

Comau Robotics
Manuale di istruzioni

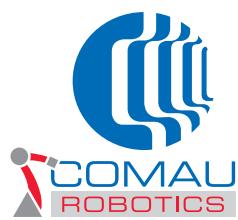


comau.com/robotics

C4G OPEN

Software di Sistema Rel. 3.2x

CR00757550_it-05/1208



Le informazioni contenute in questo manuale sono di proprietà di COMAU S.p.A.

E' vietata la riproduzione, anche parziale, senza preventiva autorizzazione scritta di COMAU S.p.A.

COMAU si riserva il diritto di modificare, senza preavviso, le caratteristiche del prodotto presentato in questo manuale.

Copyright © 2003 by COMAU - Pubblicato in data 120/2008

SOMMARIO

PREFAZIONEVII
Simbologia adottata nel manualeVII
Documentazione di riferimentoVIII
Modification HistoryIX
1. PRESCRIZIONI DI SICUREZZA GENERALI1.1
Responsabilità1.1
Prescrizioni di sicurezza1.2
Scopo1.2
Definizioni1.2
Applicabilità1.3
Modalità operative1.4
2. PRESCRIZIONI DI SICUREZZA PER C4G OPEN.....	.2.1
3. INTRODUZIONE3.1
4. SISTEMA C4G4.1
5. SISTEMA C4G OPEN5.1
6. CONFIGURAZIONE INIZIALE DEL SISTEMA6.1
7. COMUNICAZIONE7.1
Indirizzi7.1
Comunicazione stringente7.1
Avvio della comunicazione7.4
Normale comunicazione7.4
Termine della comunicazione7.4
8. PACCHETTO DI COMUNICAZIONE.....	.8.1
Struttura dei pacchetti di comunicazione8.1
Campi dei pacchetti di comunicazione8.2

Virtualizzazione dei pacchetti di comunicazione	8.3
Dimensione dei pacchetti di comunicazione	8.4
9. DAL BLIP/RAAZ AL PACCHETTO C4G OPEN	9.1
BLIP	9.1
RAAZ	9.2
Comunicazione tra SMP+, PC e DSA	9.2
10. PACCHETTO DI INIZIALIZZAZIONE.....	10.1
11. PROCEDURA DI DRIVE ON	11.1
12. SOFT CONTROL SWITCHING.....	12.1
13. GESTIONE DINAMICA DELLA MODALITÀ APERTA	13.1
14. MODALITÀ	14.1
Introduzione	14.1
Modalità 0	14.2
Funzioni	14.2
Pacchetti di trasmissione	14.2
Funzionamento.....	14.2
Note	14.3
Modalità 0'	14.3
Funzioni	14.3
Pacchetti di trasmissione	14.3
Funzionamento.....	14.4
Modalità 1	14.4
Funzioni	14.4
Pacchetti di trasmissione	14.4
Funzionamento.....	14.4
Note	14.5
Modalità 2	14.5
Funzioni	14.5
Pacchetti di trasmissione	14.5
Funzionamento.....	14.6
Note	14.6
Modalità 3	14.6
Funzioni	14.6
Pacchetti di trasmissione	14.6
Funzionamento.....	14.7

Note	14.7
Modalità 4	14.7
Funzioni	14.7
Pacchetti di trasmissione	14.8
Funzionamento	14.8
Note	14.8
Modalità 5	14.8
Funzioni	14.10
Pacchetti di trasmissione	14.10
Funzionamento	14.11
Note	14.11
Modalità 6	14.11
Funzioni	14.11
Pacchetti di trasmissione	14.12
Funzionamento	14.12
Note	14.12
Modalità 7	14.12
Funzioni	14.13
Pacchetti di trasmissione	14.13
Funzionamento	14.13
Note	14.13
Modalità 8	14.14
Funzioni	14.14
Pacchetti di trasmissione	14.14
Funzionamento	14.15
Modalità 9	14.15
Funzioni	14.15
Pacchetti di trasmissione	14.15
Funzionamento	14.16
Note	14.16
Modalità 1xx	14.17
Funzioni	14.18
Pacchetti di trasmissione	14.18
Funzionamento	14.18
Note	14.19
Modalità 101 (controllo delle accelerazioni)	14.19
Modalità 102 (visual servoing)	14.19
Modalità 103 (controllo di forza)	14.20
Modalità 201	14.21
Funzionamento	14.21
Note	14.22
Modalità 202	14.22
Funzionamento	14.22
Note	14.23
Riepilogo delle modalità	14.25
Note	14.25
Modalità e controlli di sicurezza	14.26
Controlli di sicurezza lato C4G	14.26

Controlli di sicurezza lato PC	14.27
15. MODALITÀ SPECIALI	15.1
Freezing attivo	15.1
DRIVING ON	15.1
Freezing passivo	15.2
Sblocco del server	15.2
Richiesta di DRIVE OFF	15.2
Richiesta di RESTART	15.3
Following error da PC	15.3
16. MONI DEL C4G OPEN	16.1
Moni di tipo 19	16.1
Moni di tipo 20	16.1
Moni di tipo 21	16.2
Moni di tipo 22	16.2
Moni di tipo 23	16.3
Moni di tipo 24	16.3
17. ERRORI C4G OPEN	17.1
18. VARIABILE C4GOPEN_CMD	18.1
19. TRASFORMAZIONI IN PRESENZA DI INFLUENZE CINEMATICHE	19.1
20. SEGNALE DI SINCRONISMO	20.1
21. TIMING	21.1
22. TIMING CALCOLATO DA PC	22.1
23. SWITCH	23.1
24. MACCHINA A STATI	24.1

Concetti generali	24.1
Realizzazione	24.2
Modalità 0.	24.4
Modalità 0'	24.4
Modalità 1.	24.4
Modalità 2.	24.5
Modalità 4.	24.5
Modalità 5.	24.5
Modalità 7.	24.5
Modalità 8.	24.6
Modalità 9.	24.6
Funzionamento	24.6
25. STRATI SOFTWARE	25.1
26. SOFTWARE SU PC	26.1
Introduzione	26.1
Distribuzione Fedora	26.1
Distribuzione Ubuntu	26.3
Requisiti Software	26.3
Requisiti Hardware	26.3
Schede di rete compatibili	26.3
Installazione del Kernel	26.8
Caricamento del modulo RTai	26.9
Caricamento del modulo RTNet	26.9
Esecuzione dell'applicazione C4G Open	26.11
27. DOCUMENTI DI RIFERIMENTO	27.1

PREFAZIONE

- Simbologia adottata nel manuale
- Documentazione di riferimento
- Modification History

Simbologia adottata nel manuale

Di seguito vengono riportati i simboli che rappresentano: **AVVERTENZE**, **ATTENZIONE** e **NOTE** ed il loro significato



Il simbolo indica procedure di funzionamento, informazioni tecniche e precauzioni che, se non vengono rispettate e/o correttamente eseguite, possono causare lesioni al personale.



Il simbolo indica procedure di funzionamento, informazioni tecniche e precauzioni che se non vengono rispettate e/o correttamente eseguite possono causare danni alle apparecchiature.



Il simbolo indica procedure di funzionamento, informazioni tecniche e precauzioni che è essenziale mettere in evidenza.

Documentazione di riferimento

Il presente documento si riferisce all'Unità di Controllo **C4G**.

Il set completo dei manuali del **C4G** è composto da:

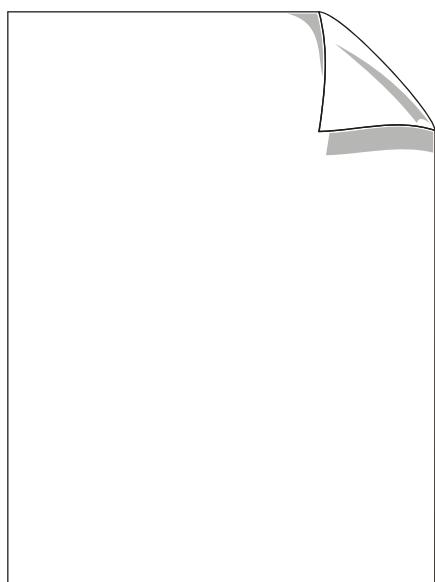
Comau	Unità di Controllo C4G	<ul style="list-style-type: none">– Specifiche Tecniche– Trasporto e installazione– Guida all'integrazione, sicurezze, I/O, comunicazioni– Manutenzione– Uso dell'Unità di Controllo.
-------	------------------------	---

Questi manuali devono essere integrati con i seguenti documenti:

Comau	Robot	<ul style="list-style-type: none">– Specifiche Tecniche– Trasporto e installazione– Manutenzione
	Programmazione	<ul style="list-style-type: none">– PDL2 Programming Language Manual– VP2 - Visual PDL2– Programmazione del movimento
	Applicativi	<ul style="list-style-type: none">– Secondo il tipo di applicativo necessario.
Sintesi	Programmazione	<ul style="list-style-type: none">– <u>C4G Open Library - User's Guide</u>

Modification History

- Nella revisione 03/0908, sono state apportate le seguenti modifiche:
 - nuovo capitolo **Prescrizioni di sicurezza per C4G Open** - ribadita la necessità, per l'utente, di operare al di fuori dello Spazio Protetto, a distanza di sicurezza dalla macchina
 - **Modalità 0'** - riservata a COMAU
 - procedura di **Istallazione del Kernel** - corretta la descrizione
 - **Modalità 5 e Modalità 7** - migliorata la descrizione.



1. PRESCRIZIONI DI SICUREZZA GENERALI

1.1 Responsabilità

- L'integratore dell'impianto deve eseguire l'installazione e la movimentazione del **Sistema Robot e Controllo** in accordo alle Norme di Sicurezza vigenti nel paese dove viene realizzata l'installazione. L'applicazione e l'utilizzo dei necessari dispositivi di protezione e sicurezza, l'emissione della dichiarazione di conformità e l'eventuale marcatura CE dell'impianto, sono a carico dell'Integratore.
- COMAU Robotics & Service declina ogni responsabilità da incidenti causati dall'uso scorretto o improprio del **Sistema Robot e Controllo**, da manomissioni di circuiti, di componenti, del software e dall'utilizzo di ricambi non presenti nella lista ricambi.
- La responsabilità dell'applicazione delle presenti Prescrizioni di Sicurezza è a carico dei preposti che dirigono / sovrintendono alle attività citate al paragrafo **Applicabilità**, i quali devono accertarsi che il **Personale Autorizzato** sia a conoscenza ed osservi scrupolosamente le prescrizioni contenute in questo documento oltre alle Norme di Sicurezza di carattere generale applicabili al **Sistema Robot e Controllo** vigenti nel Paese dove viene realizzata l'installazione.
- La mancata osservanza delle Norme di Sicurezza può causare lesioni permanenti o morte al personale e danneggiare il **Sistema Robot e Controllo**.



L' installazione deve essere eseguita da Personale qualificato all'installazione e deve essere conforme alle norme Nazionali e Locali

1.2 Prescrizioni di sicurezza

1.2.1 Scopo

Le presenti prescrizioni di sicurezza hanno lo scopo di definire una serie di comportamenti ed obblighi ai quali attenersi nell'eseguire le attività elencate al paragrafo [Applicabilità](#).

1.2.2 Definizioni

Sistema Robot e Controllo

Si definisce Sistema Robot e Controllo l'insieme funzionale costituito da: Unità di Controllo, robot, Terminale di Programmazione ed eventuali opzioni.

Spazio Protetto

Si definisce spazio protetto l'area delimitata dalle barriere di protezione e destinata all'installazione e al funzionamento del robot

Personale Autorizzato

Si definisce personale autorizzato l'insieme delle persone opportunamente istruite e delegate ad eseguire le attività elencate al paragrafo [Applicabilità](#).

Personale Preposto

Si definisce preposto il personale che dirige o sovrintende alle attività alle quali siano addetti lavoratori subordinati definiti al punto precedente

Installazione e Messa in Servizio

Si definisce installazione l'integrazione meccanica, elettrica, software del Sistema Robot e Controllo in un qualsiasi ambiente che richieda la movimentazione controllata degli assi Robot, in conformità con i requisiti di sicurezza previsti nella Nazione dove viene installato il Sistema.

Funzionamento in Programmazione

Modo operativo sotto controllo dell'operatore, che esclude il funzionamento automatico e che permette le seguenti attività: movimentazione manuale degli assi robot e programmazione di cicli di lavoro a velocità ridotta, prova del ciclo programmato a velocità ridotta e, quando ammesso, a velocità di lavoro.

Funzionamento in Auto / Remote

Modo operativo in cui il robot esegue autonomamente il ciclo programmato alla velocità di lavoro, con personale all'esterno dello spazio protetto, con barriere di protezione chiuse e inserite nel circuito di sicurezza, con avviamento/arresto locale (posto all'esterno dello spazio protetto) o remoto.

Manutenzione e Riparazione

Si definisce intervento di manutenzione e riparazione l'attività di verifica periodica e/o di sostituzione di parti (meccaniche, elettriche, software) o componenti del Sistema Robot e Controllo e l'attività per identificare la causa di un guasto sopraggiunto, che si conclude con il ripristino del Sistema Robot e Controllo nelle condizioni funzionali di progetto.

Messa Fuori Servizio e Smantellamento

Si definisce messa fuori servizio l'attività di rimozione meccanica ed elettrica del Sistema Robot e Controllo da una realtà produttiva o ambiente di studio.

Lo smantellamento consiste nell'attività di demolizione e smaltimento dei componenti che costituiscono il Sistema Robot e Controllo.

Integratore

Si definisce Integratore la figura professionale responsabile dell'installazione e messa in servizio del Sistema Robot e Controllo.

Uso Scorretto

Si definisce uso scorretto l'utilizzo del sistema al di fuori dei limiti specificati nella Documentazione Tecnica.

Campo d'Azione

Per campo d'azione del Robot si intende il volume di inviluppo dell'area occupata dal Robot e dalle sue attrezzature durante il movimento nello spazio.

1.2.3 Applicabilità

Le presenti Prescrizioni devono essere applicate nell'esecuzione delle seguenti attività:

- [Installazione e Messa in Servizio](#);
- [Funzionamento in Programmazione](#);
- [Funzionamento in Auto / Remote](#);
- [Sfrenatura degli assi robot](#);
- [Spazi di arresto \(casi limite\)](#)
- [Manutenzione e Riparazione](#);
- [Messa Fuori Servizio e Smantellamento](#)

1.2.4 Modalità operative

Installazione e Messa in Servizio

- La messa in servizio è permessa solo quando il Sistema Robot e Controllo è installato correttamente e in modo completo.
- L'installazione e messa in servizio del sistema è consentita unicamente al personale autorizzato.
- L'installazione e la messa in servizio del sistema è ammessa esclusivamente all'interno di uno spazio protetto con dimensioni adeguate ad ospitare il robot e l'attrezzatura con la quale è allestito, senza fuori uscite dalle barriere. Occorre verificare inoltre che nelle condizioni di normale movimento del robot si eviti la collisione dello stesso con parti interne allo spazio protetto (es. colonne della struttura, linee di alimentazione, ecc.) o con le barriere. Se necessario limitare l'area di lavoro del robot per mezzo di tamponi meccanici di finecorsa (vedere gruppi opzionali).
- Eventuali postazioni fisse di comando del robot devono essere posizionate fuori dallo spazio protetto e in un punto tale da consentire la più completa visione dei movimenti del robot.
- Per quanto possibile, l'area di installazione del robot deve essere sgombra da materiali che possano impedire o limitare la visuale.
- Durante le fasi di installazione, il robot e l'Unità di Controllo devono essere movimentati come indicato nella Documentazione Tecnica del prodotto; in caso di sollevamento, verificare il corretto fissaggio dei golfari ed utilizzare unicamente imbracature ed attrezzi adeguate.
- Fissare il robot al supporto di sostegno, con tutti i bulloni e le spine previsti, serrati alle coppie di serraggio riportate sulla Documentazione Tecnica del prodotto.
- Se presenti, rimuovere le staffe di fissaggio degli assi e verificare il corretto fissaggio dell'attrezzatura con cui il robot è allestito.
- Verificare che i ripari del robot siano correttamente fissati e che non vi siano particolari mobili o allentati, controllare inoltre l'integrità dei componenti dell'Unità di Controllo.
- Installare l'Unità di Controllo all'esterno dello spazio protetto: l'Unità di Controllo non deve essere utilizzata come parte delle recinzioni.
- Verificare la coerenza tra la tensione predisposta nell'Unità di Controllo indicata sull'apposita targhetta ed il valore di tensione della rete di distribuzione energia.
- Prima di procedere all'allacciamento elettrico dell'Unità di Controllo, verificare che il disgiuntore sulla rete di distribuzione sia bloccato in posizione d'apertura.
- Il collegamento tra l'Unità di Controllo e la rete di alimentazione trifase dello stabilimento, deve essere realizzato tramite un cavo armato quadripolare (3 fasi + terra) di dimensioni adeguate alla potenza installata sull'Unità di Controllo vedere la Documentazione Tecnica del prodotto.
- Il cavo d'alimentazione deve entrare nell'Unità di Controllo attraverso l'apposito passacavo ed essere correttamente bloccato.
- Collegare il conduttore di terra (PE) e di seguito collegare i conduttori di potenza all'interruttore generale.
- Collegare il cavo d'alimentazione, collegando per primo il conduttore di terra al disgiuntore sulla rete di distribuzione energia dopo avere verificato con apposito

strumento che i morsetti del disgiuntore siano fuori tensione. Si raccomanda di connettere l'armatura del cavo alla terra.

- Collegare i cavi di segnali e potenza tra Unità di Controllo e robot.
- Collegare il robot a terra o all'Unità di Controllo o ad una vicina presa di terra.
- Verificare che la/le porta/e dell'Unità di Controllo siano chiuse con l'apposita chiave.
- L'errato collegamento dei connettori può provocare danni permanenti ai componenti dell'Unità di Controllo.
- L'Unità di Controllo C4G gestisce al suo interno i principali interblocchi di sicurezza (cancelli, pulsante di abilitazione, ecc.). Collegare gli interblocchi di sicurezza dell'Unità di Controllo C4G con i circuiti di sicurezza della linea avendo cura di realizzarli come richiesto dalle Norme di Sicurezza. La sicurezza dei segnali di interblocco provenienti da linea trasferta (arresto d'emergenza, sicurezza cancelli, ecc.), ossia la realizzazione di circuiti corretti e sicuri è a carico dell'integratore del Sistema Robot e Controllo.



Nel circuito di arresto di emergenza della cella/linea è necessario includere i contatti dei pulsanti di arresto di emergenza dell'unità di controllo, disponibili su X30. I pulsanti non sono interbloccati internamente al circuito di arresto d'emergenza dell'Unità di Controllo.

- Non si garantisce la sicurezza del sistema in caso di realizzazione errata, incompleta o mancante di tali interblocchi.
- Nel circuito di sicurezza è previsto l'arresto controllato (IEC 60204-1, arresto di categoria 1) per gli ingressi di sicurezza Auto Stop/ General Stop ed Emergenza. L'arresto controllato è attivo solo in stato Automatico; in Programmazione l'esclusione della potenza (apertura dei contattori di potenza) avviene in modo immediato. La modalità per la selezione del tempo dell'arresto controllato (impostabile sulla scheda ESK) è riportato nel Manuale d'Installazione.
- Nella realizzazione delle barriere di protezione, specialmente per le barriere ottiche e le porte d'ingresso, tenere presente i tempi e gli spazi di arresto del robot in funzione della categoria di arresto (0 oppure 1) e della massa del robot.



Verificare che il tempo di arresto controllato sia coerente con il tipo di Robot collegato all'Unità di Controllo. Il tempo di arresto si seleziona attraverso i selettori SW1 e SW2 sulla scheda ESK.

- Verificare che le condizioni ambientali e operative di lavoro non eccedano i limiti specificati nella Documentazione Tecnica del prodotto specifico.
- Le operazioni di calibrazione devono essere eseguite con la massima attenzione, come riportato nella Documentazione Tecnica del prodotto specifico, e si devono concludere con la verifica della corretta posizione della macchina.

- Per le fasi di caricamento o aggiornamento del software di sistema (per esempio dopo la sostituzione di schede), utilizzare unicamente il software originale consegnato da COMAU Robotics & Service. Attenersi scrupolosamente alla procedura di caricamento del software di sistema descritta nella Documentazione Tecnica fornita con il prodotto specifico. Dopo il caricamento eseguire sempre alcune prove di movimentazione del Robot, a velocità ridotta rimanendo al di fuori dello spazio protetto.
- Verificare che le barriere dello spazio protetto siano correttamente posizionate.

Funzionamento in Programmazione

- La programmazione del robot è consentita unicamente al personale autorizzato.
- Prima di procedere alla programmazione, l'operatore deve controllare il **Sistema Robot e Controllo** per assicurarsi che non sussistano condizioni anomale potenzialmente pericolose e che nello spazio protetto non siano presenti persone.
- Per quanto possibile la programmazione deve essere comandata restando all'esterno dello spazio protetto.
- Prima di operare all'interno dello **Spazio Protetto**, l'operatore deve accertarsi, rimanendo all'esterno dello spazio protetto, che tutte le necessarie protezioni e i dispositivi di sicurezza siano presenti e funzionanti e in particolare che il Terminale di Programmazione funzioni correttamente (velocità ridotta, enabling device, dispositivo di arresto d'emergenza, ecc.).
- Durante le fasi di programmazione, la presenza all'interno dello **Spazio Protetto** è consentita al solo operatore in possesso del Terminale di Programmazione.
- Se è indispensabile la presenza di un secondo operatore nell'area di lavoro durante la verifica del programma, questi dovrà disporre di un suo enabling device (dispositivo di abilitazione) interbloccato con i dispositivi di sicurezza.
- L'attivazione dei motori (DRIVE ON) deve essere comandata sempre da posizione esterna al campo d'azione del robot, dopo aver verificato che nell'area interessata non vi sia la presenza di persone. L'operazione di attivazione motori si considera conclusa alla comparsa della relativa indicazione di stato macchina.
- Durante la programmazione l'operatore deve mantenersi ad una distanza dal robot tale da permettergli di scansare eventuali movimenti anomali della macchina, e comunque in posizione tale da evitare possibili rischi di costrizione tra il robot e parti della struttura (colonne, barriera, ecc.), o tra parti mobili del robot stesso.
- Durante la programmazione l'operatore deve evitare di trovarsi in corrispondenza di parti del robot che possono, per effetto della gravità, compiere dei movimenti verso il basso oppure verso l'alto o lateralmente (nel caso di montaggio su piano inclinato).
- La prova del ciclo programmato alla velocità di lavoro, in alcune situazioni in cui si renda necessario un controllo visivo a breve distanza, con la presenza dell'operatore all'interno dello spazio protetto, deve essere attivato solo dopo aver effettuato un ciclo completo di prova a velocità ridotta. La prova deve essere comandata da una distanza di sicurezza.
- Occorre prestare particolare attenzione quando si programma mediante Terminale di Programmazione: in tal caso, benché tutti i dispositivi di sicurezza hardware e software siano in funzione, il movimento del robot dipende comunque dall'operatore.

- La prima esecuzione di un nuovo programma può comportare il movimento del robot lungo una traiettoria diversa da quella attesa.
- La modifica di passi del programma (es. spostamento di un passo da un punto ad un altro del flusso, registrazione errata di un passo, modifica della posizione del robot fuori dalla traiettoria che raccorda due passi del programma), può dare origine a movimenti non previsti dall'operatore in fase di prova del programma stesso.
- In entrambi i casi operare con attenzione, mantenendosi comunque al di fuori del campo d'azione del robot e provare il ciclo a velocità ridotta.

Funzionamento in Auto / Remote

- L'attivazione del funzionamento in automatico (stati AUTO e REMOTE) è consentita unicamente con il **Sistema Robot e Controllo** integrato in un'area dotata di barriere di protezione correttamente interbloccate, come prescritto dalle Norme di Sicurezza vigenti nel Paese dove viene realizzata l'installazione.
- Prima di attivare il funzionamento in automatico l'operatore deve verificare il Sistema Robot e Controllo e lo spazio protetto per accertarsi che non sussistano condizioni anomale potenzialmente pericolose.
- L'operatore può attivare il funzionamento automatico solo dopo aver verificato:
 - che il Sistema Robot e Controllo non si trovi in stato di manutenzione o riparazione;
 - che le barriere di protezione siano correttamente collocate;
 - che non vi sia personale all'interno dello spazio protetto;
 - che le porte dell'Unità di Controllo siano chiuse con l'apposita chiave;
 - che i dispositivi di sicurezza (arresto d'emergenza, sicurezze delle barriere di protezione) siano funzionanti;
- Particolare attenzione deve essere posta alla selezione dello stato remote, in cui il PLC della linea può compiere operazioni automatiche di accensione motori e avvio del programma.

Sfrenatura degli assi robot

- In assenza della forza motrice, lo spostamento degli assi del robot è possibile per mezzo di dispositivi opzionali per sfrenatura e adeguati mezzi di sollevamento. Tali dispositivi permettono unicamente la disattivazione del freno di ciascun asse. In questo caso, tutte le sicurezze del sistema (compreso l'arresto d'emergenza e il pulsante di abilitazione) sono escluse inoltre gli assi robot possono muoversi verso l'alto oppure verso il basso a causa di forze generate dal sistema di bilanciamento oppure per gravità.



Prima di utilizzare i dispositivi per sfrenatura manuale si raccomanda l'imbragatura del robot oppure l'aggancio ad un carroponte.

Spazi di arresto (casi limite)

- Per ogni tipo Robot possono essere richiesti alla COMAU Robotics & Service gli spazi di arresto limite.
- Esempio: considerando il Robot NJ 370-2.7 in modalità automatico, nelle condizioni di massima estensione, massimo carico e massima velocità, a seguito pressione del pulsante di stop (fungo rosso su WiTP) si ottiene la completa fermata del Robot
NJ 370-2.7 in circa 85 ° di movimento corrispondenti a circa 3000 mm di

spostamento misurati sulla flangia TCP. Nelle condizioni indicate, il tempo di fermata del Robot NJ 370-2.7 è di 1,5 secondi.

- Considerando il robot in modalità programmazione (T1), a seguito pressione del pulsante di stop (fungo rosso su WiTP) si ottiene la completa fermata di un Robot NJ 370-2.7 in circa 0,5 secondi.

Manutenzione e Riparazione

- Al montaggio in COMAU Robotics & Service, il robot viene rifornito con lubrificanti che non contengono sostanze pericolose per la salute tuttavia in alcuni casi, l'esposizione ripetuta e prolungata al prodotto può provocare manifestazioni cutanee irritative oppure, in caso di ingestione, malessere.

Misure di Pronto Soccorso. In caso di contatto con gli occhi e con la pelle: lavare con abbondante acqua le zone contaminate; in caso persistessero fenomeni irritativi consultare un medico.

In caso di ingestione non indurre il vomito o somministrare prodotti per via orale; consultare un medico al più presto.

- Le operazioni di manutenzione, ricerca guasti e riparazione sono consentite unicamente al personale autorizzato.
- L'attività di manutenzione e riparazione in corso deve essere segnalata con apposito cartello indicante lo stato di manutenzione, posto sul pannello comandi dell'Unità di Controllo, fino ad operazione ultimata anche se temporaneamente sospesa.
- Le operazioni di manutenzione e sostituzione di componenti o dell'Unità di Controllo, devono essere eseguite con l'interruttore generale in posizione di aperto e bloccato per mezzo di un lucchetto di sicurezza.
- Anche se l'Unità di Controllo non è alimentata (interruttore generale aperto), possono essere presenti tensioni interconnesse, provenienti dal collegamento con unità periferiche o con sorgenti di alimentazioni esterne (es. input/output a 24 Vcc). Disattivare le sorgenti esterne quando si opera sulle parti del sistema interessate.
- La rimozione di pannelli, schermi protettivi, griglie ecc. è consentita solo con interruttore generale aperto e bloccato con lucchetto di sicurezza.
- I componenti guasti devono essere sostituiti con altri dello stesso codice oppure equivalenti definiti dalla COMAU Robotics & Service.



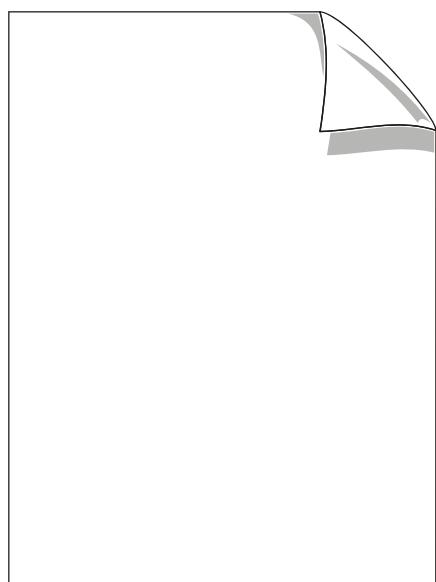
Dopo la sostituzione del modulo ESK, sul nuovo modulo verificare che l'impostazione del tempo di arresto sui selettori SW1 e SW2 sia coerente con il tipo di Robot collegato all'Unità di Controllo.

- Le attività di ricerca guasti e di manutenzione devono essere eseguite, per quanto possibile, all'esterno dello spazio protetto.
- Le attività di ricerca guasti eseguite sul controllo devono, per quanto possibile, essere eseguite in assenza di alimentazione.
- Qualora si renda necessario, nel corso dell'attività di ricerca guasti, eseguire interventi con l'Unità di Controllo alimentata, devono essere prese tutte le precauzioni richieste dalle Norme di Sicurezza quando si opera in presenza di tensioni pericolose.
- L'attività di ricerca guasti sul robot deve essere eseguita con alimentazione di potenza disattivata (DRIVE OFF).

- Al termine dell'intervento di manutenzione e ricerca guasti, devono essere ripristinate le sicurezze disattivate (pannelli, schermi protettivi, interblocchi, ecc.).
- L'intervento di manutenzione, riparazione e ricerca guasti deve essere concluso con la verifica del corretto funzionamento del **Sistema Robot e Controllo** e di tutti i dispositivi di sicurezza, eseguita restando al di fuori dello spazio protetto.
- Durante le fasi di caricamento del software (per esempio dopo la sostituzione di schede elettroniche) è necessario utilizzare il software originale consegnato da COMAU Robotics & Service. Attenersi scrupolosamente alla procedura di caricamento del software di sistema descritta nella Documentazione Tecnica del prodotto specifico; dopo il caricamento eseguire sempre un ciclo di prova per sicurezza, restando al di fuori dello spazio protetto.
- Lo smontaggio di componenti del robot (es. motori, cilindri per bilanciamento, ecc.) può provocare movimenti incontrollati degli assi in qualsiasi direzione: prima di iniziare una procedura di smontaggio è quindi necessario fare riferimento alle targhette di avvertenze applicate sul robot e alla Documentazione Tecnica fornita.
- E' assolutamente vietato rimuovere la copertura di protezione delle molle del robot.

Messa Fuori Servizio e Smantellamento

- La messa fuori servizio e la rimozione del Sistema Robot e Controllo è consentita unicamente al **Personale Autorizzato**.
- Portare il robot in posizione di trasporto e montare le staffe di bloccaggio assi (quando previsto) facendo riferimento alla targhetta applicata sul robot e alla Documentazione Tecnica del robot stesso.
- Prima di procedere alla messa fuori servizio è obbligatorio togliere la tensione di rete all'ingresso dell'Unità di Controllo (disinserire il disgiuntore sulla rete di distribuzione energia e bloccarlo in posizione aperta).
- Dopo aver verificato con apposito strumento che i morsetti siano fuori tensione, scollegare il cavo di alimentazione dal disgiuntore sulla rete di distribuzione energia, staccando prima i conduttori di potenza e successivamente quello di terra. Collegare il cavo di alimentazione dall'Unità di Controllo e rimuoverlo.
- Collegare prima i cavi di collegamento fra il robot e l'Unità di Controllo e successivamente il conduttore di terra.
- Se è presente, scollegare l'impianto pneumatico del robot dalla rete di distribuzione dell'aria.
- Verificare che il robot sia correttamente bilanciato e se necessario imbracarlo correttamente quindi smontare i bulloni di fissaggio del robot dal supporto di sostegno.
- Rimuovere il robot e l'Unità di Controllo dall'area di lavoro, adottando tutte le prescrizioni indicate nella Documentazione Tecnica dei prodotti; se si rende necessario il sollevamento, verificare il corretto fissaggio dei golfari e utilizzare unicamente imbracature ed attrezzi adeguati.
- Prima di effettuare operazioni di smantellamento (smontaggio, demolizione e smaltimento) dei componenti che costituiscono il Sistema Robot e Controllo, consultare la COMAU Robotics & Service, o una delle sue filiali, che indicherà, in funzione del tipo di robot e di Unità di Controllo, le modalità operative nel rispetto dei principi di sicurezza e di salvaguardia ambientale.
- Le operazioni di smaltimento rifiuti devono essere eseguite in accordo con la legislazione della Nazione in cui è installato il Sistema Robot e Controllo.



2. PRESCRIZIONI DI SICUREZZA PER C4G OPEN



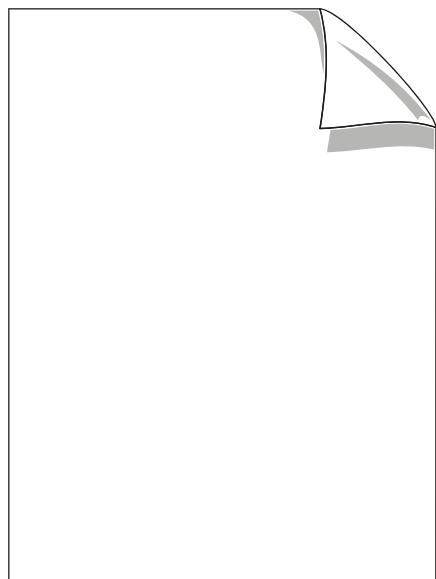
Per il corretto funzionamento del C4G Open fare riferimento alle prescrizioni di sicurezza indicate nel [Cap.1. - Prescrizioni di Sicurezza Generali](#).

In particolare si richiama all'attenzione dell'utente che, essendo il controllore C4G Open un potente strumento di ricerca/prototipazione rapida, occorre prestare massima attenzione nel movimentare il robot, poiché il pc esterno diviene l'artefice del movimento del robot stesso.

Assicurarsi inoltre che, durante le prove, non vi siano utenti nel raggio di azione del robot e che si siano prese tutte le precauzioni di sicurezza vigenti nel paese di utilizzo del sistema. COMAU declina ogni responsabilità in caso di incidenti causati dall'uso scorretto o improprio del sistema.

Si ribadisce quindi che l'operatore deve sempre stare al di fuori dello [Spazio Protetto](#) e dotare la cella di lavoro degli opportuni dispositivi di sicurezza in base alle cogenti normative di legge (es. cancelli di sicurezza anti intrusione).

All'uopo si ricorda che il funzionamento del C4G Open è attivo unicamente quando il selettore di stato è posto su "AUTO".



3. INTRODUZIONE

Sempre più spesso le università e le aziende impiegano sistemi di sviluppo e di "prototipazione veloce". A tal fine sono utilizzate tecnologie emergenti che consentono di esaminare e valutare analiticamente e rapidamente nuove idee, sviluppare prodotti innovativi e accorciare i tempi per il lancio sul mercato dei prodotti.

Nell'industria automobilistica l'uso delle tecnologie di prototipazione veloce è ormai prassi consolidata. Il time-to-market è uno dei fattori critici di successo, in un mercato altamente competitivo.

Le tre caratteristiche principali degli strumenti e delle tecnologie che si utilizzano in questa ottica sono:

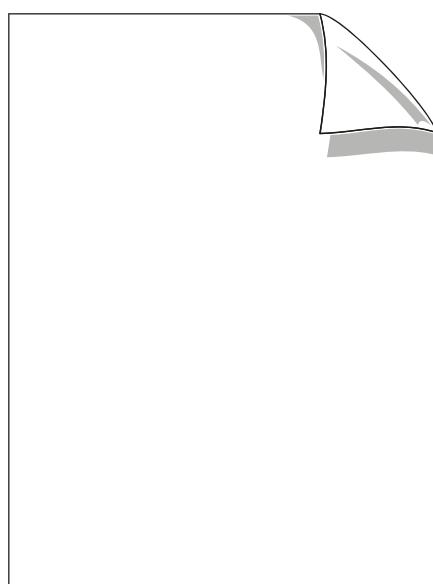
- affidabilità e sicurezza,
- facilità di impiego,
- libertà nella definizione e creazione di nuove strategie di controllo del processo produttivo.

In questo documento viene presentata l'evoluzione "**Open**" del Sistema di Controllo **COMAU C4G** per robot industriali: l'opzione software chiamata "**C4G OPEN**".

C4G Open si presenta come un'architettura software e hardware estremamente innovativa che consente una facile e sicura integrazione dell'Unità di Controllo del robot con un personal computer esterno al fine di coadiuvare la programmazione della cella automatizzata a più livelli, al fine di comandare e/o controllare dall'esterno un robot industriale con la possibilità di utilizzare misure provenienti da sensori esterni.

