

Consiglio Nazionale delle Ricerche

**ISTITUTO DI ELABORAZIONE
DELLA INFORMAZIONE**

PISA

*Contratto di collaborazione tecnico-scientifica
Aeritalia-GAT/IEI-CNR*

**PROGETTO E SVILUPPO DI UNA STRUTTURA
PER IL CND AD ULTRASUONI**

L. Azzarelli, E. Bozzi, M. Chimenti

Nota Interna B4-64

Dicembre 1990

Stampato in proprio dal Servizio Tecnografico dell'Area di Ricerca Pisana del CNR.

*Contratto di collaborazione tecnico-scientifica
Aeritalia-GAT/IEI-CNR*

**Progetto e sviluppo di una struttura per il CND
ad ultrasuoni**

Luciano Azzarelli, Edoardo Bozzi, Massimo Chimenti

Alle attività di ricerca e sviluppo software hanno contribuito:

<i>P. Andronico</i>	<i>Borsista</i>
<i>D. Borgioli</i>	<i>Borsista</i>
<i>S. Cerri</i>	<i>IEI-CNR</i>
<i>R. Evangelista</i>	<i>Borsista</i>
<i>E. Fantini</i>	<i>IEI-CNR</i>
<i>A. Marchetti.</i>	<i>Borsista</i>
<i>M. Mercatanti</i>	<i>Consul. C.O.A.P.</i>

Alle attività di ricerca e e sperimentazione hardware hanno collaborato:

<i>A. Moretto</i>	<i>IEI-CNR</i>
<i>R. Panicucci</i>	<i>IEI-CNR</i>
<i>L. Pardi</i>	<i>IEI-CNR</i>

La redazione del testo è stata curata da:

<i>M. Ballati</i>	<i>IEI-CNR</i>
-------------------	----------------

INDICE

Introduzione	pag. 3
1. Specifiche di progetto	pag. 4
1.1 Requisiti di acquisizione	pag. 5
1.2 Requisiti di gestione dei dati	pag. 6
1.3 Requisiti operativi	pag. 8
1.4 Specifiche dei componenti	pag. 8
2. Descrizione dei componenti	pag.11
2.1 Generatore-ricevitore US	pag.11
2.2 Sistema a controllo numerico per lo spostamento della sonda	pag.11
2.3 Sistema di gestione ed elaborazione	pag.12
3. Sviluppo e sperimentazione sul sistema IRB 2000	pag.16
3.1 Programmazione con tastiera	pag.16
3.2 Programmazione fuori linea	pag.18
3.3 Controllo in tempo reale	pag.19
3.4 Istruzioni di posizione	pag.20
4. Sviluppo e sperimentazione sul sistema SVP2000	pag.26
4.1 Programmi di gestione delle primitive SVP	pag.26
4.2 Definizione di un protocollo TIFF	pag.31
4.3 File Systems per immagini	pag.34
4.4 Programmi di enhancement	pag.36
4.5 Moduli per indagine statistica	pag.48
5. Definizione del sistema di acquisizione e scansione	pag.55
5.1 Banco di scansione xy	pag.55
5.2 Studio di un protocollo software	pag.56
5.3 Architettura della struttura di indagine	pag.62
5.4 Requisiti del software di scansione	pag.68
6. Discussione	pag.71
7. Appendice	pag.74

Introduzione

In questa relazione viene presentato lo stato di avanzamento delle attività di ricerca svolte nel settore del controllo non distruttivo (CND) di materiali e manufatti per uso aeronautico, nel corso della collaborazione tecnico-scientifica in atto fra l'Istituto di Elaborazione della Informazione del CNR e la Società Aeritalia GAT. Tale attività ha come obiettivo finale sia la definizione di nuove ed efficienti tecniche di analisi sia la realizzazione di stazioni di lavoro altamente specializzate per il CND, che fanno uso di differenti tecniche di indagine (onde elettromagnetiche RF, radiazioni termiche o vibrazioni ultrasonore).

Tali stazioni sono dotate di differenti procedure specializzate per esaminare i nuovi tipi di manufatti attualmente impiegati nell'industria aeronautica caratterizzati anche da geometrie complesse e diversificate, (strutture in materiale composito e strutture incollate); le stazioni potranno essere opportunamente interconnesse per formare una struttura informatizzata unica di tipo distribuito in grado quindi di gestire in modo coordinato l'intero processo di controllo sia nelle fasi di sperimentazione dei nuovi materiali sia nelle fasi di produzione dei manufatti.

Infatti a causa dei nuovi processi di realizzazione dei materiali e strutture nonché della varietà e tipologia di eventuali difetti (incollaggi scorretti, inclusioni, distacchi, ecc.), diventa necessario il poter disporre di diversificati strumenti di indagine cooperanti e interagenti, sia dal punto di vista operativo sia da quello gestionale, sia per quanto concerne l'analisi dei dati e la valutazione delle informazioni ricavate dai campioni in esame.

Questa relazione si riferisce essenzialmente alla metodologia di indagine ad ultrasuoni per la quale si deve realizzare una struttura in grado di esplorare manufatti di varia composizione e di differenti geometrie mediante spostamenti di una sonda lungo traiettorie nello spazio; conseguentemente il movimento di scansione deve essere realizzato impiegando un sistema elettromeccanico a controllo numerico con più gradi di libertà di semplice e flessibile programmabilità e con caratteristiche di notevole precisione, ripetibilità e velocità di esecuzione.

1. SPECIFICHE DI PROGETTO

Come è stato accennato nell'introduzione, la collaborazione di ricerca comporta studi, esperienze e realizzazioni di prototipi per determinare le caratteristiche di una struttura, da realizzare successivamente in una eventuale fase di ingegnerizzazione, da impiegare sia nella linea di produzione per il controllo e validazione di qualità dei manufatti, sia nell'ambiente di progettazione come strumento per la conoscenza di dettaglio e la valutazione delle caratteristiche e condizioni di difetto.

In applicazioni relative al processo di produzione, la struttura deve possedere caratteristiche tali che consentano l'esecuzione di procedure di acquisizione e diagnosi con un buon rapporto costo/efficienza; la struttura deve svolgere delle procedure fisse e predeterminate.

In applicazioni relative a studio e verifica in ambiente di progettazione, la struttura deve prevedere l'acquisizione di segnali anche multidimensionali con un alto contenuto informativo utilizzando riferimenti geometrici relativi al campione in esame definiti in precedenza durante la fase di progettazione (ad es. dati e coordinate prodotti da sistemi CAD), oppure ricavati direttamente dal campione mediante tecniche di autoapprendimento. La struttura deve svolgere delle operazioni riprogrammabili e sviluppabili in base alle esigenze delle attività di progetto dei manufatti, delle relative verifiche nonché dei risultati ottenuti con le altre tecniche di indagine non distruttiva eventualmente impiegate.

Quest'ultimo aspetto comporta la necessità di scegliere e progettare dei componenti della struttura (sistemi di calcolo, periferiche standard, dispositivi speciali e strumentazione) che consentano l'espandibilità delle capacità di calcolo e gestione nonché l'adattabilità verso le future applicazioni che possono richiedere l'interfacciamento a periferiche molto diversificate e specializzate. Questi criteri, insieme con le specifiche relative all'acquisizione dei dati, alle modalità di elaborazione, alle tecniche di rappresentazione ed ai metodi di interazione uomo-sistema, hanno guidato la scelta delle apparecchiature di base da utilizzare per realizzare la struttura.

Considerando le principali fasi che contraddistinguono il processo di CND, sono stati individuati i seguenti requisiti generali.

1.1 Requisiti di acquisizione

In relazione alla tipologia dei materiali utilizzati e agli spessori, in genere contenuti, dei campioni o manufatti di interesse nonché per consentire una precisa analisi della struttura interna del mezzo in osservazione, si è ritenuto necessario ricorrere all'impiego di tecniche di indagine basate sulla misura e sull'analisi del segnale ultrasonoro riflesso; per ottenere inoltre una sufficiente risoluzione, e conseguentemente un elevato dettaglio di rilevamento, si devono impiegare frequenze ultrasonore anche molto elevate comprese nell'intervallo 10÷100 MHz.

Tali scelte impongono almeno quattro vincoli che incidono nella complessità del processo di scansione:

- 1) la sonda deve essere spostata mantenendo costante la sua distanza dalla superficie frontale del campione da esaminare che deve essere immerso in acqua
- 2) la sonda deve essere spostata mantenendo il suo orientamento pressoché normale alla superficie frontale del campione
- 3) si devono utilizzare sonde focalizzate che impongono quindi distanze predefinite dal campione in esame (circa 50 mm)
- 4) il segnale ultrasonoro deve essere acquisito con frequenze di campionamento elevate con conseguente produzione di grandi quantità di dati che, per contenere la durata di indagine entro tempi operativamente accettabili, devono essere trasferiti a grande velocità.

Per ogni punto di misura si deve acquisire il segnale prodotto dal trasduttore in funzione dell'eco riflessa causata dal campione. Il segnale utile è compreso nell'intervallo di tempo T_v , detto tempo di volo, delimitato dal picco prodotto dalla riflessione sulla superficie anteriore ed il picco causato dalla riflessione sulla parete inferiore. In questo intervallo di tempo il segnale deve essere campionato con una frequenza di ripetizione dipendente dalla frequenza fondamentale dell'impulso emesso dal trasduttore; si sceglie una frequenza di campionamento f_s uguale a 2÷4 volte quella di eccitazione f_e . L'intervallo T_v varia in funzione delle caratteristiche acustiche del materiale attraversato dal segnale ultrasonoro

e dello spessore dei campioni da esaminare; da prove sperimentali eseguite su vari tipi di materiali questo intervallo risulta essere compreso fra 1 e 6 μsec , per cui posto:

$$T_v = 6 \mu\text{s}$$

$$f_e = 100 \text{ MHz}$$

$$f_s = 4f_e$$

il numero di campioni N ricavati dalla forma d'onda completa in ciascun punto risulta essere

$$N = T_v \cdot f_s = 2400$$

Se N è fissato a priori, ad esempio per ridurre la quantità di dati da elaborare, si può avere una limitazione del massimo spessore del provino, oppure una limitazione della massima frequenza di eccitazione utilizzabile e di conseguenza della risoluzione ottenibile.

1.2 Requisiti di gestione dei dati

Mediante la struttura di indagine US si devono poter ricavare:

- *La mappa dei tempi di volo.*

Essa è costituita da una matrice di I·J elementi ciascuno dei quali definisce la misura del tempo di volo per ciascun punto esaminato. Le dimensioni I e J dipendono dalle dimensioni Δ_x e Δ_y del campione e dal passo di campionamento P, che a sua volta dipende dalla risoluzione spaziale desiderata:

$$I = \frac{\Delta_y}{P} \quad \text{e} \quad J = \frac{\Delta_x}{P}$$

Quantizzando il tempo di volo T_v su un byte (256 livelli) e posto $T_{v\text{max}} = 6 \mu\text{s}$ si ha una risoluzione temporale $\Delta T_v \cong 0.023 \mu\text{s}$; la risoluzione in unità di spessore dipende dalla velocità di propagazione degli ultrasuoni nel mezzo e quindi dal tipo di materiale impiegato.

- *La mappa di rilevamento a soglia singola.*

Essa è costituita da una matrice di $I \cdot J$ elementi ciascuno dei quali definisce il valore di picco V_p del segnale di eco riflessa in una finestra temporale T_1 che inizia al termine del picco di segnale prodotto dalla superficie frontale.

Dato un intervallo ($V_{\min} - V_{\max}$) la risoluzione ottenibile con un convertitore ad M classi è data da $\frac{V_{\max}}{M}$; il numero delle classi M e il tipo di convertitore (lineare o logaritmico) va scelto in funzione della precisione con cui si vuole determinare la variazione della impedenza acustica Z_a che ha determinato il segnale di riflessione in modo da individuare le caratteristiche del difetto o della variazione di struttura del materiale.

- *La mappa della forma d'onda completa.*

Essa è costituita da una matrice di $I \cdot J \cdot N$ elementi ricavati dagli N campioni ottenuti per ciascuno degli $I \cdot J$ punti di misura. La mappa ottenuta dall'acquisizione della forma d'onda completa che contiene le informazioni necessarie per ricavare le mappe dei tempi di volo e delle ampiezze comporta l'inconveniente di raggiungere dimensioni molto elevate quando le dimensioni del provino da esaminare superano qualche centimetro quadrato. In generale la quantità di memoria C necessaria per immagazzinare il campionamento dell'onda completa di un'area $\Delta_x \cdot \Delta_y$ scandita a passo P è data da:

$$C = \frac{\Delta_x \cdot \Delta_y \cdot N}{P^2} \cdot M$$

dove M è il numero delle classi di quantizzazione sull'ampiezza del segnale.

I valori del passo di campionamento, della risoluzione in ampiezza e del numero N di elementi per campione dipendono dalle precisioni di misura richieste al fine di determinare in maniera esauriente le caratteristiche dei campioni; sia N che P dipendono in maniera implicita dalla frequenza del segnale ad ultrasuoni.

1.3 Requisiti operativi

I requisiti di tipo fisico elencati in precedenza devono essere conciliati con i requisiti operativi (velocità di esplorazione o di elaborazione, praticità di gestione del sistema); inoltre si devono tenere presenti anche le caratteristiche dei componenti da utilizzare (ad es. capacità di memoria, velocità di trasmissione dati ecc.).

