

Sommario

<i>AN001- Esempio di utilizzo e calibrazione del device CAMMING3</i>	3
Dichiarazione del device nella unit di configurazione	3
Corretta parametrizzazione del device	3
Movimentazione preliminare	4
Parametrizzazione uscita analogica (impostazione parametro maxvel)	5
Esempi di compilazione della cam table	5
Utilizzo di un master simulato	7

AN001- Esempio di utilizzo e calibrazione del device CAMMING3

In questa sezione si vogliono descrivere le prime operazioni che si troverà a fare l'utente nel suo primo approccio con il device CAMMING. Inoltre si descriveranno alcuni esempi di uso comune del device.

Possiamo suddividere nelle seguenti sezioni il procedere dell'operazione:

- dichiarazione del device nella unit di configurazione
- introduzione dei parametri al fine di calibrare correttamente ingressi ed uscite
- sviluppo dell'applicativo secondo le esigenze

Dichiarazione del device nella unit di configurazione

Come è stato già spiegato nella sezione di descrizione del device CAMMING, è necessario programmare correttamente la unit di configurazione dell'applicativo. È molto importante la porzione di codice in cui vengono dichiarati i device, qui infatti si dovranno indicare le risorse hardware da utilizzare per garantire un corretto funzionamento. Sarà compito del programmatore individuare e scegliere ingressi ed uscite più opportune. Ad esempio con la seguente riga di codice:

```
-----
; Dichiarazione device interni
-----
INTDEVICE
nome_device CAMMING3 TCamp CountS CountMA CountMB IntL IAZero IntLM IAZeroM InG InGInt IoutA Out
Asse CAMMING3 2 2.CNT01 2.CNT02 1.CNT01 1 2.INP01 2 2.INP02 2.INP03 5 2.AN01 2.OUT01
```

si definisce un device CAMMING3 con nome "Asse" il cui tempo di campionamento è di 2 ms. Di seguito sono dichiarate tutte le risorse hardware necessarie all'utilizzo del device, la descrizione dettagliata è presente nella documentazione del device.

Un applicativo che abbia soltanto al suo interno la dichiarazione del device nella unit di configurazione ed una unit qcl che non esegua alcuna operazione (salvo quella di WAIT forzato) consente già di compiere le prime operazioni sfruttando le funzionalità del device. Infatti dopo avere scaricato l'applicativo sullo strumento e avendolo fatto girare, sarà già possibile modificare i parametri, osservare gli stati o dare comandi al device tramite il monitor apposito da QView. Questo risulta molto comodo nelle prime fasi della programmazione quando si vogliono solo verificare alcuni funzionamenti oppure in fase di debug.

Corretta parametrizzazione del device

Una volta dichiarate correttamente le risorse hardware da utilizzare è necessario impostare alcuni parametri in base alle componenti che sono collegate al prodotto Qmove.

Introduzione di measure e pulse

Prendiamo in considerazione il caso (piuttosto diffuso) in cui il trasduttore bidirezionale sia un encoder digitale. Supponiamo che l'encoder sia direttamente calettato su un motore che debba movimentare l'asse slave. Sarà necessario impostare correttamente i parametri *measure* e *pulse* del device in modo che quest'ultimo possa interpretare gli impulsi encoder che arriveranno al QMove, lo strumento quindi potrà calcolare la posizione dell'asse. L'introduzione di *measure* e *pulse* consente di stabilire una corrispondenza tra uno spazio in una unità di misura prescelta ed un determinato numero di impulsi. Nel caso in cui l'utilizzatore conosca già lo spazio percorso in un giro encoder allora potrà procedere direttamente all'inserimento dei valori. Chiariamo questo concetto con un esempio: se l'encoder genera 1000 impulsi giro e si è a conoscenza che l'asse si muove di 5 cm quando l'encoder compie precisamente un giro allora si potranno inserire i seguenti valori:

```
AsseX:measure = 50;
AsseX:pulse = 4000
```

il valore di *measure* introdotto inoltre implica la scelta di un'unità di misura del millimetro per misurare le posizioni, nel parametro *pulse* si è introdotto un valore pari al numero di impulsi encoder moltiplicato per 4. Si ricorda che il rapporto *measure/pulse* deve avere un valore compreso tra 0.00935 e 1 (per rispetto dei limiti di precisione del device e del prodotto QMove). È importante sottolineare che i valori appena descritti sono presi come riferimento: non è necessario introdurre i parametri prendendo come riferimento un giro encoder come si andrà a descrivere di seguito.