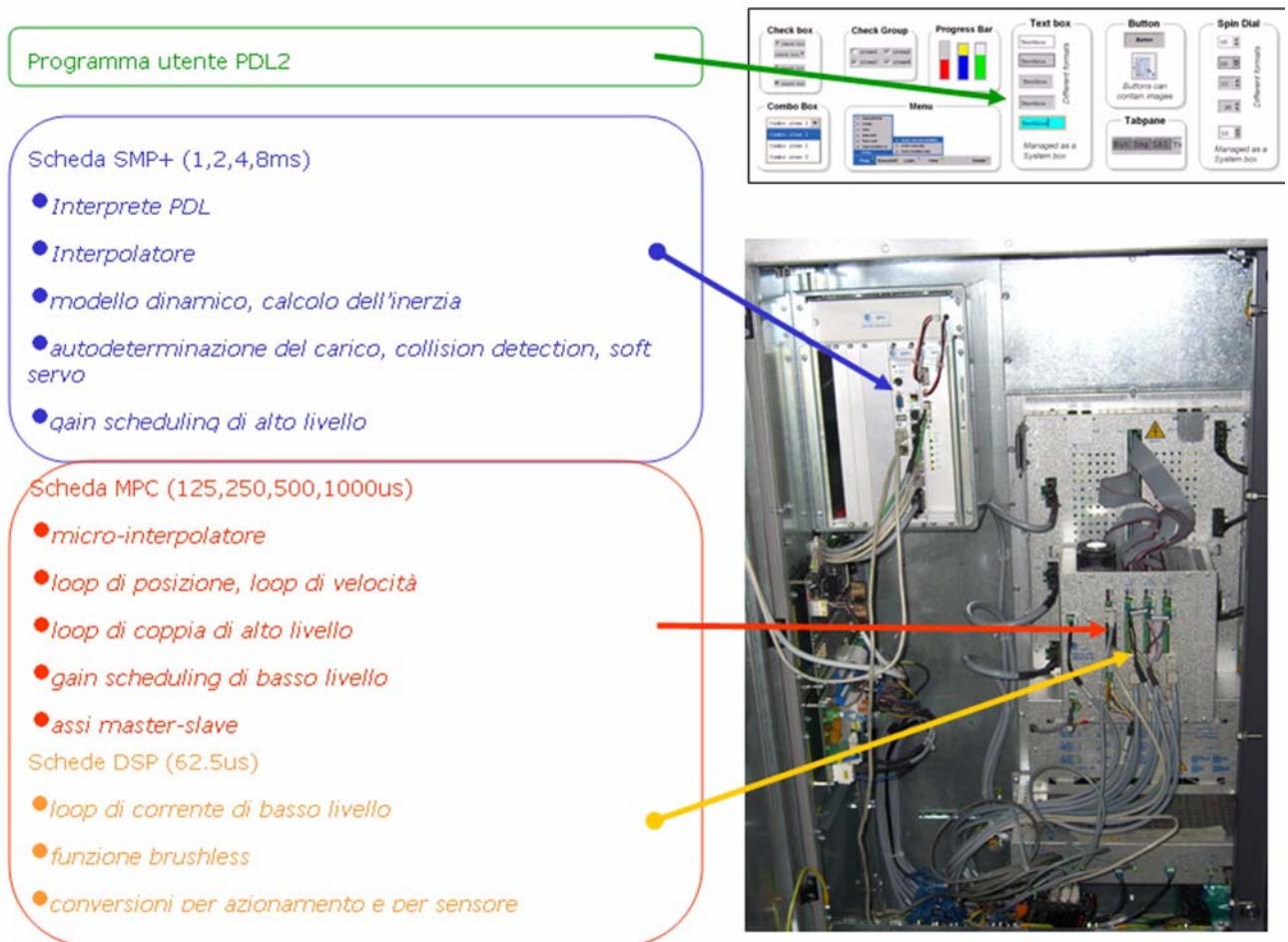
Questa architettura di controllo è stata pensata per poter allestire rapidamente e con costi contenuti nuove celle robotizzate che mirano ad utilizzare funzioni innovative di controllo del robot ed applicazioni di robotica avanzata.

Attraverso l'uso di una rete di sensori esterni, senza vincoli di tipo e di modo di impiego, vengono superati quelli che sono i tipici problemi di "chiusura" dei sistemi di controllo per robot industriali offrendo agli utilizzatori un'ampia gamma di modalità d'accesso. Per questo motivo C4G Open è considerato una architettura di controllo e un "ambiente" di lavoro affidabile, sicuro, molto interessante sia per il mondo accademico che per le industrie (integratori di celle robotizzate, aziende di robotica innovativa).



4. SISTEMA C4G

Fig. 4.1 - Il Sistema C4G



In Fig. 4.1 sono presenti gli elementi software ed hardware che compongono un sistema C4G di produzione, e le funzioni che essi svolgono.

Di interesse in questa sede sono i seguenti componenti del C4G:

- interfaccia PDL2 per la programmazione del robot
- la scheda di controllo SMP+, su cui vengono realizzati tutti i processi di alto livello:
 - interpretazione dei programmi dell'utente;
 - gestione delle interfacce operatorie (anche remote);
 - gestione delle comunicazioni di rete;
 - generazione della traiettoria;
 - calcolo del modello dinamico del manipolatore e gestione dei task ad esso riconducibili;
 - rilevamento collisioni e gestione della diagnostica di sistema;
 - processo di regolazione centralizzato di livello gerarchico alto (gestione della controllo adattativo);
 - controllo di sincronizzazione degli assi;

- gestione di tutti i dispositivi di I/O;
- la scheda di controllo MPC, interna all'azionamento multi asse, che attua:
 - la micro-interpolazione fine della traiettoria del robot;
 - il controllo adattativo di posizione del robot che è composto da 3 livelli di regolazione interagenti tra loro e con il livello di controllo centralizzato;
 - le logiche di adattamento del controllo di posizione del robot (controllo adattativo), in funzione delle informazioni disponibili a livello centralizzato;
 - la parte "real time" della diagnostica di sistema;
 - la gestione di assi "master-slave";
- le schede di controllo DSP , interne all'azionamento multi asse, che attuano:
 - il controllo delle correnti dei motori elettrici e del processo di generazione della coppia motrice (controllo ad assi indipendenti);
 - gestione dello stadio di potenza;
 - gestione del sensore di posizione e acquisizione misure di posizione angolare del motore;
 - gestione I/O digitali ed analogici ad alta velocità.

L'utente può utilizzare il C4G scrivendo in un file la sequenza di istruzioni che descrivono il compito che deve essere eseguito dal robot. Le istruzioni utilizzabili sono quelle messe a disposizione da un particolare linguaggio di programmazione molto simile al PASCAL chiamato PDL2.

I programmi PDL2 sono interpretati dal software presente sulla scheda SMP+ (System Master Processor Plus). Se una istruzione richiede un movimento, il software su SMP+ interpreta le caratteristiche generali di quel movimento tra cui i punti di partenza ed arrivo e la velocità alla quale deve essere effettuato. Successivamente, viene generata la traiettoria puntuale per eseguire il movimento richiesto.

All'interno della scheda MPC (Motorola Power PC) vengono eseguiti i loop di posizione e velocità che hanno il compito di calcolare la corrente di riferimento che permette di far compiere al robot la traiettoria imposta dal generatore di traiettoria.

Le informazioni in uscita dalla scheda MPC vengono considerate come riferimenti da attuare per le schede DSP che forniscono le correnti ai motori che fanno, infine, muovere il robot.

La scheda MPC e le schede DSP formano il DSA (Digital Servo Amplifier). In questo documento si farà riferimento generalmente a DSA senza specificare se si intende parlare della MPC o dei DSP.

5. SISTEMA C4G OPEN

C4G Open è un'integrazione del controllo robot industriale, è una opzione del prodotto standard, in cui un personal computer esterno comunica in tempo reale con il controllo robot ed interagisce a vari livelli nell'azione di controllo (servosistema) della macchina e nella generazione della traiettoria.



Il contributo esterno affianca, integra o sostituisce quello del controllo industriale; tuttavia esso mantiene sempre un ruolo attivo durante il processo. Questo permette una gestione affidabile e sicura della diagnostica di sistema e l'esecuzione di uno o più programmi applicativi.

Attraverso l'innovazione del "**controllore aperto**" la struttura hardware esterna e l'Unità di Controllo industriale formano un'architettura di controllo unica che ha potenzialità notevoli nello sviluppo di applicazioni robotiche avanzate.

Le applicazioni che possono essere realizzate attraverso questo sistema possono essere considerate operazioni di controllo robot avanzate, non completamente convenzionali. Ad esempio:

- una rete di sensori gestiti dal personal computer che esegue anche un'azione di controllo o di generazione di traiettoria a livello gerarchico elevato, oppure
- una rete di telecamere e sensori di forza in cooperazione ed a completamento delle matematiche e degli algoritmi che regolano il movimento del braccio robotizzato.

L'architettura di sistema C4G Open poggia sulla comunicazione real time su rete Ethernet, che utilizza un protocollo UDP proprietario, tra la scheda **SMP+** ("client" della rete) e la scheda **MPC** (che agisce come "server").

L'apertura del controllo robot è stata realizzata inserendo un secondo processo "server" che è il **PC** collegato in rete (si faccia riferimento alla [Fig. 5.1](#)):

- il blocco "SMP+" contiene il sistema operativo di base del C4G, l'interprete delle istruzioni PDL2 ed il generatore di traiettoria,
- il blocco "DSA" contiene il codice di controllo dei loop di posizione, velocità e corrente,
- il blocco "PC" rappresenta il dispositivo esterno che deve interfacciarsi con il sistema COMAU,
- **Switch** è uno switch hardware che permette di comunicare con i blocchi "SMP+", "DSA" e "PC".



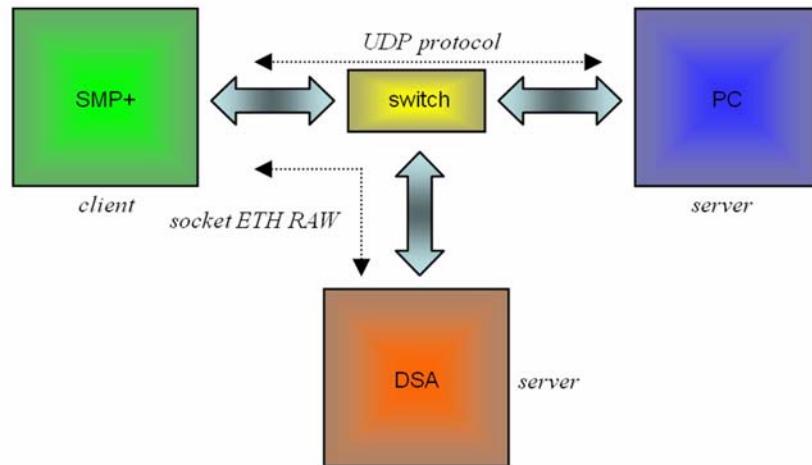
Nel presente documento con **SMP+ si intende la parte del software presente nella scheda **SMP+** che ha il compito di realizzare la comunicazione con DSA e PC. Alla parte restante del software presente sulla scheda **SMP+** si fa riferimento con il termine generale "generatore di traiettoria".**

Il protocollo di comunicazione "SMP+"- "DSA" è un protocollo dedicato non standard per ottimizzare il passaggio dei dati ed è chiamato RAW.

Il protocollo di comunicazione "SMP+"- "PC" è il classico protocollo UDP. La scelta di utilizzare il protocollo UDP per far comunicare dispositivi esterni al C4G è nata dalla

generalità di tale protocollo e dal fatto che COMAU ha già avuto esperienze operative per interfacciare il controllo C4G con azionamenti diversi da DSA.

Fig. 5.1 - Controllore aperto



Non tutte le configurazioni realizzabili con il sistema C4G di produzione possono essere riprodotte nel sistema C4G OPEN. Le restrizioni del software e dell'hardware per il sistema C4G OPEN sono le seguenti:

Tab. 5.1 - Restrizioni del C4G Open

Massimo numero di ARM	4
Massimo numero di azionamenti (DSA)	2
Massimo numero (totale di assi)	20
Massimo numero (per ogni ARM) di assi (compresi gli ausiliari)	10
Massimo numero (per ogni ARM) dei soli assi ausiliari	4

6. CONFIGURAZIONE INIZIALE DEL SISTEMA

Per poter utilizzare il C4G in modalità aperta occorre innanzitutto configurare opportunamente il sistema.

La configurazione avviene mediante i seguenti passi:

- a. esecuzione di un set di istruzioni PDL2 di settaggio della modalità aperta.
Il set di istruzioni PDL2 di settaggio sono le seguenti:
 - **\$ARM_DATA[i].C4GOPEN_JNT_MASK** : scalare contenente la maschera degli assi presenti sull'ARM i-esimo sui quali si vuole utilizzare la modalità aperta,
 - **\$ARM_DATA[i].C4GOPEN_MODE[j]** : vettore contenente, per ogni asse, la modalità aperta che si vuole utilizzare.
- b. salvataggio della configurazione del sistema. A seconda che si desideri immettere il comando da programma PDL2 o da Terminale di Programmazione, si deve agire come segue:
 - da programma PDL2 - comando **CSA: CONFIGURE SAVE ALL**
 - da Terminale di Programmazione: aprire la **SETUP Page**, scegliere la funzione **Config.** (tasto funzionale **F1**) e selezionare la voce **1 - Salva nel file di configurazione (Save)**.
- c. ripartenza del sistema. A seconda che si desideri immettere il comando da programma PDL2 o da Terminale di Programmazione, si deve agire come segue:
 - da programma PDL2 - comando **CCRC: CONFIGURE CONTROLLER RESTART COLD**
 - da Terminale di Programmazione: dalla **Home Page** scegliere la funzione **Riavvio (F4)** e selezionare la voce **1 - Completo (Cold)**.

Alla successiva ripartenza, se per il robot specifico è prevista la possibilità di comandarlo in modalità aperta e sul controllo specifico è stata attivata l'opzione software del C4G Open, il sistema si predisponde per comunicare con il dispositivo esterno.

Si noti che la funzionalità di stand-by non è supportata.

La modalità selezionata non vincola la comunicazione real-time instaurata alla ripartenza del sistema, essa può essere cambiata dinamicamente tra un movimento ed il successivo, passando anche attraverso la modalità che riserva il pieno controllo del sistema a C4G. Ciò che, invece, viene fissato una volta per tutte è la maschera degli assi da pilotare in modalità aperta. Essa viene definita in fase di configurazione e non può essere modificata se non con una successiva ripartenza.

Dopo aver configurato il sistema per la modalità aperta, alla ripartenza viene eseguita una verifica sulla correttezza della definizione della maschera degli assi per i quali è prevista la modalità aperta. Se questa non è inclusa in quella degli assi definiti dalla caratterizzazione, il software ignora gli assi non coerenti con essi.

Un asse non utilizzato in controllore aperto non viene considerato nella composizione del buffer di comunicazione: in altre parole il buffer di comunicazione viene riempito in funzione dei soli assi configurati OPEN, è cura di chi invia e chi riceve garantire la

mappatura tra posizione dell'asse nel buffer e l'indice dell'asse da pilotare.

Questa scelta permette di ottimizzare la comunicazione tra SMP+ e PC, in quanto, la lunghezza di ogni pacchetto di dati non contiene campi vuoti associati ad assi non interessati dalla modalità aperta. È chiaro, d'altro canto, che, se in fase iniziale non si conosce quale asse deve essere comandato in modalità OPEN, si possono dichiarare tutti gli assi in modalità aperta e poi cambiare strategia di controllo con un opportuno campo del pacchetto di comunicazione (vd.[Cap.8. - Pacchetto di Comunicazione](#)).

Per quanto riguarda la mappatura tra posizione dell'asse nel buffer di comunicazione e indice dell'asse da pilotare è utile introdurre la seguente nomenclatura associata, a seconda del contesto, agli indici degli assi:

- gli indici degli assi **logici** sono quelli visti direttamente dall'operatore cui fa diretto riferimento l'interfaccia uomo-macchina del C4G;
- gli indici degli assi **fisici** sono utilizzati per cablare i motori e quindi si riferiscono alla configurazione della parte dell'azionamento, della motorizzazione e del cablaggio del sistema;
- gli indici degli assi **RAW** sono utilizzati nella costruzione dei pacchetti di comunicazione tra SMP+ e DSA (il pacchetto che SMP+ invia a DSA è detto BLIP, mentre quello spedito da DSA a SMP+ è detto RAAZ). Possono essere presenti fino a 2 DSA, quindi, alcuni assi si riferiscono al DSA 1 e i restanti al DSA 2. Per ordinare le grandezze relative agli assi dei due diversi DSA e costruire due pacchetti diversi occorre ricavare gli indici assi RAW a partire dagli indici degli assi fisici;
- gli indici degli assi **OPEN** sono utilizzati nella costruzione del pacchetto di comunicazione tra SMP+ e PC. Possono riferirsi ad entrambi i DSA, quindi sono simili agli assi RAW di tutti i DSA eccetto per gli assi per i quali non è predisposta la modalità aperta.

Le trasformazioni da eseguire per passare **da assi logici ad assi RAW e viceversa** sono le seguenti:

- **pacchetto BLIP/RAAZ - assi RAW-->assi fisici-->assi logici-->assi OPEN - pacchetto per PC**
- **pacchetto da PC - assi OPEN-->assi logici-->assi fisici-->assi RAW - pacchetto BLIP/RAAZ**

Segue la dichiarazione delle variabili PDL2 per utilizzare la funzionalità C4G OPEN:

- maschera degli assi per i quali è applicata la modalità aperta:

```

SS_ENTRY C4GOPEN_JNT_MASK
  GBL_INT
  CAT_MAJOR SSL_ARM
  CAT_MINOR SSL_A_CNFG
  TITLE_MDF STE_C4GOPEN_JNT_MASK
  FIELD SST_ARM_C4GOPEN_JNT_MASK
  ATTRIBUTES PRIVRW
  TITLE Joint arm mask
  DESC
  -- Each bit, in the INTEGER data, represents whether the
  -- corresponding axis for that arm is enabled for the
  -- C4GOPEN modality.
  DEND
  RELEASE 3.1
END

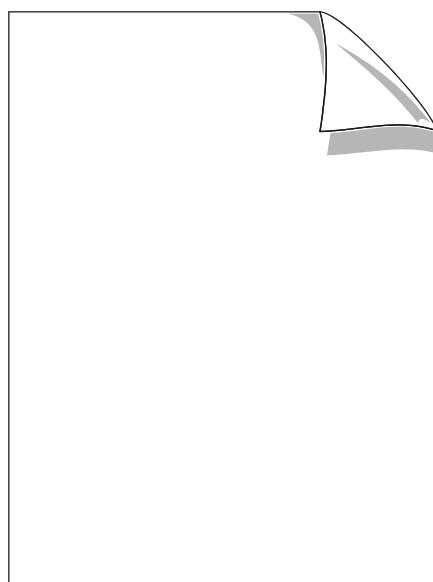
```

- modalità di C4GOPEN applicata ad ogni asse dell'ARM:

```
SS_ENTRY C4GOPEN_MODE
    GBL_AINT SSL_MARKER_NAX_ARM 0 GBZ_INT_VAL
    CAT_MAJOR SSL_ARM
    CAT_MINOR SSL_A_CNFG
    TITLE_MDF STE_C4GOPEN_MODE
    FIELD SST_ARM_C4GOPEN_MODE
    ATTRIBUTES PRIVRW
    TITLE C4GOpen modality
    DESC
    -- Each array element contains an integer which identifies
    -- the C4GOpen modality foreseen for that axis.
    DEND
    RELEASE 3.1
END
```
- bit di abilitazione della modalità C4G OPEN:

```
SSSW_ARY_HDR sx_a_along_1d; /* Arm LONG1 */
```
- opzione software C4GOPEN:

```
#define SSB_OPT_C4GOPEN      25 /* C4G OPEN */
#define SSK_OPT_C4GOPEN      (1 << SSB_OPT_C4GOPEN)
```



7. COMUNICAZIONE



Per controllare più facilmente la comunicazione che avviene sul C4G OPEN è necessario che, se si è configurato il sistema secondo le istruzioni presentate nel [Cap.6.](#), indipendentemente dal numero di assi interessati attivamente alla modalità aperta, il PC deve sempre essere attivo nella comunicazione rispondendo ai pacchetti mandati da SMP+. Questo vincolo continua a valere anche se alla partenza del sistema gli assi interessati dalla modalità aperta saranno dichiarati in "modalità 0" secondo quanto si dirà nel [par. 14.2 Modalità 0 a pag. 14-2](#).

7.1 Indirizzi

La comunicazione tra SMP+ e PC è basata sul protocollo UDP dove:

- il PC si comporta da server all'indirizzo IP 10.2.12.149 (in generale: IPADDRESS_{DSA} - 5),
 - SMP+ si comporta da client all'indirizzo IP 10.2.12.150 (in generale: IPADDRESS_{DSA} - 4).
- La comunicazione tra SMP+ e DSA è basata sul protocollo ETH-RAW dove:
- DSA si comporta da server all'indirizzo IP 10.2.12.154 (in generale: IPADDRESS_{DSA}).

7.2 Comunicazione stringente

Tra SMP+ e PC deve essere attiva una **comunicazione stringente real-time**. Con questo termine si vuole intendere una modalità di comunicazione che risponde ad alcune esigenze qui descritte.

Esigenze

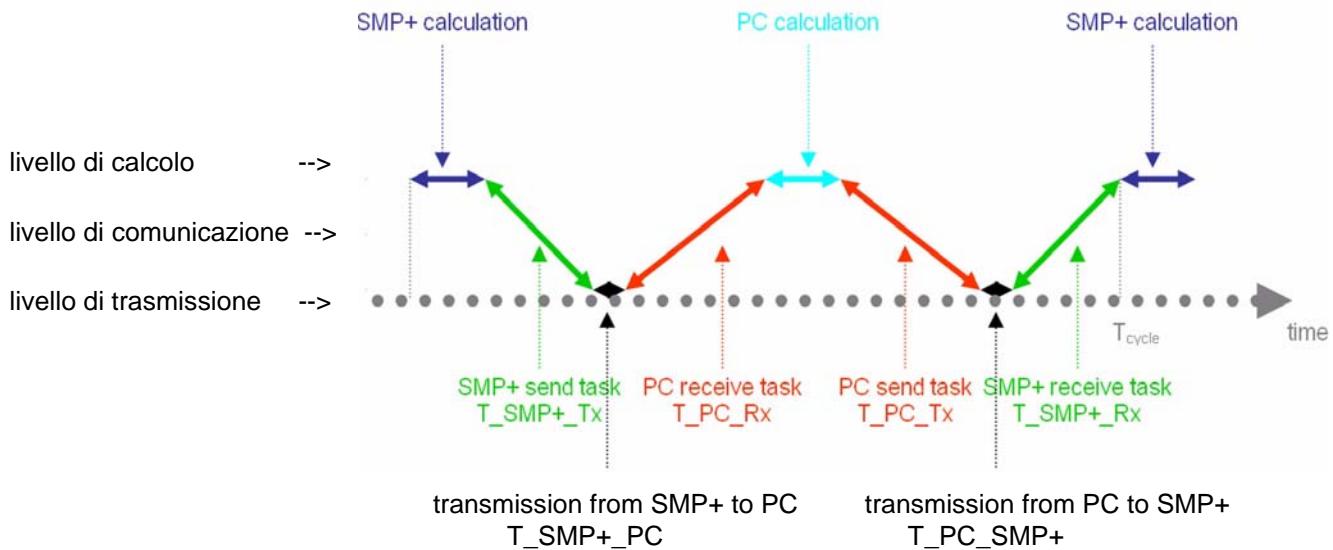
- Ad ogni pacchetto mandato da SMP+ il PC deve rispondere con un pacchetto di risposta, se questo non avviene si disabilita la prestazione di C4GOPEN con un opportuno messaggio d'errore e, a seconda dei casi, il C4G riprende il normale controllo della macchina o si verifica un errore software con priorità differente dipendente dallo stato della macchina,
- SMP+ associa un numero di protocollo al pacchetto (il primo campo del pacchetto come sarà spiegato nel [Cap.8. - Pacchetto di Comunicazione](#)) ed il PC deve rispondere con un pacchetto il cui primo campo contiene esattamente il numero di protocollo fornito da SMP+; se ciò non avviene, si incorre nella eccezione descritta al punto precedente).

Le due restrizioni permettono, da una parte, di far funzionare l'intero sistema in completa sicurezza (un fault viene individuato in un colpo di interpolatore!), dall'altra, permette di sincronizzare in maniera software i due attori della comunicazione.

In Fig. 7.1 viene riportato lo scheduling temporale dei vari task che girano su SMP+ e PC nell'ambito del C4G OPEN:

- **livello di calcolo:** processi che girano su SMP+ e PC e che eseguono gli algoritmi di calcolo (generazione della traiettoria, chiusura dei loop di controllo e così via),
- **livello di comunicazione:** processi che girano su SMP+ e PC (appartenenti ai sistemi operativi montati su SMP+ e PC) e che gestiscono la spedizione e la ricezione dei pacchetti,
- **livello di trasmissione:** segnali elettronici presenti sulla rete ETH.

Fig. 7.1 - Task che eseguono la comunicazione tra SMP+ e PC



È bene sottolineare che la comunicazione stringente real-time deve essere rispettata a tutti i livelli software/hardware!

In particolare, occorre che gli algoritmi di calcolo, i task di ricezione e spedizione dei pacchetti, l'interfaccia di rete, i cavi di collegamento e lo switch devono essere tali da rispettare le Eigenze sopra descritte, entro i tempi previsti. L'intervallo temporale massimo a disposizione del PC per rispondere alla richiesta proveniente da SMP+ verrà definito in dettaglio nel Cap.21. - Timing.

Fig. 7.2 - Scheduling reale dei processi che girano su SMP+ e PC simulato da un'altra SMP+



In Fig. 7.2 viene riportato lo scheduling dei processi misurato su un test operativo del C4G OPEN. Il sistema esaminato è costituito da due diverse schede SMP+, una che esegue il processo client proprio del C4G OPEN, un'altra che simula il processo server del C4G OPEN che deve girare sul PC remoto.

Il tempo base di comunicazione è 2ms, lo switch e la rete ETH utilizzati non sono diversi da quelli che saranno scelti per l'implementazione del sistema definitivo.

I processi che gestiscono la spedizione e la ricezione dei messaggi sia sul client che sul server sono chiamati NETT. Gli altri task che girano sul client sono principalmente quelli che si occupano della generazione della traiettoria. Il task OPEN sul server ha lo scopo di simulare il task del C4G OPEN che deve girare sul PC remoto.

Si nota che sul client girano due volte il task di ricezione e spedizione, questo è dovuto al fatto che il client deve comunicare anche con il DSA.

I primi due task NETT sono, rispettivamente, quello di spedizione del pacchetto da SMP+ a DSA e quello ricevuto da SMP+ e proveniente da DSA. È importante sottolineare che l'avvio della spedizione del pacchetto da SMP+ a DSA è scandito da un interrupt hardware deterministico, mentre, quello di ricezione dipende anche dalle operazioni eseguiti da DSA (non si discosta molto da un valore pari a 300us).

La finestra temporale che dal grafico superiore a quello inferiore si estende dal momento in cui termina l'esecuzione sul client del task di spedizione della richiesta al momento in cui termina l'esecuzione sul server del task di spedizione della risposta. Come si vede è inferiore a 100us, rispettando le [Esigenze](#) sopra descritte, nei tempi previsti dai vincoli

che sono riportati nel [Cap.21. - Timing](#).

7.3 Avvio della comunicazione

L'avvio della comunicazione avviene in 3 fasi distinte:

- **Fase 1** (avvio del PC):
 - a. il PC parte prima che sia ripartito il C4G,
 - b. il PC esegue un processo server aprendo un socket UDP,
 - c. il PC esegue un BIND alla porta destinata alla comunicazione C4GOPEN,
 - d. il PC si mette in ricezione di un pacchetto proveniente da SMP+,
- **Fase 2** (avvio del C4G):
 - e. il C4G riparte dopo che si è salvata la configurazione OPEN vista nel paragrafo 6,
 - f. il C4G apre un socket lato client all'indirizzo specificato in precedenza,
 - g. il C4G manda il primo pacchetto di comunicazione al PC,
- **Fase 3** (prima risposta del PC):
 - h. Il PC riceve il primo pacchetto spedito da SMP+ rispondendogli con un pacchetto il cui primo campo contiene il numero di protocollo comunicato da SMP+,
- **Fase 4** (avvio della normale comunicazione tra SMP+ e C4G):
 - i. se il C4G riceve nei tempi stabiliti il pacchetto di risposta dal PC i due interlocutori si sono sincronizzati e può iniziare la normale comunicazione.

7.4 Normale comunicazione

La normale comunicazione prevede che ad ogni pacchetto spedito da SMP+ il PC risponda con un pacchetto con numero di protocollo opportuno, in modo che i due dispositivi non perdano la sincronizzazione.

Gli altri campi del pacchetto sono descritti nel [Cap.8. - Pacchetto di Comunicazione](#).

7.5 Termine della comunicazione

La conclusione della comunicazione avviene dopo che si esegue una ripartenza (comando CCRC - ConfigureControllerRestartCold).

SMP+ comanda il riavvio del sistema per cui non richiede più la sincronizzazione con il PC al quale non manda più il pacchetto di comunicazione.

Il PC rileva la conclusione della comunicazione, quando non riceve più il pacchetto da SMP+.

È chiaro che se, prima del comando di ripartenza, non è stato riconfigurato il sistema (maschera degli assi per la modalità vista nel [Cap.6. - Configurazione iniziale del sistema](#)) il C4G riparte in modalità OPEN, per cui richiederà la presenza del PC per poter funzionare.

8. PACCHETTO DI COMUNICAZIONE

Di seguito sono riportati i campi costituenti i pacchetti di comunicazione che si scambiano SMP+ e PC nella modalità OPEN.

Il contenuto di alcuni campi può essere vuoto a seconda della [Modalità](#) di funzionamento; quello che deve essere rispettato è:

- l'ordine,
- il numero e
- la struttura dei vari campi.

Vengono ora trattati in dettaglio i seguenti argomenti:

- [Struttura dei pacchetti di comunicazione](#)
- [Virtualizzazione dei pacchetti di comunicazione](#)
- [Dimensione dei pacchetti di comunicazione.](#)

8.1 Struttura dei pacchetti di comunicazione

Tab. 8.1 - Struttura dei pacchetti di comunicazione

n.	tipo del campo	SMP+	PC
NUM	campo LONG per pacchetto	numero di protocollo	numero di protocollo
INFPAR	campo LONG per pacchetto	informazioni/comandi (*)	informazioni/comandi (**)
CMD	campo LONG per pacchetto	informazioni speciali (***)	informazioni speciali (****)
SMax	campo LONG per asse	modalità dell'asse	modalità dell'asse
D1ax	campo FLOAT per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo FLOAT per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo FLOAT per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo FLOAT per asse	velocità reale [delta giri motore]	contrib.in velocità[delta giri motore]
D5ax	campo FLOAT per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
EXT1ax	campo FLOAT per asse	primo campo extra per SMP+	primo campo extra per PC
EXT2ax	campo FLOAT per asse	secondo campo extra per SMP+	secondo campo extra per PC
EXT3ax	campo FLOAT per asse	terzo campo extra per SMP+	terzo campo extra per PC

(*) - segnale del DRIVE ON (C4G_STS_MOVING)

(**) - segnale di uscita dalla modalità (EXIT_FROM_OPEN)

(***) - funzionalità specifica in una data modalità

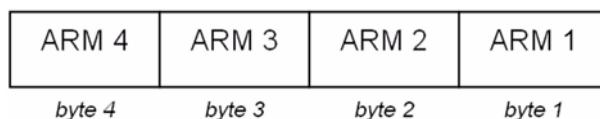
(****) - non ancora utilizzato

8.1.1 Campi dei pacchetti di comunicazione

Segue una descrizione dettagliata di ogni campo:

- per ogni pacchetto da spedire SMP+ calcola un numero di protocollo e lo inserisce nel primo campo del pacchetto (**NUM**). Come si è detto in precedenza, PC deve rispondere con un pacchetto il cui primo campo contenga il numero di protocollo mandato da SMP+ (range 0x0001- 0x7FFF).
- Il campo **INFPAR** contiene alcune informazioni che si scambiano SMP+ e PC. Grazie a questo campo SMP+ comunica a PC se è in DRIVE ON e PC comunica a SMP+ il comando di EXIT_FROM_OPEN.
Questo campo è unico per pacchetto ma fa riferimento ad una struttura che può contenere fino a quattro ARM distinti.
Per fare questo (vd. Fig. 8.1) il numero intero a 4 byte che lo rappresenta, viene suddiviso in 4 sottocampi ognuno dei quali si riferisce ad un ARM distinto e può assumere 256 valori distinti:

Fig. 8.1 - Figura 5 - Struttura del campo INFPAR



SMP+ comunica al PC il DRIVE ON sull'**ARM i** settando al valore C4G_STS_MOVING (5) il byte **i**, mentre il PC comunica a SMP+ il messaggio EXIT_FROM_OPEN sull'**ARM i** settando a C4G_EXIT_FROM_OPEN (0x01) il valore del byte **i**.

- Il campo **CMD** può contenere informazioni speciali che si scambiano SMP+ e il PC. Questo campo può essere utilizzato per comunicare al PC quale delle varie funzionalità predisposte sul sistema remoto deve essere effettivamente eseguita per una data modalità **OPEN**.

Ad esempio, questo campo risulta molto utile nel caso in cui il PC esterno abbia il compito di fornire la traiettoria al manipolatore eseguendo un percorso molto complesso.

Per portare a termine il compito, il percorso viene suddiviso in vari tratti, ognuno caratterizzato da un movimento diverso (il primo tratto da fare con movimenti sinusoidali, il secondo in lineare, il terzo in circolare e così via). La modalità **OPEN** resta sempre la stessa (la generazione del target da parte del PC esterno), mentre ciò che cambia è il tipo di movimento (sinusoidale, lineare, circolare, ...).

L'indicazione del tipo di movimento viene fornita utilizzando il campo **CMD**, ovvero, via PDL2 settando la variabile di sistema \$ARM_DATA[i_ARM].C4GOPEN_CMD[1] al valore numerico associato alla particolare funzionalità da realizzare.

Anche questo campo è unico per pacchetto ma fa riferimento ad una struttura che può contenere fino a quattro ARM distinti (similmente a quella vista in Figura 5).

- Il campo **SMax** deve contenere la modalità operativa (anche speciale) scelta per l'asse relativo, secondo quanto è spiegato nel [Cap.14. - Modalità](#).
- I campi **D1ax** e **D2ax** contengono i riferimenti in posizione e in velocità (rispettivamente, in giri motore e in delta giri motore) calcolati da SMP+ o PC a seconda della modalità dell'asse. Se la modalità non richiede al PC di generare il target, questi campi devono essere comunque presenti anche se contengono valori nulli.

- I campi **D3ax** e **D4ax** del pacchetto inviato da SMP+ contengono le misure in posizione in giri motore e in velocità in delta giri motore fornite dall'encoder. Indipendentemente dalla modalità questi campi assumono sempre valori significativi.
Le posizioni reali dei motori sono da intendersi filtrate attraverso le costanti di calibrazione per il passaggio da giri reali motore a giri ideali (lo "0" ideale del motore non potrà mai coincidere con lo "0" reale). In altri termini, il valore della posizione è conforme a quello registrato in un generico file di misure moni.log. Le costanti di calibrazione sono sempre disponibili da PDL2 o da interfaccia operatore sul controllo C4G.
Il campo **D3ax** nel pacchetto inviato da PC contiene, se lo richiede la modalità scelta, il valore della misura proveniente da PC. L'unità di misura dipende chiaramente dal tipo di misurazione effettuata. Possono esistere vari strumenti di misura quali accelerometri, encoder supplementari, strumenti di visione, sensori di forza/coppia, sensori di prossimità e così via. Ovviamente, poiché ogni misura richiede una differente procedura di signal processing, esisterà una modalità aperta per ogni tipologia di sensore.
- I campi **D4ax** e **D5ax** nel pacchetto fornito da PC contengono, se la modalità lo richiede, i valori calcolati, rispettivamente, dal loop di posizione (in delta giri motore) o dal loop di velocità (in Ampere) del controllo sviluppato sul PC. Anche in questo caso, se la modalità non richiede la consistenza di questi campi, essi devono essere comunque presenti anche se contengono valori nulli.
- Il campo **D5ax** inviato da SMP+ contiene la corrente di riferimento comprensiva del contributo del modello dinamico (in Ampere) imposta ai loop di corrente interni al DSP.
- I campi **EXT1ax**, **EXT2ax** e **EXT3ax** possono essere utilizzate per trasferire valori di grandezze extra da SMP+ a PC e viceversa.

8.2 Virtualizzazione dei pacchetti di comunicazione

Da un punto di vista concettuale, il pacchetto di comunicazione tra SMP+ e PC può essere virtualizzato con un record **C4GOpen_Packet** avente i seguenti campi:

Tab. 8.2 - Astrazione del pacchetto di comunicazione C4G OPEN

C4GOpen_Packet.NUM	numero di protocollo del pacchetto
C4GOpen_Packet.INFPAR	informazioni/parametri tra SMP+ e PC (per pacchetto)
C4GOpen_Packet.CMD	informazioni speciale tra SMP+ e PC
C4GOpen_Packet.AX(i).SM	modalità (per asse)
C4GOpen_Packet.AX(i).DATA	informazioni utili (per asse)
C4GOpen_Packet.AX(i).DATA.G1	sottocampo contenente la prima informazione utile (per asse)
C4GOpen_Packet.AX(i).DATA.G2	sottocampo contenente la seconda informazione utile (per asse)
C4GOpen_Packet.AX(i).DATA.G3	sottocampo contenente la terza informazione utile (per asse)
C4GOpen_Packet.AX(i).DATA.G4	sottocampo contenente la quarta informazione utile (per asse)
C4GOpen_Packet.AX(i).DATA.G5	sottocampo contenente la quinta informazione utile (per asse)
C4GOpen_Packet.AX(i).EXT	informazioni extra (per asse)
C4GOpen_Packet.AX(i).EXT.G1	sottocampo contenente la prima informazione extra (per asse)

Tab. 8.2 - Astrazione del pacchetto di comunicazione C4G OPEN

<i>C4GOpen_Packet.AX(i).EXT.G2</i>	sottocampo contenente la seconda informazione extra (per asse)
<i>C4GOpen_Packet.AX(i).EXT.G3</i>	sottocampo contenente la terza informazione extra (per asse)

Da un punto di vista implementativo:

```

typedef struct data_ax
{
    long SM; // modalità
    float D1; // dato utile 1
    float D2; // dato utile 2
    float D3; // dato utile 3
    float D4; // dato utile 4
    float D5; // dato utile 5
    float EXT1; // dato extra 1
    float EXT2; // dato extra 2
    float EXT3; // dato extra 3
} MVSW_DATA_AX;
#define MVZ_DATA_AX sizeof(MVSW_DATA_AX)
typedef struct data
{
    long NUM; // numero del pacchetto
    long INFPAR; // DRIVE ON e EXIT FROM OPEN
    long CMD; // informazioni speciali
    MVSW_DATA_AX AX[2*GBM_NAX_ARM]; // struttura dati per asse
} MVSW_DATA_PC;
#define MVZ_DATA_PC sizeof(MVSW_DATA_PC)

```

8.3 Dimensione dei pacchetti di comunicazione

Tab. 8.3 - Dimensione (in byte) dei pacchetti di comunicazione

n.	tipo del campo	Dimensione
HEADER	campo header del pacchetto UDP	42 byte per pacchetto
NUM	campo LONG per pacchetto	4 byte per pacchetto
INFPAR	campo LONG per pacchetto	4 byte per pacchetto
CMD	campo LONG per pacchetto	4 byte per pacchetto
SMax	campo LONG per asse	4 byte per asse
D1ax	campo FLOAT per asse	4 byte per asse
D2ax	campo FLOAT per asse	4 byte per asse
D3ax	campo FLOAT per asse	4 byte per asse
D4ax	campo FLOAT per asse	4 byte per asse
D5ax	campo FLOAT per asse	4 byte per asse
EXT1ax	campo FLOAT per asse	4 byte per asse
EXT2ax	campo FLOAT per asse	4 byte per asse
EXT3ax	campo FLOAT per asse	4 byte per asse
TOT	numero complessivo di byte per pacchetto	54 byte
	numero complessivo di byte per asse	36 byte

9. DAL BLIP/RAAZ AL PACCHETTO C4G OPEN

Come già detto ([Cap.6. - Configurazione iniziale del sistema](#)), la comunicazione tra SMP+ e DSA si basa sullo scambio dei messaggi **BLIP** e **RAAZ**: il primo viene creato da SMP+ e spedito a DSA, il secondo scritto dal DSA e ricevuto da SMP+.