Ad esempio: la scansione di un'area di $6.3 \cdot 6.3 \text{ cm}^2$ eseguita con passo P di $0,5 \text{ mm}$, con una risoluzione in ampiezza di 256 livelli e limitando il numero di campioni N a 512 per ogni punto di misura comporta l'occupazione di 8 Mby di memoria: la matrice così ricavata risulta essere già molto grande sia per l'esecuzione delle elaborazioni, sia per la rappresentazione delle mappe dei dati di acquisizione o di sintesi.

In base ai vincoli citati nel caso di controllo di qualità risulta conveniente ricavare una mappa completa di tempi di volo e/o di misure di ampiezza a soglia singola che possano essere sufficienti per rivelare e localizzare gli eventuali difetti o strutture di interesse; successivamente si può eseguire una scansione ad onda completa, limitata però alle zone di difetto rilevate in precedenza, al fine di ricavare anche il tipo o le caratteristiche fisiche del difetto.

L'acquisizione ad onda completa risulta invece essenziale nelle operazioni di analisi che servono come supporto alle azioni di progettazione dei manufatti, in questo caso la struttura di indagine presenterà delle modalità di gestione differenti da quelle richieste alla struttura quando viene impiegato nel processo di produzione.

1.4 Specifiche dei componenti

In base ai suddetti requisiti generali, la struttura per l'indagine non distruttiva ad ultrasuoni deve comprendere i seguenti componenti hardware:

- un generatore ad ultrasuoni in grado di pilotare sonde con frequenze di risonanza da 10 a 100 MHz;
- un ricevitore ad ultrasuoni in grado anche di produrre segnali di tempo di volo e di ampiezza di picco; la finestra temporale deve avere una

durata variabile tra 1 e 10 μ s e un ritardo variabile corrispondente almeno alla propagazione in un percorso d'acqua uguale alla distanza di focalizzazione delle sonde;

- un dispositivo per lo spostamento della sonda con possibilità di orientamento nello spazio lungo traiettorie definite e gestite da calcolatore in base a misure di rilevamento sui campioni o in base alle coordinate definite dal sistema CAD mediante il quale è stata eseguita la progettazione del campione da esaminare;
- un convertitore analogico-digitale con frequenza di campionamento fino a 400 MHz ed una memoria tampone la cui dimensione e tempi di accesso devono essere definiti in funzione della metodologia di scansione ed acquisizione;
- un sistema di calcolo con memoria dati di capacità opportuna (fra 8 e 16 Mby), in grado di operare in ambiente real time per la gestione e la sincronizzazione delle varie periferiche specializzate citate in precedenza; il sistema deve inoltre possedere una grande memoria di massa (1÷2 Gby);
- un dispositivo grafico pittorico con memoria video di almeno 8 Mby per la rappresentazione dei dati grezzi (mappe o grafici delle forme d'onda) e di eventuali immagini tomografiche ricavate dalla elaborazione dei dati;
- un dispositivo per la restituzione grafico-pittorica con una opportuna risoluzione e precisione.

La componente software della struttura US dovrà svolgere le seguenti funzioni fondamentali:

- 1) gestione delle risorse interne al sistema di elaborazione dati
- 2) gestione delle risorse esterne
- 3) esecuzione di procedure specializzate per il controllo di qualità o analisi di materiali
- 4) gestione dell'archiviazione e della comunicazione.

Le risorse interne sono fundamentalmente costituite dal sistema hardware e software specializzato per l'acquisizione, l'elaborazione di segnali e il trattamento di immagini.

Le risorse esterne sono costituite, oltre alle periferiche di comunicazione uomo-macchina, dalle unità di controllo programmabili delle apparecchiature per la rice-trasmissione di segnali o per la movimentazione della sonda di indagine.

Le procedure per il controllo di qualità sono costituite da insiemi di programmi interattivi o automatici che devono controllare lo svolgimento delle varie operazioni in base ai risultati delle analisi dei dati.

La gestione dell'archiviazione e della comunicazione deve poter concedere l'inserimento di dati e l'accesso agli archivi generali che raccolgono risultati e informazioni ottenute con differenti tecniche di indagine dalle varie stazioni per il controllo non distruttivo.

2. DESCRIZIONE DEI COMPONENTI

In base alle specifiche ed ai vincoli descritti in precedenza, sono stati scelti e/o utilizzati i seguenti componenti:

2.1 *Generatore-ricevitore U.S.*

Lo strumento attualmente impiegato per la generazione di impulsi US e per l'amplificazione del segnale di eco è un ECHOGRAPH-1030. Lo strumento comprende: un trasmettitore che genera un treno di impulsi in modo automatico o in seguito ad un sincronismo esterno; un ricevitore che amplifica, in modo lineare e logaritmico, il segnale ricevuto dalla sonda con eventuale filtraggio per la riduzione del rumore. Inoltre una unità di monitor-gate consente di determinare una finestra temporale con ritardo rispetto all'impulso di eccitazione e durata variabile entro un ristretto campo di valori e fornisce un segnale analogico tra 0 e 1 volt il cui valore è proporzionale al valore di picco del segnale all'interno della finestra. Lo strumento possiede anche un'unità di programmazione "Computer unit RE" che consente di eseguire alcune operazioni di misura sul segnale mediante un insieme di funzioni programmate; l'unità di elaborazione può essere controllata da un calcolatore mediante una porta seriale RS232.

2.2 *Sistema a controllo numerico per lo spostamento della sonda*

La sonda di esplorazione viene spostata mediante un sistema IRB2000 della ABB, costituito da un braccio meccanico e da un controllore. Il braccio meccanico del tipo a 6 assi, azionato da motori in corrente alterna con servo controlli, consente di spostare la sonda collegata sulla piastra porta utensili in un volume di lavoro la cui forma e le cui dimensioni dipendono dal posizionamento del robot. Nella posizione attualmente utilizzata, con il basamento fissato sul pavimento, nell'intero volume disponibile si individua un volume con forma delimitata da un parallelepipedo rettangolo con dimensioni di circa 1m•1.5m•0.5m, che definisce la zona in cui è possibile esplorare un campione, di opportuna forma, immerso in una vasca piena d'acqua.

Il braccio meccanico è azionato da un'Unità di controllo S3 ABB che può determinare spostamenti nel volume di lavoro con risoluzione e ripetibilità di 0.1 mm.

S3 è costituita da una consolle per la gestione manuale del robot, da una unità di programmazione per la scrittura di programmi nel proprio linguaggio assembler, una unità a floppy disk per la memorizzazione di programmi e da varie interfacce di collegamento verso l'esterno:

- un'interfaccia di I/O a 32 linee per segnali digitali (16 ingressi e 16 uscite)
- un'interfaccia di I/O a 8 linee per segnali analogici (4 ingressi e 4 uscite)
- un'interfaccia a relais per azionamenti di servo comandi esterni
- un'interfaccia RS232 per il collegamento ad un calcolatore ospite.

2.3 Sistema di gestione ed elaborazione

Questo sistema deve assolvere varie funzioni: gestire la generazione degli ultrasuoni; il campionamento del segnale acquisito; la scansione del campione in esame; la sincronizzazione dei vari componenti; l'acquisizione, memorizzazione e archiviazione dei dati digitali; l'elaborazione delle mappe di ingresso e di sintesi; l'estrazione delle informazioni di interesse. Queste ultime funzioni sono quelle tipiche del trattamento di segnali ed immagini che richiedono in genere, per soddisfare le necessità di velocità e interazione, architetture hardware e software molto specializzate. Per soddisfare l'esigenza di controllare i componenti esterni, in particolare il dispositivo di scansione, il sistema deve essere dotato di un ambiente operativo real time che possa coesistere con un apposito ambiente di sviluppo che consenta l'integrazione di procedure e componenti hardware e software di tipo generale oppure specificatamente realizzati per diversificate metodologie di CND.

A tale scopo è stato scelto, e in parte ridisegnato per meglio soddisfare le peculiari esigenze di input/output e di gestione dei segnali asincroni ricavabili dal processo di acquisizione, il sistema Scriba SVP2000. Tale sistema è basato su un'architettura a multi processori Motorola ed è dotato di due ambienti operativi: il System V/68 (Unix) visto come ambiente di sviluppo e gestione generale, e il System real time

VME Exec da utilizzarsi per la gestione della acquisizione, nonché per il controllo della manipolazione dei segnali ed immagini ricavati con periferiche.

L'SVP2000 si basa su una struttura multi-bus: in particolare un bus VME, sistema standard che consente la migliore espandibilità e flessibilità di sviluppo; un bus video ad alta velocità che consente la gestione di segnali digitali di tipo sincrono e asincrono.

Quest'ultimo bus V ha una banda effettiva di spostamento di 160 Mby/sec ed è preposto alla trasmissione veloce (> 20 MHz) dei dati immagine in modo asincrono e in tempo reale, e al pilotaggio di terminali video ad altissima risoluzione.

La peculiarità del sistema è determinata da una particolare scheda di acquisizione dei segnali multidimensionali e multispettrali di varia origine (telecamere di linea o di area, telecamera a colori, sensori CCD, sensori ad ultrasuoni, termocamere, rivelatori di prossimità).

Di seguito vengono sinteticamente riportate le caratteristiche fondamentali del sistema SVP2000 relativamente agli aspetti di interesse per l'applicazione specifica:

- Sistema principale CPU Motorola 68030 per l'ambiente UNIX
- Controller grafico-pittorico CPU Motorola 68020 per l'ambiente VME Exec
- Coprocessore matematico Motorola 68882
- Memoria Sistema principale (UNIX) 16 Mby shared DRAM MMU
- System real time (VME EXEXC) 1 Mby RAM
- Controller grafico-pittorico 64 Kby suddiviso in RAM EPROM-EEPROM
- Memoria video 8 Mby dual port VRAM configurabile via software

- Memoria di massa 700 Mbytes; 156 Mbytes streaming tapes
- Scheda di acquisizione segnali analogici con 3 canali paralleli
- Sistemi operativi V/68 (Unix); VME Exec; X-Window
- Linguaggi Fortran - C.

Per soddisfare le esigenze di interazione uomo-macchina e per consentire una rapida ed efficiente restituzione grafica di dati, parametri ed informazioni di sintesi, si è ritenuto opportuno integrare nella stazione un sottosistema grafico. Tale sistema è costituito dal processore grafico Motorola MVME 393 che consente la gestione di due stazioni di lavoro e che è caratterizzato dalle prestazioni di elaborazione e rappresentazione in ambiente gestione X-Window.

La sottostazione ha le seguenti caratteristiche:

- Sottosistema grafico Motorola MVME 393 dotato di:
due processori indipendenti per la gestione multitasking e window management;
2 Mby di memoria locale
- Due monitor grafici Mitsubishi 1024•1024 pixel
- Quattro piani di memoria video con 1024•2048 bit ciascuno - 64 Kb di memoria per l'esecuzione di programmi residenti.

Sulla stazione sono state inoltre inserite:

- una interfaccia GPIB-488 sul Bus VME per consentire il trasferimento veloce dei dati digitali di campionamento del segnale US e per il trasferimento e l'attenuazione di programmi residenti nei sistemi di controllo delle periferiche costituenti la struttura di indagine

- una interfaccia di I/O parallela per la ricezione e trasmissione di comandi e segnali
- una scheda con convertitori analogico-digitali e dispositivi per l'acquisizione e la generazione di segnali ricavati da diversi tipi di sensori
- un disco diagnostico da 750 MBy per l'archiviazione dei dati grezzi e di sintesi

L'inserimento di tutti questi componenti hardware nella stazione da comportato una nuova organizzazione del Bus VME modificando la ripartizione delle schede Motorola e Scriba.

3. SVILUPPO E SPERIMENTAZIONE SUL SISTEMA IRB 2000

Il braccio meccanico del robot IRB2000 può essere azionato in tre maniere:

- mediante la tastiera di programmazione manuale "teach pendant" (vedi Product Manual IRB2000 e Programming manual Robot Control System S3).
- mediante programmazione fuori linea (vedi Off-line Programming User's Guide).
- mediante controllo in tempo reale (vedi Communication Tools User's Guide).

È stata eseguita una sperimentazione per ciascuna modalità usando i diversi tipi di istruzioni di movimento

3.1 Programmazione con tastiera.

La tastiera di programmazione dell'unità di controllo consente l'accesso a un insieme di menù ad albero per la selezione di un centinaio di funzioni programmate che eseguono operazioni logiche o aritmetiche, di posizionamento, di editing, di I/O e di definizione di parametri.

Di conseguenza, è possibile scrivere, correggere, archiviare e richiamare dalla memoria, della capacità complessiva di 64KB, differenti programmi individuati da un'etichetta numerica da 1 a 9999: nella fase di scrittura o di correzione dei programmi si usano il display della tastiera, sul quale vengono mostrate le opzioni del menù corrente, e il monitor alfanumerico del sistema S3, sul quale vengono elencate le istruzioni del programma

I dati di scansione di un campione qualsiasi possono essere determinati mediante autoapprendimento delle coordinate di posizionamento o mediante calcolo di traiettorie. In entrambi i casi, è necessario fornire al sistema un insieme limitato e ordinato di posizioni, in quanto l'unità di controllo provvede a gestire i movimenti intermedi del robot in modo da spostare la sonda lungo traiettorie rettilinee (quando è stato selezionato il sistema di coordinate rettangolari), lungo traiettorie

curvilinee (coordinate Robot) oppure lungo traiettorie di minimo percorso (coordinate Modrect). È possibile inoltre concatenare fino a 10 livelli di programmi elementari.

Nel caso di autoapprendimento si seleziona il menù "Position", si aziona il joystick della tastiera in modo da orientare la sonda e portarla nella posizione desiderata e si acquisisce la coordinata di scansione premendo il tasto "position": questa operazione dev'essere ripetuta tante volte quante sono richieste dalla complessità della geometria del campione in esame. È possibile conoscere mediante il visore della tastiera i valori x,y e z di ciascuna coordinata di scansione, ma non l'orientamento della sonda.