Quando l'utilizzatore non conosce preventivamente i parametri di misura, potrà comunque effettuare la corretta calibrazione seguendo questi passi:

- dare il comando di *INIT* al device, verificare che lo stato *st_init* commuti a 1
- tramite il "device monitor" di QView visualizzare sul pc il valore del parametro *posit*
- impostare *measure* e *pulse* entrambi al valore 1
- movimentare l'asse manualmente facendogli compiere uno spostamento di una posizione facilmente misurabile

- leggere il valore di *posit*
- a questo punto inserire con la unità di misura desiderata il valore misurato nel parametro *measure* ed il valore del parametro *posit* nel parametro *pulse*.

La risoluzione dell'encoder è ora correttamente impostata.

Un'ulteriore operazione importante da compiere è impostare i parametri *maxpos* e *minpos* che definiscono rispettivamente la massima e la minima posizione raggiungibile dall'asse.



Nota:

È necessario effettuare questa procedura due volte in modo da tarare correttamente sia l'asse master che l'asse slave

Scelta dell'unità di misura della velocità

Il device CAMMING consente di misurare la sola velocità dell'asse slave. L'unità di misura della velocità istantanea dell'asse viene scelta tramite i parametri *unitvel* e *decpt*. È possibile scegliere l'unità di tempo della velocità con il parametro *unitvel*: se questo è pari a 0 allora la velocità è misurata in Um/min, se è pari a 1 allora è misurata in Um/s. Il parametro *decpt* invece stabilisce se misurare i valori di velocità in multipli dell'unità di misura fondamentale Um. Ad esempio, se l'unità di misura fondamentale Um=mm, ed *unitvel*=1 si ottiene la visualizzazione della velocità nella variabile *vel* in:

mm/s (con *decpt* = 0),
cm/s (con *decpt* = 1),
dm/s (con *decpt* = 2),
m/s (con *decpt* = 3).

In seguito, se necessario, bisognerà configurare correttamente la visualizzazione sul terminale operatore per aggiustare la corretta posizione del punto decimale.

Calibrazione uscita analogica



Attenzione:

prima di effettuare dei posizionamenti veri e propri è necessario verificare che collegamenti ed organi meccanici non siano causa di malfunzionamenti.

Esaminiamo il caso in cui il device CAMMING utilizzi un'uscita analogica implementata con un dispositivo DAC per comandare l'asse slave: questo assumerà in ingresso valori discreti con risoluzione 16 bit (perciò compresi fra -32768 e 32767) per dare in uscita una tensione analogica con range $\pm 10V$. Con la funzione di calibrazione questa uscita analogica può essere pilotata con un valore costante con lo scopo di verificare collegamenti e funzionalità.

Movimentazione preliminare

In questa sezione vengono descritte le operazioni da effettuare per verificare la correttezza dei collegamenti e la funzionalità del sistema che è stato costruito.

- dare il comando di *INIT* al device, verificare che lo stato *st_init* commuti a 1
- dare il comando *RESUME* per togliere un'eventuale condizione di emergenza (*st_emrg* = 1)
- abilitare lo stato di calibrazione asse dando il comando *CALON*, lo stato *st_cal* commuta a 1
- in queste condizioni è possibile impostare la tensione analogica tramite il parametro *vout*: il valore è espresso in decimi di Volt (perciò il range di valori introducibili è ± 100). Si consiglia di introdurre valori bassi (5, 10, 15 ...)
- poiché ora il device è utilizzato come "generatore di tensione" l'asse dovrebbe iniziare a muoversi. Se ciò non accade è opportuno verificare la correttezza dei collegamenti. Quando l'asse è in movimento il parametro *frq* indica il valore della frequenza di uno dei segnali in ingresso al contatore bidirezionale, *vel* indica la velocità dell'asse mentre *posit* la posizione secondo l'unità di misura scelta. Se dando tensione positiva la posizione decrementa è necessario invertire le fasi del trasduttore (o fisicamente spostando i cavi, oppure tramite il comando *CNTREV*) o invertire la direzione dell'azionamento
- se con tensione di uscita pari a 0 V si nota comunque che l'asse è in movimento a causa di tensioni di

offset, queste si possono controbilanciare agendo sul parametro *offset*. Per un ottimale esito della taratura, l'operazione deve essere eseguita con il sistema in regime di temperatura