Nel [Cap.8. - Pacchetto di Comunicazione](#), sono stati introdotti i campi dei pacchetti spediti e ricevuti da SMP+ e PC.

Nel presente capitolo si vede più in dettaglio come riescono a coesistere le due strutture in SMP+.

- [BLIP](#)
- [RAAZ](#)
- [Comunicazione tra SMP+, PC e DSA.](#)

9.1 BLIP

Tab. 9.1 - Struttura del BLIP

BLIP (BLocco Informativo Periodico): pacchetto di comando verso il DSA	
Descrizione (Intestazione)	
1	Codice identificativo del pacchetto (timeStamp)
1	DSA non ha ricevuto l'acknoledge precedente
Descrizione (messaggio Azionamento)	
1	Codice di Comando
1	Parametro di Comando
1	Codice del Servizio
5	Spazio di 5 word la cui strutturazione in interi e floating point dipende dal tipo del Servizio
Descrizione (messaggio Asse)	
1	Codice identificativo del comando
1	Opzione del comando
2	Posizione di riferimento corrente
2	Velocità di riferimento corrente
2	(DynFF) Guadagno del feedforward di velocità
2	(DynFF) Guadagno del feedforward di accelerazione
2	(DynFF) Guadagno del feedforward di jerk
2	Posizione obiettivo del movimento corrente
2	Contributo aggiuntivo all'anello di velocità
2	Contributo aggiuntivo all'anello di corrente
1	Limite di corrente erogata percentuale
1	Tipo del filtro di corrente utilizzato

Tab. 9.1 - Struttura del BLIP (Continua)

1	Codice della richiesta di servizio
5	Spazio di 5 word la cui strutturazione in interi e floating point dipende dal tipo della richiesta di servizio

9.2 RAAZ

Tab. 9.2 - Struttura del RAAZ

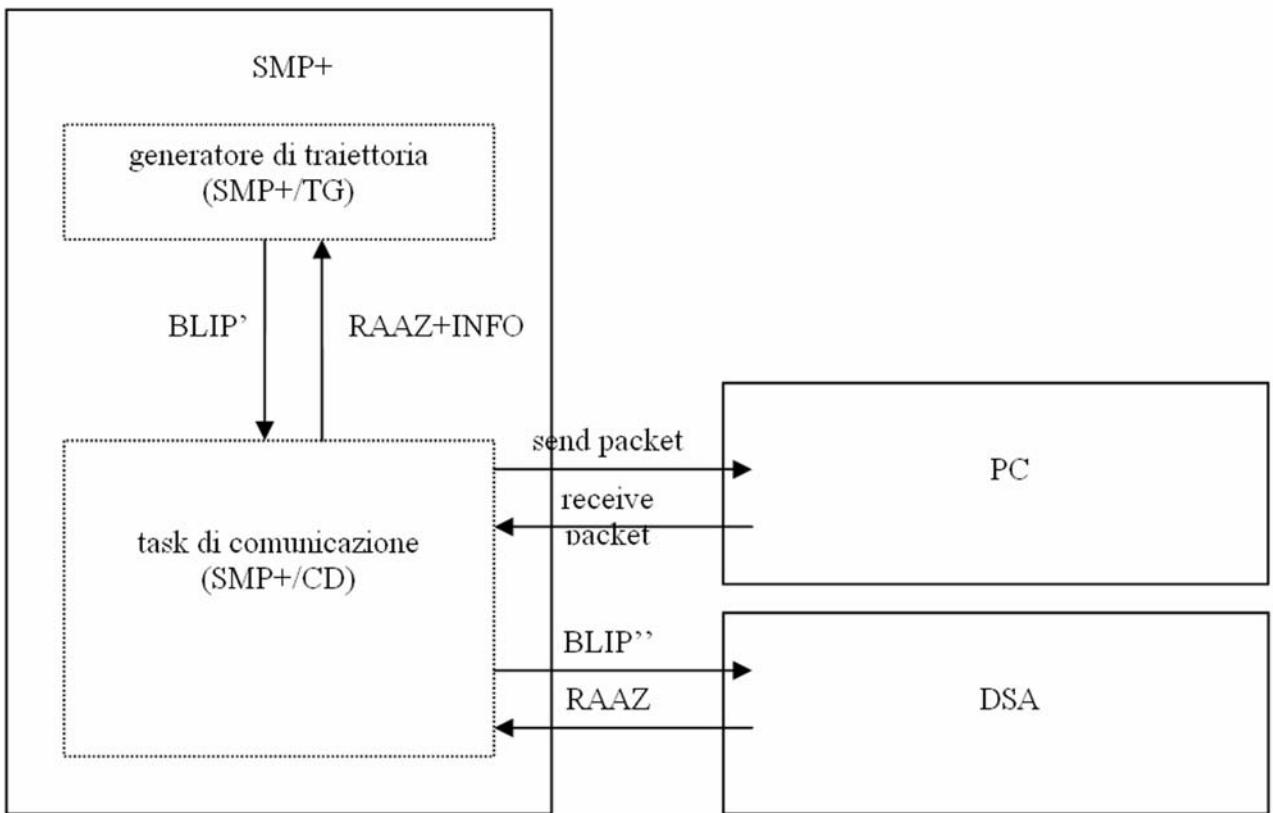
RAAZ (Risposta Automatica dell'Azionamento): pacchetto di risposta del DSA	
Descrizione (Intestazione)	
1	Codice identificativo del pacchetto (timeStamp)
1	DSA ha ricevuto il precedente blip
1	Codice dell'errore che si è verificato; il codice Zero è riservato per indicare "Nessun Errore"
1	Codice identificativo dell'oggetto che ha riconosciuto la condizione di errore
1	Identificatore posizionale del segmento di codice che ha riconosciuto l'errore
3	Tre word strutturate opportunamente per inviare parametri di errore
Descrizione (risposta Azionamento)	
1	Stato corrente di Azionamento
1	Parametro opzionale della risposta
2	Valore della grandezza configurabile
1	Codice identificativo della risposta
3	Spazio di tre word la cui strutturazione in interi e floating point dipende dal tipo della risposta
Descrizione (risposta Asse)	
1	Word di Stato dell'Asse
1	Comando di corrente (ingresso anello di corrente)
2	Posizione corrente di Asse
2	Velocità corrente di Asse
2	Errore corrente di inseguimento
2	Prima Grandezza configurabile
2	Seconda Grandezza configurabile
1	Codice della risposta al servizio
3	Spazio di tre word la cui strutturazione in interi e floating point dipende dal tipo della risposta al servizio

9.3 Comunicazione tra SMP+, PC e DSA

Nella seguente [Fig. 9.1](#) viene mostrato lo schema di comunicazione

- tra SMP+ e PC, basato sui pacchetti C4G Open, e
- tra SMP+ e DSA, basato sui pacchetti BLIP e RAAZ

Fig. 9.1 - Pacchetti C4G Open e BLIP e RAAZ



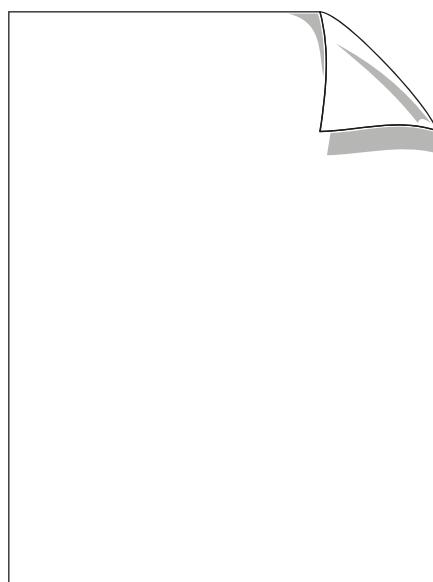
Il timing della comunicazione è il seguente (l'attore è il task di comunicazione con il PC che risiede su SMP+):

- ricezione del pacchetto da PC
- ricezione del BLIP' dal generatore di traiettoria (TG)
- spedizione del BLIP'' al DSA
- ricezione del RAAZ dal DSA
- spedizione del pacchetto a PC
- spedizione del RAAZ+INFO al generatore di traiettoria (TG)

In dettaglio, appena il generatore di traiettoria è pronto a spedire il BLIP al DSA, il task di comunicazione intercetta questo pacchetto e lo modifica in funzione delle informazioni contenute nei campi del pacchetto ricevuto dal PC.

Quando, poi, il DSA fornisce a SMP+ il RAAZ, anche questo pacchetto viene intercettato dal task di comunicazione e viene passato al PC insieme con alcune informazioni raccolte dal BLIP che il generatore di traiettoria ha fornito al passo precedente.

Al generatore di traiettoria viene restituito lo stesso RAAZ, ma questo pacchetto viene accompagnato da altre informazioni che i livelli software più elevati di SMP+ hanno bisogno di conoscere.



10. PACCHETTO DI INIZIALIZZAZIONE

All'inizio della comunicazione SMP+ manda un pacchetto contenente una serie di informazioni per permettere a PC di interpretare correttamente la struttura de [Pacchetto di Comunicazione](#).

Questa fase avviene durante la procedura di inizializzazione della comunicazione; successivamente avviene la fase di sincronizzazione tra SMP+ e PC.

Le informazioni che costituiscono il pacchetto di inizializzazione sono le seguenti:

- **1° elemento (long):** numero di protocollo del pacchetto;
- **2° elemento (long):** numero di campi per pacchetto (i): secondo quanto riportato in Tabella 2, il valore è sempre 3;
- **3° elemento (long):** numero di campi per asse (j) : secondo quanto riportato in [Tab. 8.1](#), il valore è sempre 9;
- **4° elemento (long):** numero di assi (k) interessati dalla modalità aperta (massimo 20: 4 ARM con 2 DSA) (nell'esempio in [Fig. 10.1](#) il valore è 6);
- **4 vettori di 10 elementi (long):** modalità aperta particolare impostata al riavvio i vari assi (nell'esempio in [Fig. 10.1](#) tutti gli assi hanno modalità 0 alla ripartenza);
- **4 vettori di 10 elementi (long):** elenco degli indici degli assi per i quali è prevista la modalità aperta secondo la mappatura tra indici di assi OPEN e indici di assi logici (nell'esempio in [Fig. 10.1](#) c'è corrispondenza diretta tra assi OPEN e assi logici);
- **4 vettori di 10 elementi (float):** valori delle costanti di calibrazione dei vari assi;
- **4 vettori di 10 elementi (float):** valori massimi delle correnti di riferimento assorbibili dai vari assi;
- **4 vettori di 10 elementi (float):** rapporti di trasmissione dei vari assi;
- **4 vettori di 10 elementi (float):** coefficienti di influenza cinematica dei vari assi;
- **4 vettori di 10 elementi (float):** soglie di errore di inseguimento dai vari assi;
- **successivi i elementi (long):** significato dei campi per pacchetto;
- **successivi j elementi (long):** significato dei campi per asse;
- **successivo elemento (long):** periodo di campionamento della SMP+ (in millisecondi);
- **successivo elemento (long):** versione software (major version);
- **successivo elemento (long):** versione software (minor version);
- **successivo elemento (long):** versione software (build number).

Di seguito è riportata la dichiarazione del pacchetto di inizializzazione:

```

typedef struct syncPck:
{
  long nblip;// numero protocollo
  long nFieldPck;// numero campi per pacchetto
  long nFieldAx;// numero campi per asse
  long nAxOpen;// numero assi in modalità aperta
  long arm1OpenMask[GBM_NAX_ARM]; // modalità specifica assi (ARM1)
  long arm2OpenMask[GBM_NAX_ARM]; // modalità specifica assi (ARM2)
  long arm3OpenMask[GBM_NAX_ARM]; // modalità specifica assi (ARM3)
  long arm4OpenMask[GBM_NAX_ARM]; // modalità specifica assi (ARM4)
  long num;// tipo del 1° campo per pacchetto
  long infPar;// tipo del 2° campo per pacchetto
  long cmd;// tipo del 3° campo per pacchetto
  long sMax;// tipo del 1° campo per asse
  long d1Ax;// tipo del 2° campo per asse
  long d2Ax;// tipo del 3° campo per asse
  long d3Ax;// tipo del 4° campo per asse
  long d4Ax;// tipo del 5° campo per asse
  long d5Ax;// tipo del 6° campo per asse
  long ext1Ax;// tipo del 7° campo per asse
  long ext2Ax;// tipo del 8° campo per asse
  long ext3Ax;// tipo del 9° campo per asse
  long period;// periodo di campionamento [msec]
  long arm1OpenAx[GBM_NAX_ARM]; // maschera assi OPEN (ARM1)
  long arm2OpenAx[GBM_NAX_ARM]; // maschera assi OPEN (ARM2)
  long arm3OpenAx[GBM_NAX_ARM]; // maschera assi OPEN (ARM3)
  long arm4OpenAx[GBM_NAX_ARM]; // maschera assi OPEN (ARM4)
  float arm1CalData[GBM_NAX_ARM]; // costanti di calibrazione (ARM1) [giri]
  float arm2CalData[GBM_NAX_ARM]; // costanti di calibrazione (ARM2) [giri]
  float arm3CalData[GBM_NAX_ARM]; // costanti di calibrazione (ARM3) [giri]
  float arm4CalData[GBM_NAX_ARM];// costanti di calibrazione (ARM4) [giri]
  float arm1AzLim[GBM_NAX_ARM];// correnti massime (ARM1) [A]
  float arm2AzLim[GBM_NAX_ARM];// correnti massime (ARM2) [A]
  float arm3AzLim[GBM_NAX_ARM];// correnti massime (ARM3) [A]
  float arm4AzLim[GBM_NAX_ARM];// correnti massime (ARM4) [A]
  long majorVersion;// major version del sistema operativo
  long minorVersion;// minor version del sistema operativo
  long buildNum;// build number del sistema operativo
  float arm1AxInfo[GBM_NAX_ARM];// coefficienti di influenza(ARM1) [--]
  float arm2AxInfo[GBM_NAX_ARM];// coefficienti di influenza(ARM2) [--]
  float arm3AxInfo[GBM_NAX_ARM];// coefficienti di influenza(ARM3) [--]
  float arm4AxInfo[GBM_NAX_ARM];// coefficienti di influenza(ARM4) [--]
  float arm1TxRate[GBM_NAX_ARM];// rapporti di trasmissione (ARM1) [--]
  float arm2TxRate[GBM_NAX_ARM];// rapporti di trasmissione (ARM2) [--]
  float arm3TxRate[GBM_NAX_ARM];// rapporti di trasmissione (ARM3) [--]
  float arm4TxRate[GBM_NAX_ARM];// rapporti di trasmissione (ARM4) [--]
  float arm1FollowError[GBM_NAX_ARM];// soglie di following error(ARM1) [--]
  float arm2FollowError[GBM_NAX_ARM];// soglie di following error(ARM2) [--]
  float arm3FollowError[GBM_NAX_ARM];// soglie di following error(ARM3) [--]
  float arm4FollowError[GBM_NAX_ARM];// soglie di following error(ARM4) [--]
} MVSW_SYNC_PCK;
#define MVZ_SYNC_PCK sizeof(MVSW_SYNC_PCK)

```

Fig. 10.1 - Il pacchetto di inizializzazione ricevuto dal PC

```

c4gopen@localhost:~/Desktop/C4Gopen_Linux - Shell - Konsole
Sessione Modifica Visualizza Segnalibri Impostazioni Aiuto
*****
*          C4G Open          *
*      Versione Server: 0.04      *
*****


Numero protocollo: 10
Numero campi per pacchetto: 3
Numero campi per asse: 9
Numero assi in modalit.aperta: 6
Modalita' assi ARM1:
  0 0 0 0 0 0 0 0 0
Modalita' assi ARM2:
  0 0 0 0 0 0 0 0 0
Modalita' assi ARM3:
  0 0 0 0 0 0 0 0 0
Modalita' assi ARM4:
  0 0 0 0 0 0 0 0 0
Indici assi open ARM1 (logici convenzioni noC):
  1 2 3 4 5 6 X X X
Indici assi open ARM2 (logici convenzioni noC):
  X X X X X X X X X
Indici assi open ARM3 (logici convenzioni noC):
  X X X X X X X X X
Indici assi open ARM4 (logici convenzioni noC):
  X X X X X X X X X
Costanti calibrazione ARM1:
  0.142965 0.170353 0.222660 0.405883 0.039380 0.355824 0.000000 0.000000 0.000000
Costanti calibrazione ARM2:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Costanti calibrazione ARM3:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Costanti calibrazione ARM4:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori di saturazione di corrente ARM1:
  24.500000 24.500000 24.500000 9.500000 9.500000 9.500000 0.000000 0.000000 0.000000
Valori di saturazione di corrente ARM2:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori di saturazione di corrente ARM3:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori di saturazione di corrente ARM4:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei rapporti di trasmissione ARM1:
  -181.587296 159.260864 162.473694 72.585968 78.750000 -50.000000 0.000000 0.000000 0.000000
Valori dei rapporti di trasmissione ARM2:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei rapporti di trasmissione ARM3:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei rapporti di trasmissione ARM4:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei coefficienti di influenza ARM1:
  0.000000 0.000000 -0.020000 0.020400 0.020000 0.000000 0.000000 0.000000 0.000000
Valori dei coefficienti di influenza ARM2:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei coefficienti di influenza ARM3:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori dei coefficienti di influenza ARM4:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori del Following Error ARM1:
  7.500000 7.500000 7.500000 7.500000 7.500000 0.000000 0.000000 0.000000 0.000000
Valori del Following Error ARM2:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori del Following Error ARM3:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
Valori del Following Error ARM4:
  0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
NUM: 1
INFPAR: 2
CMD: 3
SMax: 1
D1ax: 2
D2ax: 2
D3ax: 2
D4ax: 2
D4ax: 2
EXT1ax: 3
EXT2ax: 3
EXT3ax: 3
Ts: 2 [msec]
Software Version: 3.11.115

```

Per specificare la particolare modalità aperta, per ogni asse per cui è prevista, si utilizzano 4 vettori di 10 elementi ognuno.

Ogni vettore si riferisce ad un ARM (ordinati dal primo al quarto) e gli elementi del vettore si riferiscono ai 10 assi logici (ordinati dal primo al decimo) che possono essere presenti su di un ARM.

Se l' i -simo elemento del j -simo vettore assume valore pari a k , significa che per l'asse logico i sull'ARM j è prevista la modalità aperta di tipo k . In particolare, esistono tre tipologie di valori per k di seguito elencate:

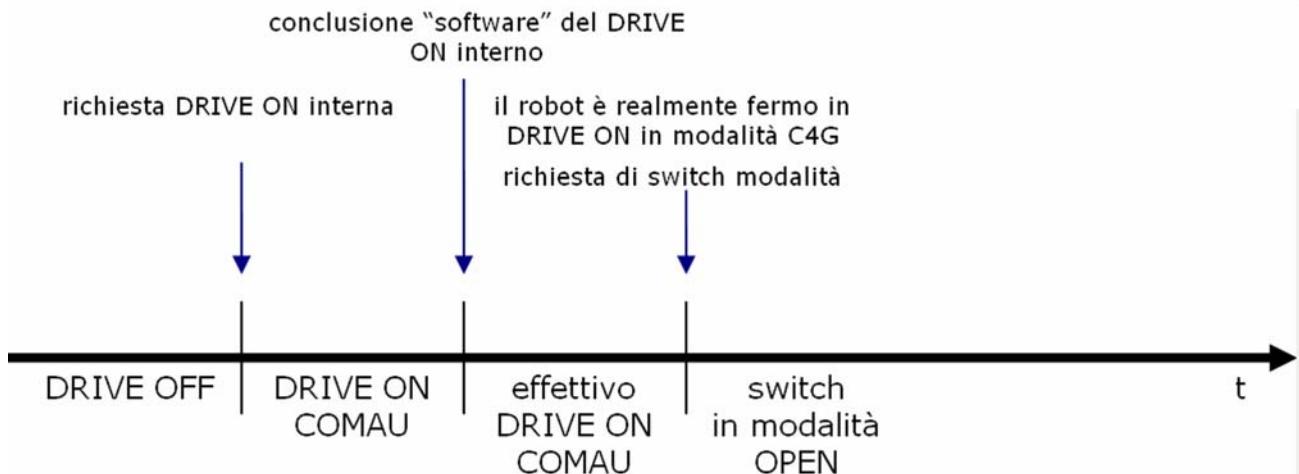
- $k=0$ --> per l'asse non è prevista la modalità OPEN,
- $k=FF$ --> per l'asse è prevista la modalità OPEN non specificata alla ripartenza del sistema,
- $0 < k < FF$ ' per l'asse è prevista la modalità k alla ripartenza del sistema (potendo comunque cambiare modalità successivamente tra un movimento ed un altro).

La sequenza degli assi con valore $k>0$ determina la relazione tra indici logici (per ARM) e indici OPEN discussa nel [Cap.6. - Configurazione iniziale del sistema](#).

11. PROCEDURA DI DRIVE ON

Fondamentale importanza riveste la procedura di DRIVE ON della macchina il cui timing viene riportato in Fig. 11.1.

Fig. 11.1 - Timing del DRIVE ON su C4G OPEN



Inizialmente il sistema parte in DRIVE OFF. Nel momento in cui viene richiesto il DRIVE ON, la procedura di attivazione si divide in due fasi.

- Prima fase - normale procedura di DRIVE ON mediante sistema C4G: richiesta di DRIVE ON, comandi e controlli software di DRIVE ON, attivazione della corrente ai motori ed, infine, al termine del transitorio della macchina, il robot è fermo in DRIVE ON controllato dai riferimenti C4G (SMP+) e dall'azionamento C4G (DSA). Durante questa fase la modalità dell'asse deve essere di tipo 0, il sistema ignora le eventuali richieste di cambiamento della modalità provenienti dal PC o da PDL2.
- Seconda fase - con l'asse abilitato è possibile commutare in una qualsiasi delle tipologie di controllore aperto. È cura del software del PC esterno garantire che, contemporaneamente al tipo di modalità, il buffer di comunicazione contenga le informazioni idonee per realizzare tale modalità (per esempio, se si prevede che il PC abbia il pieno controllo dei loop di posizione e velocità, già il primo pacchetto di comunicazione relativo alla modalità specifica deve contenere i valori di corrente opportuni).

La filosofia che sta alla base del C4G OPEN è la flessibilità, che si traduce nella possibilità di cambiare dinamicamente la modalità operativa specifica (controllo su PC o su SMP+). Deve essere, perciò, garantita la stabilità del servo-mecanismo durante le transizioni da un tipo di funzionamento all'altro. Per questo motivo, i progetti di controllo dei robot sui quali è possibile l'opzione software C4G OPEN, dovranno rispettare i seguenti requisiti:

- loop di posizione e velocità "semplici" (numero minimo di stati del controllo),
- disabilitazione del controllo in avanti ($ffv=ffa=ffj=0$),
- disabilitazione dinamica del loop di posizione ($Kp=0$),
- disabilitazione dinamica del loop di velocità ($VelGain=0$),

- disabilitazione del loop di coppia ($TcGain=0$),
- disabilitazione del modello dinamico ($DynGain=0$),
- abilitazione dei sistemi di anti-windup sia sul loop di posizione che su quello di velocità.

È bene sottolineare che, durante il DRIVE OFF e unicamente durante questa fase, può essere selezionato l'insieme degli assi abilitati al DRIVE ON.

Il comando che permette di attivare questa modalità di funzionamento è chiamato DRIVE ON DISABLE (si utilizza la funzione DRIVEON_DSBL).

Questa prestazione può essere utile nelle primissime fasi della sperimentazione per concentrarsi solo su un numero ristretto di assi, senza interagire con tutto il robot lasciando fermi gli assi non di interesse.

12. SOFT CONTROL SWITCHING

Una delle tecnologie sviluppate ad hoc per il sistema C4G OPEN è quella che va sotto il nome di **Soft Control Switching (SCS)**.

Con questo termine ci si riferisce all'insieme degli algoritmi che permettono di commutare gradualmente da un sistema di controllo standard COMAU a un sistema di controllo comandato da PC esterno.

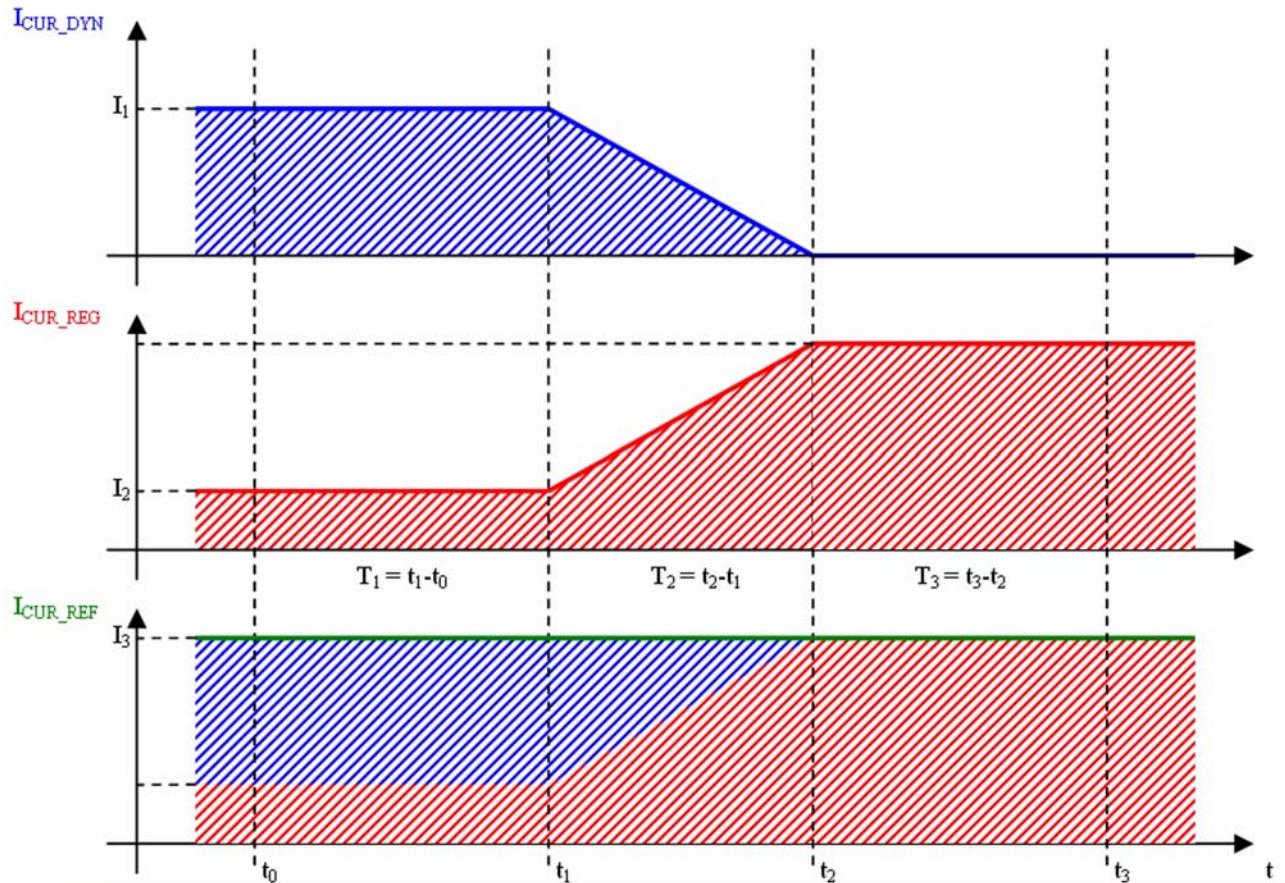
Il primo problema da risolvere è quello di utilizzare il canale del modello dinamico per fornire a DSA le correnti calcolate da PC.

Per fare questo, allora, occorre disabilitare al DRIVE ON il modello dinamico, per poter disporre liberamente di quel canale.

Per realizzare questa procedura si utilizza un approccio SCS come viene evidenziato nella [Fig. 12.1](#), dove:

- I_{CUR_DYN} è la corrente calcolata dal modello dinamico,
- I_{CUR_REG} è la corrente calcolata dal regolatore in contro-reazione,
- I_{CUR_REF} è la corrente di riferimento per i loop di corrente: $I_{CUR_REF} = I_{CUR_DYN} + I_{CUR_REG}$,
- l'intervallo di tempo **T1** è detto **pre-switching** e richiede che il robot sia in DRIVE ON e fermo, in modo che tutti i transitori si esauriscano,
- l'intervallo di tempo **T2** è la fase di **switching** vera e propria e anch'essa richiede che il robot sia in DRIVE ON e fermo, per evitare di innescare dinamiche non controllate,
- l'intervallo di tempo **T3** è detto **post-switching** e anch'essa come le precedenti richiede che il robot sia in DRIVE ON e fermo, in maniera da esaurire anche i transitori dovuti allo switching,
- le correnti **I1** e **I2** sono le correnti di riferimento totale calcolate, rispettivamente, dal modello dinamico e dai loop in contro-reazione.
- la corrente **I3** è la corrente di riferimento che, alla fine della fase di switching, è a completo carico del regolatore in contro-reazione, in quanto l'azione del modello dinamico è stata completamente annullata.

Fig. 12.1 - Soft Control Switching: disabilitazione dinamica del modello dinamico



I valori di **T1** e **T3** devono essere scelti in base ai tempi di reazione caratteristici del robot, mentre il valore di **T2** dipende anche dal modello di switching scelto.

Un modello semplice è quello lineare:

$$\begin{cases} \chi(t) = \frac{t - t_1}{t_2 - t_1} & \chi(t) \in [0,1] \\ t \in [t_1, t_2] & \eta(t) = \chi(t) & \eta(t) \in [0,1] \\ & \rho(t) = 1 - \chi(t) & \rho(t) \in [0,1] \end{cases}$$

Utilizzando questo modello, nella fase di switching il contributo del modello dinamico potrà essere modulato mediante un amplificatore a guadagno variabile:

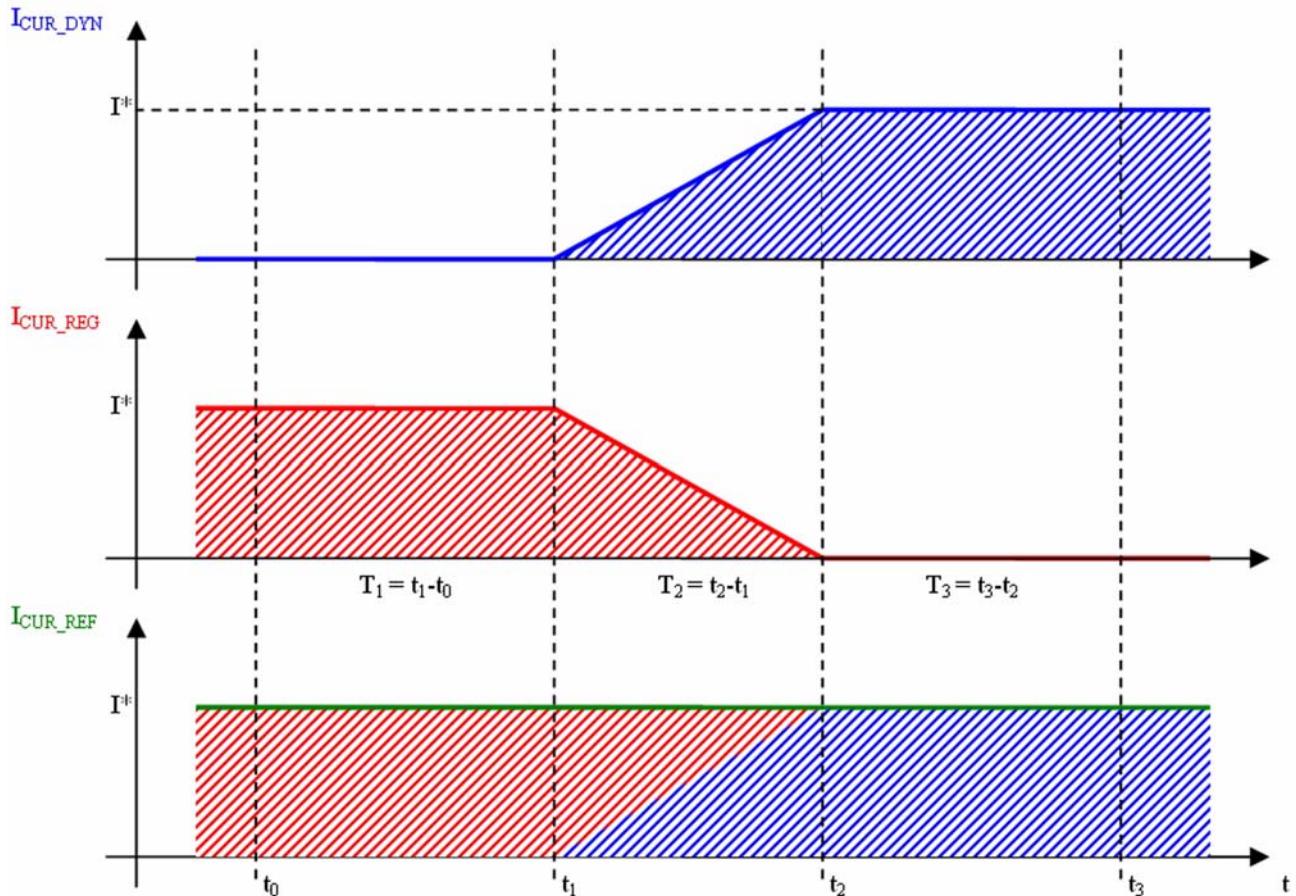
$$t \in [t_1, t_2] \quad I_{CUR_DYN}(t) = \rho(t) * I_{CUR_DYN}(t)$$

mentre la corrente ICUR_REG si adatterà di conseguenza, in quanto è frutto di algoritmi di calcolo in feedback.

La tecnologia SCS viene utilizzata anche per commutare il funzionamento di C4G OPEN da una modalità operativa che vede il pieno controllo dei loop in contro-reazione da parte di SMP+, a quella in cui il PC esterno comanda la corrente di riferimento.

Questa, una volta calcolata dal PC, viene passata a SMP+ che si fa carico di fornirla a DSA utilizzando il canale del modello dinamico.

Fig. 12.2 - Soft Control Switching: commutazione di sistemi di controllo in contro-reazione



Secondo il principio mostrato nella Fig. 12.2, nella fase di switching il contributo del modello dinamico potrà essere modulato mediante l'uso di un riduttore a guadagno variabile:

$$t \in [t_1, t_2] \quad I_{CUR_DYN}(t) = \eta(t) * I_{CUR_DYN}(t)$$

mentre il controllo su DSA potrà essere limitato mediante l'uso di un saturatore a soglia variabile:

$$t \in [t_1, t_2] \quad I_{CUR_REG}(t) = \begin{cases} |I_{CUR_REG}(t)| \leq \rho(t)(I_{\max} - I_{\min}) & I_{CUR_REG}(t) \\ |I_{CUR_REG}(t)| > \rho(t)(I_{\max} - I_{\min}) & \rho(t)(I_{\max} - I_{\min}) \operatorname{sgn}(I_{CUR_REG}(t)) \end{cases}$$

dove:

- I_{\max} è la massima corrente fornibile al sistema elettromeccanico,
- I_{\min} è la minima corrente erogabile ($I_{\min}=0$).