Nel caso di calcolo di traiettorie si inserisce la sequenza opportuna di coordinate senza spostare manualmente il robot e si eseguono le interpolazioni o richiamando le funzioni generatrici programmate, oppure eseguendo calcoli con le operazioni logiche e aritmetiche a disposizione. Nel primo caso si può usare una funzione generatrice di archi che trasla la sonda con orientamento perpendicolare al piano sul quale giacciono delle semicirconferenze ricavate da tre punti ad essa appartenenti; nel secondo caso, si possono definire coordinate di partenza e calcolare gli incrementi facendo uso dei 199 registri indirizzabili nella memoria e delle quattro operazioni elementari. Poiché la precisione dell'aritmetica è limitata dalla lunghezza dei registri (16 bit), si possono in pratica determinare con precisione solo traiettorie di tipo rettilineo; inoltre non è possibile controllare esplicitamente l'orientamento della sonda.

Durante le esperienze relative a questo tipo di azionamento del robot sono stati realizzati una decina di programmi elementari di scansione. Il programma sposta la sonda e genera su una porta di uscita dell'unità di controllo S3 un impulso utilizzabile come trigger per l'emissione del segnale a ultrasuoni; il programma prosegue nell'esecuzione dopo il riconoscimento di un flag operazione eseguita letto su una linea d'ingresso.

In appendice sono riportati i listati e le descrizioni dei programmi che sono stati scritti per sperimentare la movimentazione e la metodologia di programmazione.

3.2 Programmazione fuori linea

Il controllo S3 comunica con un calcolatore ospite (Superior Controller) attraverso una porta seriale RS-232, attivata selezionando sulla tastiera di programmazione il modo di funzionamento REMOTE; il calcolatore ospite può trasmettere a S3 funzioni o programmi in formato eseguibile.

Nel caso di programmazione fuori linea si dispone del software "ABB Robotics Off-line programming" (OLP), operante in ambiente MS-DOS, che presenta un menù di opzioni per la scrittura, la compilazione e il trasferimento di programmi nella memoria del sistema di controllo; è possibile inoltre richiamare programmi da S3, decompilarli e modificarli mediante un editor.

I programmi vengono scritti nel linguaggio ARLA, che possiede un insieme di istruzioni coincidente con le funzioni richiamabili mediante la tastiera di programmazione, tranne che per i comandi per la gestione dell'unità a disco di S3; si dispone quindi di istruzioni per operazioni logiche, aritmetiche o di I/O e per la definizione delle posizioni del robot meccanico.

A differenza del modo di programmazione locale, le posizioni del robot vengono definite in maniera completa specificando le coordinate rettangolari x,y,z (location) del "centro flangia" (Tool Center Point) e l'orientamento dell'utensile: quest'ultimo può essere espresso o mediante quaternioni o mediante angoli di Eulero (vedi par. successivo).

Per determinare la traiettoria del robot si scrive un programma con la sequenza di istruzioni di posizione, con etichette che determinano l'accesso ad un file (Position file) contenente la sequenza numerata di posizioni espresse in uno dei due modi suddetti. È dunque possibile specificare le posizioni del centro flangia con la precisione offerta dal sistema, che è di 1/65136 per le coordinate rettangolari e di 1/32567 per l'orientamento. Una traiettoria comunque complicata può essere seguita con la precisione suddetta pur di calcolare i punti necessari e di trascriverli in un apposito Position File di OLP.

Nella fase di compilazione il programma sorgente e il rispettivo Position File vengono trasformati in un programma eseguibile, che viene

poi trasmesso al sistema di controllo con un numero di riconoscimento definito tra 1 e 9999. L'operazione di trasferimento è gestita da OLP; l'esecuzione del programma trasferito può essere attivata tramite un comando sulla tastiera di programmazione o mediante una linea di controllo su un'interfaccia di sistema (non disponibile nella configurazione attuale di S3).

In appendice sono riportati i programmi sviluppati nel corso dell'esperienza condotta sulla programmazione fuori linea.

3.3 Controllo in tempo reale

Il software "ABB Robotics Communication Tools", installato su un personal computer collegato al controllo S3 mediante una porta seriale RS-232 consente diversi tipi di interazione tra il PC ed il robot. Infatti, il Communication Tools consente di scrivere in linguaggio C programmi per eseguire funzioni che sono attivate direttamente oppure in seguito alle richieste generate da un programma in esecuzione nel controllo del robot; queste funzioni permettono la lettura dello stato del robot, il trasferimento dei dati, l'attivazione, l'arresto o il trasferimento di programmi robot.

Nel processo di interazione più semplice il robot ha un comportamento di tipo passivo: i movimenti meccanici possono infatti essere determinati da sequenze di comandi di movimento (cmd, move) inviati dal Communication Tools, senza che ci sia un programma in esecuzione in S3: in questo caso viene sostanzialmente simulato il controllo manuale ottenuto con la tastiera di programmazione.

In un processo più complicato, il robot si può muovere sotto il controllo di un programma residente in S3, mentre sul calcolatore ospite si può conoscere lo stato del robot e si può arrestare il programma in esecuzione e attivarne un altro residente nella memoria di programmi di S3 o in un'area di memoria del PC: in questo caso il calcolatore ospite esegue un compito di controllo.

Nel tipo più complesso di interazione si hanno ancora due programmi in esecuzione, uno in S3 e uno sul PC. Il programma in corso su S3 contiene istruzioni SUCTRL, che fanno al calcolatore ospite delle richieste interpretate dal Communication Tools, al fine di attivare una

trasmissione di dati o di programmi in formato eseguibile: in questo caso il calcolatore ospite integra le capacità elaborative di S3 in modo da seguire traiettorie comunque complesse, assicurando al contempo la sincronizzazione con le operazioni svolte da altre periferiche.

In appendice sono riportati i programmi sviluppati nel corso dell'esperienza di programmazione col Communication Tools..

3.4 Istruzioni di posizione

Si riportano in maniera sintetica i risultati di uno studio sperimentale sui modi di controllo del robot; la relazione completa è contenuta in [1].

I movimenti elementari del braccio meccanico sono riportati nella tabella I.

Asse 1 - rotazione dell'intero braccio da $+179^\circ$ a -179°

Asse 2 - rotazione del braccio inferiore da $+100^\circ$ a -110°

Asse 3 - rotazione del braccio superiore da $+60^\circ$ a -60°

Asse 4 - rotazione del polso da $+200^\circ$ a -200°

Asse 5 - rotazione del polso da $+120^\circ$ a -120°

Asse 6 - rotazione della flangia da $+200^\circ$ a -200° .

Tabella I - Gradi di libertà del robot IRB 2000

La figura 1 mostra i sistemi di riferimento impiegati per la misura delle coordinate spaziali e degli orientamenti: si ha un sistema fisso $Oxyz$ (Base Coordinate System), e un sistema di mobile $Ox'y'z'$ (Hand Coordinate System); l'asse z di $Oxyz$ coincide con l'asse 1 del robot, l'origine di $Ox'y'z'$ coincide col centro della flangia P_0 mentre la sonda è rappresentabile mediante un vettore v sull'asse x' , di estremi P_0 e P_1 .

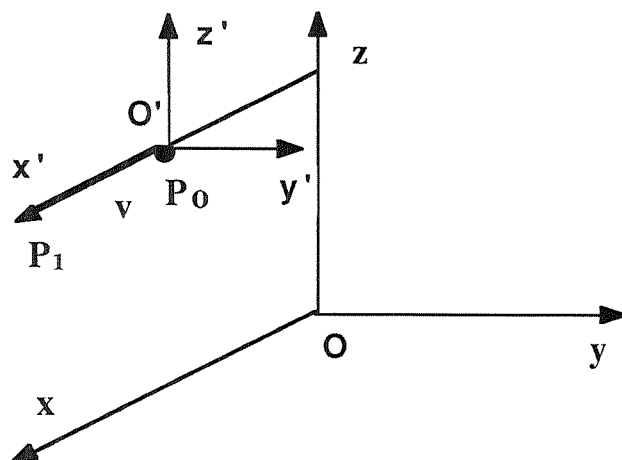


Figura 1 - Sistemi di riferimento.

Mediante il comando di sincronizzazione l'angolo di rotazione di ciascun asse viene portato a zero. Nella posizione di reset le coordinate di P_0 valgono $x_0=950$, $y_0=0$, $z_0=1585$ e i sistemi $Oxyz$ e $Ox'y'z'$ hanno gli assi paralleli e concordi (vedi figura 1).

Per compiere la scansione di un campione si deve controllare il movimento del robot orientando in maniera opportuna la sonda e portandola nella posizione corretta: il movimento complessivo consiste in una rotazione di $Ox'y'z'$ intorno a P_0 e in una traslazione di P_0

L'orientamento della sonda, coincidente con l'orientamento di un vettore unitario v , è ottenuto tramite una rotazione di $Ox'y'z'$, che può essere a sua volta determinata mediante istruzioni di posizione contenenti valori di quaternioni o di angoli di Eulero.

Infatti, una rotazione di un angolo θ intorno ad un asse n può essere associata ad un quaternion definito da:

$$Q = \text{Rot}(\mathbf{n}, \theta) = [\cos(\theta/2) + \mathbf{n} \sin(\theta/2)] \quad (1)$$

θ è misurato secondo la regola della mano destra intorno all'asse orientato, assunto di lunghezza unitaria; si ha:

$$\mathbf{n} = n_1 \mathbf{i} + n_2 \mathbf{j} + n_3 \mathbf{k}; \quad |\mathbf{n}| = 1 \quad (2)$$

In modo alternativo una trasformazione di coordinate per rotazione intorno all'origine può essere descritta introducendo le tre rotazioni successive specificate dagli angoli di Eulero φ, θ, ψ

Con riferimento alla fig.2 i piani x, y e x', y' si intersecano lungo l'asse o linea dei nodi \mathbf{n} ; \mathbf{n} è normale al piano zz' ed è orientato in modo che una rotazione intorno ad esso in verso antiorario porta l'asse z su z' .

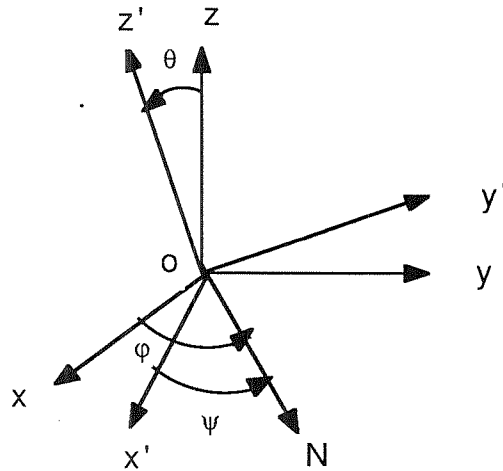


Figura 2 - Sistemi di riferimento fisso e mobile per la definizione di angoli di Eulero.

L'angolo φ tra x e \mathbf{n} è detto angolo di rotazione propria; φ è misurato in verso antiorario intorno a z ; si ha: $0 \leq \varphi < 2\pi$,

L'angolo θ tra z e z' è detto angolo di nutazione; θ è misurato in verso antiorario intorno a \mathbf{n} ; si ha: $0 < \theta < \pi$

L'angolo ψ tra \mathbf{n} e x' è detto angolo di precessione; ψ è misurato in verso antiorario intorno a z' ; si ha: $0 \leq \psi < 2\pi$

Gli angoli θ, φ, ψ sono tra loro indipendenti e determinano in modo univoco la trasformazione di coordinate.

In generale, in funzione di θ, φ, ψ i coseni direttori degli assi del sistema ruotato sono dati da:

$$\begin{aligned}
a_1 &= \cos\varphi \cos\psi - \sin\varphi \sin\psi \cos\theta \\
b_1 &= \cos\varphi \sin\psi + \sin\varphi \cos\psi \cos\theta \\
c_1 &= \sin\varphi \sin\theta \\
a_2 &= -\sin\varphi \cos\psi - \cos\varphi \sin\psi \cos\theta \\
b_2 &= -\sin\varphi \sin\psi + \cos\varphi \cos\psi \cos\theta \\
c_2 &= \cos\varphi \sin\theta \\
a_3 &= \sin\psi \sin\theta \\
b_3 &= -\cos\psi \sin\theta \\
c_3 &= \cos\theta
\end{aligned} \tag{3}$$

Si può dimostrare che nel sistema $Oxyz$ le coordinate del punto P_1 , corrispondente all'estremità del vettore v , sono date da:

$$\begin{aligned}
x &= \cos\varphi \cos\psi - \sin\varphi \sin\psi \cos\theta \\
y &= \sin\varphi \cos\psi + \cos\varphi \sin\psi \cos\theta \\
z &= \sin\psi \sin\theta
\end{aligned} \tag{4}$$

Al variare di φ , θ , ψ il punto P_1 si sposta sulla superficie di una sfera di raggio unitario con centro sull'origine del sistema di riferimento fisso.

Le operazioni necessarie per portare la sonda nella corretta posizione di misura relativa al punto P'_1 sulla superficie del campione sono descritte nella seguente procedura.

Si determinano inizialmente le coordinate x_t, y_t, z_t di P'_1 nel sistema $Oxyz$ e, presi due punti vicini a P'_1 , si calcolano i coseni direttori della normale alla superficie del campione passante per P'_1 , secondo l'equazione (1).

$$\begin{aligned}
\cos \alpha &= a/\rho \\
\cos \beta &= b/\rho \\
\cos \gamma &= c/\rho
\end{aligned}$$

Si ricava così l'istruzione di posizione da passare al controllo del robot. per ottenere il movimento di rotazione della sonda rispetto all'origine del sistema di riferimento mobile in condizioni di reset e la successiva traslazione della sonda orientata (vedi fig. 3).

