di P0 dipende dal motore e dall'encoder.

Non vengono effettuate rampe.

I comandi +/- se superiori a P8 producono errore.

Questi comandi sono comandi relativi e non assoluti, quindi se vengono usati in un asse cartesiano abbiamo un asse infinito.

Esempio: +3000 .....muovi di 3000passi avanti, -2756 .....muovi di 2756passi indietro.

## **Tabelle:**

E' possibile memorizzare in EEROM 5 tabelle parametri (da 0 a 4) per 5 tipi di motore diversi, ogni tabella ha 25 parametri (da 0 a 24) , ogni parametro è a 16 bit con segno. La tabella operativa è la numero 0 (zero). Solo la tabella 0 ha una copia in ram, il programma durante il funzionamento legge I parametri dalla ram e non dalla eerom. Il comando Read **Rn** copia la tabella n nella 0, Il comando **Wn** scrive la ram nella tabella n.

Esempio R4 legge la tabella 4 e la copia nella 0 (ram) , comunque se si toglie l'alimentazione al ritorno nella ram viene copiato la tabella 0 della EEROM, quindi se si vuole rendere operativa definitivamente la tabella 4 bisogna fare R4 e poi W0.

## **Parametri:**

Il parametro Gear **P6** è usato solo in m3/m11 si imposta **(ppr/spr) x 1024** dove ppr=pulse per revolution encoder, spr sono I passi per revolution che si vogliono ottenere, ad esempio per un encoder da 500ppr (2000cpr in quadratura) e 400 passi all'ingresso step impostiamo 1280. In questo modo un giro completo del motore avviene con 400 passi all'ingresso step e quindi si può emulare uno stepper funzionante a mezzo passo. P6 max=65535. P6 min=256 che corrisponde spr=cpr ma volendo si può impostare spr ad un numero superiore a cpr , in tal caso alcuni passi non eseguono un effettivo movimento del motore, invece con p6 molto grande ad ogni passo corrisponde un movimento troppo grande del motore, sono tutte 2 comportamenti anomali del motore, un valore normale va tra 256 e 2560 cioè tra 1/4 e 2.5 volte il gear ratio.

Il parametro Position error **P8** si riferisce ai cpr in quadratura e non ai ppr.

La variazione di un qualsiasi parametro viene applicata istantaneamente perchè viene memorizzato in RAM, per rendere la variazione permanente anche quando si toglie e si ridà l'alimentazione usare il comando W\_.

### **Errore:**

La condizione di errore accende in modo fisso il led errore sulla scheda, manda un messaggio di errore sulla rs232 e quindi se è collegato il Terminal-TTL viene visualizzato sul display, e attiva il fotoaccoppiatore di errore. Quando è in errore il motore viene scollegato. Per ripristinare l'errore basta inviare in rs232 la modalità operativa desiderata oppure spegnere e accendere il driver.

Il led sulla scheda segnala anche il superamento del Limite di coppia mediante lampeggi veloci.

### **Dip-switch EERom:**

Il parametro **P15** è un dip switch software in eeprom a 16bit (16levette):

**Bit0(DIP-DIR)** = 1 = Inverte il senso di rotazione del motore, funziona in m3/m11 in step/dir , in m4/m12, in m5/m13 e m6/m14.

**Bit1(DIP-ENA)** = 1 = Scollega il motore quando il driver non è abilitato.

### **Terminal-TTL:**

Il Terminal-TTL non è altro che un terminale rs232-ttl che si collega al driver mediante 4 fili (TX-RX-GND-5VCC), ha un tastierino con 16 tasti e un display a 16 caratteri 2 righe. Serve per visualizzare/modificare I parametri e visualizzare I RPM motore e gli errori del driver.

**Modalità di funzionamento comando M:**  
**DC = DC motor BL = Brushless**

**m0 = DC in tensione:** S varia la velocità...da -240 a 240, ha pochissima coppia a basso numero di giri perchè non ha il controllo di posizione con encoder e nemmeno quello di coppia. E' praticamente un normale controllo PWM.

La velocità massima dipende dalla tensione di alimentazione e non va mai in errore.  
Con MOTORE RS a 12V=1800rpm a 24V=3600rpm a 30V=4000rpm.

**m1 = DC in coppia:** Il comando S varia la coppia...da -240 a 240, la velocità è data dalla tensione di alimentazione, se il valore di coppia è troppo basso il motore rallenta o si ferma, non ha il controllo di posizione con encoder e non va mai in errore.

Con MOTORE RS a 12V=1800rpm a 24V=3600rpm a 30V=4000rpm.

Impostare il **P5** =  $BEMF[V/krpm] / ppr \text{ encoder} / \text{Tensione alim} * 3840$

**m2 = DC in velocità:** Il comando S varia la velocità...da circa -1023 a 1023, il valore massimo di S dipende dalla tensione di alimentazione, l'encoder funziona, se la coppia resistente supera il valore di Limite di coppia in **P7** si accende il led di errore, e se l'errore di coppia permane per un tempo maggiore di **P9** il motore va in errore. Ogni unità di P9 corrisponde a circa 0.3msec.

Con MOTORE RS ogni unità di S corrisponde circa a 4rpm (3.91) (con **P1**=6550), In questa modalità anche con S=1 la coppia è massima. Praticamente con P1 si può regolare il range di fondo scala.

Con MOTORE RS: 12V=Smax=460/1800rpm    24V=Smax=920/3600rpm

32V=Smax=1023/4000rpm    36V=Smax=1200/4700rpm

Se si aumenta P1 a 8000 per raggiungere i 4000rpm bisogna impostare S=1250 mentre con P1=4000 per raggiungere i 4000rpm basta impostare S=625.

Con P1=2560 per 4000rpm si imposta S=400.

**m3 = DC in posizione:** Funzionano gli ingressi step/dir con l' encoder e i comandi +/- ma non il comando S.

Il Limite di velocità **P0** o il Limite di coppia **P7** troppo bassi potrebbe far andare in errore il driver. Anche il Position error **P8** o il Tempo di errore coppia **P9** troppo bassi fanno andare in errore. Se l'errore è di posizione aumentare P0 e/o P8 , se l'errore è di coppia aumentare P7 e/o P9.

La freq max di step/sec arriva a 100Khz mentre il contatore dell'encoder può conteggiare fino a 133Khz in quadratura.

Le rampe devono venire fatte dal generatore di step.

Con MOTORE RS I dati rilevati sono:

a 12V=>12.000step/sec max(1800rpm) la coppia però è bassa, tensione troppo bassa

a 24V=>24.000step/sec max(3600rpm) la coppia è sostenuta

a 32V=>26.666step/sec max(4000rpm) la coppia è massima, quella di targa.

**m4 = DC in tensione analogica:** come m0 ma la tensione viene presa dall'ingresso analogico e non dal parametro S.

**m5 = DC in coppia analogica:** come m1 ma la coppia viene presa dall'ingresso analogico e non dal parametro S.

**m6 = DC in velocità analogica:** come m2 ma la velocità viene presa dall'ingresso analogico e non dal parametro S.

**m7** = motore scollegato

-----

**m8 = BL in tensione:** come m0  
**m9 = BL in coppia:** come m1  
**m10 = BL in velocità:** come m2  
**m11 = BL in posizione:** come m3  
**m12 = BL in tensione analogica:** come m4  
**m13 = BL in coppia analogica:** come m5  
**m14 = BL in velocità analogica:** come m6  
**m15 = motore scollegato**