Per eseguire la commutazione inversa, da controllo completo dal PC a controllo completo dai regolatori implementati in DSA, basta fornire, nella fase di switching, i

seguenti andamenti delle correnti:

$$t \in [t_1, t_2] \quad I_{CUR_DYN}(t) = \rho(t)^* I_{CUR_DYN}(t)$$

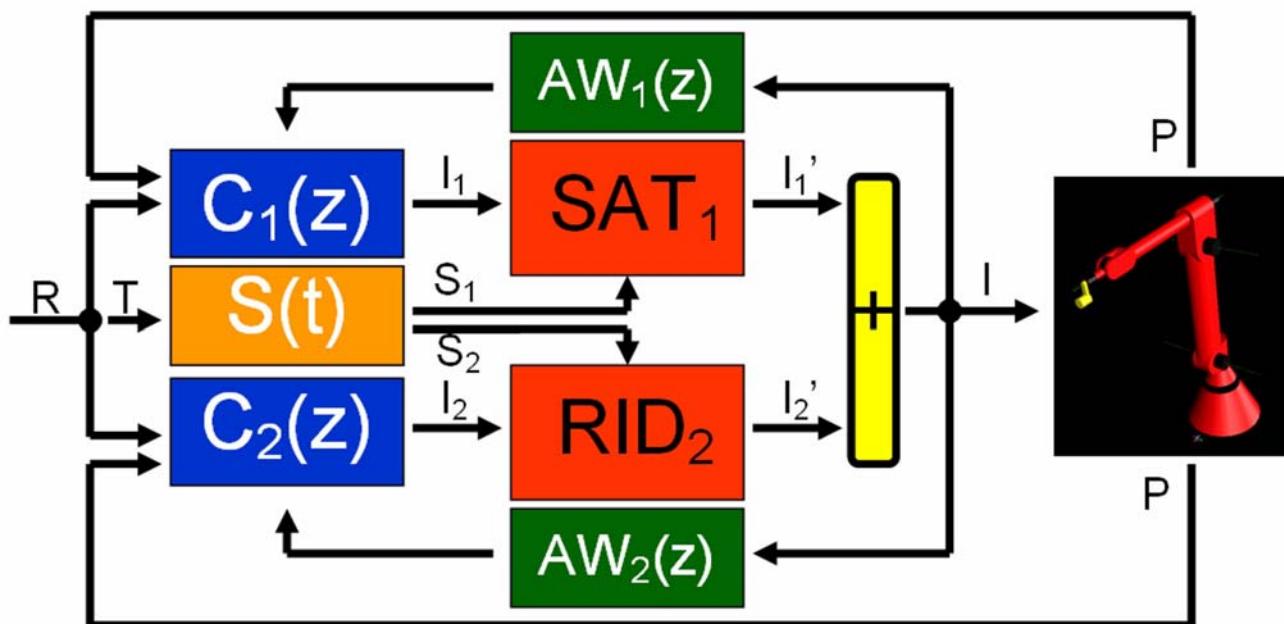
$$t \in [t_1, t_2] \quad I_{CUR_REG}(t) = \begin{cases} |I_{CUR_REG}(t)| \leq \eta(t)(I_{\max} - I_{\min}) & I_{CUR_REG}(t) \\ |I_{CUR_REG}(t)| > \eta(t)(I_{\max} - I_{\min}) & \eta(t)(I_{\max} - I_{\min}) \operatorname{sgn}(I_{CUR_REG}(t)) \end{cases}$$

Risulta chiaro che la tecnologia SCS si basa sull'utilizzo di un riduttore e di un saturatore che limitano le correnti, sia del modello dinamico sia del regolatore in contro-reazione.



Per evitare bruschi comportamenti della macchina, ogni sistema di controllo deve essere accompagnato da un sistema di anti-windup.
Il controllo C4G dispone già di sistemi anti-windup, mentre deve essere considerato come requisito vincolante quello di realizzare, sul PC esterno, loop in contro-reazione anch'essi dotati di anti-windup.

Fig. 12.3 - Soft Control Switching: schema a blocchi



In Fig. 12.3 viene riportato lo schema a blocchi che permette di realizzare la SCS, dove:

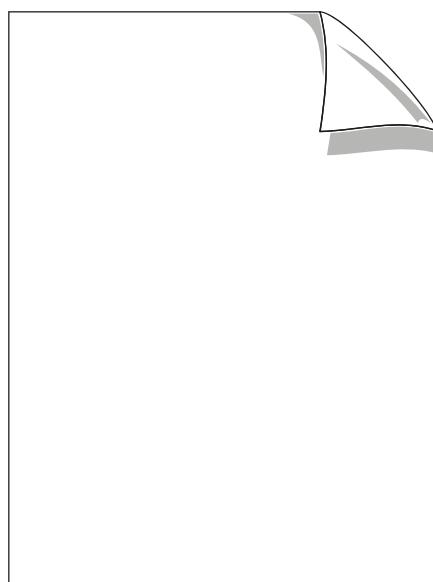
- $C_1(z)$ e $C_2(z)$ si riferiscono ai sistemi di controllo, rispettivamente, dei regolatori interni a DSA e di quelli realizzati su PC,
- SAT_1 si riferisce ai saturatori a soglia variabile dei regolatori interni a DSA,
- RID_2 si riferisce ai riduttori a guadagno variabile dei regolatori realizzati su PC,
- $AW_1(z)$ e $AW_2(z)$ si riferiscono ai sistemi di anti-windup, rispettivamente, per i regolatori interni a DSA e per quelli realizzati su PC,
- $I_1=I_1(t)$ e $I_2=I_2(t)$ si riferiscono alle correnti calcolate, rispettivamente, dai regolatori interni a DSA e da quelli realizzati su PC,
- $I_1'=I_1'(t)$ e $I_2'=I_2'(t)$ si riferiscono alle correnti a valle dei saturatori, rispettivamente, dei regolatori interni a DSA e di quelli realizzati su PC,
- $R=R(t)$ sono i riferimenti in posizione da far realizzare ad ogni giunto del robot,

- $P=P(t)$ sono le posizioni dei giunti misurate dagli encoder,
- $I=I(t)=I_1'(t)+I_2'(t)$ sono le correnti di riferimento per i loop di corrente,
- $S(t)$ è il termine $\chi(t)$ calcolato da SCS,
- $S_1=S_1(t)$ è il termine $\rho(t)$ calcolato da SCS,
- $S_2=S_2(t)$ è il termine $\eta(t)$ calcolato da SCS,
- $T=T(t)$ è il timing interno a SCS.



Per poter sfruttare a pieno la tecnologia SCS, bisogna rispettare alcune regole per la scrittura del programma PDL2 che comanda il controllo C4G OPEN:

- la prima istruzione di movimento deve essere preceduta da un'istruzione **DELAY** (cioè di ritardo) per un tempo di $T=T_1+T_2+T_3$ secondi per consentire la disabilitazione dinamica del modello dinamico,
- ogni istruzione che richiede un cambio di modalità operativa dal controllo gestito da DSA al controllo comandato da PC, deve essere preceduta e seguita da un'istruzione **DELAY** (cioè di ritardo) per un tempo di $T=T_1+T_2+T_3$ per consentire l'esaurimento dei transitori di cui sopra.



13. GESTIONE DINAMICA DELLA MODALITÀ APERTA



Una delle caratteristiche più importanti del C4G OPEN è la possibilità di cambiare dinamicamente la modalità operativa, senza richiedere la procedura di DRIVE OFF.

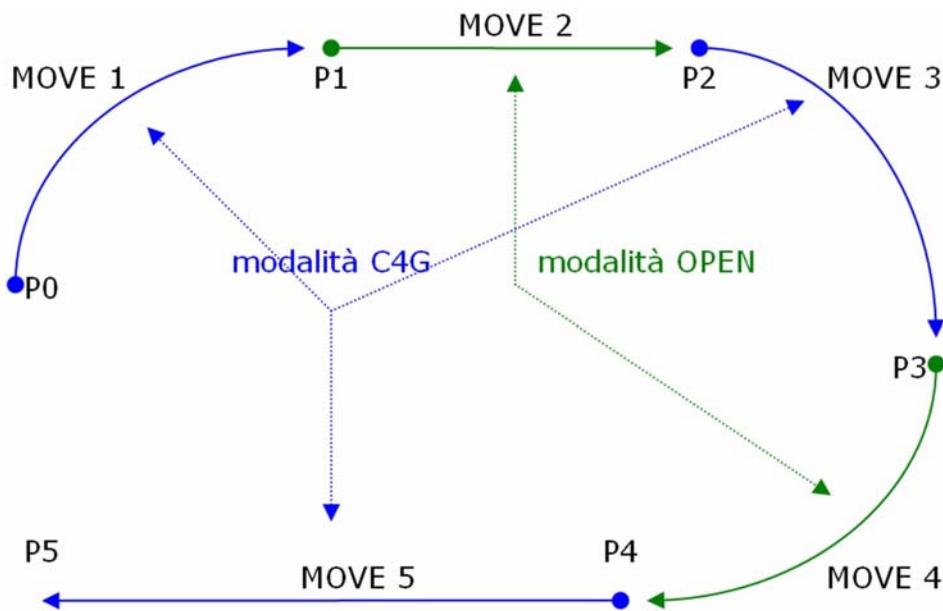
L'unico vincolo da rispettare è che il cambiamento di modalità operativa sia effettuato unicamente tra una MOVE e la successiva.

La realizzazione dell'opzione "Open" del controllo industriale COMAU C4G, permette di creare applicazioni in cui i processi di controllo e generazione di traiettoria standard (SMP+ e MPC), interagiscano in modo estremamente flessibile con funzioni addizionali di controllo e generazione della traiettoria programmate e realizzate esternamente su PC.

Sono previste diverse modalità di intervento del processo esterno sul controllo del manipolatore; segue un elenco di alcune tra modalità disponibili:

- controllo di posizione additivo (correzioni);
- controllo di posizione;
- controllo additivo di corrente;
- gestione della traiettoria;
- modifica della traiettoria programmata da C4G.

Un esempio di traiettoria mista, che evidenzia la possibilità di alternare e di affiancare le azioni di controllo tra C4G e PC esterno, è mostrato in [Fig. 13.1](#).

Fig. 13.1 - Movimenti in modalità di controllo differenti

Un possibile programma di movimento in PDL2, che realizza il path di Fig. 13.1, potrebbe essere il seguente:

```

MOVE TO P1
GO OPEN TYPE A
WAIT FOR PC
MOVE TO P3
GO OPEN TYPE B
MOVE TO P4
WAIT FOR PC
MOVE TO P5

```

Inizialmente il robot si trova nel punto P0 e, mediante l'istruzione "MOVE TO P1" esegue il movimento verso P1 in modalità normale, cioè con il pieno controllo di C4G.

Raggiunto P1, si esegue l'istruzione "GO OPEN TYPE A" che fa entrare il sistema in modalità aperta: SMP+ comunica a PC questo comando attraverso l'opportuno settaggio del campo, nel pacchetto di comunicazione (vd. Cap.8. - Pacchetto di Comunicazione).

Il PC esegue le operazioni in modalità aperta, riempiendo opportunamente i campi del pacchetto che invia a SMP+, in funzione delle modalità specificate nel Cap.14. - Modalità (ad esempio, fornendo i riferimenti di posizione ai motori, per spostare il robot fino al punto P2). Intanto, mediante l'istruzione "WAIT FOR PC", l'esecuzione del programma PDL2 viene interrotta.

Il programma PDL2 riprende solo quando il PC, terminate le sue operazioni in modalità aperta, riporta il sistema in modalità normale, fornendo a SMP+ il segnale "EXIT_FROM_OPEN" con un opportuno settaggio del campo INFPAR nel pacchetto di comunicazione.

Il programma PDL2 continua eseguendo l'istruzione "MOVE TO P3" in modalità normale.

Quando il robot raggiunge P3, il sistema entra nuovamente in modalità aperta con l'istruzione "GO OPEN TYPE B".

Il programma PDL2 non si arresta ma esegue l'istruzione "MOVE TO P4" (ad esempio, ricevendo dal PC i riferimenti in corrente per i motori) e successivamente attende il segnale da PC per ritornare in modalità normale.

Infine, in modalità normale, viene spostato il robot sul punto finale P5.



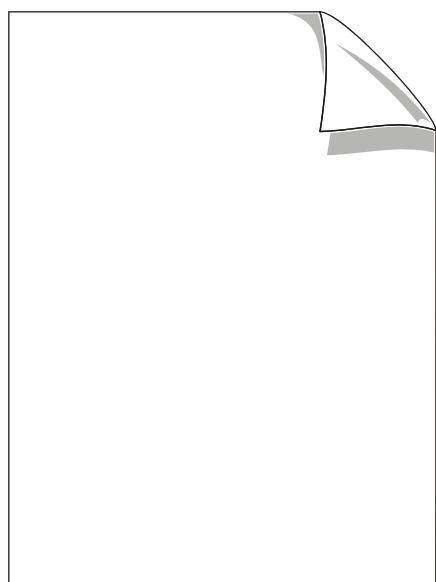
Si noti la sostanziale differenza tra l'istruzione "GO OPEN TYPE A" e l'istruzione "GO OPEN TYPE B":

- la prima non richiede a SMP+ la schedulazione della traiettoria durante la modalità aperta; alla fine di questa fase, quindi, la traiettoria viene determinata unicamente da PC.
- con l'istruzione "GO OPEN TYPE B", invece, SMP+ schedula la traiettoria eventualmente insieme al PC.

La terminazione della modalità aperta viene determinata da due eventi distinti:

- SMP+ deve terminare la sua traiettoria e
- il PC deve trasmettere il segnale "EXIT_FROM_OPEN".

Il meccanismo deve funzionare indipendentemente dall'ordine con cui i due eventi avvengono.



14. MODALITÀ

14.1 Introduzione

Con il termine **modalità** si intende la tipologia di funzionamento specifico nell'ambito della prestazione C4G OPEN.

Esistono varie modalità di funzionamento che verranno elencate nei paragrafi successivi; la differenza tra una modalità e l'altra viene identificata dal valore del campo **SMax** ("modalità dell'asse") all'interno del pacchetto di comunicazione tra SMP+ e C4G.

Sono previste delle modalità che permettono di controllare un asse, mediante algoritmi sviluppati su PC esterno, altre che permettono di generare la traiettoria di movimento per un asse, mediante un profilo calcolato da PC esterno. Infine, sono previste anche delle modalità che utilizzano un PC come interfaccia verso un sensore esterno, al fine di usare l'informazione acquisita, per sintetizzare una migliore strategia di controllo del moto da adottare.

Il cambiamento di modalità viene comunicato dal sistema portando ad 1 la variabile `$ARM_DATA[i_ARM].C4GOPEN_CMD[2]`.



Si noti che questa variabile non viene resettata dal sistema; quindi, se deve essere monitorata, occorre avere l'accortezza di resettarla (cioè metterla a 0), non appena passa al valore 1.

Viene ora fornita una descrizione dettagliata dei seguenti argomenti:

- [Modalità 0](#)
- [Modalità 0'](#)
- [Modalità 1](#)
- [Modalità 2](#)
- [Modalità 3](#)
- [Modalità 4](#)
- [Modalità 5](#)
- [Modalità 6](#)
- [Modalità 7](#)
- [Modalità 8](#)
- [Modalità 9](#)
- [Modalità 1xx](#)
- [Modalità 101 \(controllo delle accelerazioni\)](#)
- [Modalità 102 \(visual servoing\)](#)
- [Modalità 103 \(controllo di forza\)](#)
- [Modalità 201](#)

- Modalità 202
- Riepilogo delle modalità
- Modalità e controlli di sicurezza

14.2 Modalità 0

È la modalità di default del controllore aperto ed è il funzionamento standard del sistema.

- Funzioni
- Pacchetti di trasmissione
- Funzionamento
- Note.

14.2.1 Funzioni

- **SMP+**: generazione della traiettoria, controllo del moto e scambio di informazioni con DSA
- **PC**: nessuna operazione significativa

14.2.2 Pacchetti di trasmissione



Nella seguente tabella vengono riportati in ROSSO i pacchetti **NECESSARI** non vuoti, in BLU quelli **NON NECESSARI**.

Tab. 14.1 - Pacchetto di comunicazione in modalità 0

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.2.3 Funzionamento

- a. SMP+ manda i riferimenti di posizione e velocità al PC e a DSA;
- b. DSA calcola il following error in funzione della posizione misurata,
- c. DSA chiude i loop di posizione e velocità

- d. DSA restituisce a SMP+ tutte queste informazioni
- e. SMP+ le elabora (bloccando la macchina nel caso di fault)
- f. SMP+ fornisce le informazioni al PC.

14.2.4 Note

Il PC, anche se non prende parte attivamente al movimento del robot, deve essere presente e rispondere ad ogni pacchetto inviatogli da SMP+.

14.3 Modalità 0'



Modalità non disponibile per l'utente: riservata COMAU.

È una modalità di debug del sistema che permette di verificare lo scambio di informazioni tra SMP+ e PC.

Il PC deve prendere i riferimenti di posizione e velocità e, senza elaborarli, passarli nuovamente a SMP+ che, infine, li fornisce a DSA.

Questo permette di testare la comunicazione tra SMP+ e il PC tralasciando completamente qualsiasi altra elaborazione da parte del PC stesso.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento.](#)

14.3.1 Funzioni

- **SMP+:** generazione della traiettoria, controllo del moto e scambio di informazioni con DSA
- **PC:** copia e rispedisce le informazioni che riceve da SMP+

14.3.2 Pacchetti di trasmissione

Tab. 14.2 - Pacchetto di comunicazione in modalità 0'

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.3.3 Funzionamento

- a. SMP+ calcola i riferimenti di posizione e velocità
- b. SMP+ li passa al PC
- c. PC ripassa a SMP+ le informazioni ricevute
- d. SMP+ fornisce a DSA le informazioni ricevute da PC.

14.4 Modalità 1

È la modalità aperta più completa: permette di dare pieno controllo al PC esterno.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.4.1 Funzioni

- **SMP+**: scambio di informazioni con DSA
- **PC**: generazione della traiettoria e controllo del moto

14.4.2 Pacchetti di trasmissione

Tab. 14.3 - Pacchetto di comunicazione in modalità 1

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.4.3 Funzionamento

- a. PC manda a SMP+ i riferimenti di posizione e velocità
- b. SMP+, a sua volta, li manda a DSA;
- c. PC calcola il riferimento di corrente e/o di velocità
- d. PC manda il dato a SMP+
- e. SMP+ lo manda a DSA

- f. DSA calcola il following error in funzione della posizione misurata
- g. DSA restituisce le informazioni a SMP+
- h. SMP+ le elabora (bloccando la macchina nel caso di fault)
- i. SMP+ le fornisce al PC.

14.4.4 Note

- È necessario, per ragioni di sicurezza, che ci sia un controllo del following error e una saturazione del massimo valore delle correnti a livello PC;
- per uscire da questa modalità si può utilizzare come segnale l'indicazione della modalità 1' (11); il numero tra parentesi è il codice corrispettivo di implementazione da utilizzare nei programmi PDL2 per fare riferimento a questa modalità. Vd. anche [Cap.24. - Macchina a stati.](#)

14.5 Modalità 2

È la modalità che permette di chiudere i loop di controllo sul PC esterno, mentre SMP+ si preoccupa di generare la traiettoria.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.5.1 Funzioni

- **SMP+:** generazione della traiettoria e scambio di informazioni con DSA
- **PC:** controllo del movimento

14.5.2 Pacchetti di trasmissione

Tab. 14.4 - Pacchetto di comunicazione in modalità 2

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.5.3 Funzionamento

- a. SMP+ manda i riferimenti di posizione e velocità al PC e al DSA;
- b. PC calcola il riferimento di corrente e/o di velocità
- c. PC manda il dato a SMP+
- d. SMP+ lo fornisce a DSA
- e. DSA calcola il following error in funzione della posizione misurata
- f. DSA manda le informazioni a SMP+
- g. SMP+ le elabora (bloccando la macchina nel caso di fault)
- h. SMP+ le fornisce al PC.

14.5.4 Note

- È necessario, per ragioni di sicurezza, che ci sia un controllo del following error e una saturazione del massimo valore delle correnti a livello PC;
- per uscire da questa modalità si può utilizzare come segnale l'indicazione della modalità 2' (12); il numero tra parentesi è il codice corrispettivo di implementazione da utilizzare nei programmi PDL2 per fare riferimento a questa modalità. Vd. anche [Cap.24. - Macchina a stati](#).

14.6 Modalità 3

È la modalità che permette di controllare il robot in velocità dal PC esterno, mentre SMP+ si preoccupa di generare la traiettoria in sola velocità.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note](#).

14.6.1 Funzioni

- **SMP+:** generazione della traiettoria in sola velocità e scambio di informazioni con DSA
- **PC:** controllo in velocità

14.6.2 Pacchetti di trasmissione

Tab. 14.5 - Pacchetto di comunicazione in modalità 3

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]

Tab. 14.5 - Pacchetto di comunicazione in modalità 3 (Continua)

n.	tipo del campo	SMP+	PC
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.6.3 Funzionamento

- a. SMP+ manda i riferimenti di velocità al PC e al DSA
- b. PC manda il contributo in corrente a SMP+
- c. SMP+, a sua volta, lo fornisce a DSA
- d. DSA misura la posizione degli encoder
- e. DSA restituisce le informazioni a SMP+
- f. SMP+ le elabora (bloccando la macchina nel caso di fault) e le deriva
- g. SMP+ fornisce le velocità ricostruite al PC.

14.6.4 Note



Nell'attuale versione del software questa modalità non è presente. Essa è prevista per sviluppi futuri.

14.7 Modalità 4

È la modalità di funzionamento che permette al PC di generare la traiettoria in posizione, in modo assoluto.

- Funzioni
- Pacchetti di trasmissione
- Funzionamento
- Note.

14.7.1 Funzioni

- **SMP+:** controllo del moto e scambio di informazioni con DSA
- **PC:** generazione della traiettoria

14.7.2 Pacchetti di trasmissione

Tab. 14.6 - Pacchetto di comunicazione in modalità 4

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.7.3 Funzionamento

- PC manda a SMP+ i riferimenti di posizione assoluta e velocità
- SMP+, a sua volta, li manda a DSA
- DSA calcola il following error in funzione della posizione misurata
- DSA calcola i loop di posizione e velocità
- DSA restituisce a SMP+ queste informazioni
- SMP+ le elabora (bloccando la macchina nel caso di fault)
- SMP+ le fornisce al PC.

14.7.4 Note

- è preferibile che, per sicurezza, ci sia un ulteriore controllo del following error anche a livello PC per avere una gestione dei fault ridondante;
- per uscire da questa modalità si può utilizzare come segnale l'indicazione della modalità 4' (14); il numero tra parentesi è il codice corrispettivo di implementazione da utilizzare nei programmi PDL2 per fare riferimento a questa modalità. Vd. anche Cap.24. - Macchina a stati.



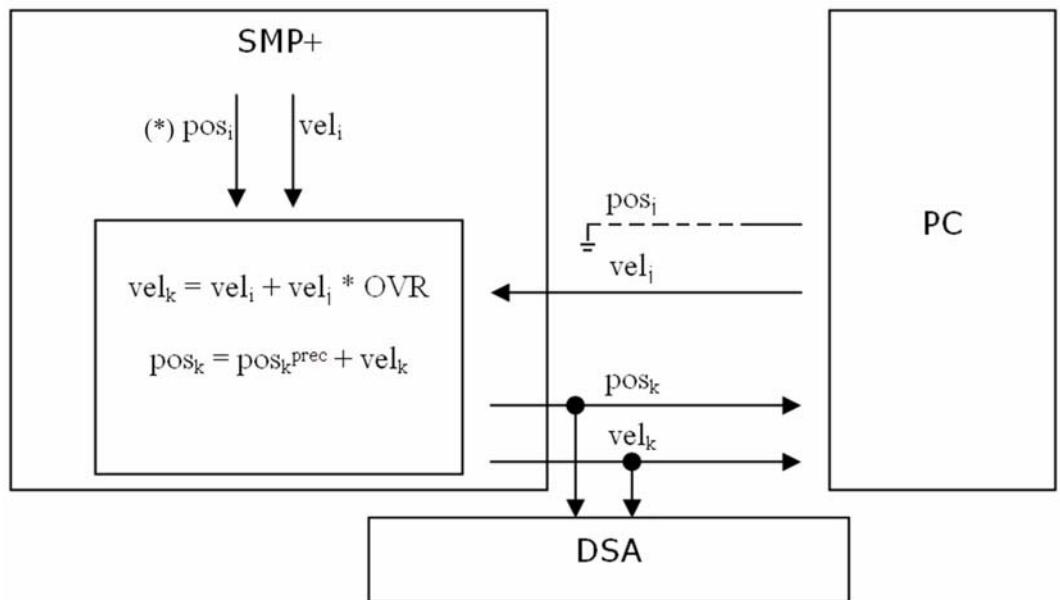
Dal momento che il riferimento in posizione deve essere fornito in maniera assoluta occorre che:

- le costanti di calibrazione del robot devono essere tenute in debito conto,
- prima di entrare in questa modalità operativa occorre portare il robot sul punto iniziale della traiettoria da eseguire in modalità 4.

14.8 Modalità 5

È la modalità di funzionamento che permette al PC di generare la traiettoria in posizione in modo relativo.

Fig. 14.1 - Traiettoria generata come delta di posizioni filtrate con l'override generale



(*) pos_i è utilizzato solo per determinare le condizioni iniziali dell'integratore pos_k

Nella Fig. 14.1 viene riportato lo schema a blocchi funzionale che realizza la particolare configurazione della modalità 5 che si basa sulle posizioni relative, fornite sia da PC che da SMP+.

Dal generatore di traiettoria del C4G standard vengono schedulate le nuove posizioni pos_i e velocità vel_i di riferimento. Il PC calcola le nuove posizioni pos_j e velocità vel_j di riferimento. Le posizioni di riferimento complessive pos_k da inviare a DSA non tengono in conto delle posizioni di riferimento fornite dal generatore di traiettoria del C4G e dal PC, bensì vengono calcolate in funzione delle posizioni di riferimento complessive ricavate al passo precedente e delle velocità di riferimento complessive vel_k.

Le velocità vel_k sono calcolate sommando le velocità di riferimento fornite dal generatore di traiettoria vel_i e le velocità di riferimento calcolate dal PC vel_j opportunamente scalate del fattore di override generale della macchina.

Al termine del movimento congiunto, cioè in uscita dalla modalità aperta, il generatore di traiettoria C4G aggiorna pos_i ponendolo uguale al valore accumulato pos_k e azzera pos_k. A partire da quel momento la traiettoria pos_i inviata al DSA terrà conto delle correzioni calcolate dal PC ed accumulate nel passato durante la generazione della traiettoria in modo relativo (modalità aperta).

**ATTENZIONE!**

Ci sono condizioni di anomalia o emergenza che possono causare una uscita della modalità aperta prima che si siano verificate le condizioni normali e corrette di uscita.

Può avvenire che l'aggiornamento di pos_i con il valore fino a quel momento accumulato in pos_k non possa essere eseguito. In questi casi la macchina può presentare movimenti repentini ed inattesi causati dalla discontinuità esistente tra i valori correnti di pos_i ed i valori correnti di pos_k. PRESTARE LA MASSIMA ATTENZIONE!

Le operazioni da eseguire per riportare il sistema al corretto funzionamento sono:

- a. sospendere l'esecuzione della prova/programma utente nella modalità corrente del C4G Open;
- b. impostare la variabile \$C4G_PA:=1;
- c. impostare le system variable legate al cambio di modalità
(\$ARM_DATA[ind_arm].C4GOPEN_MODE[ind_asse] := 0
che imposta gli assi interessati alla modalità di default);
- d. eseguire un Configure Save All (comando CSA);
- e. eseguire una ripartenza a freddo (comando CCRC);
- f. eventualmente riportare il robot nella posizione di riposo, PRESTANDO LA MASSIMA ATTENZIONE!

Le azioni **b.**, **c.** e **d.** non occorrono nel caso in cui l'utente non abbia salvato (comando CSA) valori diversi da quelli di default, per quelle system variable.

- Funzioni
- Pacchetti di trasmissione
- Funzionamento
- Note.

14.8.1 Funzioni

- **SMP+**: controllo del moto e scambio di informazioni con DSA
- **PC**: generazione della traiettoria

14.8.2 Pacchetti di trasmissione

Tab. 14.7 - Pacchetto di comunicazione in modalità 5

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]

Tab. 14.7 - Pacchetto di comunicazione in modalità 5

n.	tipo del campo	SMP+	PC
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.8.3 Funzionamento

- a. PC manda a SMP+ i riferimenti di posizione relativa e velocità
- b. SMP+, a sua volta, li trasforma in posizioni assolute
- c. SMP+ manda le informazioni a DSA
- d. DSA calcola il following error in funzione della posizione misurata
- e. DSA calcola i loop di posizione e velocità
- f. DSA restituisce le informazioni a SMP+
- g. SMP+ le elabora (bloccando la macchina nel caso di fault)
- h. SMP+ le fornisce al PC.

14.8.4 Note

- È preferibile che, per sicurezza, ci sia un ulteriore controllo del following error anche a livello PC, per avere una gestione ridondante dei fault ;
- per uscire da questa modalità si può utilizzare come segnale l'indicazione della modalità 5' (15); il numero tra parentesi è il codice corrispettivo di implementazione da utilizzare nei programmi PDL2 per fare riferimento a questa modalità. Vd. anche [Cap.24. - Macchina a stati..](#)

14.9 Modalità 6

È la modalità di funzionamento che permette al PC di generare la traiettoria in velocità.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.9.1 Funzioni

- **SMP+:** controllo del moto e scambio di informazioni con DSA
- **PC:** generazione della traiettoria in velocità

14.9.2 Pacchetti di trasmissione

Tab. 14.8 - Pacchetto di comunicazione in modalità 6

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.9.3 Funzionamento

- PC manda a SMP+ i riferimenti di velocità
- SMP+, a sua volta, li manda a DSA
- DSA esegue la misura della posizione dagli encoder
- DSA calcola i loop di velocità
- DSA restituisce a SMP+ queste informazioni
- SMP+ le elabora (bloccando la macchina nel caso di fault)
- SMP+ le fornisce a PC.

14.9.4 Note



Nell'attuale versione del software questa modalità non è presente; essa è prevista in versioni future.

14.10 Modalità 7



Questa modalità è una generalizzazione della [Modalità 5](#), quindi valgono tutte le considerazioni fatte per tale modalità, più quelle contenute nel presente paragrafo. Leggere con cura la nota [ATTENZIONE!](#) presente nella descrizione della [Modalità 5](#).

La differenza sostanziale tra le modalità 5 e 7 consiste nel fatto che la prima non permette che venga schedulato un movimento da SMP+ durante l'esecuzione della modalità.

È la modalità di funzionamento che permette al PC di fornire un contributo **aggiuntivo**

in posizione, alla traiettoria generata da SMP+.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.10.1 Funzioni

- **SMP+**: generazione della traiettoria, controllo del movimento e scambio di informazioni con DSA
- **PC**: calcolo del contributo in posizione aggiuntivo

14.10.2 Pacchetti di trasmissione

Tab. 14.9 - Pacchetto di comunicazione in modalità 7

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.10.3 Funzionamento

- a. PC manda il contributo aggiuntivo in posizione a SMP+
- b. SMP+ manda i riferimenti di posizione (con il contributo aggiuntivo ricevuto dal PC) e velocità a DSA
- c. DSA calcola il following error in funzione della posizione misurata
- d. DSA chiude i loop di posizione e velocità (il primo riceve in ingresso anche il contributo calcolato dal PC)
- e. DSA restituisce a SMP+ queste informazioni
- f. SMP+ le elabora (bloccando la macchina nel caso di fault)
- g. SMP+ le fornisce al PC;

14.10.4 Note

- per uscire da questa modalità si può utilizzare come segnale l'indicazione della modalità 7' (17); il numero tra parentesi è il codice corrispettivo di implementazione

da utilizzare nei programmi PDL2 per fare riferimento a questa modalità. Vd. anche [Cap.24. - Macchina a stati](#).

- questa modalità operativa torna utile quando bisogna correggere la traiettoria in funzione delle informazioni provenienti dal PC esterno.
La correzione può avvenire in due fasi distinte:
 - durante il percorso: in questo caso, C4G e PC contemporaneamente prendono parte attiva al movimento
 - nel posizionamento finale: in questo caso, viene verificata la doppia entrata in soglia sul punto, è prevista la gestione della correzione del movimento da parte del PC dopo che SMP+ abbia terminato la generazione della traiettoria, ovvero, la continuazione della generazione della traiettoria da SMP+ anche dopo la terminazione della correzione apportata dal PC.
- Il contributo aggiuntivo in posizione fornito dal PC, ad ogni step viene memorizzato nella posizione corrente dall'SMP+, secondo lo schema di [Fig. 14.1](#), per cui il PC esterno si limita a fornire unicamente l'ultima correzione relativa alla posizione già aggiornata dall'SMP+ al colpo precedente.

14.11 Modalità 8

È la modalità che permette di aggiungere un contributo in velocità calcolato dal PC esterno mentre SMP+ si preoccupa di generare la traiettoria.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)

14.11.1 Funzioni

- **SMP+:** generazione della traiettoria, controllo del moto e scambio di informazioni con DSA
- **PC:** calcolo del contributo in velocità aggiuntivo

14.11.2 Pacchetti di trasmissione

Tab. 14.10- Pacchetto di comunicazione in modalità 8

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.11.3 Funzionamento

- a. PC manda il contributo aggiuntivo in velocità a SMP+
- b. SMP+ manda i riferimenti di posizione e velocità (con il contributo aggiuntivo ricevuto dal PC) a DSA
- c. DSA calcola il following error in funzione della posizione misurata
- d. DSA chiude i loop di posizione e velocità (questo ultimo riceve in ingresso anche il contributo calcolato dal PC)
- e. DSA restituisce a SMP+ queste informazioni
- f. SMP+ le elabora (bloccando la macchina nel caso di fault)
- g. SMP+ le fornisce a PC.

14.12 Modalità 9

È la modalità che permette di aggiungere un contributo in corrente calcolato da PC esterno, mentre SMP+ genera la traiettoria.

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.12.1 Funzioni

- **SMP+:** generazione della traiettoria, controllo del moto e scambio di informazioni con DSA
- **PC:** calcolo del contributo in corrente aggiuntivo

14.12.2 Pacchetti di trasmissione

Tab. 14.11- Pacchetto di comunicazione in modalità 9

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]
E1ax	campo singolo per asse	modello dinamico [Ampere]	
E2ax	campo singolo per asse	inerzia diagonale [Kg*m ²]	

14.12.3 Funzionamento

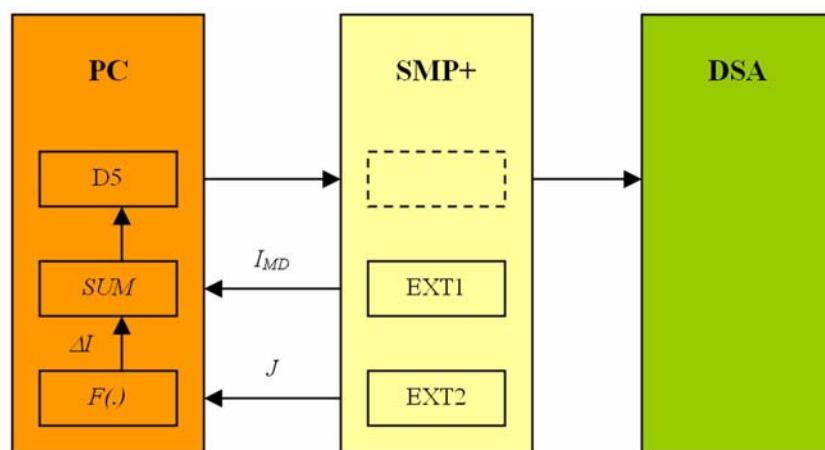
- a. PC manda il contributo aggiuntivo in corrente a SMP+
- b. SMP+ manda i riferimenti di posizione e velocità ed il contributo aggiuntivo in corrente (ricevuto da PC) a DSA;
- c. DSA calcola il following error in funzione della posizione misurata
- d. DSA chiude i loop di posizione e velocità (quest'ultimo ha in ingresso anche il contributo calcolato da PC)
- e. DSA restituisce a SMP+ queste informazioni
- f. SMP+ le elabora (bloccando la macchina nel caso di fault)
- g. SMP+ le fornisce a PC.

14.12.4 Note

- In modalità aperta il modello dinamico calcolato da SMP+ non viene più mandato a DSA, in quanto il canale è utilizzato per fornire a DSA i dati calcolati da PC. Se in questa modalità è necessario utilizzare il modello dinamico, PC deve fornire a SMP+ anche questa informazione. La corrente del modello dinamico può essere calcolata da PC oppure PC può farla tramite ricevendola da SMP+. L'informazione è contenuta nel campo EXT1 del pacchetto che SMP+ manda a PC (vd. [Tab. 14.11](#)). Nel campo EXT2 viene fornita l'inerzia diagonale calcolata da SMP+, anche questa informazione può essere utilizzata da PC per i propri conti.

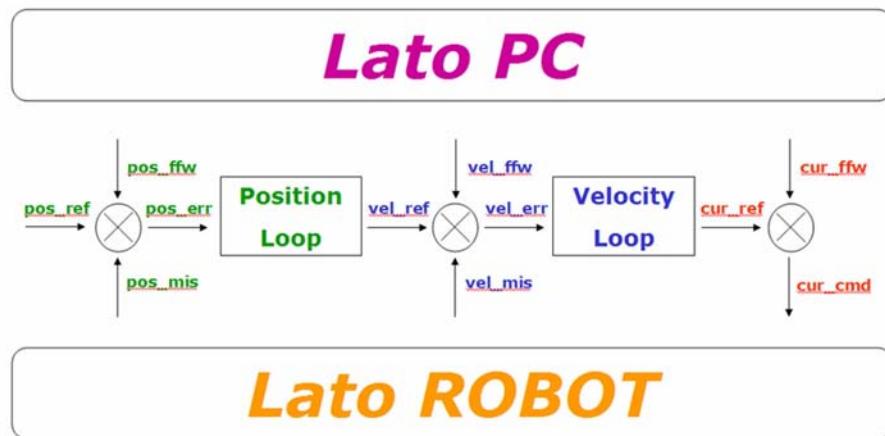
Il seguente schema a blocchi descrive questi aspetti (vd. [Fig. 14.2](#)).

Fig. 14.2 - Contributo aggiuntivo in corrente da PC che utilizza il modello dinamico calcolato da SMP+.



Per distinguere l'effetto dei vari contributi aggiuntivi che il PC remoto può fornire al controllo del movimento, in [Fig. 14.3](#) è riportata la struttura dei blocchi di controllo all'interno di DSA.

Fig. 14.3 - Azione dei contributi aggiuntivi in posizione, velocità e corrente



In ingresso al loop di controllo in posizione, viene passato il contributo aggiuntivo proveniente da PC, insieme al riferimento in posizione calcolato da SMP+ confrontato con la posizione misurata dagli encoder.