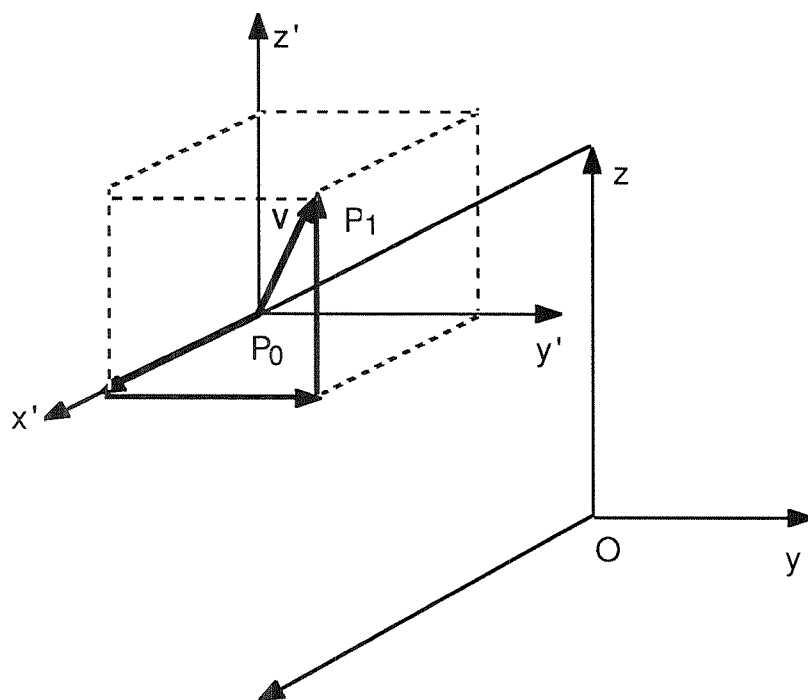


Figura 3 - Posizione della sonda dopo la rotazione.

Usando i quaternioni nella forma di rotazione composta del tipo $\text{Rot}(y, \gamma) \cdot \text{Rot}(z, \alpha)$ l'istruzione che definisce la posizione di P_0 e l'orientamento di v è espressa nella forma:

$$(x_t, y_t, z_t, q_1, q_2, q_3, q_4)$$

dove la locazione di P_0 è data da (x_t, y_t, z_t) e l'orientamento di v è dato da:

$$\begin{aligned} q_1 &= \cos((\pi/2 - \gamma)/2) \cdot \cos(\sin^{-1}[\cos \beta / \sin \gamma])/2 \\ q_2 &= \sin((\pi/2 - \gamma)/2) \cdot \sin(\sin^{-1}[\cos \beta / \sin \gamma])/2 \\ q_3 &= \sin((\pi/2 - \gamma)/2) \cdot \cos(\sin^{-1}[\cos \beta / \sin \gamma])/2 \\ q_4 &= \cos((\pi/2 - \gamma)/2) \cdot \sin(\sin^{-1}[\cos \beta / \sin \gamma])/2 \end{aligned}$$

Usando la notazione degli angoli di Eulero l'istruzione che definisce la posizione di P_0 e l'orientamento di v è espressa nella forma:

$$(x_t, y_t, z_t, \sin^{-1}[\cos \beta / \sin \gamma], 90^\circ, 90^\circ - \gamma)$$

Si definisce poi un TCP tale da portare la sonda alla distanza d rispetto alla superficie del campione; posto $v = |v|$ si ha:

$$\Delta x = v + d$$

$$\Delta y = 0$$

$$\Delta z = 0$$

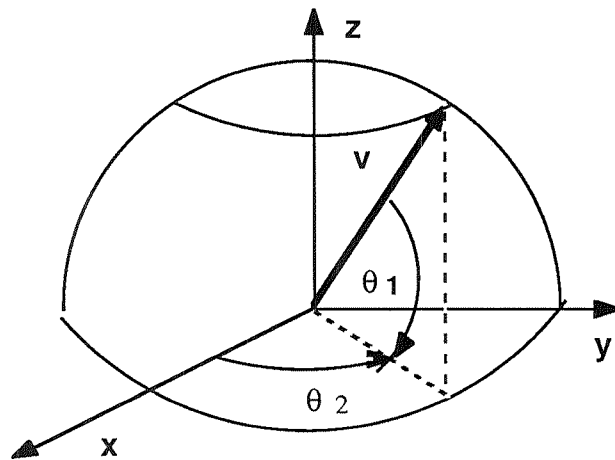


Figura 4 - Orientamento mediante latitudine e longitudine

L'orientamento della sonda può essere desunto anche mediante la definizione di un angolo di latitudine θ_1 e un angolo di longitudine θ_2 . (vedi fig 4). In questo caso si ha il quaternioni:

$$q_1 = \cos(\theta_1/2) \cos(\theta_2/2)$$

$$q_2 = \sin(\theta_1/2) \sin(\theta_2/2)$$

$$q_3 = \sin(\theta_1/2) \cos(\theta_2/2)$$

$$q_4 = \cos(\theta_1/2) \sin(\theta_2/2)$$

oppure l'istruzione definita dagli angoli di Eulero:

$$(\theta_2, 90^\circ, \theta_1)$$

Nell'applicazione in esame la sonda deve essere orientata verso il basso e quindi si prendono in considerazione solo le rotazioni che portano P_1 sulla superficie della semisfera i cui punti hanno $z < z_0$; si ha:

$$0 < \theta_1 < 180^\circ$$

$$-180^\circ < \theta_2 < 180^\circ$$

4. SVILUPPO E SPERIMENTAZIONE SUL SISTEMA SVP2000

L'attività svolta sul sistema SVP2000 ha comportato lo studio e lo sviluppo di:

- programmi di gestione delle primitive SVP
- libreria di gestione file immagini con protocollo TIFF
- File Systems per immagini
- programmi per l'enhancement di immagini
- software per misure.

4.1 Programmi di gestione delle primitive SVP

Il sistema SVP2000 è una stazione di lavoro specializzata per l'acquisizione ed il trattamento di immagini e segnali aperta sia come architettura interna (inserimento di CPU specializzate, ampliamenti di memoria video e dati, sottosistemi elaborativi, software orientato), sia come interfacciamento verso sistemi e dispositivi periferici standard o specializzati (generatori ed acquisitore ad US, telecamera di area, telecamera di linea, termocamere, strumentazioni di misura, dispositivi a controllo numerico, ecc.).

Questa grande flessibilità ha comportato di conseguenza: un'architettura software basata sull'impiego di più sistemi operativi per la gestione generale (Unix, VME Exec, X-Window) e gestori specializzati delle risorse orientate al trattamento di segnali e immagini; librerie di funzioni e comandi per il colloquio uomo-macchina e la con figurazione dinamica delle risorse interne.

L'impiego di un sistema aperto di questo tipo presenta daltronde l'inconveniente di richiedere una dettagliata conoscenza dell'architettura del sistema sia hardware che software e la necessità di concatenare molti comandi elementari per realizzare funzioni anche relativamente semplici.

Ad esempio per eseguire la funzione di acquisizione da una telecamera di area è necessario svolgere la seguente serie di azioni elementari:

- posizionarsi sul range selezionato
- selezionare il canale di input.
- selezionare l'origine e il tipo del sincronismo di input
- selezionare i parametri dei convertitori analogici digitali (DACs)
- selezionare il tipo di telecamera (di linea o di area)
- configurare la memoria video (VRAM)
- selezionare lo standard del monitor
- selezionare il frame su cui acquisire
- attivare la visualizzazione
- acquisire in VRAM
- disattivare la visualizzazione

Per ovviare a questo inconveniente, lasciando intatte le potenzialità del sistema, si è pensato di sviluppare diversi ambienti operativi ciascuno dei quali realizzasse un sottoinsieme delle possibilità offerte dal sistema per soddisfare in modo mirato tutte e sole le esigenze elaborative di un particolare tipo di applicazione.

Tenendo presenti le applicazioni previste dalla collaborazione scientifica è prevedibile la necessità di acquisizione segnali di tipo mono o pluridimensionali relativi ad immagini monocromatiche o multicromatiche di formati diversificati. In particolare è prevista l'acquisizione di segnali video monocromatici con standard CCIR o RS170, e segnali US di tipo asincrono.

Nel primo caso il formato dei dati è caratterizzato da matrici di 512•512 elementi espressi su 8 bit, nel secondo caso invece da matrici 2D o 3D di grandi dimensioni comunque variabili.

Si sono quindi individuati tre diversi ambienti operativi:

- un ambiente per la gestione di immagini monocromatiche con formato max di 512•512 elementi di 8 bit;
- un ambiente per la gestione di immagini tricromatiche, con formato max di 512•512 elementi di 24 bit;
- un ambiente per la gestione di immagini o segnali monocromatici con formato max di 512•512•32 elementi di 8 bit.

Per realizzare questi tre diversi ambienti, sono stati definiti dei macrocomandi interagenti da terminale con il sistema, i quali a partire da

quegli elementi disponibili mirano a raggrupparli, riducendo il numero dei parametri da essi richiesti, e ad eliminare tutte le opzioni e le ridondanze che nel particolare ambiente non servono. Dove questa operazione non era possibile, sono stati creati dei nuovi comandi specializzati e ottimizzati per le funzioni di sintesi.

Questi macrocomandi sono richiamabili da un'interfaccia sotto UNIX a menù gerarchici per accedere all'ambiente in maniera guidata. La figura seguente dà un'idea del lavoro svolto per l'ambiente monocromatico, mostrando i quattro menù principali e riportati in fig. 5.

In questo nuovo ambiente per eseguire un'operazione di acquisizione occorre:

- selezionare il canale di input
- selezionare il frame su cui acquisire
- selezionare i parametri dei DACs
- visualizzare e acquisire

Nel prossimo futuro si prevede di sostituire questa interfaccia gerarchica con un'interfaccia grafica, basata su X-Window e Motif; a tale scopo sono state acquistate per il sistema SVP2000 oltre al software di base necessario anche due stazioni grafiche dotate ciascuna di tastiera, mouse e monitor 1024•768 16 colori.

La figura 6 mostra la configurazione dei posti di lavoro e delle risorse primarie del sistema SVP2000.