- ora è possibile disabilitare lo stato di calibrazione con il comando *CALOFF* (lo stato *st_cal* commuta a 0)

Parametrizzazione uscita analogica (impostazione parametro *maxvel*)

Il device EANPOS genera il valore di tensione dell'uscita analogica sulla base di una proporzione tra la velocità massima dell'asse e la massima tensione di uscita. Per fare ciò è necessario impostare il parametro *maxvel*, ovvero la velocità con cui si muove l'asse quando viene data tensione massima all'azionamento. Ovviamente l'asse deve avere un comportamento simmetrico rispetto al valore zero di tensione analogica, quindi la velocità deve essere la stessa (in modulo) alla tensione massima sia positiva che negativa.

Per conoscere la velocità massima esistono due vie: il "metodo teorico" presuppone di conoscere la velocità massima del motore (giri massimi dichiarati) da cui si può ricavare facilmente la velocità lineare.

Se non si è a conoscenza della velocità massima dichiarata del motore si deve procedere in questo modo:

- entrare nella modalità di calibrazione (come descritto precedentemente)
- se il sistema lo permette fornire all'azionamento tensione massima e leggere il valore del parametro *vel*
- è possibile anche fornire una tensione minore e calcolare la velocità massima con la proporzione *vout* : 10

$$V = vel : maxvel$$

Ora è quindi possibile introdurre il valore della velocità massima nel parametro *maxvel*.

Prima movimentazione



Attenzione:
prima di movimentare l'asse, verificare il corretto funzionamento dei dispositivi d'emergenza e protezione.

Le procedure fin qui descritte hanno permesso di completare la prima fase di parametrizzazione del device. Ora è possibile eseguire una semplice movimentazione dell'asse. Seguire ad esempio i seguenti passi:

- dare il comando di *INIT* al device, verificare che lo stato *st_init* commuti a 1
- spostare l'asse in una posizione tale per cui si possa compiere un determinato spazio senza incontrare finecorsa
- impostare i parametri *tacc* e *tdec*
- impostare la velocità di posizionamento con il parametro *setvel*
- impostare la quota di posizionamento con il parametro *setpos*
- impostare il parametro *feedfw* a 1000 (100%)
- resettare l'eventuale stato di emergenza con il comando *RESUME*
- avviare il posizionamento con il comando *START*, per arrestare il movimento dare il comando *STOP* (oppure *EMRG*).

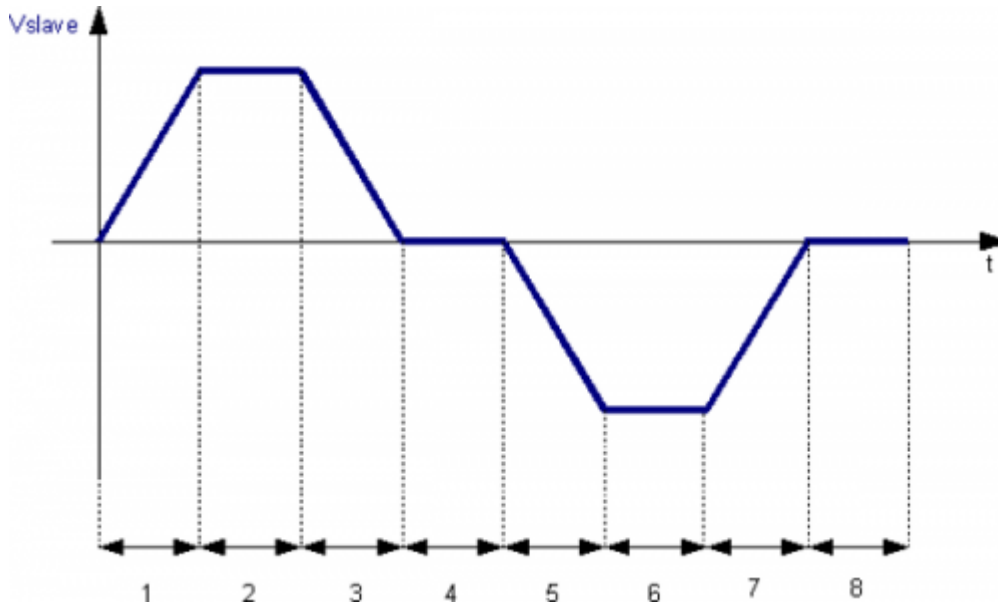
Questa prima movimentazione è stata eseguita senza attivare l'anello di reazione di spazio, quindi qualsiasi errore introdotto da valori di tensione di offset o da agenti esterni non viene corretto.