Discorso analogo vale per il loop di velocità che riceve come riferimento l'uscita del loop di posizione.

Infine, a DSP arriva il comando di corrente dall'uscita del loop di velocità e il comando di corrente calcolato da PC.

14.13 Modalità 1xx

È la modalità di funzionamento utilizzata per interfacciare il C4G con un sensore esterno.

In questa Modalità, il PC può trasferire una sola misura a SMP+, ma ciò non vieta di gestire più sensori, facendo uso del PC, per poi fornire l'elaborazione finale a SMP+.

Nel sistema C4G OPEN occorre distinguere tra connessione **fisica** e **architettura di controllo estesa**.

Da un punto di vista "fisico", il C4G OPEN basa il suo funzionamento sulla realizzazione di una rete Ethernet real-time a "T" in cui ci sono solo 3 dispositivi che comunicano:

- SMP+ che è l'unico "client" presente sulla rete,
- DSA e PC che sono i due "server" del client.

Il mondo esterno a C4G deve passare attraverso il PC che può essere visto come il componente che integra un numero anche elevato di sensori, di natura diversa, con tipologia di connessioni varie (USB, seriale, schede PCI, etc.) e con passi di campionamento a volte diversi.

È pertanto il PC che può integrare una rete estesa e con diversi sensori.

Questo implementa il massimo grado di flessibilità ed apertura e comporta che l'intelligenza per acquisire, interpretare, elaborare ed integrare l'informazione proveniente dai sensori esterni, risieda su PC.

Anche se a livello di connessione **fisica** il PC è un **server**, da un punto di vista dell'architettura di controllo estesa e **logica**, l'intelligenza del sistema è distribuita e risiede sia su SMP+ che su PC. La connessione fisica va soprattutto vista come puro canale aperto per l'attuazione.

L'architettura così descritta, consente di realizzare applicazioni di controllo robot avanzato.

A titolo di esempio, riporteremo più avanti tre esempi di applicazioni di controllo con sensori esterni:

- controllo con integrazione di un modulo IMU (navigazione inerziale) (vd. [par. 14.13.5 Modalità 101 \(controllo delle accelerazioni\)](#) a pag. 14-19);
- controllo con integrazione di un sistema di visione (vd. [par. 14.13.6 Modalità 102 \(visual servoing\)](#) a pag. 14-19);
- controllo con sensore di forza (vd. [par. 14.13.7 Modalità 103 \(controllo di forza\)](#) a pag. 14-20).

Vengono ora descritti i seguenti argomenti:

- [Funzioni](#)
- [Pacchetti di trasmissione](#)
- [Funzionamento](#)
- [Note.](#)

14.13.1 Funzioni

- **SMP+:** generazione della traiettoria, controllo del moto e scambio di informazioni con DSA
- **PC:** interfaccia con il sensore esterno

14.13.2 Pacchetti di trasmissione

Tab. 14.12- Pacchetto di comunicazione in modalità 1xx

n.	tipo del campo	SMP+	PC
D1ax	campo singolo per asse	posizione target [giri motore]	posizione target [giri motore]
D2ax	campo singolo per asse	velocità target [delta giri motore]	velocità target [delta giri motore]
D3ax	campo singolo per asse	posizione reale [giri motore]	misura [varie]
D4ax	campo singolo per asse	velocità reale [delta giri motore]	contributo in velocità [delta giri motore]
D5ax	campo singolo per asse	corrente di riferimento [Ampere]	contributo in corrente [Ampere]

14.13.3 Funzionamento

- a. PC fornisce a SMP+ la misura proveniente dal sensore esterno
- b. SMP+ manda i riferimenti di posizione e velocità a PC e a DSA considerando la misura proveniente da PC (se la modalità lo richiede)
- c. DSA calcola il following error in funzione della posizione misurata
- d. DSA calcola i loop di posizione e velocità in funzione della misura proveniente da PC (se la modalità lo richiede)
- e. DSA restituisce a SMP+ queste informazioni

- f. SMP+ le elabora (bloccando la macchina nel caso di fault)
- g. SMP+ le fornisce a PC.

14.13.4 Note

- Si sottolinea che la misura può essere utilizzata per la generazione della traiettoria oppure per chiudere i loop di posizione e velocità o per tutte e due le funzioni. Per discriminare i vari casi di funzionamento si deve specificare la modalità aperta descritta nei paragrafi seguenti.

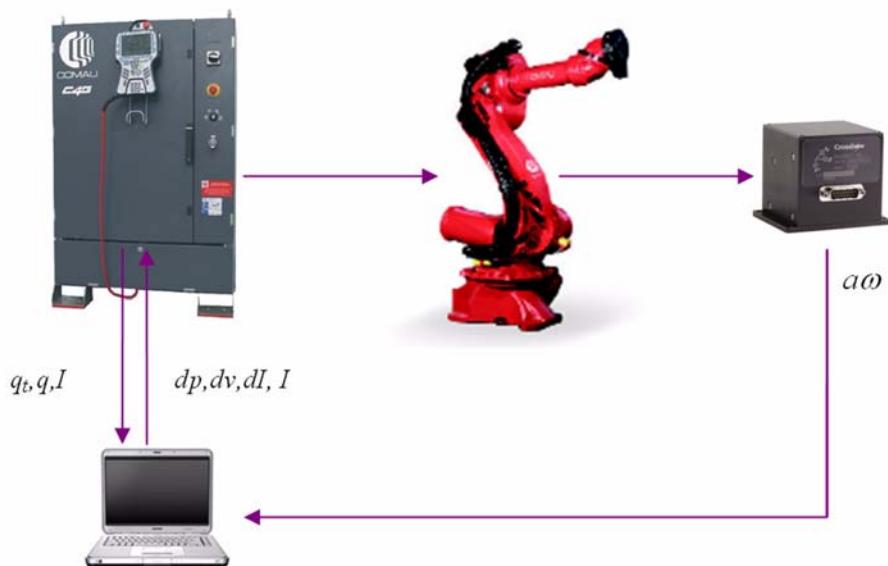


Nell'attuale versione del software questa modalità non è presente; essa è prevista in versioni future.

14.13.5 Modalità 101 (controllo delle accelerazioni)

Il sensore esterno è un accelerometro, per cui l'informazione viene gestita a livello di loop di controllo. In [Fig. 14.4](#) viene presentata una possibile configurazione che fa uso di accelerometro.

Fig. 14.4 - Esempio di architettura C4G OPEN con sensore di accelerazione.



14.13.6 Modalità 102 (visual servoing)

Il sensore esterno è una telecamera, per cui l'informazione viene gestita a livello di generazione della traiettoria, per aggiornare il target in funzione dello scostamento tra posizione prevista dagli encoder e posizione del tool. In [Fig. 14.5](#) e in [Fig. 14.6](#) viene presentata una possibile configurazione del C4G OPEN che fa uso del sensore esterno di visione.

Fig. 14.5 - Esempio di architettura C4G OPEN con telecamera (a)

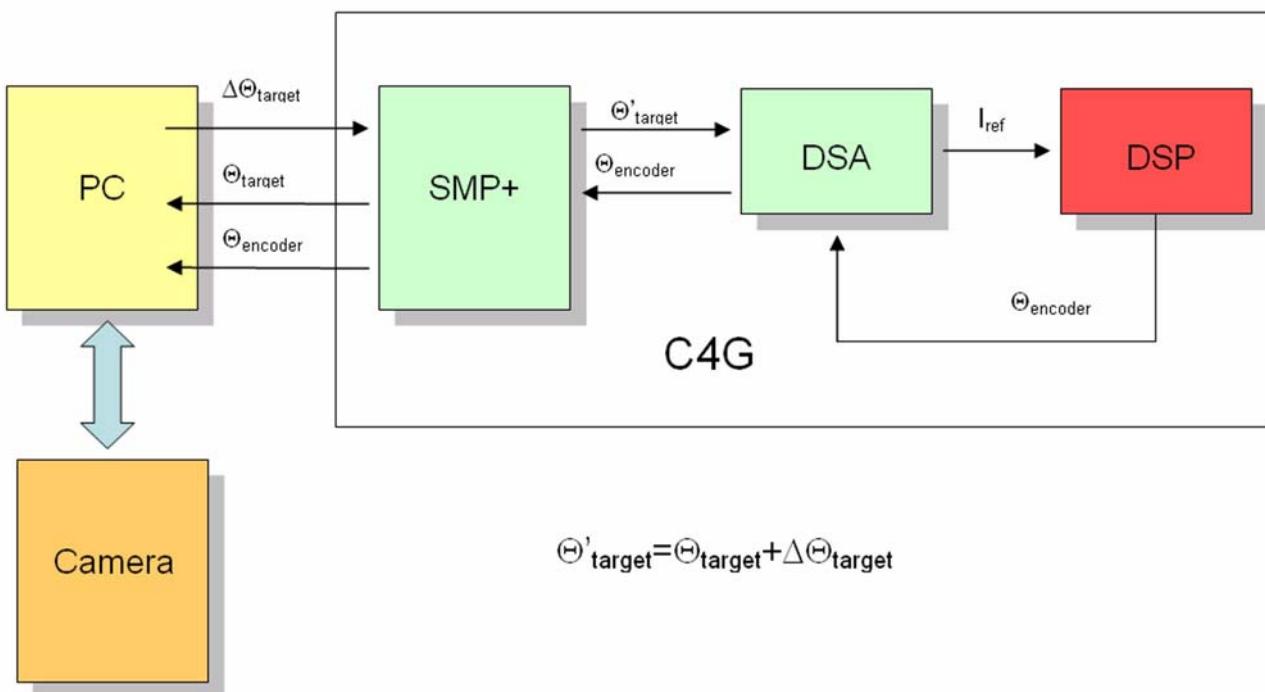
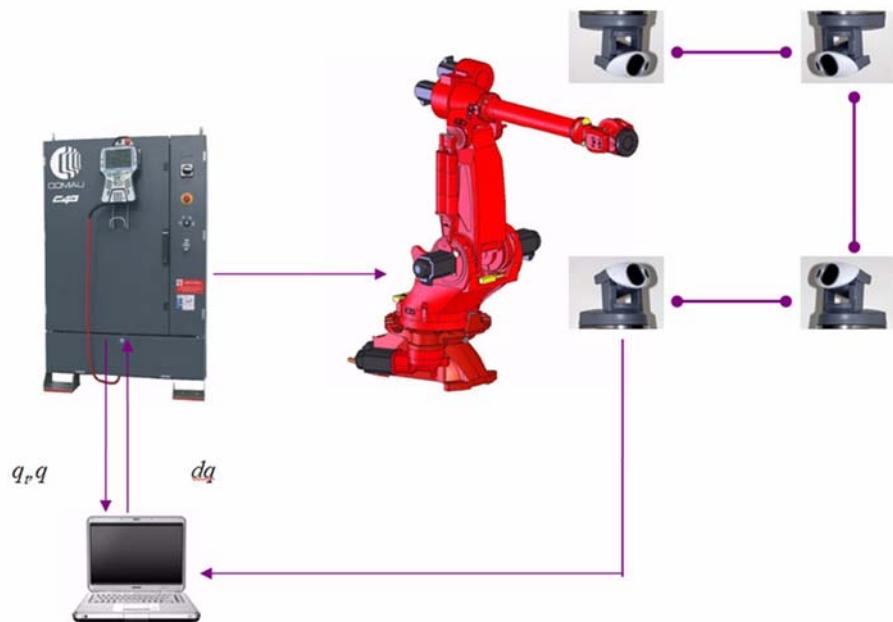


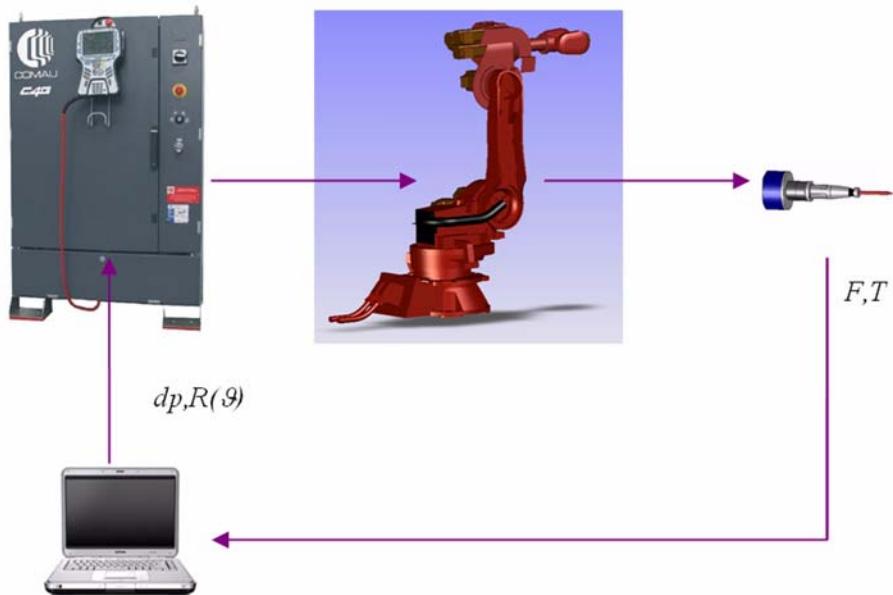
Fig. 14.6 - Esempio di architettura C4G OPEN con telecamera (b)



14.13.7 Modalità 103 (controllo di forza)

Il sensore esterno è un sensore di forza/coppia, per cui verosimilmente l'informazione viene gestita a livello di generazione della traiettoria per aggiornare il target in funzione della forza di contatto tra il robot e l'ambiente estero. In Fig. 14.7 viene riportata una possibile architettura che utilizza il C4G OPEN con un sensore di forza/coppia.

Fig. 14.7 - Esempio di architettura C4G OPEN con sensore di forza



14.14 Modalità 201

Questa modalità consente di generare una traiettoria cartesiana calcolata da PC esterno. In questo caso si propone una revisione completa del C4G OPEN.

Fino ad ora, infatti, la connessione del PC è avvenuta tra SMP+ e DSA, ovvero, l'influenza del PC si è limitata ai riferimenti di posizione, velocità o corrente a livello giunti.

Con questa modalità, invece, PC può fornire una traiettoria espressa nel mondo cartesiano. Per realizzarla, c'è bisogno dell'inversione cinematica presente nel software di SMP+.

- [Funzionamento](#)
- [Note.](#)

14.14.1 Funzionamento

- a. PC manda il riferimento cartesiano a SMP+
- b. SMP+ fornisce, ai livelli software superiori della generazione di traiettoria, il riferimento cartesiano ricevuto da PC,
- c. SMP+ lo trasforma in riferimento espresso in giunti
- d. SMP+ manda i riferimenti di posizione e velocità a DSA
- e. DSA calcola il following error in funzione della posizione misurata
- f. DSA chiude i loop di posizione e velocità
- g. DSA restituisce a SMP+ queste informazioni

- h. SMP+ le elabora (bloccando la macchina nel caso di fault) e le trasforma in posizioni nel mondo cartesiano
- i. SMP+ le fornisce a PC.

14.14.2 Note

- Poiché il flusso dati comporta l'utilizzo degli strati software del generatore di traiettoria, è molto grande il ritardo tra riferimento imposto e attuazione vera e propria.
Questo è uno dei motivi per cui conviene far lavorare il C4G OPEN unicamente nel mondo giunti, eventualmente eseguendo il passaggio dal dominio giunti a quello cartesiano e viceversa, nel PC remoto.
- Vd. [par. 14.15.2 Note a pag. 14-23](#) per ulteriori informazioni.



Nell'attuale versione del software questa modalità non è presente; essa è prevista in versioni future.

14.15 Modalità 202

Questa modalità consente di fornire un contributo aggiuntivo in posizione calcolato da PC esterno nel mondo cartesiano.

Il funzionamento è molto simile a quello visto nella modalità precedente. L'unica differenza è che il riferimento fornito da PC non è assoluto ma deve essere sommato al riferimento cartesiano calcolato da SMP+.

- [Funzionamento](#)
- [Note.](#)

14.15.1 Funzionamento

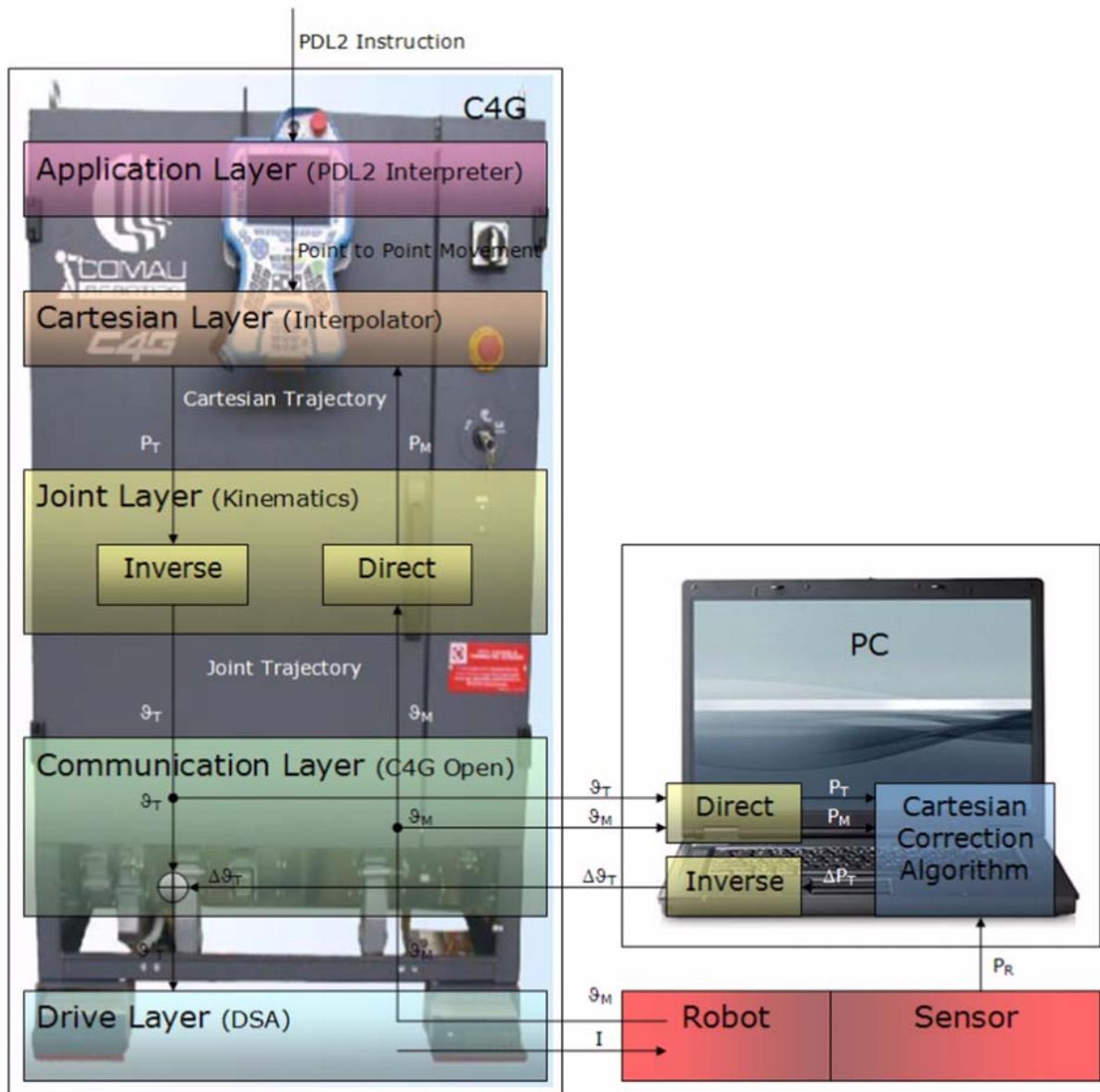
- a. PC manda il contributo aggiuntivo in posizione a SMP+
- b. SMP+ fornisce questa informazione ai livelli software superiori della generazione di traiettoria, per costruire il riferimento cartesiano assoluto e trasformarlo nel riferimento giunti
- c. SMP+ manda i riferimenti di posizione e velocità a DSA
- d. DSA calcola il following error in funzione della posizione misurata
- e. DSA chiude i loop di posizione e velocità
- f. DSA restituisce a SMP+ queste informazioni
- g. SMP+ le elabora (bloccando la macchina nel caso di fault) e le trasforma in posizioni nel mondo cartesiano
- h. SMP+ le fornisce a PC.

14.15.2 Note

- Poiché il flusso dati comporta l'utilizzo degli strati software del generatore di traiettoria, è molto grande il ritardo tra riferimento imposto e attuazione vera e propria.

Questo è uno dei motivi per cui conviene far lavorare il C4G OPEN unicamente nel mondo giunti, eventualmente eseguendo il passaggio dal dominio giunti a quello cartesiano e viceversa, nel PC remoto.

Un modo per riportare il funzionamento del C4G Open dal dominio cartesiano a quello giunti, e viceversa, è schematizzato nella figura seguente:



Nella figura si riportano tutti i livelli di gestione del movimento all'interno del software del C4G:

- "Application Layer": a questo livello c'è il software di interpretazione del linguaggio di programmazione robot PDL2 che riceve in ingresso le istruzioni di movimento (tipo MOVE TO P1 e MOVE TO P2) e restituisce in uscita le indicazioni generali del movimento (movimento cartesiano o giunti, punto P1 e punto P2, velocità di movimento, ecc...);

- "Cartesian Layer": a questo livello lavora il software di generazione della traiettoria nel dominio cartesiano che riceve in ingresso le indicazioni generali del movimento fornite dal livello superiore e restituisce in uscita la traiettoria cartesiana in real-time ad ogni passo dell'interpolatore;
- "Joint Layer": a questo livello è presente il software di generazione della traiettoria nel dominio giunti che riceve in ingresso la traiettoria cartesiana dal livello superiore e in real-time ad ogni passo dell'interpolatore la trasforma in traiettoria ai giunti, il software esegue anche la trasformazione inversa: dalle posizioni giunti reali misurate dagli encoder calcola le posizioni cartesiane mediante la cinematica diretta;
- "Communication Layer": a questo livello è eseguito il software di comunicazione che riceve in ingresso dal livello superiore la traiettoria giunti real-time, governa la comunicazione tra SMP+ e DSA (ricevendo le posizioni misurate dagli encoder e fornendo le posizioni di riferimento) e, nel caso del C4G Open, governa la comunicazione tra SMP+ e PC esterno (fornendo le posizioni dei motori e la traiettoria giunti e ricevendo le correzioni sulla traiettoria giunti);
- "Drive Layer": a questo livello è eseguito il software dell'azionamento che, ricevendo in ingresso dal livello superiore la traiettoria giunti complessiva real-time, calcola le correnti da fornire ai motori restituendo le posizioni dei motori misurate dagli encoder.

Nella figura si riporta anche il set minimo di livelli software che devono girare sul PC esterno per gestire il sensore esterno nel "dominio cartesiano":

- "Direct": modulo software che implementa la cinematica diretta (trasformazione dal dominio giunti a quello cartesiano delle posizioni misurate e di quelle di riferimento) coerentemente con quella eseguita da C4G;
- "Inverse": modulo software che implementa la cinematica inversa (trasformazione dal dominio cartesiano a quello giunti delle correzioni alla posizione cartesiana) coerentemente con quella eseguita da C4G;
- "Cartesian Correction Algorithm": modulo software che comunica con il sensore, riceve dalla cinematica diretta le posizioni cartesiane "reali" (ricostruite, cioè, a partire dalle posizioni dei motori) e le posizioni cartesiane di riferimento (ricostruite dalle posizioni di riferimento ai giunti calcolate da C4G), applica il controllo specifico da realizzare (impedenza, forza, ibrido, ...) e fornisce la correzione alla posizione cartesiana al modulo che si occupa della cinematica inversa.

Alla luce della presentazione degli strati software attivi su C4G e su PC esterno, si possono trarre le due seguenti importanti conclusioni:

- a. risulta ovvia la scelta di calare la tecnologia del C4G Open nel dominio giunti piuttosto che in quello cartesiano: il modulo appartiene al livello software di comunicazione;
- b. relativamente al software del PC, a parte i moduli che eseguono le cinematiche diretta e inversa (che sono ben noti e fissati con la classe di appartenenza del robot), l'utente si deve preoccupare unicamente di comunicare con il sensore ed implementare l'opportuno algoritmo che realizza l'applicazione specifica richiesta.



Nell'attuale versione del software questa modalità non è presente; essa è prevista in versioni future.

14.16 Riepilogo delle modalità

Nella tabella seguente si riporta una panoramica di tutte le modalità previste per il C4G OPEN. Sono indicati anche i vincoli che il sistema deve rispettare per realizzare una particolare modalità.

Tab. 14.13- Riepilogo delle varie modalità del C4G OPEN

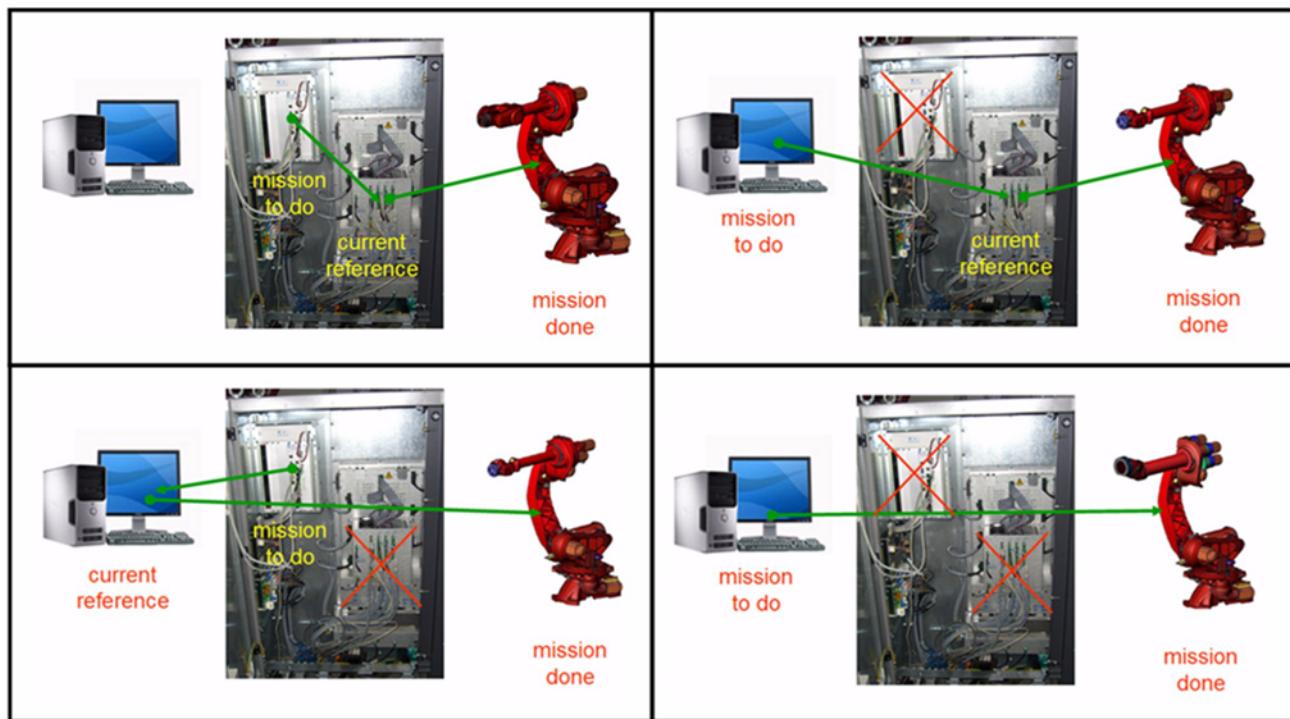
Modo	PC	SMP+	NOTE
0	--	target e loop di posizione e velocità	modalità per default
0'	Riservata COMAU		
1	target e loop di posizione e velocità	--	a., b., c., d., e., f., p.
2	loop di posizione e velocità	target di posizione	b., f., o., p.
3	loop di velocità	target di velocità	b., c., e., f., g., o.
4	target di posizione assoluta	loop di posizione e velocità	a., d., e., f.
5	target di posizione relativa	loop di posizione e velocità	a., d., e., f.
6	target di velocità	loop di velocità	a., c., d., e., f., g.
7	contributo aggiuntivo in posizione	target e loop di posizione e velocità	c., d., e., f., o.
8	contributo aggiuntivo in velocità	target e loop di posizione e velocità	c., d., f., h., o.
9	contributo aggiuntivo in corrente	target e loop di posizione e velocità	f., o.
1xx	misure generiche	target e loop di posizione e velocità	l.
101	misure dall'accelerometro	target e loop di posizione e velocità	f., i., o.
102	misure dalla videocamera	target e loop di posizione e velocità	c., d., e., f., j.
103	misure dal sensore di forza/coppia	target e loop di posizione e velocità	c., d., e., j., k.
201	target di posizione cartesiana	loop di posizione e velocità	d., m.
202	delta target di posizione cartesiana	loop di posizione e velocità	d., m., n.

14.16.1 Note

- a. movimento via PDL2 assente,
- b. saturazione dell'uscita in corrente,
- c. gestione dell'errore di inseguimento,
- d. controllo della traiettoria esterna,
- e. adattamento della posizione finale,
- f. modello dinamico eliminabile con una ripartenza del C4G,
- g. inibizione del loop di posizione,
- h. la media del contributo in velocità deve essere nulla,
- i. contributo in corrente o in velocità o in delta posizione,
- j. contributo in delta posizione,
- k. gestione della informazione su SMP+ mediante il modello di impedenza,

- I. gestione della misura su SMP+,
- m. inversione cinematica su SMP+,
- n. somma dei contributi cartesiani su SMP+,
- o. loop di controllo "semplici",
- p. controllo del following error e del valore massimo delle correnti dal PC.

Fig. 14.8 - Architettura ideale "equivalente" alle modalità OPEN più importanti



In alto a sinistra: modalità 0
 in alto a destra: modalità 5
 in basso a sinistra: modalità 2
 in basso a destra: modalità 1

14.17 Modalità e controlli di sicurezza

- Controlli di sicurezza lato C4G
- Controlli di sicurezza lato PC.

14.17.1 Controlli di sicurezza lato C4G

Nella tabella seguente si riporta una panoramica di tutte le modalità previste per il C4G OPEN ed i corrispondenti controlli di sicurezza attivi interni al C4G.

Tab. 14.14- Riepilogo dei controlli previsti per le varie modalità del C4G OPEN

Modo	Controlli		
	comunicazione PC-SMP+	valore limite di corrente	errore di inseguimento
0	attivo	attivo	attivo
0'	riservata COMAU		
1	attivo	attivo	non attivo (**)
2	attivo	attivo	attivo
3	modalità non ancora sviluppata		
4	attivo	attivo	attivo (*)
5	attivo	attivo	attivo (*)
6	modalità non ancora sviluppata		
7	attivo	attivo	attivo (***)
8	attivo	attivo	attivo (****)
9	attivo	attivo	attivo (*****)
1xx	modalità non ancora sviluppata		
101	modalità non ancora sviluppata		
102	modalità non ancora sviluppata		
103	modalità non ancora sviluppata		
201	modalità non ancora sviluppata		
202	modalità non ancora sviluppata		

Per tutte le modalità è attivo il controllo della comunicazione: se la comunicazione tra PC e SMP+ cade, se vengono persi dei pacchetti di comunicazione o se il PC esterno non risponde nei tempi prestabiliti, il robot è posto in stato di sicurezza a fronte della generazione dell'opportuno errore di comunicazione.

Il normale funzionamento del sistema è possibile unicamente a fronte di una ripartenza del C4G.

14.17.2 Controlli di sicurezza lato PC



Si noti che i controlli qui descritti sono a carico dell'utilizzatore del C4G Open.

Per tutte le modalità in cui il PC esterno fornisce completamente o parzialmente la traiettoria, giacché il C4G non ha modo di verificarne la correttezza, occorre che l'opportuno controllo dei fine-corsa sia implementato nel software del PC esterno. Inoltre, è indispensabile che il software del PC esterno effettui una verifica sulla

continuità relativa ai riferimenti calcolati.



In particolare, nel caso in cui il PC esterno fornisca completamente o parzialmente la traiettoria, ma non chiuda i loop di posizione e velocità, l'aggiornamento del riferimento di posizione e di velocità per l'interpolatore C4G avviene alla fine del movimento.

Ciò significa che, in caso di anomalia o emergenza, potrebbe esserci un'uscita dalla modalità aperta prima che si siano verificate le condizioni normali e corrette di uscita, quindi anche prima che sia terminato il movimento.

Può quindi avvenire che l'aggiornamento di pos con il valore fino a quel momento accumulato in posk, non venga eseguito.

In questi casi la macchina può presentare transitori di spostamento veloci causati dalla discontinuità esistente tra i valori correnti di pos e i valori correnti di posk.
PRESTARE LA MASSIMA ATTENZIONE!

Le operazioni da eseguire per riportare il sistema al corretto funzionamento sono:

- a. sospendere l'esecuzione della prova/programma utente nella modalità corrente del C4G Open;
- b. impostare la variabile \$C4G_PA:=1;
- c. impostare le system variable legate al cambio di modalità
(\$ARM_DATA[ind_arm].C4GOPEN_MODE[ind_asse] := 0
che imposta gli assi interessati alla modalità di default);
- d. eseguire un Configure Save All (comando CSA);
- e. eseguire una ripartenza a freddo (comando CCRC);
- f. eventualmente riportare il robot nella posizione di riposo, PRESTANDO LA MASSIMA ATTENZIONE!

Le azioni b., c. e d. non occorrono nel caso in cui l'utente non abbia salvato (comando CSA) valori diversi da quelli di default, per quelle system variable.

Nelle modalità operative in cui **PC fornisce in modo esclusivo la traiettoria del robot** (casi contrassegnati con un solo asterisco (*)), il controllo sull'errore di inseguimento si riferisce a quella traiettoria; quindi, se essa non è corretta, il C4G non ha modo di rilevare l'errore. Resta attivo, comunque, il controllo del valore massimo della corrente erogabile ai vari motori mediante saturazione della stessa.

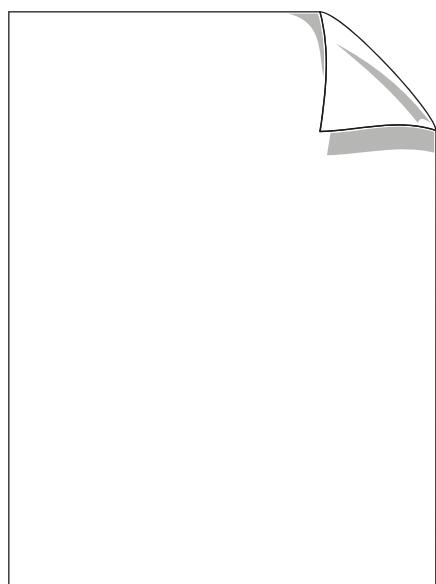
Nelle modalità operative in cui **PC gestisce la chiusura dei loop di posizione e velocità, fornendo a SMP+ la corrente** (casi contrassegnati con due asterischi (**)), è attivo il solo controllo del valore massimo della corrente erogabile ai vari motori mediante saturazione della stessa.

Nelle modalità operative in cui **PC fornisce la traiettoria del robot in modo additivo a SMP+** (casi contrassegnati con tre asterischi (**)), il controllo sull'errore di inseguimento si riferisce alla traiettoria complessiva quindi, se essa non è corretta, il C4G non ha modo di rilevare l'errore. Resta attivo, comunque, il controllo del valore massimo della corrente erogabile ai vari motori mediante saturazione della stessa e lo scostamento massimo tra un punto della traiettoria di riferimento e quella schedulata al passo precedente.

Nelle modalità operative in cui **PC fornisce la velocità del robot in modo additivo a**

SMP+ (casi contrassegnati con quattro asterischi (****)), il controllo sull'errore di inseguimento si riferisce alla traiettoria schedulata da SMP+ a meno del contributo in velocità. Resta attivo, comunque, il controllo del valore massimo della corrente erogabile ai vari motori mediante saturazione della stessa e lo scostamento massimo tra un punto della traiettoria di riferimento e quella schedulata al passo precedente.

Nelle modalità operative in cui **PC fornisce la corrente ai motori in modo additivo a SMP+** (casi contrassegnati con cinque asterischi (*****)) resta attivo il controllo del valore massimo della corrente erogabile ai vari motori mediante saturazione della stessa e lo scostamento massimo tra un punto della traiettoria di riferimento e quella schedulata al passo precedente.



15. MODALITÀ SPECIALI

Attraverso il canale utilizzato per la variazione della modalità C4G OPEN desiderata, si possono attivare ulteriori funzionalità qui descritte:

- Freezing attivo
- DRIVING ON
- Freezing passivo
- Sblocco del server
- Richiesta di DRIVE OFF
- Richiesta di RESTART
- Following error da PC.

15.1 Freezing attivo

Utilizzando la modalità speciale *C4GOPEN_MOD_FREEZING_A* (500), SMP+ non va più a testare la comunicazione con il PC. In questo modo, possono essere eseguite operazioni prolungate non real-time sul PC (come la lettura di un file) evitando che il sistema si fermi perché viene perso qualche pacchetto di comunicazione.

Viene chiamato freezing **attivo** perché è il PC che congela la comunicazione e, successivamente, la riattiva.

Quando SMP+ riceve il segnale *C4GOPEN_MOD_FREEZING_A*, considera valido il pacchetto corrente, ma non fa scattare l'errore fatale se, al successivo colpo di interpolatore, PC non fornisce il pacchetto.

Quando poi SMP+ riceve un nuovo pacchetto da PC e questo pacchetto contiene il valore *EXIT_FROM_OPEN* nel campo INFPAR, riattiva la comunicazione, manda il sistema in modalità 0 e dal successivo pacchetto ripristina la normale comunicazione con il PC.



Attenzione - Questa particolare procedura deve essere eseguita predisponendo il sistema in Modalità 0 che è l'unica modalità in cui il contenuto del pacchetto proveniente da PC non è significativo per le azioni da intraprendere su SMP+.