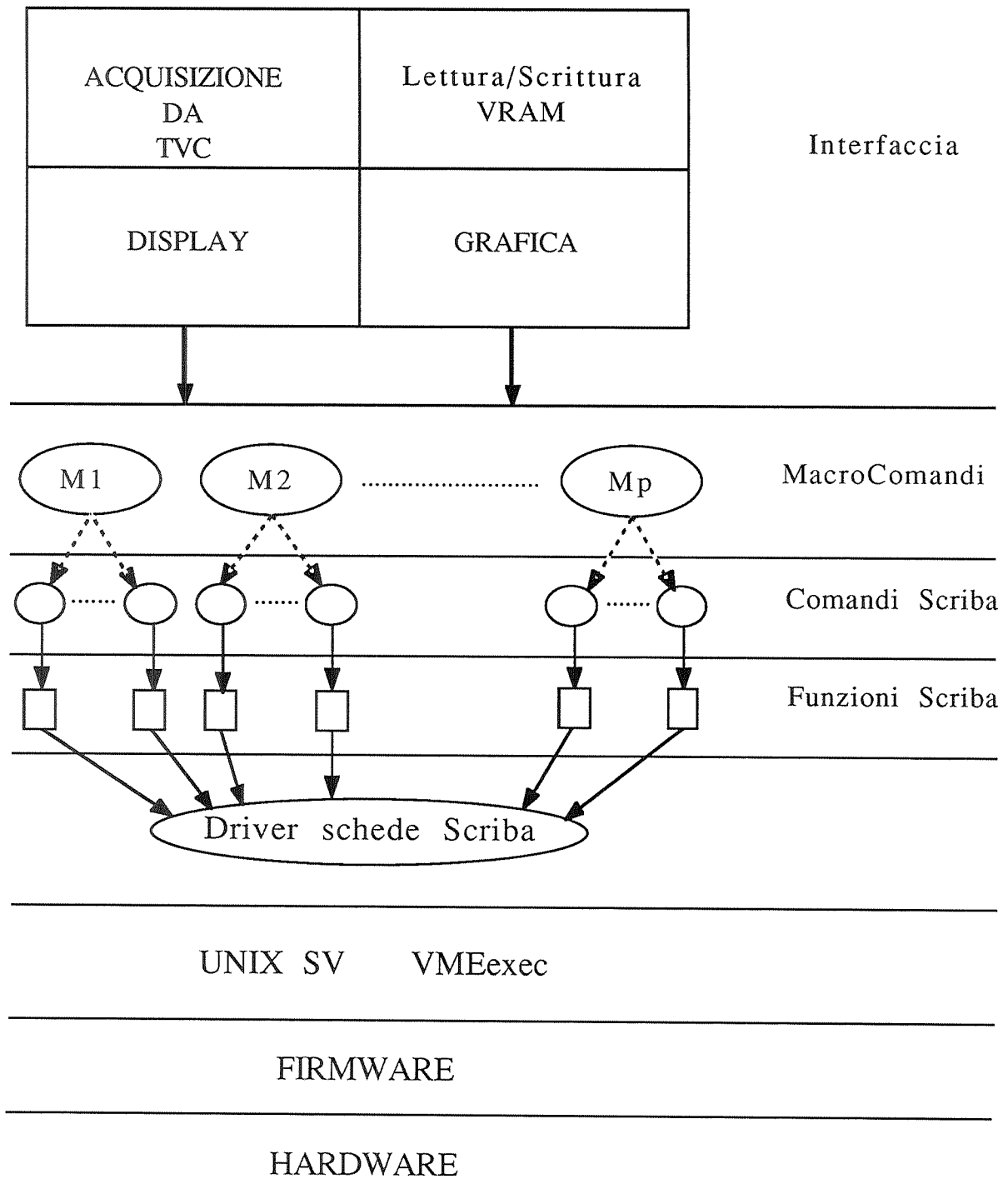


Figura 5

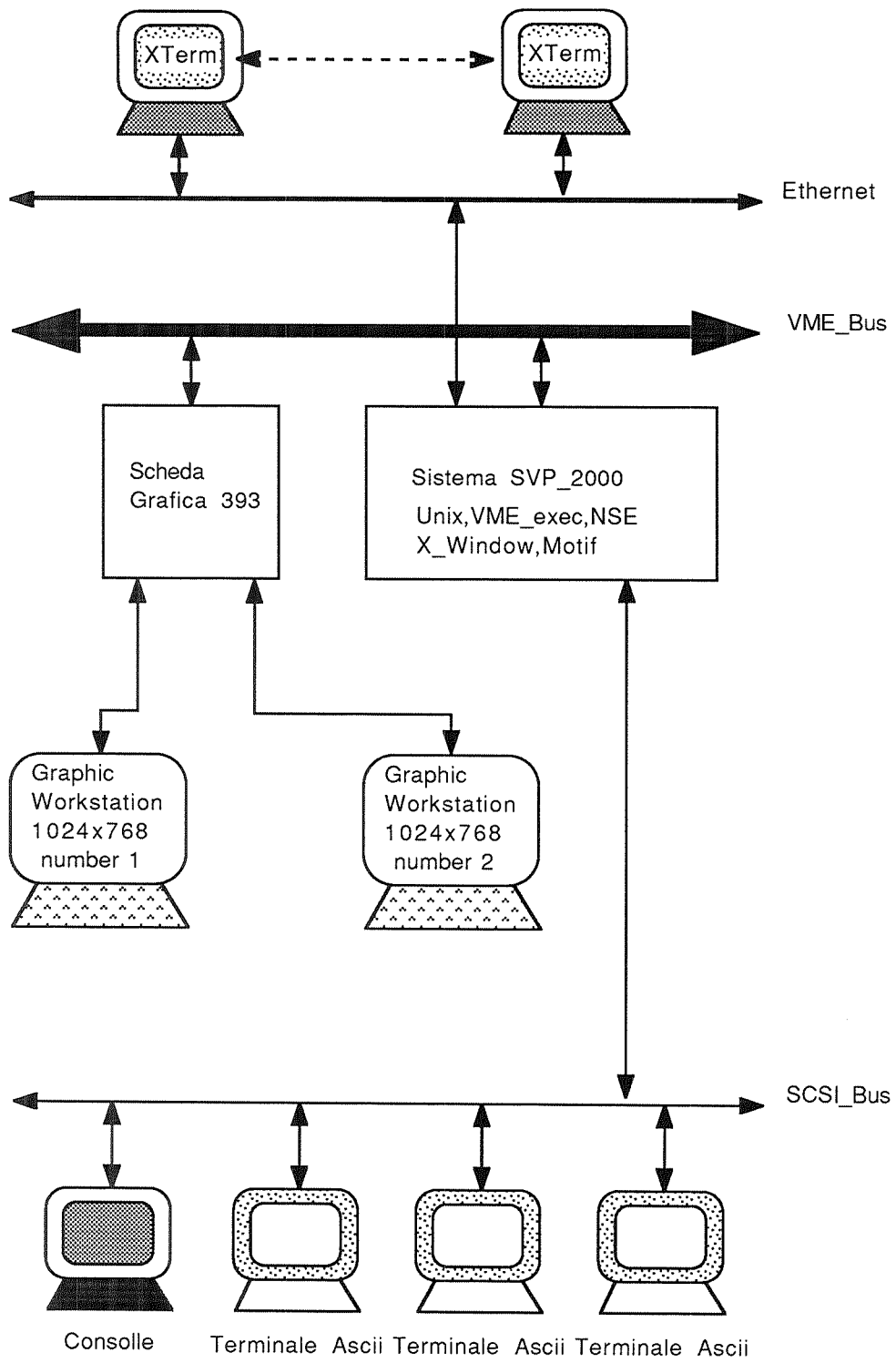


Figura 6

4.2 Definizione di un protocollo TIFF

Il sistema SVP2000 viene usato per il trattamento di dati, segnali ed immagini ricavate con tecniche diverse; i dati di varia provenienza e formato devono essere organizzati ed archiviati secondo un opportuno standard.

Il primo problema da affrontare è la scelta di un formato di memorizzazione per i dati ricavati. Nel nostro caso è stato scelto il formato T.I.F.F. (Tag Image File Format) in quanto sta diventando uno standard per i formati grafici pittorici, il che garantisce un facile interscambio di dati con altri sistemi, ed inoltre la sua estrema flessibilità consente di adattarlo a tutti i tipi di immagini che si prevede di trattare.

La struttura generica di un file Tiff consta essenzialmente di tre parti logiche (v. fig. 7a, 7b):

- un Header che serve ad identificare il file come un'immagine in formato Tiff.
- un numero variabile di Image File Directory ognuna delle quali descrive un'immagine. In genere vi sarà una sola I.F.D. in quanto nella maggioranza dei casi in un file Tiff si memorizza una sola immagine. D'altronde la possibilità di avere più I.F.D. consente di memorizzare in solo file tutta una qualsiasi sequenza logica d'immagini.
- i dati fisici costituenti l'immagine.

Struttura Logica del formato TIFF

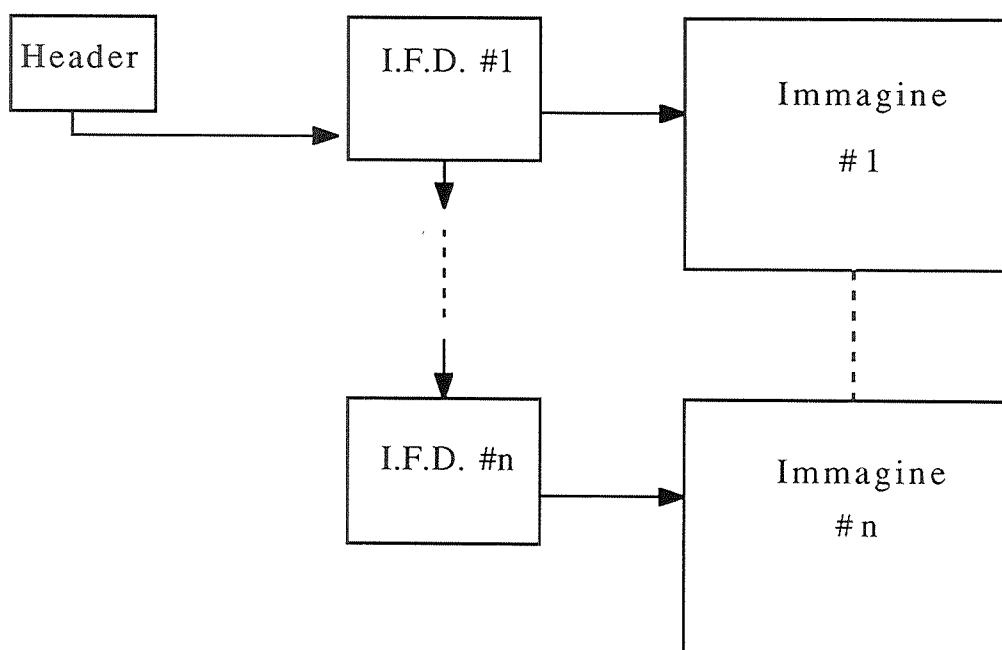
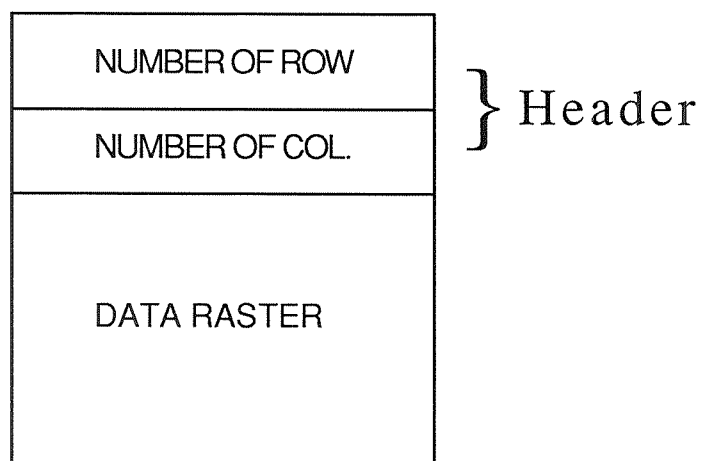


Figura 7a

Software Tiff

Per la generazione e la gestione di file Tiff è stato progettato un pacchetto software strutturato a tre livelli: high, mid e low levels:

Low level: libreria di funzioni per la creazione, lettura, modifica e cancellazione di file Tiff.

Mid level: insieme di utility interattive per il controllo, l'interpretazione, l'editing, il dumping dei file Tiff ed un pacchetto di funzioni e/o applicativi che eseguono conversioni da Tiff ad altri formati grafici e viceversa

High level: libreria di funzioni strettamente collegata con il File System inerenti operazioni sui singoli file Tiff.

4.3 File Systems per immagini

In generale un sistema di Image Processing riceve in input una immagine espressa come una matrice intera bidimensionale, su di essa esegue le sue operazioni ed in uscita può fornire o una nuova matrice immagine, o una elaborazione parziale costituita da una immagine reale, oppure tutto un corollario di dati estratti dall'immagine in esame come look-up table, contorni dell'immagine, parametri statistici, commenti etc.

Tutti questi dati possono a loro volta essere elaborati in maniera ricorsiva, generando nuovi dati che comunque rientrano sempre nelle medesime classi che possiamo identificare nelle seguenti:

- matrici immagine 2D intere
- matrici immagine 2D reali
- vettori 1D e 2D (lut, contorni etc.)
- informazioni testuali

Per quanto riguarda i tipi di immagine trattati, sono state previste le seguenti classi:

- Immagini Singole o Monocromatiche rappresentate con matrici 1D intere.
- Immagini Multispettrali, consistenti di insiemi di immagini singole, connesse nel dominio delle frequenze.

- Immagini Sequenziali Temporali, consistenti di insiemi di immagini singole o multispettrali, connesse nel dominio del tempo.
- Immagini Grafiche, costituite da patterns di coordinate 3D.

È evidente quindi, che uno dei problemi da risolvere in un sistema di Image Processing, è la gestione di tutti questi dati e la loro archiviazione su supporti di memoria permanente (dischi magnetici, dischi ottici).

Assumiamo che ciascun dato logico con ogni informazione utile per identificarlo sia memorizzato su di un singolo dato fisico (file o record). Questo implica che bisogna definire per ogni dato trattato un formato di memorizzazione. Per quanto riguarda il dato immagine pittorica, che sicuramente rappresenta la struttura più ricorrente, è stato scelto di utilizzare lo standard TIFF (si veda il paragrafo relativo), per gli altri tipi di dati; matrici immagini reali, tabelle di LUT, informazioni testuali ed altri, si potrà utilizzare un formato privato molto semplice.

Con l'assunzione di associare il dato logico al dato fisico (file) implicitamente si dà una soluzione sia al modo di ordinare tra loro i dati che ai meccanismi di recupero dell'informazione, infatti per tutte queste operazioni si può utilizzare, con pochi ritocchi, la libreria di routine e comandi associata al file system del sistema operativo su cui verranno realizzati gli applicativi che nel caso del sistema SVP2000 è UNIX SV.

Vediamo una descrizione di questo File System dal punto di vista dell'utente.

Sotto UNIX i file sono organizzati gerarchicamente in una struttura ad albero in cui i nodi sono directory e le foglie sono i file stessi.

Il cammino per arrivare dalla directory radice ad un file è detto "pathname". Ogni file è quindi identificato da un pathname, da un nome e da una estensione (esempio: pathname=/user/image nome=retina estensione=mono). L'estensione può essere utilizzata per identificare la classe di appartenenza del dato logico contenuto nel file.

Per i nostri dati si utilizzeranno queste estensioni:

-Immagine Singola	*.mono
-Immagine Multispettrale	*.multi
-Immagine Sequenziale	*.seq <u>n</u> ove <u>n</u> =numero delle immagini
-Immagine Grafica	*.graf
-Matrici immagine reale	*.real
-Tabelle di LUT	*.lut
-Informazioni testuali	*.text

Inoltre nel file system di UNIX ad ogni file sono associati dei diritti di accesso e la data di creazione.

Rispettando queste poche regole è facilissimo poter navigare in una tale struttura considerando solo i file di nostro interesse.

4.4 Programmi di enhancement

Il sistema SVP2000 è dotato di tavole di look-up di uscita mediante le quali si può modificare immediatamente l'immagine rappresentata sul monitor senza alterare i dati in memoria: in questa maniera si può compiere agevolmente un'operazione di enhancement che può facilitare l'interpretazione dell'immagine e si possono modificare vari parametri sino a raggiungere il risultato migliore: è possibile a questo punto modificare anche i dati in memoria per archiviare l'immagine manipolata.