Esempi di compilazione della cam table

Camma per spandifilo

Consideriamo ad esempio un'applicazione per spandifilo, i passi che dovrà seguire l'asse slave sono:

- partenza con rampa di accelerazione
- raggiungimento di una velocità proporzionale a quella del master
- mantenimento della velocità raggiunta per un percorso predeterminato
- arresto con rampa di decelerazione
- stop dell'asse per un certo spazio del master
- ritorno al punto di partenza con le stesse modalità del tratto d'andata.



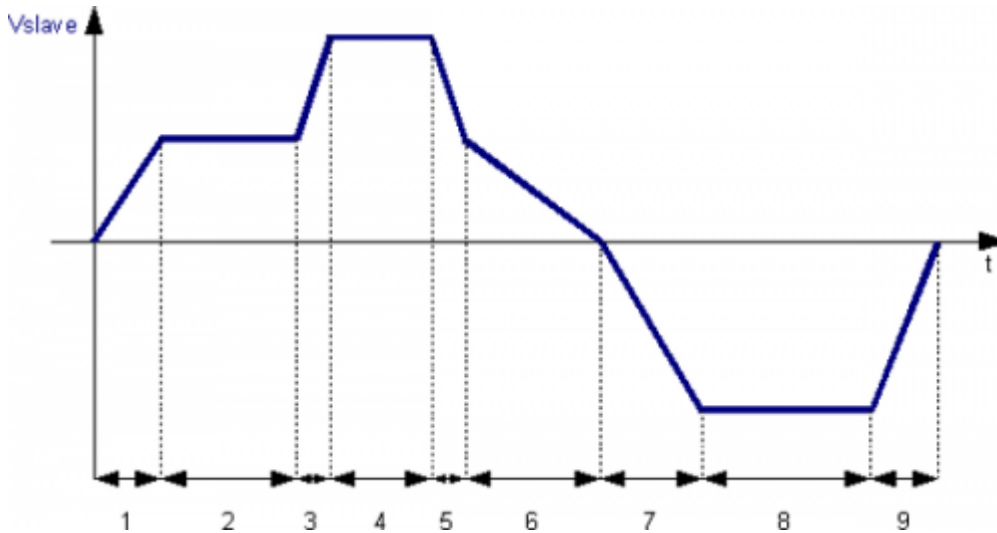
Settore 1	accelerazione con partenza da velocità nulla e arrivo a velocità preimpostata (<i>codeG 131</i>) con incremento positivo della posizione slave; è necessario calcolare correttamente il rapporto fra le quote master e slave perché la velocità di uscita sia corretta
Settore 2	tratto a velocità costante (<i>codeG 133</i>) con incremento positivo della posizione slave
Settore 3	decelerazione con velocità finale 0 (<i>codeG 135</i>) con incremento positivo della posizione slave
Settore 4	arresto dell'asse slave (<i>codeG 133</i>): si programma la quota master come desiderata mentre si lascia nulla quella slave
Settore 5	accelerazione con partenza da velocità nulla e arrivo a velocità preimpostata (<i>codeG 131</i>) con decremento della posizione slave; è necessario calcolare correttamente il rapporto fra le quote master e slave perché la velocità di uscita sia corretta (è possibile utilizzare gli stessi valori del settore 1 con quota slave negativa)
Settore 6	tratto a velocità costante (<i>codeG 133</i>) con decremento della posizione slave
Settore 7	decelerazione con velocità finale 0 (<i>codeG 135</i>) con decremento della posizione slave
Settore 8	arresto dell'asse slave (<i>codeG 133</i>): si programma la quota master come desiderata mentre si lascia nulla quella slave

Dopo l'esecuzione del settore 8 si potranno eseguire delle funzioni che eseguono il ri-fasamento delle posizioni master e slave in impulsi primari sottraendo lo spazio percorso fino a fine settore; in seguito sarà possibile ri-eseguire automaticamente le camma.