15.2 DRIVING ON

Durante la procedura di DRIVE ON non può essere imposta (da PC o da PDL2) una modalità diversa dalla 0.

Per far questo, SMP+, durante il DRIVE ON forza la modalità speciale *C4GOPEN_MOD_DRIVING_ON* (501) ignorando qualsiasi richiesta proveniente dal server.

Quando la procedura di DRIVE ON è finita, la modalità non è più *C4GOPEN_MOD_DRIVING_ON* bensì quella impostata da PC o PDL2.

Si noti che il server riesce a rilevare l'istante in cui la procedura di DRIVE ON ha termine,

semplicemente individuando il momento in cui il campo SMax del pacchetto in ricezione passa dal valore *C4GOPEN_MOD_DRIVING_ON* a quello scelto da PC o da PDL2.

15.3 Freezing passivo

Utilizzando la modalità speciale *C4GOPEN_MOD_FREEZING_P_IN* (502), SMP+ non manda più i pacchetti a PC, congelando l'attività con PC stesso.

In questo modo, possono essere eseguite operazioni prolungate non real-time sul C4G (esempio: lettura moni.log) evitando che il sistema si fermi perché viene perso qualche pacchetto di comunicazione.

Viene chiamato **freezing passivo** perché la riattivazione della normale comunicazione avviene settando la modalità speciale *C4GOPEN_MOD_FREEZING_P_OUT* (503) ancora da SMP+.

Il PC in realtà non si accorge di nulla, in quanto, rimane in attesa di un pacchetto che, nell'intervallo di tempo tra l'istante in cui è stata comandata la modalità *C4GOPEN_MOD_FREEZING_P_IN* e l'istante di tempo in cui è stata comandata la modalità *C4GOPEN_MOD_FREEZING_P_OUT*, non viene mandato da SMP+.



Attenzione - Questa particolare procedura deve essere eseguita predisponendo il sistema in Modalità 0 che è l'unica modalità in cui il contenuto del pacchetto proveniente da PC non è significativo per le azioni da intraprendere su SMP+.

15.4 Sblocco del server

Si è detto in precedenza che il processo server che gira su PC deve essere attivato prima che venga attivato il C4G.

Inoltre, qualsiasi richiesta faccia SMP+, il server deve essere pronto a rispondere in maniera coerente.

Quando bisogna riottenere il controllo totale del PC, si può utilizzare la modalità speciale *C4GOPEN_MOD_EXIT_OPEN* (504). Questa modalità è stata predisposta proprio per far cessare definitivamente la verifica della comunicazione real-time tra PC e SMP+.

Quando SMP+ riceve questa richiesta (via PDL2 o mediante CCRC), la manda al PC. In questo modo, al colpo successivo di interpolatore SMP+ non va più a testare la modalità real-time, mentre il processo server che gira su PC può uscire dal ciclo infinito di servizi alle richieste del client.



Attenzione - Questa particolare procedura deve essere eseguita predisponendo il sistema in Modalità 0 che è l'unica modalità in cui il contenuto del pacchetto proveniente da PC non è significativo per le azioni da intraprendere su SMP+.

15.5 Richiesta di DRIVE OFF

È prevista una modalità speciale che permette a PC di richiedere che SMP+ esegua la procedura di DRIVE OFF per fermare il robot.

La modalità operativa è *C4GOPEN_MOD_DRIVE_OFF* (505): quando il PC esterno manda questo valore nel pacchetto di comunicazione, il sistema schedula il DRIVE OFF e poi si porta nella **Modalità 0**.

15.6 Richiesta di RESTART

Durante l'implementazione di una nuova applicazione che gira su PC remoto, è utile riavviare spesso il server, ma evitare di dover riavviare ogni volta il C4G.

Esiste quindi una modalità speciale che permette di riavviare il solo servizio client su SMP+ senza far ripartire il C4G.

Questa modalità operativa è **C4GOPEN_MOD_SVR_RESTART** (506) che può essere settata mediante istruzione PDL2.

Quando la comunicazione riprende, ovviamente, il primo pacchetto trasmesso è quello di inizializzazione. Si noti che non vengono rispediti i pacchetti di sincronizzazione.

15.7 Following error da PC

Per la **Modalità 1** e la **Modalità 2** è necessario che il PC esterno controlli il valore dell'errore di inseguimento e delle correnti di riferimento per i motori.

In dettaglio, i controlli da eseguire sono i seguenti:

$$\begin{aligned} \text{err}(i) &= |pos_{ref}(i) - pos_{mis}(i)| \leq \text{err}^{\max}(i) \\ |\text{cur}_{ref}(i)| &\leq \text{cur}^{\max}(i) \end{aligned}$$

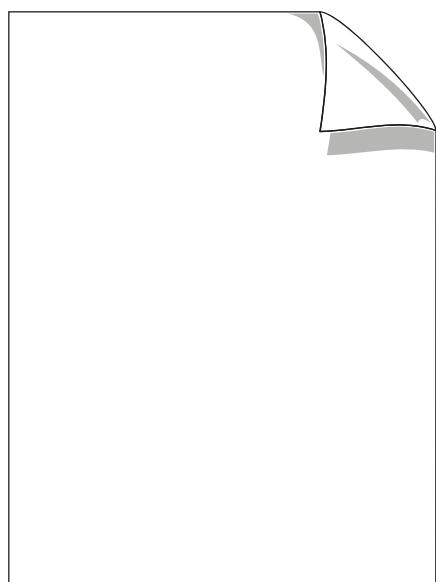
dove:

- **pos_{ref}(i)** è la posizione di riferimento attuale dell'asse **i**,
- **pos_{mis}(i)** è la posizione misurata attuale dell'asse **i**,
- **err(i)** è l'errore di inseguimento attuale dell'asse **i**,
- **err_{max}(i)** è la massima soglia di errore di inseguimento consentita per l'asse **i**,
- **cur_{ref}(i)** è la corrente di riferimento attuale dell'asse **i**,
- **cur_{max}(i)** è il massimo valore di corrente di riferimento consentita per l'asse **i**.

Nel caso in cui l'errore di inseguimento superi la soglia massima consentita (per almeno un asse), PC richiede a SMP+ di segnalare un errore, grazie alla modalità speciale **C4GOPEN_MOD_FOLLOW_ERROR** (508).

Ovviamente, SMP+ dopo aver segnalato l'errore manda in DRIVE OFF il sistema.

Nel caso in cui la corrente di riferimento superi il valore limite, PC fornisce a SMP+ il valore massimo consentito per quel dato asse, eseguendo la saturazione di corrente remota.



16. MONI DEL C4G OPEN

Per monitorare il funzionamento del sistema C4G OPEN, ovvero, le operazioni eseguite da SMP+, i comandi forniti da PC e le risposte che fornisce il robot in termini di correnti di riferimento ai motori e posizioni misurate dagli encoder, sono state predisposte diverse di tipologie di **moni** che registrano i contenuti dei pacchetti, scambiati tra SMP+ e PC, ad ogni colpo di interpolatore.



- **Con la parola MONI si intende lo strumento COMAU per analizzare i più svariati segnali real-time, interni a C4G.**
- **Nel seguito, con i termini PACCHETTO RX e PACCHETTO TX, si intendono, rispettivamente, il pacchetto ricevuto da SMP+ e quello trasmesso da SMP+.**

Vengono ora descritti dettagliatamente:

- [Moni di tipo 19](#)
- [Moni di tipo 20](#)
- [Moni di tipo 21](#)
- [Moni di tipo 22](#)
- [Moni di tipo 23](#)
- [Moni di tipo 24](#).

16.1 Moni di tipo 19

Per ogni asse campionario vengono fornite le seguenti informazioni:

- pacchetto RX - Campo d1 [-]
- pacchetto RX - Campo d2 [-]
- pacchetto RX - Campo d3 [-]
- pacchetto RX - Campo d4 [-]
- pacchetto RX - Campo d5 [-]
- modalità C4G OPEN [-]



Questo tipo di moni viene utilizzato quando occorre analizzare l'intero pacchetto proveniente da PC.

16.2 Moni di tipo 20

Per ogni asse campionario vengono fornite le seguenti informazioni:

- pacchetto TX - Campo d1 [-]
- pacchetto TX - Campo d2 [-]

- pacchetto TX - Campo d3 [--]
- pacchetto TX - Campo d4 [--]
- pacchetto TX - Campo d5 [--]
- modalità C4G OPEN [--]



Questo tipo di moni viene utilizzato quando occorre analizzare l'intero pacchetto proveniente da SMP+.

16.3 Moni di tipo 21

Per ogni asse campionato vengono fornite le seguenti informazioni:

- pacchetto RX - Campo d1 [--]
- pacchetto RX - Campo d2 [--]
- pacchetto TX - Campo d1 [--]
- pacchetto TX - Campo d3 [--]
- pacchetto TX - Campo d5 [--]
- modalità C4G OPEN [--]



Questo tipo di moni è misto: ci sono informazioni contenute nel pacchetto proveniente da SMP+ e informazioni provenienti da PC.

È utile per analizzare la Modalità 5, in quanto, le prime due informazioni sono i target, rispettivamente, di posizione e velocità calcolati da PC, mentre le ultime due informazioni sono, rispettivamente, la posizione reale fornita da quest'ultimo e la corrente di riferimento inviata a DSA, mentre la terza informazione è il target calcolato da SMP+.

16.4 Moni di tipo 22

Per ogni asse campionato vengono fornite le seguenti informazioni:

- pacchetto RX - Campo d4 [--]
- pacchetto RX - Campo d5 [--]
- pacchetto TX - Campo d1 [--]
- pacchetto TX - Campo d3 [--]
- pacchetto TX - Campo d5 [--]
- modalità C4G OPEN [--]



Questo tipo di moni è **misto**: ci sono informazioni contenute nel pacchetto proveniente da SMP+ e informazioni provenienti da PC.

È utile per analizzare la **Modalità 2**, in quanto, la terza informazione è il target calcolato da SMP+, la quarta informazione è la posizione reale misurata dagli encoder, la prima e la seconda informazione sono i contributi in velocità e corrente calcolati da PC, mentre l'ultima informazione è la corrente di riferimento ai motori.

16.5 Moni di tipo 23

Per ogni asse campionato vengono fornite le seguenti informazioni:

- pacchetto TX - Campo d1 [--]
- pacchetto TX - Campo d5 [--]
- pacchetto TX - Campo e1 [--]
- pacchetto TX - Campo e2 [--]
- pacchetto RX - Campo d5 [--]
- modalità C4G OPEN [--]



Questo tipo di moni è **misto**: ci sono informazioni contenute nel pacchetto proveniente da SMP+ e informazioni provenienti da PC, inoltre memorizza dati contenuti anche nei campi EXT.

È utile per analizzare la **Modalità 9**, in quanto, le prime quattro informazione (target di posizione, corrente di riferimento, corrente del modello dinamico e inerzia diagonale) servono per permettere a PC di calcolare la corrente aggiuntiva da fornire a DSA. Questo dato è registrato nel moni nella quinta informazione.

16.6 Moni di tipo 24

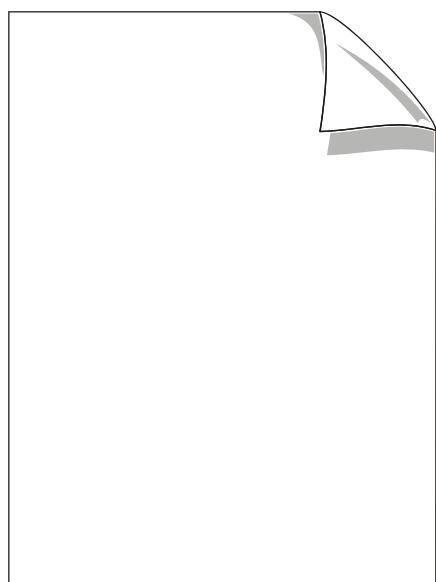
Per ogni asse campionato vengono fornite le seguenti informazioni:

- pacchetto TX - Campo d1 [--]
- pacchetto TX - Campo d5 [--]
- pacchetto TX - Campo e1 [--]
- pacchetto TX - Campo e2 [--]
- pacchetto RX - Campo d4 [--]
- modalità C4G OPEN [--]



Questo tipo di moni è **misto**: ci sono informazioni contenute nel pacchetto proveniente da SMP+ e informazioni provenienti da PC, inoltre memorizza dati contenuti anche nei campi EXT.

È utile per analizzare la **Modalità 8**, in quanto, le prime quattro informazione (target di posizione, corrente di riferimento, corrente del modello dinamico e inerzia diagonale) servono per permettere a PC di calcolare la velocità aggiuntiva da fornire a DSA. Questo dato è registrato nel moni nella quinta informazione.



17. ERRORI C4G OPEN

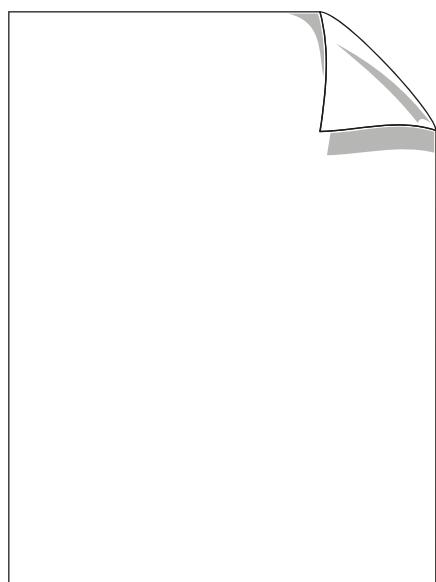
Di seguito si riporta la lista dei messaggi di errore generati da SMP+ nel caso di condizioni di funzionamento anomalo del C4G Open.



Nel caso in cui si verifichino altri errori, si rimanda ai manuali del Sistema C4G standard.

Tab. 17.1 - Errori del C4G OPEN

Errore	Messaggio	Descrizione
ERR_C4GOPEN_SKT_UDP	<i>C4GOpen: errore apertura socket UDP porta 1000</i>	Non è possibile stabilire la comunicazione con il server, in quanto non è possibile aprire una socket UDP sulla porta 1000
ERR_C4GOPEN_UDP	<i>C4GOpen: errore di comunicazione</i>	Il C4G non ha ricevuto per 10 volte consecutive il pacchetto UDP da PC
ERR_C4GOPEN	<i>C4GOpen: errore di configurazione</i>	Si stanno configurando troppi assi in modalità aperta
ERR_C4GOPEN_CONFIGURATION_AX	<i>C4GOpen: ARM %d Asse %d non presente</i>	L'utente vuole configurare in modalità aperta un asse non presente
ERR_C4GOPEN_UDP_DEBUG	<i>C4GOpen: Rx Debug</i>	Il C4G non ha ricevuto il pacchetto UDP da PC
ERR_C4GOPEN_TX_UDP	<i>C4GOpen: errore in trasmissione socket UDP</i>	Non è possibile inviare il pacchetto UDP
ERR_C4GOPEN_FOLLOW_ERROR	<i>C4GOpen: ARM %d asse %d errore di inseguimento fuori limite</i>	Following error rilevato da PC
ERR_C4GOPEN_OPT_HW	<i>C4GOpen: Opzione software non consentita per ARM %d</i>	Non è possibile abilitare l'opzione C4GOpen su questa macchina
ERR_C4GOPEN_DRIVEOFF_SERVER	<i>C4GOpen: Drive Off da Pc</i>	Comando di drive off ricevuto da PC



18. VARIABILE C4OPEN_CMD

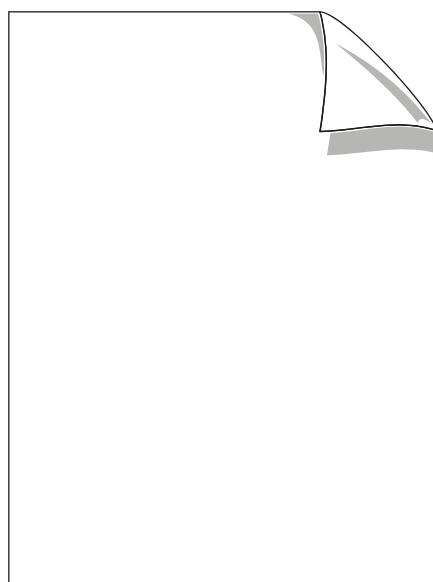
La variabile di sistema `$ARM_DATA[i_ARM].C4OPEN_CMD` è un array di 16 elementi che può essere utilizzato per alcune funzioni di servizio.

In questo paragrafo si riassume il significato degli elementi attualmente utilizzati:

- `$ARM_DATA[i_ARM].C4OPEN_CMD[1]`: indicazione del tipo di movimento all'interno di un percorso complesso da eseguire in un'unica modalità;
- `$ARM_DATA[i_ARM].C4OPEN_CMD[2]`: indicazione del cambiamento di modalità



ATTENZIONE: questa variabile non viene resettata dal sistema, quindi, se deve essere monitorata, occorre avere l'accortezza di cambiarne il valore a 0 subito dopo che passa al valore 1



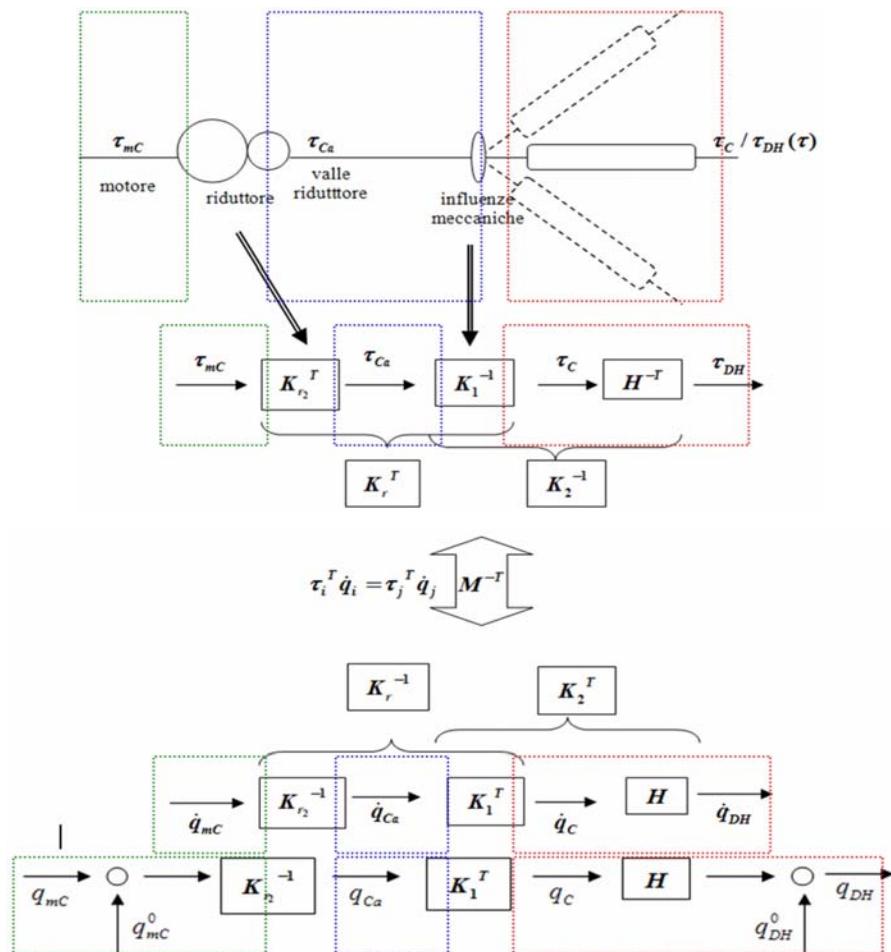
19. TRASFORMAZIONI IN PRESENZA DI INFLUENZE CINEMATICHE

In Fig. 19.1 viene riportato il disegno schematico delle influenze cinematiche che possono esistere nei polsi di alcuni robot COMAU.

Per esprimere analiticamente le relazioni che esistono tra le grandezze calcolate in vari punti della catena cinematica, è bene introdurre le seguenti definizioni:

- **variabili lato motore ("mC")**: sono grandezze calcolate a monte della trasmissione, in generale espresse secondo le convenzioni COMAU;
- **variabili di attuazione ("Ca")**: sono grandezze calcolate a valle del riduttore, ma a monte delle influenze cinematiche, in generale espresse secondo le convenzioni COMAU;
- **variabili lato link ("C" o "DH")**: sono le grandezze al link, cioè a valle del riduttore e delle influenze cinematiche e, a seconda dei casi, possono essere espresse in convenzioni COMAU ("C") o Denavit-Hartenberg ("DH").

Fig. 19.1 - Disegno schematico delle influenze cinematiche del polso



Inoltre va sottolineato che, nel caso in cui le grandezze siano delle posizioni, le misure degli encoder (ovviamente espresse lato motore) devono essere depurate dalle costanti di calibrazione \mathbf{q}^0_{mc} calcolate in giri motore.

Infine, relativamente alle posizioni link, le relazioni tra le convenzioni COMAU e quelle Denavit-Hartenberg, devono tenere in conto degli offset tra i due sistemi di riferimento, oltre che della matrice di trasformazione H.

La matrice K_{r_2} che mette in relazione le grandezze lato motore con quelle all'attuazione viene chiamata **matrice dei rapporti di trasmissione puri** e, per un robot a sei assi, assume la forma seguente:

$$K_{r_2} = \begin{bmatrix} k_{r_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{r_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{r_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{r_4} & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{r_5} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{r_6} \end{bmatrix}$$

dove i k_{r_i} sono i rapporti di trasmissione puri (con segno) dei riduttori o degli ingranaggi utilizzati dai vari assi.

La matrice K_1 che mette in relazione le grandezze all'attuazione con quelle lato link viene chiamata **matrice delle influenze cinematiche** e, per un robot a sei assi, assume la forma seguente:

$$K_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & k_{54} & k_{64} \\ 0 & 0 & 0 & 0 & 1 & k_{65} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

dove:

- k_{54} è il coefficiente di influenza dell'asse 4 sull'asse 5 (muovendo l'asse 4 ruota anche l'asse 5 secondo il fattore k_{54}),
- k_{64} è il coefficiente di influenza dell'asse 4 sull'asse 6 (muovendo l'asse 4 ruota anche l'asse 6 secondo il fattore k_{64}),
- k_{65} è il coefficiente di influenza dell'asse 5 sull'asse 6 (muovendo l'asse 6 ruota anche l'asse 6 secondo il fattore k_{65}).

Il loro valore è determinato in funzione dei salti intermedi di riduzione che sono presenti sulla catena cinematica e sono determinabili analizzando i seguenti campi:

```

arm1AxInfo[ GBM_NAX_ARM ]
arm2AxInfo[ GBM_NAX_ARM ]
arm3AxInfo[ GBM_NAX_ARM ]
arm4AxInfo[ GBM_NAX_ARM ]

```

del pacchetto di inizializzazione.

In dettaglio:

```

k54 = arm*AxInfo[ 4 ]
k64 = arm*AxInfo[ 5 ]
k65 = arm*AxInfo[ 6 ]

```

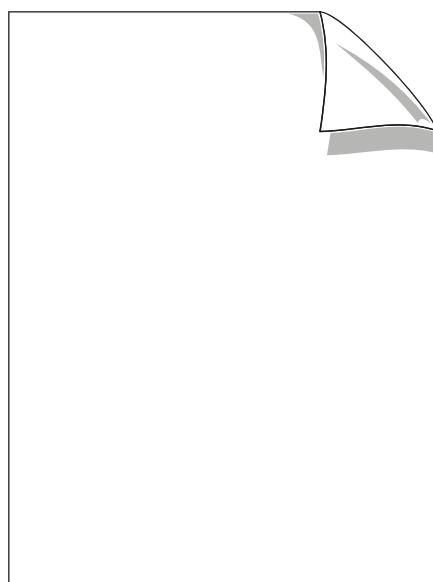
La matrice \mathbf{K}_r che mette in relazione direttamente le grandezze lato motore con quelle lato link viene chiamata **matrice di trasmissione** e viene calcolata nel seguente modo:

$$K_r = K_{r_2} K_1^{-T} :$$

$$K_r = \begin{bmatrix} k_{r_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{r_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{r_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{r_4} & 0 & 0 \\ 0 & 0 & 0 & -k_{r_5}k_{54} & k_{r_5} & 0 \\ 0 & 0 & 0 & k_{r_6}(k_{65}k_{54} - k_{64}) & -k_{r_6}k_{65} & k_{r_6} \end{bmatrix}$$

Ad esempio, per trasformare le posizioni ai link in radianti in posizioni ai motori in giri (comprensivi delle costanti di calibrazione) occorre utilizzare la seguente formula:

$$q_{m_C} = K_r \frac{q_C}{2\pi} + q_{m_C}^0$$



20. SEGNALE DI SINCRONISMO

Affinché la base tempi di PC e C4G siano agganciate in modo molto preciso, si può fornire a PC il segnale di sincronismo (SYNC) della rete che è prodotto da SMP+ ed utilizzato da DSA.

Questo meccanismo di sincronizzazione hardware si aggiunge a quello software, basato sulla comunicazione stringente di pacchetti tra SMP+ e PC.

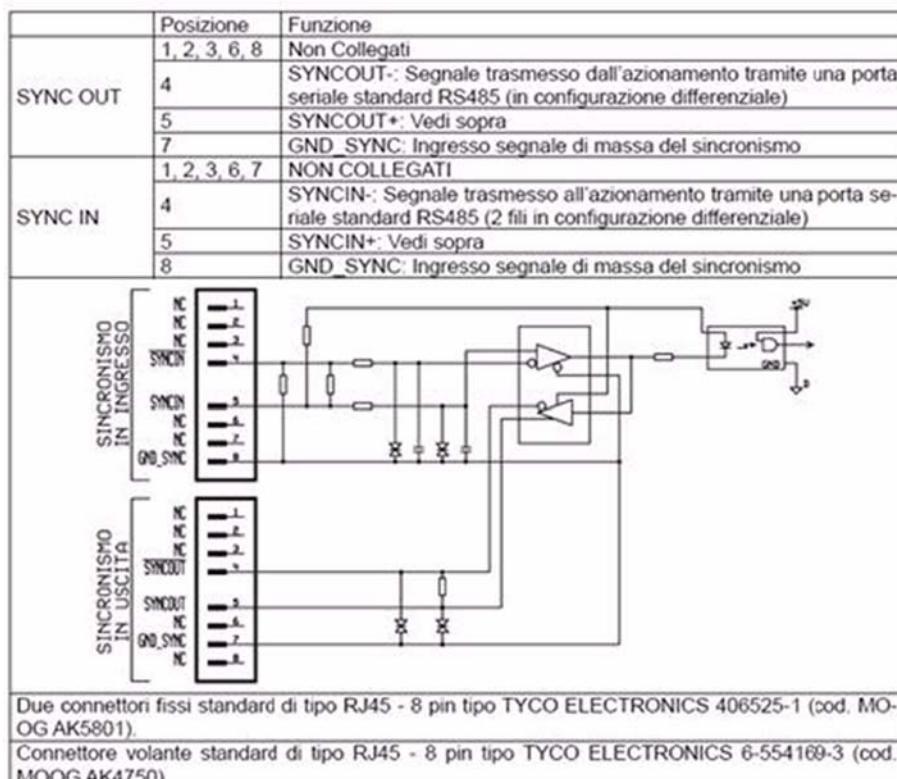
Il segnale SYNC è una fronte d'onda quadra a 8KHz (125us).

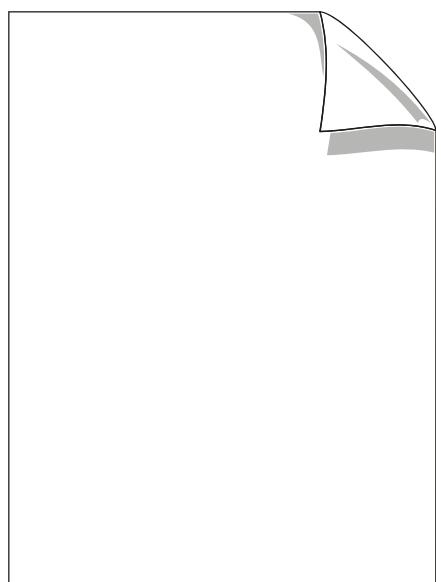
Fisicamente il segnale viene trasmesso mediante una connessione a tre fili: due in differenziale per il segnale, uno per lo 0V dei segnali.

Il connettore SYNC IN (X6) (vd. Fig. 20.1) è usato per l'ingresso del clock esterno e lo porta internamente all'azionamento, il connettore SYNC OUT (X5) è l'uscita dello stesso segnale di clock in modo tale da rendere disponibile il segnale di sincronismo ad un possibile altro azionamento o, nel caso del C4G OPEN, al PC esterno.

Ovviamente, se esistono più dispositivi che hanno bisogno del segnale di sincronizzazione hardware, occorrerà che essi si passino il segnale di SYNC essendo predisposti alla configurazione DAISY CHAIN.

Fig. 20.1 - Schema elettrico del segnale SYNC.



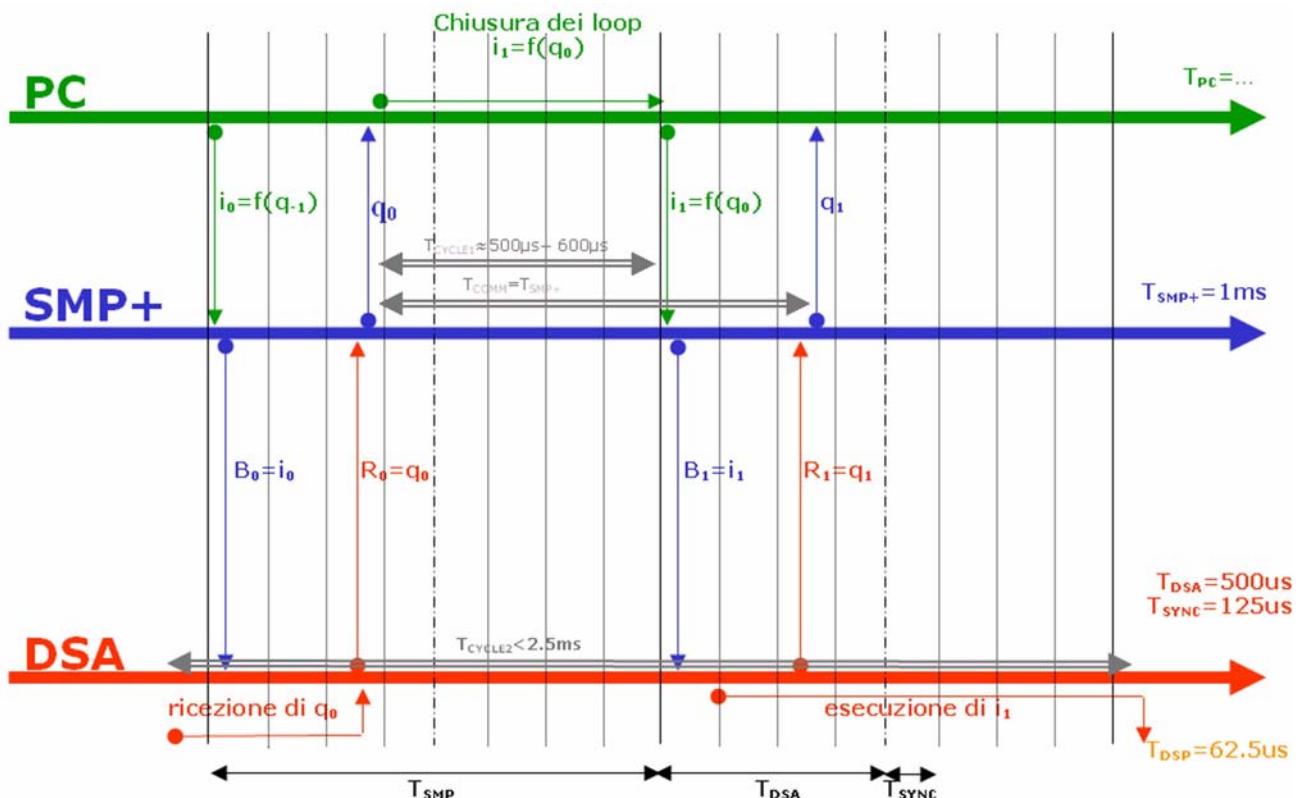


21. TIMING

Prima di parlare del timing che sta alla base del funzionamento del C4G OPEN è opportuno introdurre alcune definizioni:

- periodo base del PC T_{PC} : è il periodo base di esecuzione delle istruzioni su PC;
- periodo base di sincronizzazione T_{SYNC} : è il periodo del segnale di sincronizzazione tra SMP+ e DSA pari a 125usec (8KHz) (vd. Cap.20. - Segnale di sincronismo);
- periodo base di SMP+: T_{SMP+} : è il periodo di campionamento dell'interpolatore pari a 1msec (1KHz) nel caso si voglia utilizzare il C4G in modalità aperta;
- periodo base di DSA T_{DSA} : è il periodo di campionamento dei loop di posizione e velocità internamente a DSA ed è pari a 500usec (2KHz) nel caso del C4G OPEN;
- periodo base di DSP T_{DSP} : è il periodo di campionamento dei loop di corrente internamente ai DSP ed è pari a 62.5usec (16KHz);
- intervallo T_{CYCLE2} : è la distanza temporale massima (inferiore a 2.5ms) tra la ricezione delle posizioni misurate dagli encoder e l'istante in cui viene effettivamente eseguita da DSA l'azione calcolata da PC;
- intervallo T_{CYCLE1} : è la finestra temporale entro cui il PC deve rispondere al pacchetto mandato da SMP+ ed è pari a circa 500-600usec;
- il ritardo robot T_{ROBOT} : è il ritardo con cui risponde la meccanica del robot ai riferimenti in posizione.

Fig. 21.1 - Timing di C4G OPEN



In [Fig. 21.1](#) viene riportato il diagramma temporale che sta alla base della comunicazione tra SMP+, DSA e PC.

È stato considerato il caso in cui il PC deve elaborare le posizioni misurate dagli encoder ed, in funzione di queste, calcolare generici contributi (correnti di riferimento, riferimenti di posizione, riferimenti di velocità, contributi del modello dinamico, ...) da inviare a DSA.

Una volta che l'encoder ha misurato la posizione del motore, DSA non trasferisce immediatamente questa informazione a SMP+ ma solo dopo il secondo colpo successivo di sincronizzazione (T_{SYNC}), ovvero, prima della metà del colpo successivo di SMP+ (T_{SMP+}) nel pacchetto DSA-->SMP+ (RAAZ).

Appena SMP+ riceve l'informazione, la trasferisce a PC. All'inizio del successivo colpo d'interpolazione, SMP+ (T_{SMP+}) deve fornire le informazioni a DSA nel pacchetto SMP+-->DSA (BLIP).

Ovviamente, se PC ha risposto in tempo, SMP+ può fornire a DSA le informazioni da esso mandate.

Per questo motivo, il PC deve rispondere al messaggio mandato da SMP+ in un tempo inferiore a T_{CYCLE1} .

Quando DSA riceve il BLIP da SMP+, l'informazione in essa contenuta non viene processata da DSP istantaneamente, ma all'inizio del secondo colpo successivo di DSA (T_{DSA}), ovvero, all'inizio del colpo successivo di SMP+ (T_{SMP+}).



Si noti che da quando viene misurata la posizione del motore a quando viene eseguita una azione frutto di una elaborazione del PC passa il tempo T_{CYCLE2} .

Riassumendo, il funzionamento del sistema C4G OPEN si basa sul seguente ciclo infinito di azioni da eseguire:

- a. PC riceve il pacchetto con le ultime quote da SMP+ (questo messaggio funge anche da interrupt e sincronizzazione software per PC);
- b. PC ha un intervallo di tempo pari a T_{CYCLE1} per elaborare le informazioni prima di spedirle a SMP+ (oltre questo tempo, il ciclo si interrompe generando un errore fatale);
- c. SMP+ riceve il pacchetto da PC, lo elabora (in funzione della particolare modalità aperta) e lo passa a DSA;
- d. DSA riceve da SMP+ il pacchetto (BLIP) e risponde con le misure effettuate;
- e. SMP+ elabora il pacchetto ricevuto (RAAZ) e lo passa a PC;
- f. il ciclo riprende dal punto a.

Risulta chiaro che il tempo di esecuzione dell'intero ciclo è pari a $T_{COMM}=T_{SMP+}=1\text{ms}$.

Ciò comporta che, al fine di ottimizzare la gestione delle azioni su PC, occorra dividere la loro elaborazione in **due fasi** (vd. [Fig. 21.2](#)):

- **Fase di "calcolo algebrico"**: in questa fase si effettuano i calcoli basati sulla conoscenza dell'ultima posizione dei motori ottenuta da DSP. Questa fase deve avere una durata inferiore a T_{CYCLE1} .
- **Fase di "calcolo dinamico"**: in questa fase possono essere aggiornate tutte le grandezze che si basano sullo stato del sistema, indipendente dall'ultima posizione fornita dai DSP. Questa fase può durare almeno $T_{COMM}-T_{CYCLE1}$.