4.4.1 Modifica della tavole di look-up

Il sistema SVP2000 ha a disposizione un insieme limitato predefinito ma permette la realizzazione e l'archiviazione di un gran numero di tavole disponibili dall'utente in funzione dei requisiti applicativi. Sono stati pertanto definiti e sviluppati:

- un insieme di LUT fisse ottenibili attraverso un menù specifico
- un insieme di programmi per l'attivazione e il controllo di LUT modificabili interattivamente
- un insieme di programmi per la generazione, l'archiviazione e il caricamento di LUT.

Tutti i programmi sono stati scritti in linguaggio C sotto sistema operativo UNIX System KSH: di seguito si riporta una descrizione sintetica dei programmi realizzati.

FIXLUT: - Il programma abilita il caricamento di tavole di look-up predefinite; le tavole selezionate sono generate *run-time*.

Dati in Ingresso:

Numtav = codice di selezione della tavola.

Le tavole selezionabili sono:

- 1 - Warm Body
- 2 - Cold Body
- 3 - Stend
- 4 - Alpha
- 5 - Positive Ramp
- 6 - Negative Ramp
- 7 - Logarithmic

Solo per i codici 4, 5, 6 e 7:

Icolor = colore della look-up in cui viene caricata la tavola prescelta.

Sono possibili le combinazioni:

- 1 - red
- 2 - green
- 3 - blue
- 4 - red + green
- 5 - red + blue
- 6 - green + blue
- 7 - red + green + blue

Dopo l'esecuzione del comando il programma si pone in attesa di un nuovo dato in ingresso.

Rispondendo 'Yes' alla domanda '**Continue?:**' è possibile caricare una nuova tavola di look-up.

SHILUT: - Il programma abilita il caricamento di tavole di look-up predefinite e il loro scorrimento spaziale sulla scala dei valori (0 ÷ 255); sono caricate due tavole distinte, di cui una fissa e l'altra dinamica.

Dati in Ingresso:

Frame = frame-buffer di lavoro.

Rispondendo 'Yes' alla domanda 'Colour scale? (y/n):' viene visualizzata in basso sul monitor grafico-pittorico la scala iniziale di riferimento dei livelli di rappresentazione dal nero al bianco.

Iscal = codice di selezione della scala di riferimento fissa:

- 1 = none
- 2 = positive ramp
- 3 = negative ramp
- 4 = arbitrary

Se *Iscal* = 4 è possibile costruirsi *run-time* una tavola arbitraria.

Se invece *Iscal* ≠ 4, allora:

Iscall = codice di selezione della scala *dinamica*:

- 1 = grey scale
- 2 = colour scale

Se *Iscall* = 1 allora:

IQ = scala di quantizzazione (1-256)

IEL = elementi per intervallo (nota: $IQ \times IEL$ deve essere ≤ 256)

Se *Iscall* = 2 allora:

ICS = tipo di scala:

- 1 = saturated colours
- 2 = non saturated colours

Se *ICS* = 1 allora:

IELI = elementi per colore (1-2-4-8-16-32)

Se ICS=2 allora:

IQC = colori di quantizzazione (1 ÷ 16)

IEL2 = elementi per colore (nota: $IQC \times IEL2$ deve essere ≤ 256)

ICC = codice di selezione della sequenza dei colori sulla tavola dinamica. È visualizzata sul monitor alfanumerico di lavoro una lista di codici associati ai colori possibili. È quindi possibile selezionare la sequenza in modo predefinito od arbitrario. In quest'ultimo caso, la sequenza dei codici di colori deve essere specificata esplicitamente.

Rispondendo 'Yes' alla domanda 'Table slipping? (y/n):' viene richiesto anche il parametro

ISS = passo di scorrimento dinamico (1 ÷ 256).

Lo scorrimento delle tavole, a destra od a sinistra è controllato dalle frecce direzionali sulla tastiera.

Premendo il tasto <CR> si esce dalla fase di scorrimento delle tavole dinamiche e viene chiesto se si vuol modificare il passo di spostamento: in tal caso deve essere specificato il nuovo passo (1 ÷ 256) ed il programma riprende dalla fase di scorrimento.

Rispondendo 'Yes' alla domanda finale 'Continue?:' è possibile eseguire nuovamente il programma

SATLUT: - Il programma permette di caricare tavole di look-up prestabilite e di spostarne dinamicamente una per ottenere un viraggio dei colori.

Dati in Ingresso:

ICOD = codice di selezione della sequenza dei colori primari (R,G,B). La sequenza può essere scelta tra le seguenti:

- 1 = red, green, blue
- 2 = red, blue, green
- 3 = green, red, blue
- 4 = green, blue, red
- 5 = blue, green, red
- 6 = blue, red, green

LUTH = soglia di luminosità ($0 \div 255$)

Rispondendo 'Yes' alla domanda 'Level 0 = black; Level 255 = white, Ok? (y/n):' il valore 0 viene rappresentato NERO ed il valore 255 BIANCO.

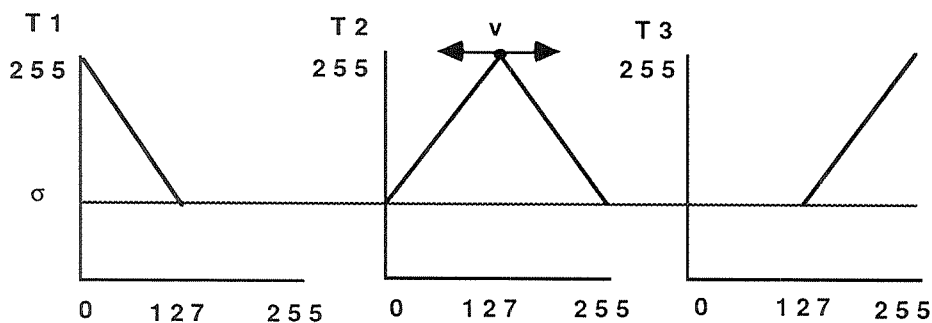


Figura 8

Nella figura 8 è mostrato come vengono costruite le tavole di look-up per ogni colore primario. T1 indica la prima tavola selezionata, T2 la seconda e T3 l'ultima.

T1 e T3 sono fisse e statiche, mentre T2 può essere variata dinamicamente con passo unitario, spostando il suo vertice *v* nei due sensi con le frecce direzionali.

Premendo il tasto <CR> si esce dalla fase di scorrimento delle tavole dinamiche

Rispondendo 'Yes' alla domanda finale 'Continue?:' è possibile eseguire nuovamente il programma.

STRIP: - Il programma permette di visualizzare la scala cromatica di riferimento sul monitor grafico-pittorico.

Dati in Ingresso:

FRAME = frame-buffer di lavoro.

Dopo la selezione del frame-buffer sul quale si vuole visualizzare la scala cromatica viene fatta la richiesta '**Clear frame-buffer? (y/n):**', rispondendo '**Yes**' a tale domanda la memoria video di lavoro viene automaticamente azzerata.

DIRLUT: - Il programma visualizza sul monitor alfanumerico di lavoro la lista dei programmi disponibili che operano sulle tavole di look-up.

ERASELUT: - Il programma cancella una tavola di look-up dall'archivio ARCLLOT di sistema.

Dati in Ingresso:

FN = nome della look-up che si vuol cancellare.

Nell'eventualità che la tavola non sia presente nell'archivio ARCLOCK di sistema, viene segnalato un messaggio di errore.

RENAMELUT: - Il programma permette il cambiamento del nome associato ad una tavola di look-up.

Dati in Ingresso:

OLDNAME = nome della look-up di cui si vuol cambiare il nome.

NEWNAME = nuovo nome da associare alla look-up.

Nell'eventualità che la tavola non sia presente nell'archivio ARCLUT di sistema viene segnalato un messaggio di errore.

BUILLUT: - Il programma permette la costruzione di tavole di look-up arbitrarie, in maniera grafica, su monitor, tramite joystick, e ne consente o il caricamento diretto o la memorizzazione nell'archivio specifico ARCLUT.

Dati in Ingresso:

FRAME = frame-buffer della scheda grafica di sistema prescelto per l'esecuzione.

Dati in Uscita:

ICOL = codice di colore prescelto per il caricamento:

- 1 = red
- 2 = green
- 3 = blue
- 4 = red + green + blue
- 5 = file su disco

Nel caso *ICOL*=5:

LUT = nome della look-up da inserire in ARCLUT

All'inizio della esecuzione, viene data all'utente la possibilità di azzerare il frame-buffer di lavoro su cui il programma visualizza un reticolo ed un cursore, da pilotarsi con il joy-stick, per la costruzione della LUT. Sul monitor alfanumerico vengono stampate le modalità d'uso dello strumento.

Per ogni intervallo di creazione della tavola determinato, viene stampato l'indirizzo dell'ultima cella riempita ed il valore del suo contenuto. Sul monitor grafico-pittorico appare il segmento relativo alle celle riempite con i rispettivi valori dei quali viene chiesta conferma.

In caso di errore è possibile ripetere l'operazione a partire dall'ultima cella riempita convalidata. La costruzione della tavola termina con l'assegnamento di un valore all'ultimo indirizzo del reticolo.

PRINTLUT: - Il programma stampa sul monitor alfanumerico di lavoro i valori di una tavola di look-up letta dall'archivio ARCLUT di sistema e, se richiesto, ne disegna il grafico sul monitor pittorico.

Dati in Ingresso:

LUT = nome della look-up da selezionare.

Rispondendo 'Yes' alla domanda 'Look-up table graphics? (y/n):'

FRAME = frame-buffer di lavoro

Nel caso di richiesta del grafico della tavola, è possibile inizializzare il frame-buffer di lavoro.

4.4.2 Modifica dei dati in memoria

Un altro pacchetto di programmi consente di modificare in modo interattivo l'intera immagine o porzioni di essa, sempre ai fini di una migliore interpretazione visiva. La modifica viene fatta agendo sull'istogramma, sulla scala dei livelli di uscita oppure mediante una convoluzione della matrice in memoria con maschere selezionabili all'interno di un menù.

Di seguito si riporta una sintetica descrizione dei programmi; la relazione completa è contenuta in [2].

HISMOD: - Hismod sta per "HIStogram MODification", ovvero la modifica dell'istogramma, un metodo abbastanza potente per aumentare il contrasto in determinate zone della dinamica. L'elaborazione consiste

nell'accumulare i livelli di grigio l in varie classi in modo che l'istogramma risultante approssimi il più possibile una data curva $H(l)$.

Le curve approssimabili con Hismod sono riportate in tab. I:

Curva	H	n
(1)	$H(l) = k \cdot l^n$	$n < 0$
(2)	$H(l) = k \cdot l^n$	$n > 1$
(3)	$H(l) = k \cdot (l_{\max} + l_{\min} - l)^n$	$0 < n < 1$
(4)	$H(l) = k \cdot l^n$	$0 < n < 1$
(5)	$H(l) = k \cdot \ln(l)$	
(6)	$H(l) = k \cdot l^n$	$n = 0$
(7)	$H(l) = k \cdot l^n$	$n = 1$
(8)	$H(l) = k \cdot (l_{\max} + l_{\min} - l)^n$	$n = 1$

Tab. I

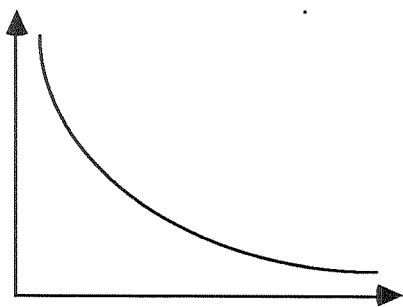
dove k è una costante e n il parametro modificabile interattivamente.

In fig.9 sono riportati i grafici delle curve $H(l)$.

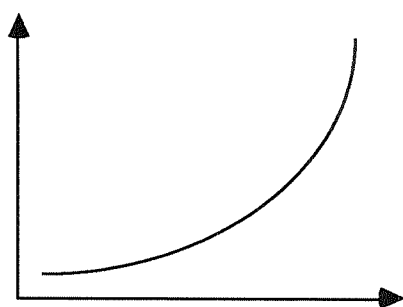
Hismod, in base alla funzione e al valore del parametro scelti, produce un'immagine modificata agendo sulle tavole di look-up d'uscita del sistema.

In ogni momento è possibile:

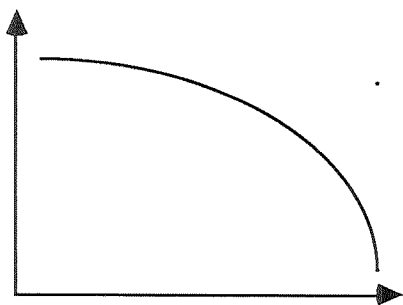
- selezionare frame-buffer
- selezionare il numero dei livelli di grigio in uscita
- selezionare la funzione H
- modificare il parametro n
- rappresentare l'istogramma attuale
- rappresentare l'immagine originale e richiamare il suo istogramma
- scrivere in memoria video l'immagine modificata.