Camma per taglio al volo con extravelocità

Consideriamo ad esempio un'applicazione per un semplice taglio al volo, i passi che dovrà seguire l'asse slave sono:

- partenza dell'asse slave con rampa di accelerazione
- raggiungimento della velocità dell'asse master
- mantenimento della velocità raggiunta per la durata dell'operazione di taglio
- concluso il taglio l'asse slave deve accelerare per portarsi ad una velocità maggiore e mantenerla per un certo spazio
- arresto dell'asse slave con rampa di decelerazione
- ritorno dell'asse slave al punto di partenza eseguendo rampe di accelerazione e decelerazione.



Settore 1: accelerazione con partenza da velocità nulla e arrivo alla stessa velocità del master (*codeG 132*) con incremento positivo della posizione slave. Settore 2: tratto intermedio a velocità costante (*codeG 133*) con incremento positivo della posizione slave. In questo settore master e slave percorreranno lo stesso spazio. Settore 3: accelerazione e incremento positivo della posizione dell'asse slave. Il settore impostato non è di accelerazione (*codeG 133*), per fare accelerare lo slave si dovrà impostare uno spazio maggiore rispetto al master. Settore 4: tratto a velocità costante e incremento positivo della posizione slave (*codeG 133*). Settore 5: decelerazione con incremento positivo della posizione slave (*codeG 133*). Si riporta lo slave alla stessa velocità del master, perciò la quota dovrà essere calcolata correttamente. Settore 6: decelerazione con velocità finale 0 (*codeG 135*) con decremento della posizione slave. Settore 7: accelerazione con partenza da velocità nulla e arrivo a velocità preimpostata (*codeG 131*) con decremento della posizione slave. Settore 8: tratto a velocità costante (*codeG 133*) con decremento della posizione slave. Settore 9: decelerazione con velocità finale 0 (*codeG 135*) con decremento della posizione slave.

Dopo l'esecuzione del settore 9 si potranno eseguire delle funzioni che eseguono il ri-fasamento delle posizioni master e slave in impulsi primari sottraendo lo spazio percorso fino a fine settore; in seguito sarà possibile ri-eseguire automaticamente le camma.

Utilizzo di un master simulato

Come è stato spiegato nella descrizione del device CAMMING è possibile che l'asse master sia reale oppure simulata, una via per realizzare un master simulato è dichiarare per il device CAMMING un indirizzo di un trasduttore simulato utilizzando un device di posizionamento (ad esempio EANPOS) dichiarato con il contatore sullo slot 1 (normalmente riservato alla CPU del sistema) e tutte le altre risorse disabilitate.

```

;-----
; Dichiarazione device interni
;-----
INTDEVICE
<nome_device> EANPOS TCamp ICont IntL IAZero IOutA
Master EANPOS 2 1.CNT01 X X.X X.X

```

Il device così configurato viene considerato come un master simulato, esso viene comandato come fosse un normale device tenendo presente che il loop di regolazione deve essere aperto e di conseguenza non serve parametrizzare il PID, è sufficiente impostare il feedforward a 100%.

Esempio di programmazione

Si vuole utilizzare il device EANPOS configurato nell'esempio appena descritto e si ipotizza che il master simulato debba continuare il suo movimento all'infinito.

```

;-----
; Gestione del master simulato
;-----
Master:measure = 1000 ;impostazioni parametri master
Master:pulse = 4000
Master:decpt = 0
Master:unitvel = 1
Master:maxvel = 1000
Master:taccdec = 100
Master:maxpos = 999999
Master:minpos = -999999
INIT Master ;inizializzazione master simulato
WAIT Master:st_init
LOOPFF Master
WAIT NOT Master:st_loopen
RESUME Master
WAIT NOT Master:st_emrg

MAIN:
IF sf01 ;flag che da lo start al master

```

```
IF Master: st_still
  Master:posit = 0
  Master:setvel = 500
  Master:setpos = 999999
  START Master          ;comando di START
ENDIF
IF Master:posit GE 500000 ;aggiorno la posizione per non causare
                          ;un overflow
  Master:posit = 0
ENDIF
ELSE
  IF NOT Master:st_still  ;stop del master simulato
    STOP Master
  ENDIF
ENDIF
WAIT 1
JUMP MAIN
END
```

Documento generato automaticamente da **Qem Wiki** - <https://wiki.qem.it/>

Il contenuto wiki è costantemente aggiornato dal team di sviluppo, è quindi possibile che la versione online contenga informazioni più recenti di questo documento.