Fig. 21.2 - Fase di pre-calcoli e di calcoli sul PC

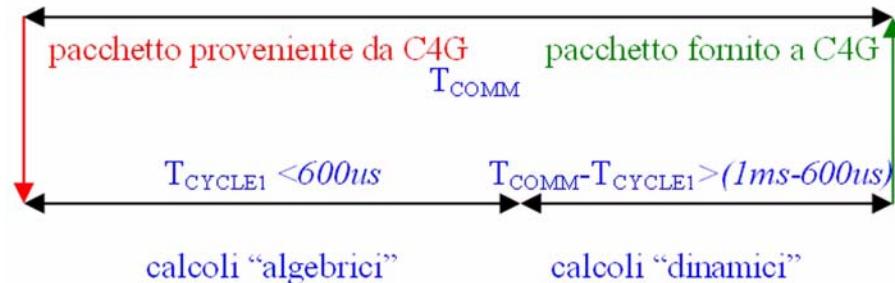
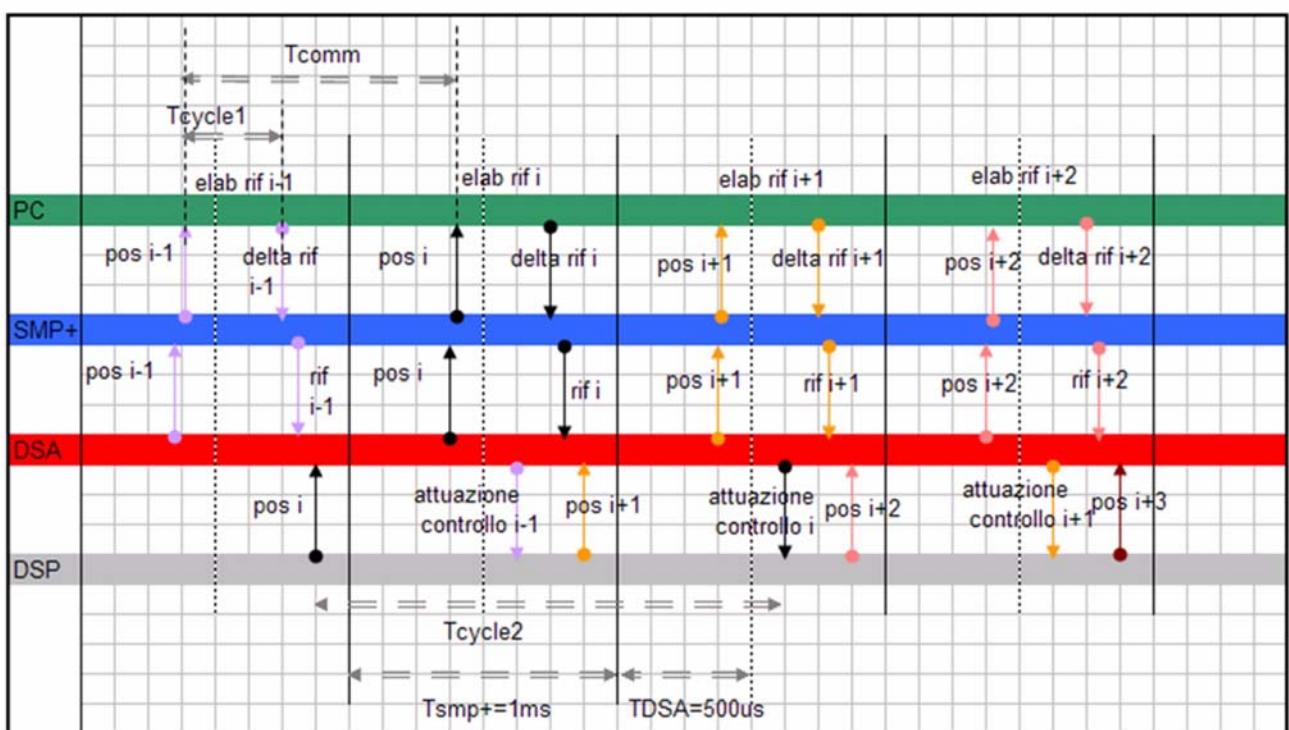
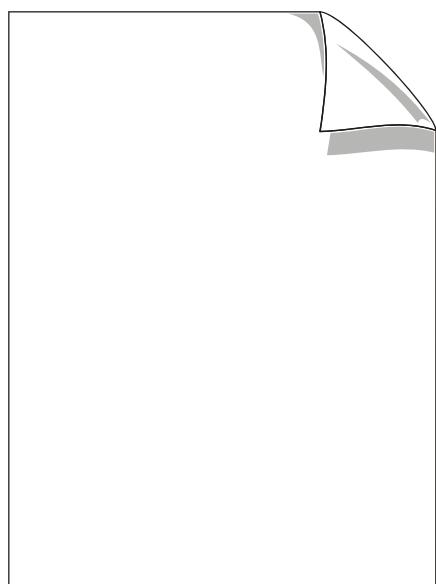


Fig. 21.3 - Scheduling temporale delle informazioni nel modello C4G OPEN





22. TIMING CALCOLATO DA PC

Di seguito viene riportata una parte del codice che può essere eseguito da PC per verificare le tempistiche della comunicazione con SMP+.

```

// ****
// AT BEGINNING
// ****

// definition of packet period
double interPacketsPeriod = 1e-3; // C4G Open <-> PC Inter-Packets Period [secs]

// ****
// IN CYCLE
// ****

// receiving
si_sts = mvfi_ReceiveFromSocketUDP(&(sx_conn));
t1 = rt_get_time_ns();

// calculation

// sending
si_sts = mvfi_SendFromSocketUDP(&(sx_conn));
t2 = rt_get_time_ns();

// cycle time
lastCycleTime = (double)(t2 - t1) / 1000000000.0;

// inter-step time
lastInterStepTime = ((double)(t1 - t1old)) / 1000000000.0;

// updating statistics
worstCycleTime = max(lastCycleTime);
minInterStepTime = min(lastInterStepTime);
maxInterStepTime = max(lastInterStepTime);

// ****
// AT END
// ****

// final statistics
double minJitter = minInterStepTime - interPacketsPeriod;
double maxJitter = maxInterStepTime - interPacketsPeriod;
double minPercJitter = minJitter / interPacketsPeriod * 100.0;
double maxPercJitter = maxJitter / interPacketsPeriod * 100.0;
// display statistics
printf("Nominal Inter-Packets Period = %.9f s\n", interPacketsPeriod);
printf("Worst cycle time = %.9f s\n", worstCycleTime);

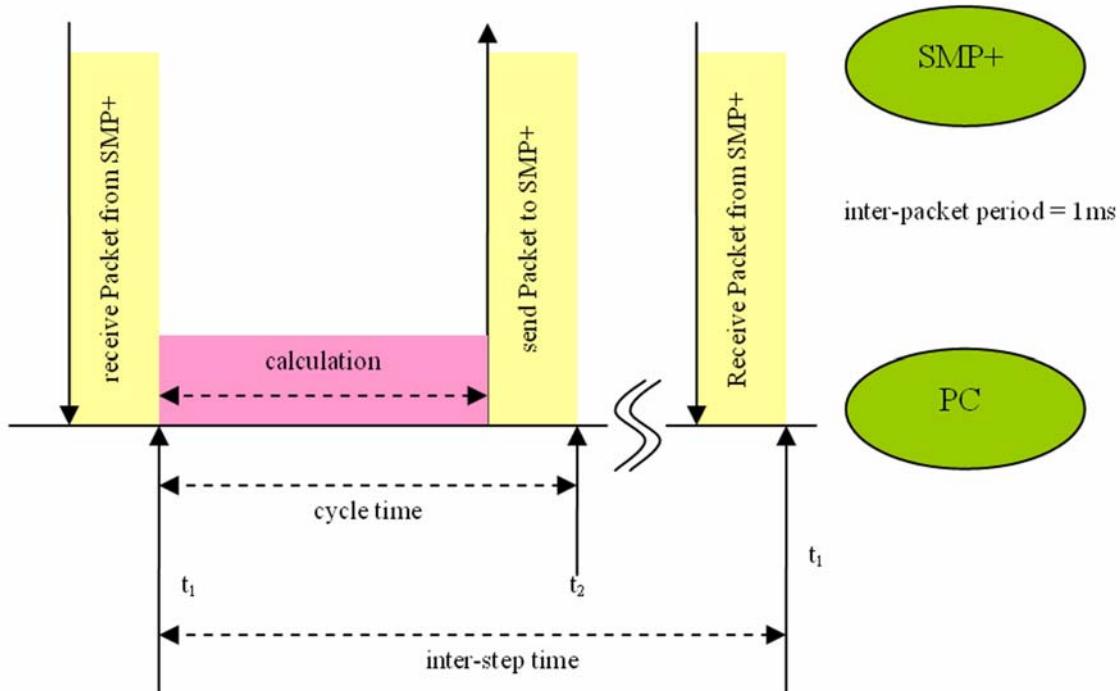
```

```

printf("Measured Inter-Packets Period range = [%.9f; %.9f] s\n",
minInterStepTime, maxInterStepTime);
printf("Jitter range = [%.9f; %.9f] s\n", minJitter, maxJitter);
printf("Percentage jitter range           = [%.9f; %.9f] %\n\n",
minPercJitter, maxPercJitter);

```

Fig. 22.1 - Timing esaminato dal punto di vista di PC



La Fig. 22.1 mostra nel dettaglio il timing delle operazioni eseguite da PC.
Si noti che:

$$\left\{ \begin{array}{l} \text{last_cycle_time} = t_2(k) - t_1(k) \\ \text{worst_cycle_time} = \max(\text{last_cycle_time}) \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{last_inter_step_time} = t_1(k) - t_1(k-1) \\ \text{min_inter_step_time} = \min(\text{last_inter_step_time}) \\ \text{max_inter_step_time} = \max(\text{last_inter_step_time}) \end{array} \right.$$

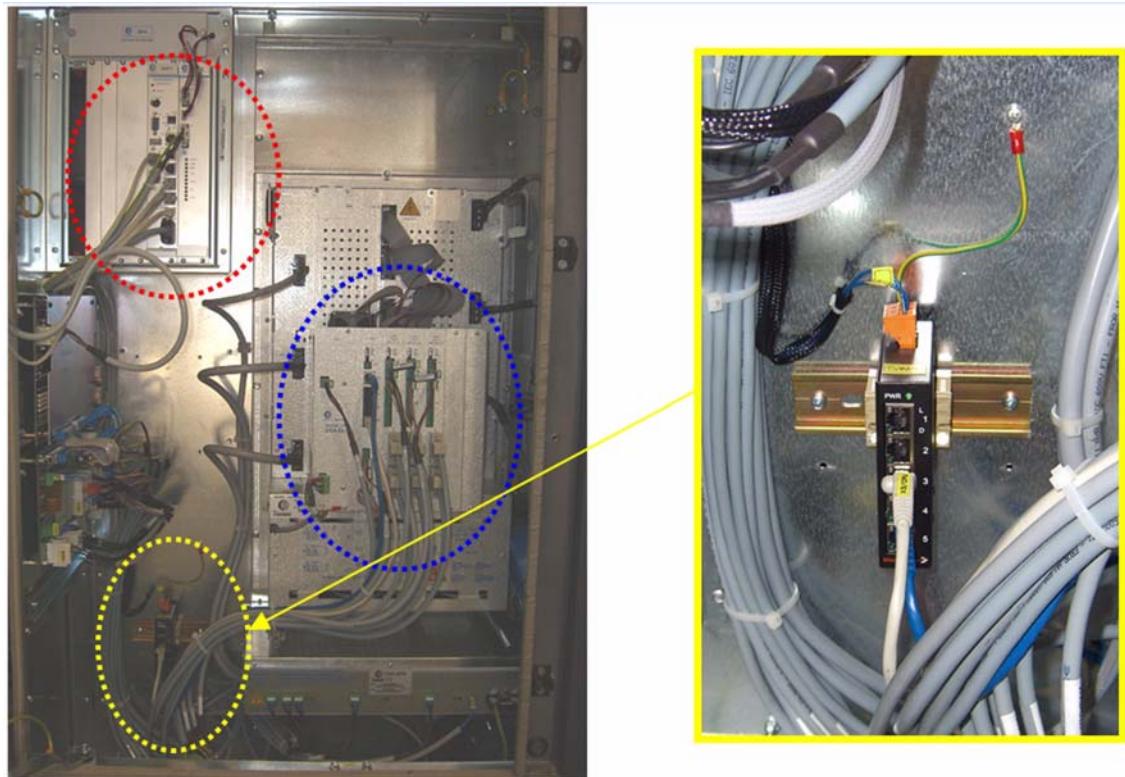
Il termine *last_cycle_time* rappresenta il tempo che impiega il PC per rispondere a SMP+ una volta ricevuto da questo il pacchetto di richiesta. Il tempo che passa dalla ricezione di una richiesta alla successiva è misurato dal termine *last_inter_step_time*. A meno di un jitter minimo (le operazioni sono cadenzate da interrupt hardware), i valori assunti da *last_inter_step_time* tendono a *inter_packet_period* che corrisponde al periodo di campionamento di SMP+.

23. SWITCH

In [Fig. 23.1](#) è riportata l'implementazione hardware dell'architettura funzionale dei blocchi presenti in [Fig. 5.1 - Controllore aperto](#).

Nella parte destra della figura è mostrato, nel dettaglio, lo switch standard utilizzato in COMAU per lo sviluppo del C4G OPEN.

Fig. 23.1 - Lo switch del C4G Open



Lo schema elettrico di tale switch è proposto in [Fig. 23.2](#).

Fig. 23.2 - Switch WEIDMULLER mod. IE-SW5-ECO (*)

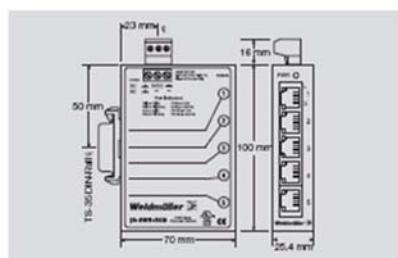
IE-SW5-ECO

5 Ports



Technical data

Housing	Aluminium
Ports	5xRJ45
AC Input voltage, min.-max.	8-24 V AC
DC Input voltage, min.-max.	10-36 V DC
AC Input power/DC Input power	5 VA AC/5 Watt DC
Input frequency	47 - 63 Hz
Operating temperature, min.-max.	0 °C ... 60 °C
Storage temperature, min.-max.	-40 °C ... 85 °C
Installation	DIN Rail Mount TS 35
Ingress Protection Class	IP 20
Standard	ANSI / IEEE 802.3
Data rate	10BASE-T or 100BASE-TX
Segment length	Copper Cable 100m max.
Functionality	Autonegotiation, Autocrossing
Flow control	semi-duplex/full duplex
Status indication	Power; Connection/Activity
Approvals	UL508/CE/EMC/EN 55024 und EN 55022
Aging	300 s
Length x width x height	mm 70 x 25.5 x 100



(*) - estratto dal documento **5652480000_Ethernet-Unmanaged_Switches.pdf**, liberamente scaricabile dal sito della WEIDMULLER.

24. MACCHINA A STATI

Il presente capitolo descrive in dettaglio la macchina a stati finiti relativa al sistema C4G Open.

Viene qui fornita la trattazione di tale argomento, secondo i seguenti paragrafi:

- Concetti generali
- Realizzazione
- Funzionamento.

24.1 Concetti generali

In Fig. 24.1 viene presentata la macchina a stati finiti che governa il funzionamento del C4G OPEN, evidenziando la gestione delle transizioni tra

- modalità aperta classica (controllo via PC),
- modalità chiusa (controllo standard via SMP+),
- modalità mista (controllo via PC e SMP+).

Gli stati sono definiti di seguito:

- S_{00}) sistema in DRIVE ON ma robot fermo (**STOP**) (tutti gli interpolatori fermi al nodo di default)
- S_{10}) robot in movimento imposto da SMP+ (**CLOSED**)
- S_{01}) robot in movimento imposto da PC (**OPEN TYPE A**)
- S_{11}) robot in movimento imposto da SMP+ e da PC (**OPEN TYPE B**)

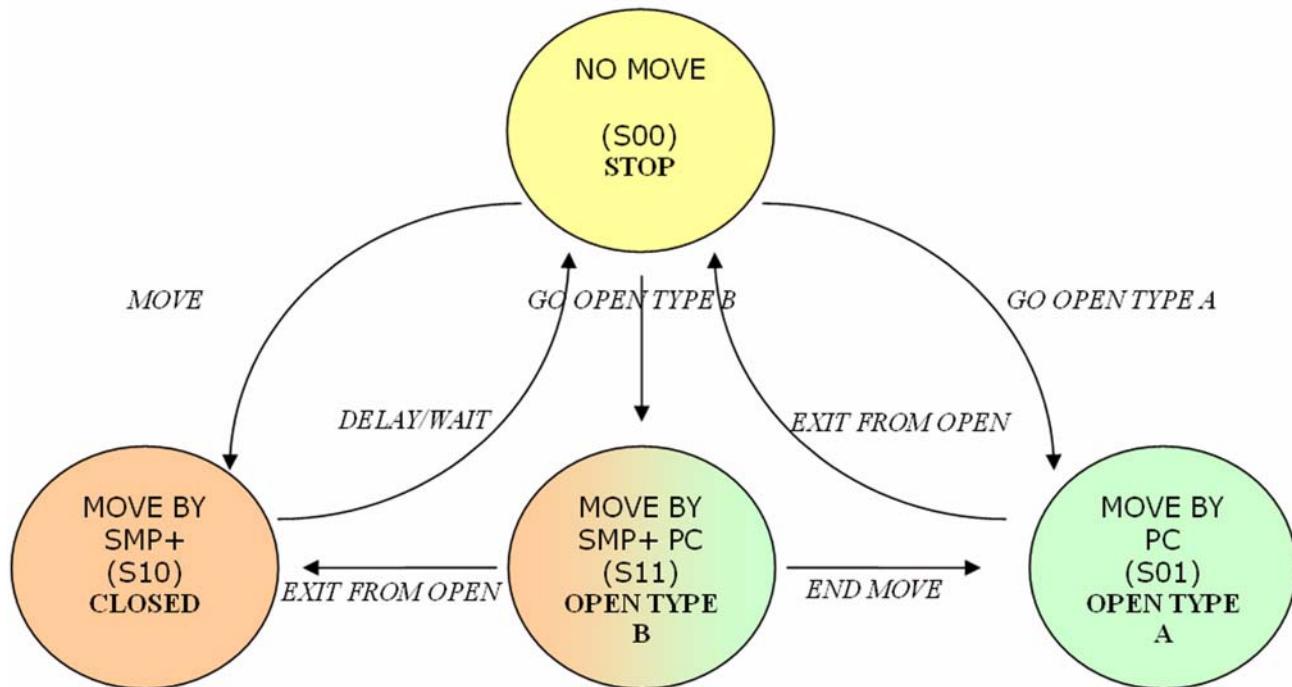
Le transizioni sono le seguenti:

- A_{0010}) **MOVE** via PDL2
- A_{1000}) **DELAY/WAIT FOR SIGNAL**, ...
- A_{0001}) **GO OPEN TYPE A** (modalità aperta senza alcuna traiettoria fornita da SMP+)
- A_{0100}) **EXIT FROM OPEN** (segnale proveniente da PC)
- A_{0011}) **GO OPEN TYPE B** (modalità aperta con traiettoria fornita anche da SMP+)
- A_{1100}) NA (occorrono due eventi distinti: fine traiettoria da SMP+ e fine attività da PC)
- A_{1011}) NA (modalità mista attivabile solo da fermo)
- A_{1110}) **EXIT FROM OPEN** (segnale proveniente da PC)
- A_{1101}) **END MOVE** (terminazione del movimento imposto da SMP+)
- A_{0111}) NA (modalità mista attivabile solo da fermo)
- A_{1001}) NA (modalità aperta attivabile solo da fermo)
- A_{0110}) NA (uscita dalla modalità aperta realizzabile solo da fermo).

Come si nota, alcune transizioni di stato non sono permesse. Ciò è dovuto alle

restrizioni imposte dal funzionamento del C4G OPEN che richiede di passare sempre dalla configurazione di quiete (macchina ferma in DRIVE ON) prima di cambiare la modalità operativa della macchina, come spiegato dettagliatamente nel [Cap.13. - Gestione dinamica della modalità aperta](#).

Fig. 24.1 - FSM che governa il C4G OPEN



24.2 Realizzazione

Quella riportata sopra è l'idea concettuale che sta alla base del funzionamento del C4G Open.

Nel presente paragrafo viene illustrata la realizzazione vera e propria della macchina a stati.

Innanzitutto, occorre introdurre le "entità software" che sono coinvolte nella implementazione del C4G Open:

- **Server PC** (nel seguito **SW_PC**): codice del server che gira su PC e che implementa tutte le azioni a carico del PC esterno,
- **Programma PDL2** (nel seguito **SW_PDL**): software che gira su C4G e che implementa tutte le istruzioni PDL2 per la gestione complessiva delle funzionalità del C4G Open,
- **Task SMP+** (nel seguito **SW_SMP**): task che gira su SMP+ e che implementa nel software di sistema il C4G Open.

Le tre entità software di cui sopra eseguono tutte o alcune delle seguenti operazioni:

- sincronizzazione tra i vari software,
- generazione della traiettoria,
- controllo dell'asse,

- cambiamento di una modalità operativa.

In dettaglio, con il termine sincronizzazione, in questo paragrafo si intende l'insieme dei segnali che permettono di sincronizzare tra loro le varie entità software:

- **SW_PC:**
 - *segnali spediti*: segnale EXIT_FROM_OPEN di uscita da una modalità;
 - *segnali ricevuti*: varie modalità aperte settate dalle altre entità software;
- **SW_PDL2:**
 - *segnali spediti*: routine di settaggio delle varie modalità aperte;
 - *segnali ricevuti*: ricezione di una "SIGNAL" su una "WAIT" associata ad un semaforo settato sulla **Modalità 0** (modalità settata automaticamente, tranne alcuni casi, da SW_SMP quando si riceve il segnale EXIT_FROM_OPEN mandato da SW_PC);
- **SW_SMP:**
 - *segnali spediti*: settaggio della **Modalità 0** (o altre) a seguito di un segnale EXIT_FROM_OPEN ricevuto da SW_PC;
 - *segnali ricevuti*: ricezione del segnale EXIT_FROM_OPEN mandato da SW_PC e delle varie modalità aperte settate da SW_PDL.

Per quanto riguarda la generazione della traiettoria, è importante sottolineare che l'entità software che, in un dato momento, fornisce la traiettoria, governa anche la sincronizzazione per le altre entità software.

In dettaglio:

- **SW_PC**: può generare la traiettoria in maniera autonoma (in **Modalità 1**, **Modalità 4**, **Modalità 5**) o insieme a SW_PDL2 (per esempio in **Modalità 7**);
- **SW_PDL2**: può, mediante l'istruzione MOVE, generare la traiettoria in maniera autonoma o insieme a SW_PC (in **Modalità 7**);
- **SW_SMP**: non genera mai la traiettoria, per cui non può mai sincronizzare le altre entità senza ricevere prima un segnale di sincronizzazione da una di esse.

Per quanto riguarda il controllo dell'asse, questo viene effettuato sempre dal software che gira su DSA, a meno che non si entri in **Modalità 1** o **Modalità 2**.

La **Modalità 1** permette a PC di generare anche la traiettoria, quindi, in questo caso il pieno controllo della sincronizzazione è affidato a SW_PC.

Viceversa, nella **Modalità 2**, il controllo dell'asse è affidato a SW_PC ma non la generazione della traiettoria. In questo ultimo caso, allora, la sincronizzazione delle operazioni è affidata a SW_PDL.

Per quanto riguarda, infine, il cambiamento di una modalità operativa, in generale, questa coinvolge tre fasi distinte:

- transitorio iniziale di ingresso alla modalità (**START**),
- esecuzione a "regime" della modalità (**CRUISE**),
- transitorio finale di uscita alla modalità (**END**).

Segue una descrizione dettagliata dei segnali di sincronizzazione che prevedono l'inizio e/o la fine di ognuna di queste fasi per le varie modalità operative.

- **Modalità 0**
- **Modalità 0'**
- **Modalità 1**
- **Modalità 2**

- Modalità 4
- Modalità 5
- Modalità 7
- Modalità 8
- Modalità 9.

24.2.1 Modalità 0

- START: inizia con il settaggio della modalità da parte di SW_PDL oppure dall'uscita dalle altre modalità operative secondo le tecniche descritte più avanti.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità viene eseguita attraverso il settaggio di una altra modalità da parte di SW_PDL.

24.2.2 Modalità 0'

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità viene eseguita attraverso il settaggio di una altra modalità da parte di SW_PDL.

24.2.3 Modalità 1

- START: inizia con il settaggio della modalità da parte di SW_PDL. In questa fase il controllo dell'asse passa dal DSA al PC esterno. Esiste un contatore nel SW_SMP che governa questo passaggio, questo viene incrementato fino ad un valore tale da rendere stabile il passaggio del controllo. Il SW_PC deve fornire una corrente tale da mantenere il robot fermo per un tempo correlato al contatore del SW_SMP per fare in modo che il passaggio del controllo avvenga a macchina ferma. Infatti, in questa modalità il SW_PC genera la traiettoria e, quindi, governa la sincronizzazione.
- CRUISE: inizia quando il contatore interno a SW_SMP raggiunge il valore massimo e quando SW_PC incomincia a generare una traiettoria che muove il robot.
- END: il transitorio in uscita viene stabilito da SW_PC, unico detentore della sincronizzazione in quanto generante la traiettoria, attraverso la spedizione del messaggio EXIT_FROM_OPEN. Ciò comporta che si passa nella modalità ausiliaria 1 STAND_BY durante la quale SW_PC deve fornire una corrente tale da mantenere fermo il robot, mentre SW_SMP incomincia ad effettuare il passaggio del controllo dal PC esterno al DSA decrementando il contatore che inizialmente aveva incrementato. Quando questo contatore raggiunge il valore nullo, si esce dalla modalità e si ritorna in modalità 0.

24.2.4 Modalità 2

- START: inizia con il settaggio della modalità da parte di SW_PDL. In questa fase il controllo dell'asse passa dal DSA al PC esterno. Esiste un contatore nel SW_SMP che governa questo passaggio, questo viene incrementato fino ad un valore tale da rendere stabile il passaggio del controllo. Il SW_PDL deve contenere un DELAY settato in maniera congrua con il contatore del SW_SMP per fare in modo che il passaggio del controllo avvenga a macchina ferma. Infatti, in questa modalità il SW_PDL genera la traiettoria e, quindi, governa la sincronizzazione.
- CRUISE: inizia quando il contatore interno a SW_SMP raggiunge il valore massimo e quando terminano i DELAY settati da SW_PDL che, quindi, incomincia a generare una traiettoria che muove il robot.
- END: il transitorio in uscita viene stabilito dal SW_PDL, unico detentore della sincronizzazione in quanto generante la traiettoria, attraverso il settaggio della modalità ausiliaria 2 STAND_BY. Durante questa modalità il SW_PDL si ferma su un DELAY settato in maniera simile a quello precedente permettendo il ritorno al pieno controllo dell'asse da parte del DSA. Il SW_SMP esegue il passaggio del controllo decrementando il contatore che inizialmente aveva incrementato. Quando questo contatore raggiunge il valore nullo, si esce dalla modalità e si ritorna in modalità 0.

24.2.5 Modalità 4

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità è eseguita mediante il segnale EXIT_FROM_OPEN fornito da SW_PC.

24.2.6 Modalità 5

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità è eseguita mediante il segnale EXIT_FROM_OPEN fornito da SW_PC.

24.2.7 Modalità 7

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità inizia attivando il segnale EXIT_FROM_OPEN fornito da SW_PC (in quanto, SW_PC può generare la traiettoria insieme a SW_PDL), si possono avere due comportamenti diversi:
 - se SW_PDL ha terminato il movimento previsto, il sistema entra subito nella modalità ausiliaria 7 STAND_BY_EXIT ed, infine, nella modalità 0,

- se SW_PDL non ha terminato il movimento previsto, il sistema passa per la modalità ausiliaria 7 STAND_BY. In questa modalità, SW_PC non fornisce più la traiettoria, per cui il pieno controllo di questa è, quindi, della sincronizzazione è affidato a SW_PDL. Appena il movimento previsto termina, il sistema entra nella modalità ausiliaria 7 STAND_BY_EXIT ed, infine, nella modalità 0.

24.2.8 Modalità 8

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità è eseguita mediante il segnale EXIT_FROM_OPEN fornito da SW_PC.

24.2.9 Modalità 9

- START: inizia con il settaggio della modalità da parte di SW_PDL.
- CRUISE: non esiste, in quanto, l'esecuzione a "regime" inizia appena si entra nella modalità.
- END: l'uscita dalla modalità è eseguita mediante il segnale EXIT_FROM_OPEN fornito da SW_PC.

24.3 Funzionamento

I concetti introdotti circa sincronizzazione, generazione della traiettoria, controllo dell'asse e cambiamento della modalità relativi alle tre entità software coinvolte nell'opzione C4G OPEN (server che gira sul PC esterno, programma di movimenti PDL2 e task che gira su SMP+), sono illustrati in modo schematico nelle figure seguenti ([Fig. 24.2](#), [Fig. 24.3](#), [Fig. 24.4](#)).

La prima mostra la macchina a stati che governa SW_SMP, la seconda le azioni che vengono eseguite da SW_PC e la terza i comandi impartiti da SW_PDL.



Si noti la scelta dei colori adottata:

- in **blu** sono indicati gli stati della macchina a stati del SW_SMP e gli eventi scaturiti dai contatori adottati da questo software per le sincronizzazioni,
- in **rosso** sono riportati i settaggi delle varie modalità ad opera del SW_PDL,
- in **nero** sono segnate le azioni eseguite dal SW_PC in funzione delle particolare modalità attiva in un dato momento,
- in **verde** sono indicati i segnali EXIT_FROM_OPEN generati dal SW_PC.

Fig. 24.2 - Macchina a stati vista da SW_SMP

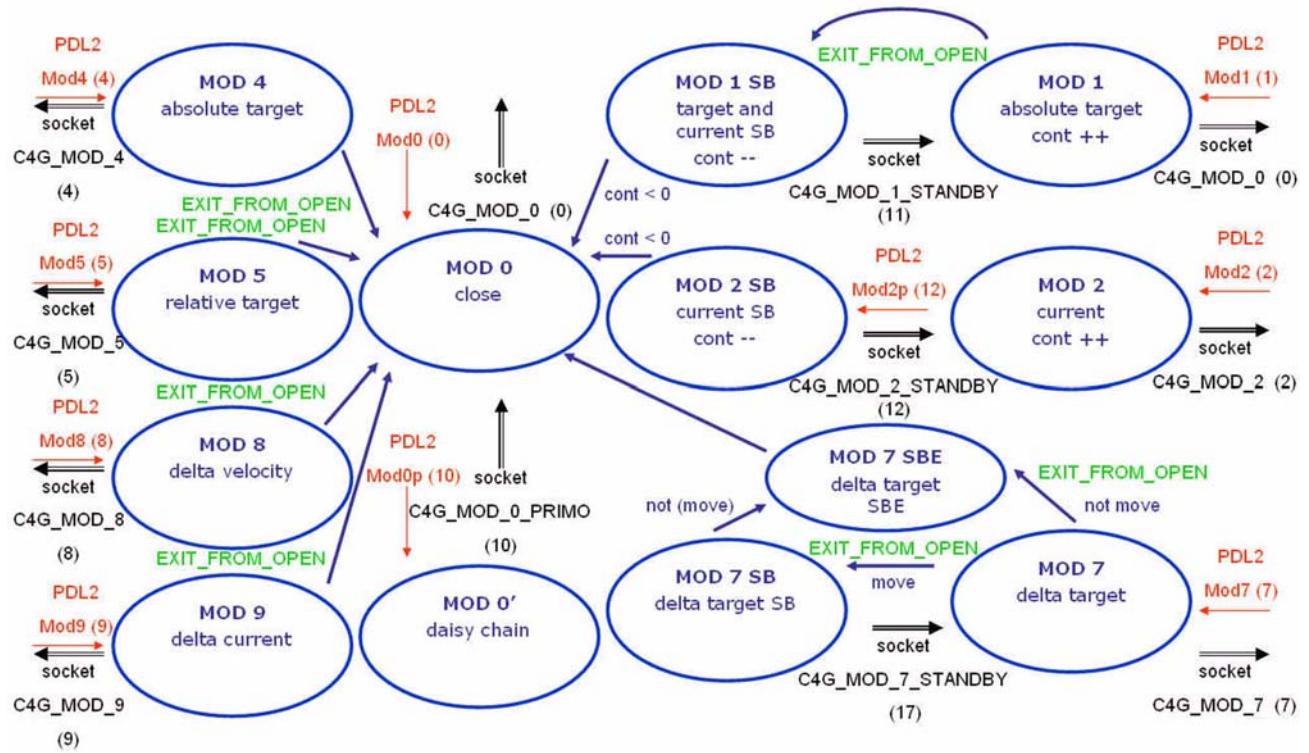


Fig. 24.3 - Azioni eseguite da SW_PC

```

switch(sx_C4GOpen_Packet_Rx.AX[0].SM)
- case C4G_MOD_0 (0): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_0
- replicazione del pacchetto;
- case C4G_MOD_0_PRIMO (10): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_0_PRIMO
- replicazione del pacchetto;
- case C4G_MOD_1 (1): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_1
- al primo passo si inizializzano gli stati e i parametri di controllo,
- in una prima fase viene generata una traiettoria nulla controllandola,
- successivamente viene generata una traiettoria non nulla controllandola e verificandone l'errore di inseguimento,
all'ultimo passo si crea il messaggio C4G_EXIT_FROM_OPEN generando una traiettoria nulla e controllandola;
- case C4G_MOD_1_STANDBY (11): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_1_STANDBY
- viene generata una traiettoria nulla controllandola;
- case C4G_MOD_2 (2): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_2
- al primo passo si inizializzano gli stati e i parametri di controllo,
- viene controllata la traiettoria;
- case C4G_MOD_2_STANDBY (12): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_2_STANDBY
- viene controllata la traiettoria;
- case C4G_MOD_4 (4): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_4
- al primo passo si inizializzano gli stati,
- vengono generati il riferimento di posizione e velocità,
- all'ultimo passo si crea il messaggio C4G_EXIT_FROM_OPEN e si mandano i riferimenti ricevuti;
- case C4G_MOD_5 (5): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_5
- al primo passo si inizializzano gli errori di inseguimento,
- in funzione di sx_C4GOpen_Packet_Rx.CMD si generano i riferimenti di velocità (e posizione) e si controlla l'errore di inseguimento,
- all'ultimo passo si eseguono una serie di campioni con un riferimento fisso (l'ultimo) in funzione di sx_C4GOpen_Packet_Rx.CMD,
- finita la serie di campioni si crea il messaggio C4G_EXIT_FROM_OPEN e si rimanda il pacchetto ricevuto;
- case C4G_MOD_7 (7): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_7
- in una prima fase viene fornito il delta target,
- successivamente si crea il messaggio C4G_EXIT_FROM_OPEN fornendo delta target nullo;
- case C4G_MOD_7_STANDBY (17): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_7_STANDBY
- si fornisce un delta target nullo;
- case C4G_MOD_7_STANDBY_EXIT (170): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_7_STANDBY_EXIT
- si forniscono riferimenti nulli;
- case C4G_MOD_8 (8): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_8
- al primo passo si inizializzano i parametri necessari,
- in una prima fase si fornisce il delta velocità,
- successivamente si crea il messaggio C4G_EXIT_FROM_OPEN e si forniscono riferimenti nulli;
- case C4G_MOD_9 (9): sx_C4GOpen_Packet_Tx.AX[AX_OPEN].SM = C4G_MOD_9
- al primo passo si inizializzano i parametri necessari,
- in una prima fase si fornisce il delta corrente,
- successivamente si crea il messaggio C4G_EXIT_FROM_OPEN e si forniscono riferimenti nulli;
- default:
- nessuna operazione viene eseguita.

```

Fig. 24.4 - Comandi forniti da SW_PDL

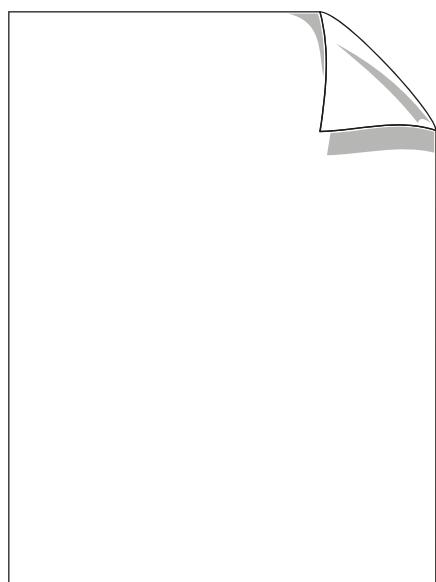
```

-- Modalità 0
setta_modalita(mod0)
MOVE APPROCCIO(ind_stato)
DELAY time_intermod
-- Modalità 0p
setta_modalita(mod0p)
MOVE APPROCCIO(ind_stato)
DELAY time_intermod
-- Modalità 1
setta_modalita(mod1)
DELAY time_mod1_input -- transitorio in ingresso
DELAY time_mod1_change -- saturazione variabile
DELAY time_mod1_output -- transitorio in uscita
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
WAIT FOR segnale_go
DELAY time_intermod
-- Modalità 2/2p
setta_modalita(mod2)
DELAY time_mod2_input -- transitorio in ingresso
DELAY time_mod2_change -- saturazione variabile
DELAY time_mod2_output -- transitorio in uscita
DELAY time_premove
MOVE (nulla o significativa)
setta_modalita(mod12)
DELAY time_mod2_input -- transitorio in uscita
DELAY time_mod2_change -- saturazione variabile
DELAY time_mod2_output -- transitorio in ingresso
DELAY time_intermod

-- Modalità 4
MOVE APPROCCIO
DELAY time_premove
setta_modalita(mod4)
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
WAIT FOR segnale_go
DELAY time_intermod
-- Modalità 5
$ARM_DATA[ind_arm].C4GOPEN_CMD['] := ind_stato
setta_modalita(mod5)
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
WAIT FOR segnale_go
DELAY time_intermod
-- Modalità 7
setta_modalita(mod7)
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
MOVE (lunga o breve)
WAIT FOR segnale_go
DELAY time_intermod

-- Modalità 8
setta_modalita(mod8)
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
MOVE GENERICA
WAIT FOR segnale_go
DELAY time_intermod
-- Modalità 9
setta_modalita(mod9)
segnale_go := FALSE
ENABLE CONDITION[cond_mod0]
MOVE GENERICA
WAIT FOR segnale_go
DELAY time_intermod

```



25. STRATI SOFTWARE

Per il C4G OPEN è prevista la realizzazione e la gestione dei seguenti strati software/hardware sul PC remoto collegato al C4G riportata in [Tab. 25.1](#).