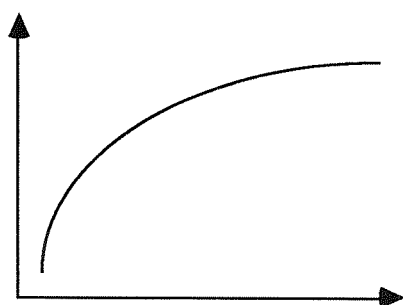
(1)



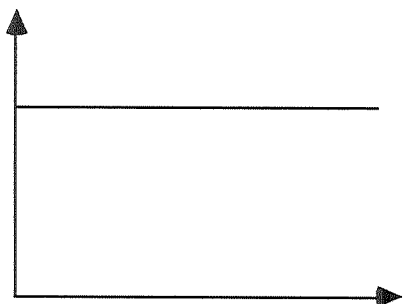
(2)



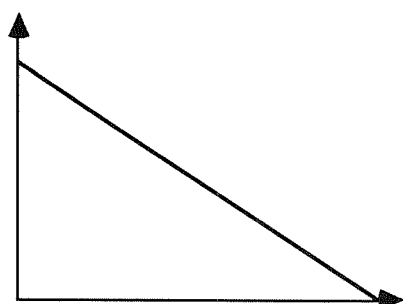
(3)



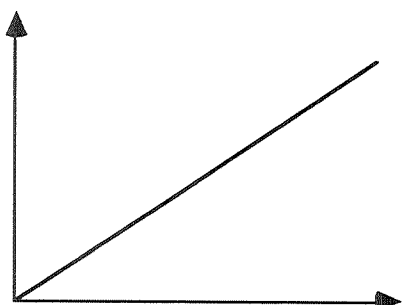
(4)



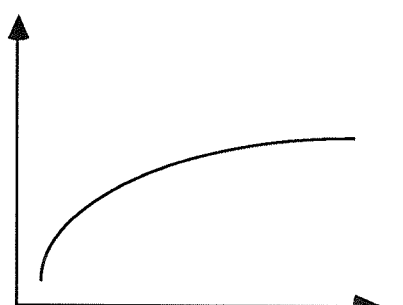
(5)



(6)



(7)



(8)

Figura 9

WINMOD: - Winmod sta per "WINdow MODification", ed è la versione di Hismod per finestre sull'immagine, e quindi per Winmod valgono le stesse considerazioni fatte per Hismod; tra le funzionalità va aggiunta quella di definizione di una finestra rettangolare sull'immagine.

LINEAR: - Linear sta per "LINEAR modification", ovvero modifiche lineari del valore di luminanza x dei pixel. Il programma Linear permette di scegliere una qualsiasi spezzata (crescente) come funzione di mapping dei pixel: l'intervallo di definizione x è compreso tra 0 e 255, così come l'intervallo dei valori di uscita y .

La fig. 10 mostra la funzione lineare di mapping $y=x$ caricata all'accensione del sistema, mentre la fig. 11 mostra un esempio di mapping con funzione continua a tratto. In questo caso i livelli d'ingresso $x \leq x_1$ sono proiettati sul livello d'uscita $y=0$ e i livelli $x \geq x_2$ sono proiettati in $y=255$; si perdono quindi le informazioni relative agli intervalli di x suddetti ma si esalta al massimo il contrasto e quindi si migliora la comprensibilità per i valori dei pixel compresi tra x_1 e x_2 .

Linear permette la definizione di una qualsiasi spezzata muovendo una coppia di cursori min e max che individuano i punti x_1 e x_2 definiti in precedenza. Si possono perciò verificare immediatamente sul monitor i risultati di compressioni/espansioni di scala sul range (x_1, x_2) . Normalmente, sono visualizzati in blu i pixel a inizio scala (0) e in rosso quelli con valore massimo (255).

In ogni momento sono possibili le seguenti operazioni:

- selezione del frame-buffer
- spostamento del cursore min
- spostamento del cursore max
- shift dei valori di luminanza
- visualizzazione dell'istogramma risultante e del grafico della funzione di mapping
- abilitazione/disabilitazione dei falsi colori per evidenziare gli estremi della scala
- richiamo dell'immagine originale e selezione del suo istogramma
- scrittura in memoria video dell'immagine attuale.

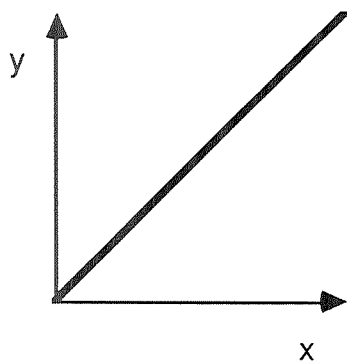


Figura 10

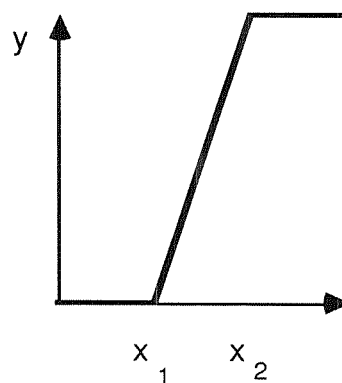


Figura 11

WINLIN: - Winlin sta per "WINdow LINear modification" ed è la versione di Linear operante su finestre d'immagine. Sono possibili le stesse operazioni di Linear.

LOG: - Log sta "LOGaritmic transformation", ovvero la modifica secondo una legge logaritmica dei valori dei pixel.

Le funzioni di trasformazione tra cui scegliere sono 2:

$$T_1(l) = \log_{10}(l)$$

$$T_2(l) = -\log_{10}(l)$$

dove l sono i livelli di grigio d'ingresso; i risultati della trasformazione sono normalizzati tra 0 e 255.

In ogni momento è possibile:

- selezionare frame buffer
- selezionare funzione
- rappresentare l'istogramma e il grafico di T
- richiamare l'immagine originale e rappresentare il suo istogramma
- scrivere in memoria video l'immagine attuale.

Come si può notare, questo tipo di trasformazione non ha parametri modificabili.

WINCON: - Wincon sta per "WINdow CONvolution", cioè convoluzione tra una sottomatrice dell'immagine delimitata da una finestra mobile e una maschera definita da una matrice $N \cdot N$, con $N=1, 3, 5, 7$. Wincon permette l'editing, l'archiviazione e la prova di maschere da utilizzarsi per vari tipi di filtraggio bidimensionale.

Con Wincon è perciò possibile:

- selezionare frame-buffer
- definire e muovere le finestre entro cui avviene la convoluzione
- editare una maschera
- archiviare la maschera corrente
- leggere dall'archivio una maschera
- scrivere in memoria video l'immagine attuale.

4.5 Moduli per indagine statistica

Nell'analisi di immagini digitali si è rivelato di fondamentale importanza lo studio dei parametri statistici calcolati sui valori radiometrici dell'immagine considerata.

È stato quindi progettato e sviluppato un pacchetto software costituito da una serie di moduli che consentono l'esecuzione di diversificati metodi di indagine e di analisi.

Il software è stato realizzato e sviluppato con Personal PC IBM compatibile al fine di sperimentarne l'efficienza ed è attualmente in fase di trasferimento sul sistema SVP2000.

I moduli sono stati progettati ed implementati in modo da permetterne un semplice utilizzo da parte dell'utente grazie anche all'uso di dispositivi di interazione come il mouse. Per far ciò, infatti, oltre a cercare di ridurre al minimo le opzioni selezionabili, che sfruttano controlli automatici interni al software stesso, le stesse scelte possono essere direttamente selezionabili sia da tastiera che da mouse.

Le diverse procedure sono state implementate in linguaggio C.

Il pacchetto "Moduli elaborativi per l'indagine statistica" è attualmente composto da otto moduli suddivisi in due classi:

- 1) misure dei valori radiometrici dell'immagine memorizzata su Video Ram e loro rappresentazione grafica;
- 2) calcolo di parametri statistici e loro rappresentazione grafica.

Alla prima classe appartengono i moduli "Nlinea", "Stereo", "Polyline" e "Polygon", mentre della seconda fanno parte i moduli "Histo", "Momenti", "Mappa" e "Histo2".

Una semplice interfaccia ingloba gli applicativi costituenti il pacchetto, in modo da permettere rapidi ed efficaci passaggi fra i diversi moduli e quindi di poter ottenere un'analisi completa dell'immagine considerata.

Parametri statistici

I parametri statistici calcolati su di una immagine monotonale nel campo spaziale, sono essenzialmente delle misure di funzioni discrete.

Un'immagine monotonale di dimensioni 512•512 con 256 possibili livelli di grigio è descritta da una funzione discreta a due variabili

$$L_{i,j} = \text{luminanza del pixel } (i,j) \quad 0 < i,j < 511 \quad 0 \leq L_{i,j} \leq 255$$

Dalla funzione $L_{i,j}$ si può ricavare un'altra funzione discreta unidimensionale $I(l)$ così definita

$$I(l) = \text{Numero dei pixel } (i,j) \text{ t.c. } L_{i,j} = l$$

La funzione $I(l)$ rappresenta la distribuzione della frequenza dei valori radiometrici dell'immagine ed il suo grafico è detto istogramma. Al posto di $I(l)$ si può considerare la funzione $F(l) = \frac{I(l)}{M}$ dove M è il numero complessivo dei pixel dell'immagine. $F(l)$ rappresenta la distribuzione della frequenza relativa dei valori radiometrici.

Dalle funzioni $L_{i,j}$ e $I(l)$ si possono ricavare diversi parametri, che descrivono la forma dei rispettivi grafici e da cui si intuiscono diverse caratteristiche dell'immagine.

Parametri ricavati da $L_{i,j}$

- 1) $\min L_{i,j}$: minimo valore radiometrico
- 2) $\max L_{i,j}$: massimo valore radiometrico
- 3) $\sum_{i,j=0}^{511} L(i,j)$: integrale

Parametri ricavati da $I(l)$

- 1) l t.c. $I(l)$ è minimo e $\neq 0$
- 2) l t.c. $I(l)$ è massimo: moda
- 3) $\bar{L} = \sum_{l=0}^{255} l \cdot F(l)$ media aritmetica
- 4) $s = \sqrt{\sum_{l=0}^{255} (l - \bar{L})^2 \cdot F(l)}$ scarto quadratico medio
- 5) $a_3 = \frac{\sum_{l=0}^{255} (l - \bar{L})^3 \cdot F(l)}{s^3}$ skewness
- 6) $a_4 = \frac{\sum_{l=0}^{255} (l - \bar{L})^4 \cdot F(l)}{s^4}$ kurtosis

STEREO: - Questo pacchetto visualizza in falsa tridimensionalità lo stereogramma dei valori radiometrici di una finestra dell'immagine precedentemente selezionata dall'utente.

L'utente stesso ha la possibilità di scegliere due modi di scansione della finestra da analizzare:

- il primo, *NORMALIZZATO*, prevede la visualizzazione di due righe su tre della porzione di immagine definita e, per ogni riga, di un pixel ogni due;
- il secondo, *ZOOMX2*, visualizza tutte le righe e tutte le colonne di una finestra fissa, che per questioni di capacità massime dell'area destinata al grafico, ha dimensione 256•341 pixel.

Un'altra opzione fa sì che l'utente possa scegliere quattro diversi orientamenti nello spazio (*BASSO*, *ALTO*, *DESTRA*, *SINISTRA*), per la visualizzazione dello stereogramma. La lettura della matrice immagine sarà quindi condizionata anche da tale scelta.

NLINEA: - Dopo aver selezionato un segmento di retta sull'immagine digitale scelta per l'analisi, il programma disegna il grafico relativo ai livelli di grigio dei pixel considerati.

Le dimensioni del segmento possono variare da un minimo di 1 pixel ad un massimo di 512 e il segmento stesso può essere orientato a piacere al momento della definizione.

La memoria di immagine non viene modificata in quanto, sia nella visualizzazione che nella cancellazione del segmento stesso, si usa fare uno XOR con il valore del pixel corrispondente della matrice immagine.

Prima di visualizzare il grafico relativo, si cancella il cursore di linea sull'immagine e si memorizzano in un array i valori radiometrici dei punti appartenenti alla linea sfruttando l'algoritmo di Bresenham. Quindi, dopo aver visualizzato il grafico, si calcolano i parametri statistici che verranno riportati nell'area di schermo apposita.

HISTO: - Il modulo Histo disegna la curva relativa alla distribuzione dei valori radiometrici presenti in un'area dell'immagine digitale precedentemente selezionata.

Nella parte di schermo destinata a contenere il grafico, viene visualizzato solo l'asse delle ascisse con i valori possibili per i pixel.

La curva delle frequenze viene calcolata e disegnata solo dopo aver cancellato la finestra selezionata sull'immagine, per non modificare i dati contenuti nella memoria di quadro.

L'occorrenza massima, insieme ad altri parametri, viene riportata nella parte di schermo destinata alle informazioni numeriche utili all'utente.

MOMENTI: - Lo scopo di questo modulo è quello di calcolare e visualizzare contemporaneamente i grafici relativi ai parametri statistici *sigma*, *skewness* e *kurtosis* relativi ad un'area dell'immagini di lavoro precedentemente scelta dall'utente.

La parte di schermo destinata ai grafici si presenta suddivisa in tre zone, una per ogni curva, con i relativi assi cartesiani. Per una maggiore semplicità di comprensione e di visualizzazione, si è scelto di usare tre colori diversi per i grafici, e, a fianco di ciascuno di essi è riportato il nome del parametro a cui fanno riferimento.

Nell'area di schermo destinata ai valori numerici vengono invece visualizzati gli stessi parametri statistici relativi questa volta all'intera finestra selezionata sull'immagine. Per la selezione dell'area di immagine su cui lavorare valgono le stesse ipotesi fatte per le procedure precedenti.

Oltre ai parametri di tipo statistico, si hanno anche i dati numerici relativi ai vertici della finestra selezionata, alle sue dimensioni e all'area in pixel corrispondente, e al minimo e al massimo valore di pixel fra tutti quelli esaminati.

MAPPA: - Il modulo visualizza la mappa degli istogrammi di tutte le righe o di tutte le colonne di una finestra scelta su un'immagine digitale.

L'asse x del grafico è di riferimento per i valori possibili dell'istogramma, mentre l'asse y riporta il numero di righe o di colonne considerati sull'immagine digitale.

Oltre alla mappa degli istogrammi, il modulo rappresenta graficamente anche l'istogramma dell'intera immagine (512•512 pixel) e l'istogramma di una riga o colonna dell'immagine scelta dall'utente. Questi due ultimi grafici non pretendono di dare informazioni accurate, ma vogliono solo essere di supporto alla mappa per una visione più completa e generale del problema.