Tab. 25.1 - Strati software/hardware sul PC remoto

4	Applicazioni	programma applicativo
3	Funzioni	funzioni base FSM pacchetti C4G OPEN
2	Comunicazione	API per le funzioni da fornire
1	Sistema Operativo	driver di comunicazione
0	Hardware	Linux + librerie RTAI

Per la natura real time del C4G OPEN il sistema operativo che gira su PC remoto deve essere anch'esso di tipo real time. Per questo è stato scelto di utilizzare il sistema operativo Linux compilato con librerie RTAI dedicate all'elaborazione real time.

In ambiente Linux devono, poi, essere sviluppati i driver di comunicazione UDP/IP che permettono lo scambio di pacchetti tra SMP+ e PC.

Dal momento che si utilizza una distribuzione open source di Linux, anche le librerie di comunicazione sono distribuite in modalità open.

Al di sopra dello strato di comunicazione, c'è il livello delle funzioni.

Questo livello può essere suddiviso in tre sottolivelli dal più alto al più basso:

- Funzionalità di base,
- Macchina a stati finiti (FSM),
- Gestione dei pacchetti C4G OPEN.

Il sottolivello che si occupa della gestione dei pacchetti C4G OPEN, ha il compito di creare, spedire e ricevere i pacchetti C4G OPEN lato PC.

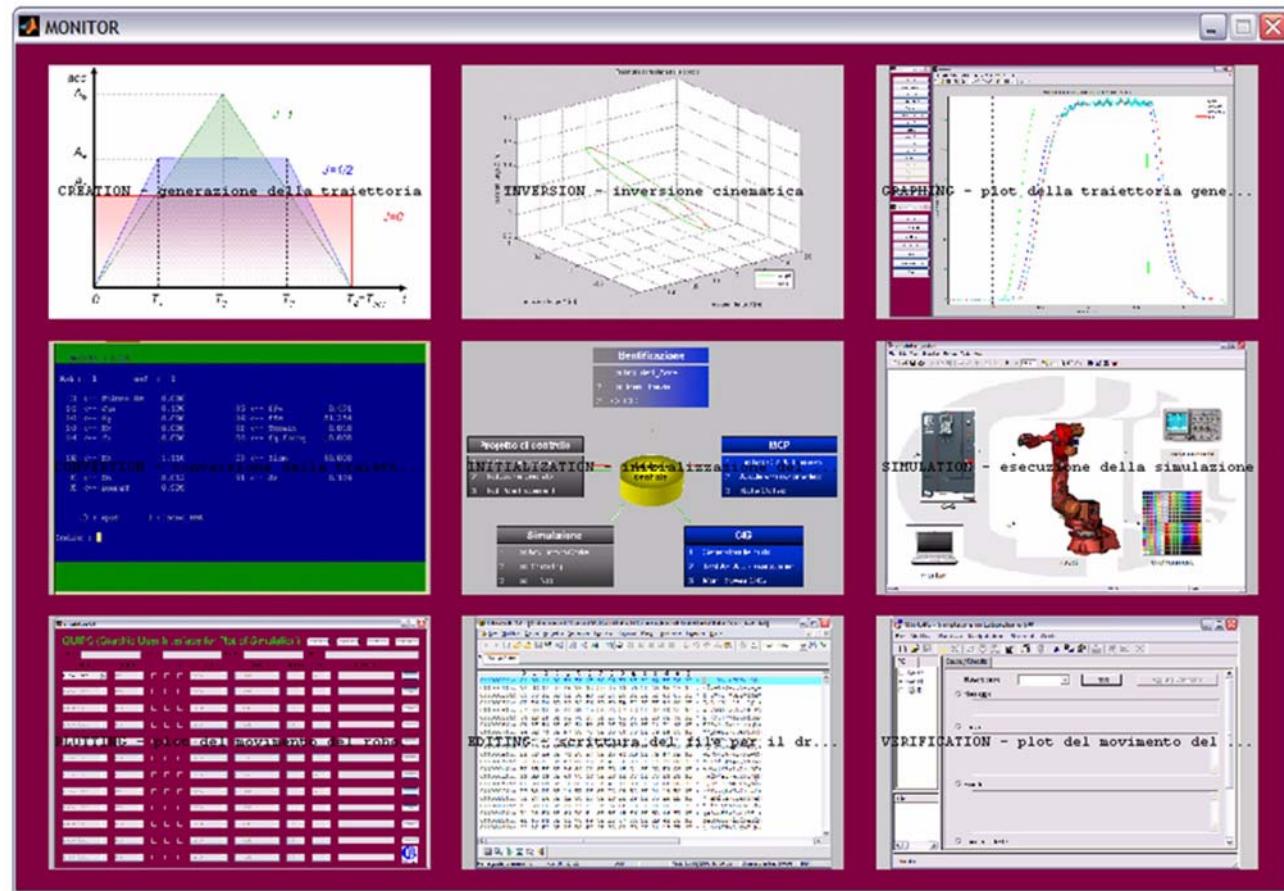
Lato SMP+ svolge le stesse funzioni ma anche quella di filtrare il [BLIP](#) e il [RAAZ](#) e reperire/inserire in questi pacchetti le informazioni da riversare/fornire ai pacchetti C4G OPEN verso PC secondo quanto spiegato nel [Cap.9](#).

La macchina a stati finiti, invece, si occupa di gestire le modalità OPEN e il passaggio da una modalità operativa ad un'altra.

Le funzionalità di base, invece, sono una serie di primitive di base che devono essere fornite ai livelli software superiori.

A questo livello devono essere presenti le procedure che permettono la gestione degli assi per i quali è prevista la modalità aperta, che scelgono il tipo di modalità aperta specifica, che schedulano gli algoritmi di generazione della traiettoria o di controllo del moto da eseguire in funzione delle richieste provenienti da SMP+, che ricevono le informazioni provenienti dai sensori standard e così via. Questo strato software deve prevedere anche la procedura di HOLD da PC a seguito del comando imposto da SMP+ e la gestione ridondante del following error dal PC.

Fig. 25.1 - Esempio di software applicativo per C4G OPEN



Il progetto del sistema C4G OPEN prevede come obiettivo principale quello di poter utilizzare (nel suo significato più generale possibile) un robot COMAU mediante comandi impartiti da PC.

In questo senso, il livello applicativo può essere sviluppato da ogni singolo utente secondo le sue particolari necessità.

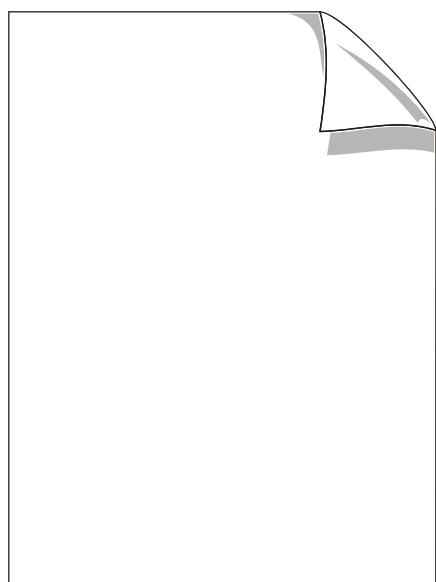
D'altro canto, per verificare che tutta la pila di strati software funzionino, occorre realizzare un software applicativo di base che, seppur con funzionalità limitate, permetta di usare in maniera facile ed immediata l'intero sistema.

In Fig. 25.1 è mostrata, a titolo di esempio, l'interfaccia grafica realizzata in MatLab/SimuLink di un possibile programma applicativo che utilizza il C4G OPEN per pilotare un robot mediante dei target calcolati in precedenza su PC.

L'esempio è utile per capire quali debbano essere le funzionalità che un tale programma applicativo deve svolgere:

- **generazione di traiettoria:** un modulo che, in funzione della caratterizzazione del robot, carica un file contenente i movimenti che deve eseguire la macchina e genera la sequenza di punti più appropriata per eseguire la traiettoria richiesta;
- **strumento di simulazione:** un modulo che, conoscendo la caratterizzazione del robot e le condizioni al contorno quali il carico presente alla flangia e il tipo di montaggio (a pavimento, soffitto o inclinato), simula il comportamento del robot durante la traiettoria calcolata dal modulo precedente;
- **programma di traduzione:** un modulo che traduce la traiettoria calcolata in precedenza dal formato proprio del PC a quello compatibile con il controllo C4G;

- **console di comando:** un modulo che permette di tenere sotto controllo ciò che avviene sul C4G, in particolare questo modulo deve poter fornire le seguenti funzionalità:
 - **visualizzazione:** permette di esaminare lo stato del sistema eventualmente mostrando graficamente il movimento del manipolatore ricostruito sulla base delle misure reali delle posizioni degli encoder;
 - **controllo:** permette di far eseguire semplici compiti al robot (esempio piccoli spostamenti dei giunti del robot) impartendo i comandi mediante l'interfaccia grafica.



26. SOFTWARE SU PC

26.1 Introduzione

Nel presente capitolo sono descritti i seguenti argomenti:

- [Distribuzione Fedora](#)
- [Distribuzione Ubuntu](#).



ATTENZIONE - Insieme all'opzione C4GOpen vengono rilasciati:

- la libreria “C4G Open Library for Linux and OrchestraC4GLibrary v1.17.00” per l’interfacciamento del C4GOpen sotto LINUX,
- un esempio di applicazione dimostrativa sviluppata per Robot SIX (vd. directory “C4G Open Example”).

Tutto ciò è distribuito a puro titolo dimostrativo e per offrire una base allo sviluppo di nuove applicazioni.

Comau non si assume alcuna responsabilità circa il funzionamento di applicazioni che l’utente sviluppi a partire da tali esempi.



NOTA - Per architetture con chip AMD e/o a 64 bit, le librerie C4G Open non sono attualmente utilizzabili a causa dei limiti della distribuzione RTai.

Il funzionamento del software C4GOpen e delle librerie ad esso collegate, è garantito per chip set INTEL Pentium a 32 bit.

26.2 Distribuzione Fedora

In questo capitolo sono riportate alcune note per la configurazione software del PC esterno che comunica con SMP+.

I dettagli sono riportati nel documento **C4G Open Setup HW & SW "lato PC"** (vd. [Tab. 27.1 - Documenti di riferimento](#)).

[Distribuzione di Linux Fedora Core 4:](#)

- <http://fedora.redhat.com/>
- <http://redhat.download.fedoraproject.org/pub/fedora/linux/core/4/i386/>
I file ISO da scaricare e poi masterizzare su 4 CD:
 - FC4-i386-disc1.iso,
 - FC4-i386-disc2.iso,
 - FC4-i386-disc3.iso,
 - FC4-i386-

[Kernel vanilla 2.6.16:](#)

- <http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.16.tar.bz2>

RTAI 3.4:

- <https://www.rtai.org/RTAI/rtai-3.4.tar.bz2>
 script di caricamento:
`#!/bin/sh`

```
MOD_DIR=/usr/realtme/modules
```

```
sync
make -f /usr/src/rtai/GNUmakefile dev
insmod ${MOD_DIR}/rtai_hal.ko
insmod ${MOD_DIR}/rtai_sched.ko
insmod ${MOD_DIR}/rtai_fifos.ko
insmod ${MOD_DIR}/rtai_sem.ko
insmod ${MOD_DIR}/rtai_mbx.ko
insmod ${MOD_DIR}/rtai_msg.ko
insmod ${MOD_DIR}/rtai_netrpc.ko ThisNode="127.0.0.1"
insmod ${MOD_DIR}/rtai_math.ko
sync
```

RTnet 0.97:

- <http://www.rts.uni-hannover.de/rtnet/download/rtnet-0.9.7.tar.bz2>
 script di caricamento:
`#!/bin/sh`

```
MOD_DIR=/usr/local/rtnet/modules
```

```
sync
mknod /dev/rtnet c 10 240
insmod /usr/realtme/modules/rtai_rtdm.ko
insmod ${MOD_DIR}/rtnet.ko
insmod ${MOD_DIR}/rtipv4.ko
insmod ${MOD_DIR}/rtpacket.ko
insmod ${MOD_DIR}/rt_loopback.ko
insmod ${MOD_DIR}/rt_eopro100.ko
insmod ${MOD_DIR}/rtcap.ko
/usr/local/rtnet/sbin/rtifconfig rtlo up 127.0.0.1
/usr/local/rtnet/sbin/rtifconfig rteth0 up 10.2.12.149
/usr/local/rtnet/sbin/rtroute solicit 10.2.12.150 dev rteth0
sync
```

Compilatore:

```
[c4gopen@localhost ~]$ gcc -v
Using built-in specs.
Target: i386-redhat-linux
Configured with: ..../configure --prefix=/usr
--infodir=/usr/share/info --enable-shared
--enable-checking=release --with-system-zlib
--disable-libunwind-exceptions
--enable-languages=c,c++,objc,java,f95,ada
--with-java-home=/usr/lib/jvm/java-1.4.2-gcj-1.4.2.0/jre --host=i386-redhat-linux
Thread model: posix
gcc version 4.0.0 20050519 (Red Hat 4.0.0-8)
```

Indicazione iniziale a video: Fedora Core 2.6.11-1.1369_FC4

Indicazione finale a video: Fedora Core 2.6.16-rtai.

26.3 Distribuzione Ubuntu

Vengono ora descritti dettagliatamente i seguenti argomenti al riguardo:

- [Requisiti Software](#)
- [Requisiti Hardware](#)
- [Schede di rete compatibili](#)
- [Installazione del Kernel](#)
- [Caricamento del modulo RTai](#)
- [Caricamento del modulo RTNet](#)
- [Esecuzione dell'applicazione C4G Open.](#)

26.3.1 Requisiti Software

- **Ubuntu 7.04 "The Feisty Fawn" CD x86 edition:** Ubuntu Website - <http://www.ubuntu.com>
- **Linux vanilla kernel 2.6.19:** The Linux Kernel Archives - <http://www.kernel.org>
- **RTAI 3.5:** RTAI Website - <http://www.rtai.org>
- **RTnet 0.9.9:** RTnet Website - <http://www.rtnet.org>
- **GNU g++ 4.1.2:** The GNU Project - <http://www.gnu.org>
- **GNU bash 3.2.13(1):** The GNU Project - <http://www.gnu.org>

Indicazione iniziale a video: Ubuntu Kernel 2.6.20_15_generic

Indicazione finale a video: Ubuntu Kernel 2.6.19_rtai_generic.

26.3.2 Requisiti Hardware

- **un PC con processore x86-compatibile** almeno di classe Pentium
- **SOLO architettura 32 bit**
- **tre cavi di rete UTP cat.5 o superiore**
- **uno switch Fast Ethernet**
- **una scheda di rete RTnet-compliant** (10/100 Mbps) (vedere il sottoparagrafo successivo).

26.3.3 Schede di rete compatibili

Per far funzionare il C4G Open occorre che la scheda di rete sia predisposta per la comunicazione hard real-time tra il PC esterno e il C4G.

Per sapere se la propria scheda di rete permette la comunicazione hard real-time, la distribuzione Ubuntu fornisce le informazioni sul NIC, inclusi il modello del chipset e delle IRQ utilizzando il seguente comando

```
lspci -v
```

Ecco un esempio dell'output del comando su una delle schede di rete testate dall'azienda *Sintesi Scpa* di Bari che collabora con COMAU nello sviluppo del software per il PC, nel progetto C4G Open:

```
00:0b.0 Ethernet controller: Digital Equipment Corporation DECchip 21140 [FasterNet] (rev 22)
  Subsystem: D-Link System Inc DFE-500TX Fast Ethernet
  Flags: bus master, medium devsel, latency 32, IRQ 10
  I/O ports at c800 [size=128]
  Memory at e8143000 (32-bit, non-prefetchable) [size=128]
    [virtual] Expansion ROM at 30000000 [disabled] [size=256K]
```

Si possono ricavare ulteriori informazioni circa la configurazione hardware utilizzando l'utility *hwinfo* messa a disposizione dal pacchetto *Synaptic Package Manager* (richiamabile dal comando *sudo synaptic*).

In particolare, per ricavare le informazioni di tutti i dispositivi PCI installati, può essere utile immettere il seguente comando

```
hwinfo --pci
```



NOTA - nel caso in cui il suddetto comando non venga eseguito correttamente, provvedere ad eseguire il passo C, descritto al par. 26.3.4 Installazione del Kernel a pag. 26-8.

Nella figura seguente, l'output dell'esecuzione del comando su un hardware testato.

Si noti che si ottengono anche le informazioni sul nome del driver Linux standard dei vari dispositivi:

```
19: PCI 0b.0: 0200 Ethernet controller
[Created at pci.281]
UDI: /org/freedesktop/Hal/devices/pci_1011_9
Unique ID: JNkJ.AujCtKsDPG3
SysFS ID: /devices/pci0000:00/0000:00:0b.0
SysFS BusID: 0000:00:0b.0
Hardware Class: network
Model: "D-Link DFE-500TX Fast Ethernet"
Vendor: pci 0x1011 "Digital Equipment Corporation"
Device: pci 0x0009 "DECchip 21140 [FasterNet]"
SubVendor: pci 0x1186 "D-Link System Inc"
SubDevice: pci 0x1100 "DFE-500TX Fast Ethernet"
Revision: 0x22
Driver: "tulip"
Driver Modules: "tulip"
Device File: eth1
I/O Ports: 0xc800–0xc87f (rw)
Memory Range: 0xe8143000–0xe814307f (rw,non-prefetchable)
Memory Range: 0x30000000–0x3003ffff (ro,prefetchable,disabled)
IRQ: 10 (8 events)
HW Address: 00:40:05:37:36:e4
Module Alias: "pci:v00001011d00000009sv00001186sd00001100bc02sc00i00"
Driver Info #0:
    Driver Status: tulip is active
    Driver Activation Cmd: "modprobe tulip"
Config Status: cfg=new, avail=yes, need=no, active=unknown
```

È importante accertarsi che la IRQ utilizzata dal NIC chipset non sia condivisa con altri dispositivi per non pregiudicare la comunicazione hard real-time del sistema.
Per verificare ciò basta digitare il comando *cat /proc/interrupts*.

In caso di condivisione della IRQ bisogna cercare di modificare lo slot PCI della propria scheda di rete o disabilitare gli altri device o rimuovere i moduli del kernel associati a questi ultimi.

Nella figura seguente, l'output del comando *cat* sull'hardware testato:

0:	291251	XT-PIC-XT	timer
1:	2	XT-PIC-XT	i8042
2:	0	XT-PIC-XT	cascade
3:	119	XT-PIC-XT	SiS SI7012
5:	16674	XT-PIC-XT	eth0
6:	5	XT-PIC-XT	floppy
7:	1	XT-PIC-XT	parport0
8:	3	XT-PIC-XT	rtc
9:	0	XT-PIC-XT	ehci_hcd:usb2
10:	8	XT-PIC-XT	ohci_hcd:usb1, eth1
11:	67755	XT-PIC-XT	ohci_hcd:usb3, radeon@pci:0000:01:00.0
14:	27017	XT-PIC-XT	ide0

Per esempio, se la scheda di rete condivide la linea IRQ 10 con il modulo *ohci_hcd*, occorre rimuoverla digitando il seguente comando da ROOT o in modalità SUDO:

```
rmmmod ohci_hcd
```



Occorrerebbe utilizzare unicamente mouse e tastiera PS/2 per evitare che i dispositivi USB vadano in conflitti con il canale IRQ della scheda.

Le tabelle che seguono, contengono la lista delle schede compatibili RTNet hard real-time al momento in cui viene scritto questo documento:

- [Tab. 26.1 - Schede compatibili RTNet hard real-time \(1\)](#)
- [Tab. 26.2 - Schede compatibili RTNet hard real-time \(2\)](#)

Tab. 26.1 - Schede compatibili RTNet hard real-time (1)

RTnet driver name	Vendor ID	Device ID	Manufacturer	Chip number
rt_tulip	0x1011	0x0002	DEC	DC21040
	0x1011	0x0014	DEC	DC21041
	0x1011	0x0009	DEC	DC21140
	0x1011	0x0019	DEC	DC21142/3
	0x11ad	0x0002	Lite-On Communications	NGMC169B
	0x11ad	0xe115	Lite-On Communications	LC82C115
	0x10d9	0x0512	Macronix	MX98713
	0x10d9	0x0531	Macronix	MX98715/25
	0x1317	0x0981	ADMtek	AN981
	0x1317	0x0985	ADMtek	AN983B
	0x1317	0x1985	ADMtek	AN985
	0x1317	0x9511	ADMtek	ADM9511
	0x104a	0x0981	STMicroelectronics	21x4x
	0x104a	0x2774	STMicroelectronics	21x4x
	0x11f6	0x9881	Powermatic	TXA9881
	0x1282	0x9102	Davicom	DM9102/A/AF
	0x1113	0x1216	Accton	EN5251BE
	0x1113	0x1217	Accton	EN2242
	0x1113	0x9511	Accton	SMC EN5251BE
rt_3c59x (experimental)	0x10b7	0x5900	3Com	3C590
	0x10b7	0x5920	3Com	3C592
	0x10b7	0x5970	3Com	3C597
	0x10b7	0x5950	3Com	3C595
	0x10b7	0x5951	3Com	3C595
	0x10b7	0x5952	3Com	3C595
	0x10b7	0x9000	3Com	3C900
	0x10b7	0x9001	3Com	3C900
	0x10b7	0x9004	3Com	3C900B-TPO
	0x10b7	0x9005	3Com	3C900B-Combo
	0x10b7	0x9006	3Com	3C900B-TPC
	0x10b7	0x900a	3Com	3C900B-FL
	0x10b7	0x9050	3Com	3C905
	0x10b7	0x9051	3Com	3C905
	0x10b7	0x9055	3Com	3C905B
	0x10b7	0x9058	3Com	3C905B
	0x10b7	0x905a	3Com	3C905B-FX
	0x10b7	0x9200	3Com	3C905C-TX/TX-M
	0x10b7	0x9800	3Com	3C980-TX
	0x10b7	0x9805	3Com	3C980-C
	0x10b7	0x7646	3Com	3CSOHO100-TX
	0x10b7	0x5055	3Com	3C555
	0x10b7	0x6055	3Com	3C556
	0x10b7	0x6056	3Com	3C556B
	0x10b7	0x5b57	3Com	3C595
	0x10b7	0x5057	3Com	3C575
	0x10b7	0x5157	3Com	3CCFE575BT
	0x10b7	0x5257	3Com	3CCFE575CT
	0x10b7	0x6560	3Com	3CCFE656
	0x10b7	0x6562	3Com	3CCFEM656B
	0x10b7	0x6564	3Com	3CXFEM656C
	0x10b7	0x4500	3Com	3C450

Tab. 26.2 - Schede compatibili RTNet hard real-time (2)

RTnet driver name	Vendor ID	Device ID	Manufacturer	Chip number
rt_8139too	0x10ec	0x8139	RealTek	RTL8139/8139C/8139C+
	0x10ec	0x8138	RealTek	RTL8139 (B/C)
	0x4033	0x1360	Delta Networks	RTL8139
	0x1186	0x1300	D-Link	RTL8139
	0x1186	0x1340	D-Link	DFE-690TXD
	0x13d1	0xab06	AboCom	RTL8139
	0x1259	0xa117	Allied Tsyn	RTL81xx
rt_eopro100	0x8086	0x1229	Intel	82557/8/9/0/1
	0x8086	0x1209	Intel	8255xER/IT
	0x8086	0x1029	Intel	82559
	0x8086	0x1030	Intel	825593
	0x8086	0x1031	Intel	82801CAM
	0x8086	0x1032	Intel	82801CAM
	0x8086	0x1033	Intel	82801CAM
	0x8086	0x1034	Intel	82801CAM
	0x8086	0x1035	Intel	82562EH
	0x8086	0x1036	Intel	82562EH
	0x8086	0x1037	Intel	82801CAM
	0x8086	0x1038	Intel	82801CAM
	0x8086	0x1039	Intel	82801DB
	0x8086	0x103A	Intel	82801DB
	0x8086	0x103B	Intel	82801DB
	0x8086	0x103C	Intel	82801DB
	0x8086	0x103D	Intel	82801DB
	0x8086	0x103E	Intel	82801DB
	0x8086	0x1227	Intel	82865g
	0x8086	0x1228	Intel	82556
	0x8086	0x2449	Intel	82559ER
	0x8086	0x2459	Intel	82801E
	0x8086	0x245D	Intel	82801E
	0x8086	0x5200	Intel	EE PRO/100
rt_natsemi	0x100b	0x0001	National Semiconductors	DP83810
	0x100b	0x0020	National Semiconductors	DP83815/16

26.3.4 Installazione del Kernel

Dopo che è stata installata la versione 7.04 di Ubuntu, l'installazione del Kernel prevede i seguenti passi:

a. passo A

- sudo dpkg -i linux-headers-2.6.19-rtai-generic_r1_i386.deb (RTAI Kernel)
- sudo dpkg -i linux-source-2.6.19-rtai-generic_r1_all.deb (RTAI Kernel Source)
- sudo dpkg -i linux-image-2.6.19-rtai-generic_r1_i386.deb (RTAI Kernel)
- sudo dpkg -i libfreetype6_2.2.1-5ubuntu1.1_i386.deb
- sudo dpkg -i libgl1-mesa-glx_6.5.2-3ubuntu8_i386.deb
- sudo dpkg -i libgl1-mesa-dri_6.5.2-3ubuntu8_i386.deb
- sudo dpkg -i libglu1-mesa_6.5.2-3ubuntu8_i386.deb

b. passo B

- tar xjvf rtai-base_1.0-3_i386_full.tar.bz2
- sudo dpkg -i rtai-base_1.0-3_i386_full/*.deb (RTAI Base)

c. passo C

- sudo dpkg -i libhd13_13.11-3_i386.deb (Informazioni dell'hardware)
- sudo dpkg -i hwinfo_13.11-3_i386.deb (Informazioni dell'hardware)

- d. **passo D**
 - sudo reboot (scegliendo ubuntu-rtai)
- e. **passo E**
 - cd /usr/src/rtai-base
 - sudo make
 - sudo make upgrade

26.3.5 Caricamento del modulo RTai

Il primo passo nell'uso del C4G Open consiste nel caricamento del modulo RTai, in maniera da fornire al computer le capacità real-time di elaborazione. Occorre eseguire lo script loadRTAI (riportato qui sotto) che si trova nella directory **/usr/local/C4Gopen/bin**, eseguendo il comando da ROOT o in modalità SUDO: ./loadRTAI

```
#!/bin/bash

RTAI_SRC_DIR=/usr/src/rtai
RTAI_MODULES_DIR=/usr/realtime/modules

sync
make -f ${RTAI_SRC_DIR}/GNUmakefile dev
insmod ${RTAI_MODULES_DIR}/rtai_hal.ko
insmod ${RTAI_MODULES_DIR}/rtai_lxrt.ko
insmod ${RTAI_MODULES_DIR}/rtai_fifo.ko
insmod ${RTAI_MODULES_DIR}/rtai_sem.ko
insmod ${RTAI_MODULES_DIR}/rtai_mbx.ko
insmod ${RTAI_MODULES_DIR}/rtai_msg.ko
insmod ${RTAI_MODULES_DIR}/rtai_rtdm.ko
insmod ${RTAI_MODULES_DIR}/rtai_netrpc.k
ThisNode="127.0.0.1"
sync
```



Il caricamento del modulo RTAI deve essere eseguito ad ogni riavvio del computer.

26.3.6 Caricamento del modulo RTNet

Allo scopo di **fornire al PC i requisiti hard real-time per la comunicazione**, occorre caricare i moduli del kernel RTNet e configurare opportunamente la sottorete per il C4G Open.

Bisogna sostituire lo script loadRTnet che si trova nella directory **/usr/local/C4Gopen/bin**, con le istruzioni sotto riportate, dopo aver inserito il nome corretto dei driver Linux e RTNet (esempio in **rosso**). Per un corretto utilizzo delle istruzioni sottoelencate, seguire le **NOTE**:

```
#!/bin/bash

RTAI_MOD_DIR=/usr/realtime/modules
```

```

RTNET_MOD_DIR=/usr/local/rtnet/modules
RTNET_BIN_DIR=/usr/local/rtnet/sbin

ETH_DRV=linux_driver_name (e.g. 8139too <- to remove (*))
RT_ETH_DRV=rtnet_driver_name(e.g. rt_8139too <- to remove(*) )

LOCAL_IP=10.2.12.149
TARGET_IP=10.2.12.150

mknod /dev/rtnet c 10 240

rmmod ${ETH_DRV}

insmod ${RTNET_MOD_DIR}/rtnet.ko
insmod ${RTNET_MOD_DIR}/rtipv4.ko
insmod ${RTNET_MOD_DIR}/rtpacket.ko
insmod ${RTNET_MOD_DIR}/rt_loopback.ko
insmod ${RTNET_MOD_DIR}/${RT_ETH_DRV}.ko

${RTNET_BIN_DIR}/rtifconfig rtlo up 127.0.0.1
${RTNET_BIN_DIR}/rtifconfig rteth0 up ${LOCAL_IP}

${RTNET_BIN_DIR}/rtroute solicit ${TARGET_IP} dev rteth0

```

**NOTE**

- (*) - l'esempio in rosso deve essere cancellato prima di utilizzare le istruzioni sopraelencate
- Si noti che la variabile ETH_DRV si riferisce al nome del modulo del kernel del driver della propria scheda di rete, mentre, la variabile RT_ETH_DRV si riferisce al nome del modulo del kernel RTnet del driver hard real-time della propria scheda di rete.

Come già detto, si veda il [par. 26.3.2 Requisiti Hardware a pag. 26-3](#) per reperire le informazioni sul NIC installato sul PC e per individuare il nome del driver della propria scheda di rete.

Per il nome del driver RTnet, ci si può riferire alle precedenti [Tab. 26.1 - Schede compatibili RTNet hard real-time \(1\)](#) e [Tab. 26.2 - Schede compatibili RTNet hard real-time \(2\)](#).

Ad esempio, se il proprio chipset è RTL8139, il nome del driver standard Linux è 8139too, mentre il driver RTNet è rt_8139too, per cui le istruzioni da inserire saranno:

```

ETH_DRV=8139too
RT_ETH_DRV=rt_8139too

```

- Si noti che il valore della variabile LOCAL_IP è "10.2.12.149", mentre quello della variabile TARGET_IP è "10.2.12.150", questi sono, rispettivamente, l'indirizzo IP del PC e l'indirizzo IP della scheda SMP+ all'interno della sottorete del C4G Open.

Dopo aver modificato e salvato lo script, occorre eseguirlo, digitando il seguente comando dalla directory **/usr/local/C4Gopen/bin** da ROOT o in modalità SUDO:

```
./loadRTnet
```

A questo punto il PC è pronto per comunicare con il C4G Open.



Il caricamento del modulo RTNet deve essere eseguito ad ogni riavvio del computer.

26.3.7 Esecuzione dell'applicazione C4G Open

- a. Scompattare Socket_C4G_Open.zip in una directory dove si hanno gli accessi in lettura e scrittura.
- b. Accedere alla directory e compilare il codice con il comando:

```
make
```

- c. Verificare che la compilazione non dia errori e quindi eseguire l'applicazione con il comando:

```
sudo ./socketC4Gopen
```

- d. A questo punto il software del C4GOpen è in esecuzione, aspettando il primo messaggio proveniente dal C4G.
- e. Quando questo arriva, vengono mostrate a video le informazioni di inizializzazione riportate in [Fig. 10.1](#). Da questo momento in poi, il sistema è pronto per eseguire una delle modalità aperte, previste dall'applicazione da realizzare.



Si noti che il software che gira sul PC esterno è un server, quindi deve partire prima che il C4G riparta per la prima volta, dopo aver settato le variabili opportune per abilitare gli assi alla modalità aperta, secondo quanto riportato nel [Cap.6. - Configurazione iniziale del sistema](#).



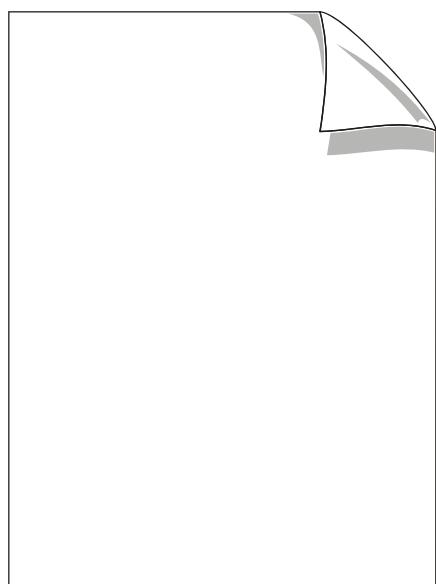
La compilazione del codice deve essere eseguita ogni qualvolta si modifichi il programma.

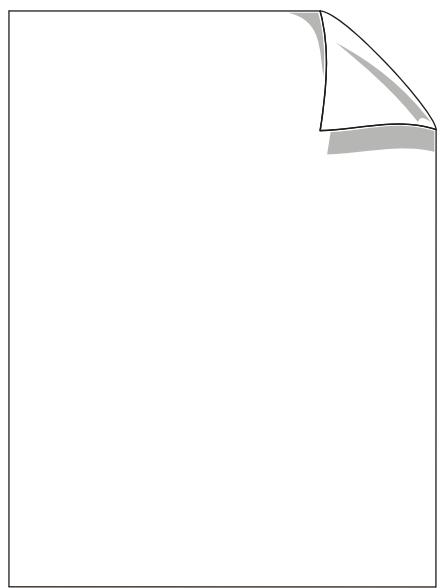
27. DOCUMENTI DI RIFERIMENTO

In [Tab. 27.1](#) sono riportati i documenti di riferimento al C4G OPEN.

Tab. 27.1 - Documenti di riferimento

N°	Nome file	Titolo
1	RT_CRPCSX01-07_A C4G Open Setup HW-SW lato PC.pdf	C4G Open Setup HW & SW "lato PC"
2	Troubleshooting RTAI Comedi rev0.2.pdf	Troubleshooting RTAI & COMEDI







Comau in the World

**COMAU S.p.A.
Headquarters**
Via Rivalta, 30
10095 Grugliasco - TO (Italy)
Tel. +39-011-0049111

Powertrain Machining & Assembly
Via Rivalta, 30-49
10095 Grugliasco - TO (Italy)
Tel. +39-011-0049111
Telefax +39-011-0049688

Body Welding & Assembly
Strada Borgaretto, 22
10092 Borgaretto di Beinasco - TO (Italy)
Tel. +39-011-0049111
Telefax +39-011-0048672

Robotics & Service
Via Rivalta, 30
10095 Grugliasco - TO (Italy)
Tel. +39-011-0049111
Telefax +39-011-0049866

Engineering, Injection Moulds & Dies
Via Bistagno, 10
10136 Torino (Italy)
Tel. +39-011-0051711
Telefax +39-011-0051882

Comau France S.A.
5-7, rue Albert Einstein
78197 Trappes Cedex (France)
Tel. +33-1-30166100
Telefax +33-1-30166209

Comau Estil
10, Midland Road
Luton, Bedfordshire LU2 0HR (UK)
Tel. +44-1582-817600
Telefax +44-1582-817700

Comau Deutschland GmbH
Monzastrasse 4D
D-63225 Langen (Germany)
Tel. +49-6103-31035-0
Telefax +49-6103-31035-29

German Intec GmbH & Co. KG

Im Riedgrund 1
74078 Heilbronn (Germany)
Tel. +49-7131 28 22-0
Telefax +49-731 28 22-400

Mecaner S.A.

Calle Aita Gotzon 37
48610 Urduliz - Vizcaya (Spain)
Tel. +34-94-6769100
Telefax +34-94-6769132

Comau Poland Sp. ,Z.O.O.

Ul. Turyńska 100
43-100 Tychy (Poland)
Tel. +48-32-2179404
Telefax +48-32-2179440

Comau Romania S.R.L.

Oradea, 3700 Bihor
Str. Berzei nr.5 Suite E (Romania)
Tel. +40-59-414759
Telefax +40-59-479840

Comau Russia S.R.L.

Ul. Bolshaya Dmitrovka 32/4
107031 Moscow (Russian Federation)
Tel. +7-495-7885265
Telefax +7-495-7885266

Comau SPA Turkiye Bursa Isyeri

Panayir Mah. Buttimis İş Merkezi
C Block Kat 5 no.1494
16250 Osmangazi/Bursa (Turkey)
Tel. +90-0224-2112873
Telefax +90-0224-2112834

Comau Inc.

21000 Telegraph Road
Southfield, MI 48034 (USA)
Tel. +1-248-3538888
Telefax +1-248-3682531

Comau Pico Mexico S. de R.L. de C.V.

Av. Acceso Lotes 12 y 13
Col. Fracc. Ind. El Trébol 2º Secc.
C.P. 54610, Tepotzotlán (Mexico)
Tel. +1-52-5 8760644
Telefax +1-52-5 8761837

Comau Canada Inc.

4325 Division Road Unit # 15
Ontario N9A 6J3 (Canada)
Tel. +1-519-9727535
Telefax +1-519-9720809

Comau do Brasil Ind. e Com. Ltda.

Rua Do Paraíso, 148 - 4º Andar
Paraíso - Cep. 04103-000
São Paulo - SP (Brazil)
Tel. +55-11-21262424
Telefax +55-11-32668799

Comau Argentina S.A.

Ruta 9, Km 695
5020 - Ferreyra
Córdoba (Argentina)
Tel. +54-351-4503996
Telefax +54-351-4503909

Comau SA Body Systems (Pty)

Hendrik van Eck Drive
Riverside Industrial Area
Uitenhage 6229 (South Africa)
Tel. +27-41-9953600
Telefax +27-41-9229652

Comau (Shanghai) Automotive Equipment Co., Ltd.

Pudong, Kang Qiao Dong Road Nr. 1300
Block 2 - Kang Qiao
201319 Shanghai (P.R.China)
Tel. +86-21-68139900
Telefax +86-21-68139622

Comau India Pvt. Ltd.

33Km Milestone Pune-Nagar Road
Shikrapur, Pune - 412208 (India)
Tel. +91.2137.678100
Telefax +91.2137.678110

COMAU Robotics services

Repair: repairs.robots@comau.com

Training: training.robots@comau.com

Spare parts: spares.robots@comau.com

Technical service: service.robots@comau.com

comau.com/robotics