POLYLINE: - Il modulo Polyline calcola e visualizza l'istogramma dei valori radiometrici di una poligonale chiusa o aperta selezionabile dall'utente.

I vertici della poligonale vengono definiti utilizzando il mouse o i tasti freccia da tastiera.

Oltre alla visualizzazione dell'istogramma, vengono calcolati alcuni parametri statistici relativi al grafico presentato.

POLYGON: - Lo scopo di questo modulo è quello di calcolare e visualizzare l'istogramma dell'area racchiusa dentro una poligonale.

Per far questo, dopo aver selezionato e memorizzato in un array i vertici della poligonale (fino ad un massimo di 20) l'algoritmo calcola l'istogramma prendendo in considerazione tutti i punti interni alla spezzata chiusa.

HISTO2: - Consideriamo un'area dell'immagine. Se prendiamo una seconda area di uguale dimensione sulla stessa immagine o su una immagine diversa, possiamo mettere in relazione fra loro le coppie di pixel che hanno la stessa posizione relativamente all'area selezionata.

L'istogramma del secondo ordine calcola il numero di volte N che una certa coppia (a,b) di valori è presente nell'area esaminata.

I dati calcolati vengono memorizzati in una matrice detta "Matrice delle occorrenze congiunte" o "Matrice di dipendenza dei livelli di grigio".

L'algoritmo visualizza il grafico relativo ai dati contenuti nella matrice suddetta.

Se le coppie di entro un'immagine sono altamente connesse, l'istogramma si svilupperà entro la diagonale principale.

Quindi, calcolato l'istogramma del secondo ordine $P(a,b)$, dato da:

$$P(a,b) = \frac{N(a,b)}{M}$$

dove $N(a,b)$ = frequenza della coppia (a,b) , e M = numero totale di coppie di pixel prese in considerazione, si possono fare alcune misure sulla dispersione intorno alla diagonale. Riportiamo di seguito le formule relative.

$$B_A = \sum_{a=0}^{255} \sum_{b=0}^{255} a b P(a,b) = \frac{\sum_{a=0}^{255} \sum_{a=0}^{255} a b N(a,b)}{M} \quad \text{autocorrelazione}$$

$$B_C = \sum_{a=0}^{255} \sum_{b=0}^{255} (a-a) (b-b) P(a,b) \quad \text{covarianza}$$

$$B_I = \sum_{a=0}^{255} \sum_{b=0}^{255} (a-b)^2 P(a,b) \quad \text{inerzia}$$

$$B_V = \sum_{a=0}^{255} \sum_{b=0}^{255} |a-b| P(a,b) \quad \text{valore assoluto}$$

$$B_D = \sum_{a=0}^{255} \sum_{b=0}^{255} \frac{P(a,b)}{1+(a-b)^2} \quad \text{differenza inversa}$$

$$B_N = \sum_{a=0}^{255} \sum_{b=0}^{255} P(a,b)^2 \quad \text{energia}$$

$$B_E = - \sum_{a=0}^{255} \sum_{b=0}^{255} P(a,b) \log_2 P(a,b) \quad \text{entropia}$$

5. *DEFINIZIONE DEL SISTEMA DI ACQUISIZIONE E SCANSIONE*

In base alle sperimentazioni realizzate sui singoli componenti è stata definita più in dettaglio l'architettura della struttura d'indagine US e sono stati individuati i requisiti operativi del software da utilizzare per la scansione e l'acquisizione dei segnali.

È stato inoltre realizzato un dispositivo di scansione ridotto da utilizzare per studi sul segnale a ultrasuoni e per mettere a punto le procedure di scansione.

5.1 *Banco di scansione xy*

È stato progettato e realizzato un dispositivo con due gradi di libertà per la scansione mediante ultrasuoni di campioni con superfici piane.

Il dispositivo è costituito da due guide ortogonali con carrelli traslatori: una guida è fissata ad un basamento e determina la direzione di scansione x, mentre la seconda, che si sposta lungo la precedente mediante un carrello, determina la direzione di scansione y. Sulla guida dell'asse y è montato un carrello con un supporto regolabile per l'alloggiamento di un tubo portasonda; sul basamento è sistemata una vasca per l'immersione dei campioni da esplorare.

Il movimento su ciascun asse è ottenuto mediante un motore passo-passo, una cinghia di trasmissione e una puleggia di rinvio: è possibile variare manualmente di pochi gradi la direzione della sonda rispetto alla normale z al piano xy.

I motori sono pilotati attraverso un'interfaccia digitale I/O da un calcolatore di tipo personal; il calcolatore riceve i dati di acquisizione mediante una porta parallela IEE-488.

Il dispositivo è in grado di esplorare un'area di 30cm•30cm con la risoluzione spaziale di 1mm che può essere facilmente aumentata variando il rapporto di riduzione delle pulegge; il tipo di dato acquisito e la sua risoluzione dipendono dall'apparecchiatura usata per la rivelazione e la quantizzazione del segnale a ultrasuoni.

Il dispositivo verrà impiegato per compiere sperimentazioni e studi in parallelo rispetto all'attività svolta facendo uso del braccio meccanico del robot. È infatti semplice realizzare programmi per l'acquisizione di mappe di segnali ricavate dalla scansione di campioni con geometrie piane, e di conseguenza il dispositivo di scansione xy verrà usato soprattutto per valutare le variazioni di risposta in funzione del passo di campionamento e della direzione della sonda rispetto alla superficie del campione, usando i risultati ottenuti verranno utilizzati per sviluppare i programmi di scansione che fanno uso del sistema IRB 2000.

5.2 Studio di un protocollo software di gestione interattiva in linea

Il sistema IRB 2000 è dotato di software per lo sviluppo di programmi, ma non è dotato di programmi operativi per il movimento del braccio che devono essere sviluppati o in S3 o sfruttando la risorsa di calcolo esterna.

La realizzazione di procedure di scansione, mediante l'utilizzo diretto delle risorse software interne al controllore del robot, è possibile quando si devono effettuare movimenti relativamente semplici: si deve cioè acquisire per autoapprendimento una sequenza di posizioni e selezionare il modo di interpolazione fra le coppie di punti. Una volta definita la traiettoria, il programma può ripeterla indefinitamente e si può dunque esaminare un qualsiasi numero di campioni identici, come avviene in genere in un controllo di qualità.

Se la traiettoria da seguire durante la scansione non è di tipo rettilineo, il numero dei valori di posizione, da acquisire manualmente nella operazione di auto apprendimento, può crescere notevolmente: è infatti necessario definire porzioni elementari di traiettoria costituite da segmenti di rette che meglio approssimino la traiettoria ideale. Si deve inoltre evitare che la sonda entri in contatto col campione in esame o che la direzione della sonda non si discosti eccessivamente dalla normale alla superficie in ciascun punto di misura.

Queste condizioni possono essere valutate con precisione solo quando la superficie del campione è definita in maniera analitica oppure in maniera discreta come dato di progettazione CAD: in entrambi i casi è possibile sviluppare dei programmi generatori di posizione per la sintesi di superfici poliedriche sovrapponibili alle superfici del campione; le superfici devono essere costruite in base a vincoli sulle modalità di scansione (passo di campionamento, limiti nell'orientamento del robot, margine di sicurezza, ecc.); inoltre l'orientamento della sonda nei punti di misura giacenti su ciascuna faccia elementare può scostarsi dalla normale alla superficie effettiva del campione entro i limiti determinati in fase di sperimentazione.

Il Position File così determinato non può essere inserito manualmente nella memoria dell'unità di controllo del robot ma deve essere trasmesso attraverso il Computer Link.

Di conseguenza si ritiene opportuno sviluppare un sistema basato su un calcolatore ospite in modo da realizzare procedure di scansione che, sfruttando integralmente le potenzialità del robot, permettano di seguire traiettorie complesse con la precisione necessaria, e che nel contempo consentano una gestione semplice e interattiva.

Attualmente l'unità di controllo può essere collegata attraverso una porta seriale RS232 soltanto ad un calcolatore con un sistema operativo MS/DOS; in attesa di sviluppi, da parte della ABB o di altri, che permettano di collegare l'unità S3 ad altri sistemi di calcolo, si è ritenuto conveniente e opportuno definire un'architettura comprendente un calcolatore di tipo personal nel quale sono operativi i pacchetti software ABB per lo sviluppo di programmi robot e la gestione della comunicazione.

Sul calcolatore vanno sviluppati gli strumenti software necessari per individuare i vincoli e le prestazioni della componente software da realizzare successivamente nella versione finale della struttura US da usare per operazioni di controllo di qualità e di sperimentazione.

A questo scopo sono stati studiati e definiti alcuni package software, inseriti in una opportuna architettura di gestione, che compiono le seguenti funzioni:

a) *Supervisione dei movimenti robot.*

I movimenti possibili del robot rispetto al sistema di riferimento individuato dalla sua collocazione (nel caso attuale con basamento fissato al suolo) devono essere limitati tenendo conto del tipo di applicazione previsto. Infatti i campioni da esaminare vanno immersi in una vasca ed è quindi opportuno, anche per ragioni di sicurezza, limitare, sia in maniera hardware sia software, i possibili spostamenti del robot in modo che la sonda si trovi immersa alla giusta profondità nell'acqua e che il corpo del robot non possa entrare in contatto colle pareti della vasca o con l'acqua.

Si devono quindi determinare i confini di uno spazio di lavoro ammissibile, e imporre che in qualsiasi posizione di scansione le coordinate del TCP non escano da questa zona.

A questi vincoli, determinati dalle dimensioni della vasca e dalla posizione rispetto al robot, si devono aggiungere altri vincoli che dipendono dalla forma del campione da esaminare; infatti, nel tentativo di disporre la sonda in maniera perpendicolare rispetto alla superficie tangente al punto di scansione, il tubo portasonda potrebbe urtare il campione se questo presenta sporgenze o rientranze con piccoli raggi di curvatura. Si deve allora conoscere a priori la traiettoria da seguire sull'intero campione e determinare se, con certe dimensioni del tubo porta sonda, ci sia la possibilità di urtare il campione in determinati punti di scansione. In tal caso si deve rinunciare all'ortogonalità della sonda, accettando, se possibile, lo scostamento della perpendicolarità determinata dalla forma del campione.

Si deve in sostanza determinare e sviluppare una libreria di routines di supervisione che, noti i vincoli fissi e quelli dipendenti dalla geometria del campione e del tubo porta sonda, controllino le posizioni che il braccio dovrà raggiungere durante la scansione e facciano avviare la scansione vera e propria solo quando tali vincoli sono rispettati; in forma alternativa questo controllo può essere fatto durante la scansione stessa, imponendo un arresto se il movimento da eseguire per compiere il successivo passo di scansione viola i limiti di sicurezza determinati.

La possibilità di esplorare un campione con geometria complessa può dipendere anche dalla posizione del campione stesso all'interno della vasca; è quindi opportuno conoscere dimensioni e forma dei campioni in modo da trovare la posizione ottimale in cui il campione va collocato.

b) *Controllo interattivo dei movimenti robot.*

In numerose applicazioni di ricerca si devono effettuare scansioni di campioni con facce piane o a forma di cuneo, in modo quindi che la sonda si sposti lungo l'intero campione con un orientamento costante. Risulta allora utile poter definire in maniera semplice un insieme limitato di parametri che definiscono integralmente la scansione, senza dover ricorrere di volta in volta alla scrittura di programmi ed al rilevamento dell'area di scansione mediante spostamento controllato dal joystick della tastiera di programmazione. Si possono quindi simulare le possibilità offerte dalla Unità di controllo nel modo di funzionamento locale facendo uso di due tipi di programmi interattivi sul calcolatore ospite:

- programmi in cui le coordinate x , y , z degli estremi dell'area di scansione su una superficie piana, l'angolo costante di orientamento della sonda e il valore del passo incrementale di scansione vengono inseriti mediante tastiera;
- programmi in cui lo spostamento del braccio, è controllato mediante tasti funzionali (freccie direzionali sulla tastiera o mouse).

c) *Sintesi di un modello di scansione complessa.*

I movimenti del robot nello spostamento fra una coppia di posizioni successive, comportano una fase di accelerazione per raggiungere una velocità di regime e una fase di decelerazione per l'arresto sulla coordinata di arrivo. Le rampe di accelerazione sono gestite automaticamente dalla Unità di Controllo e dipendono dalla distanza tra i punti; la velocità raggiunge il massimo valore possibile (2 m/s) solo quando i punti sono sufficientemente distanti. La lunghezza dell'intervallo in cui la sonda si muove con moto uniforme alla

massima velocità è anch'essa funzione della distanza fra la coppia di posizioni.

In una procedura di indagine ad ultrasuoni per il controllo di qualità in una linea di produzione è opportuno ridurre il tempo necessario per la scansione di un campione al minimo valore compatibile con i requisiti di risoluzione, precisione di misura e flusso di trasferimento dei dati ricavati. Questo risultato può essere ottenuto imponendo che il braccio del robot si sposti lungo un intervallo (P_1, P_4) con estremi sufficientemente distanti in modo da raggiungere la massima velocità all'interno di un intervallo (P_2, P_3) contenuto in (P_1, P_4) , e acquisendo i segnali di eco nei punti di scansione interni a (P_2, P_3) . La lunghezza dell'intervallo (P_2, P_3) va scelta in maniera tale che la distanza tra i punti di (P_2, P_3) e la superficie S del campione non si allontani eccessivamente dal valore di distanza di focalizzazione della sonda; inoltre l'angolo fra la retta r che definisce l'orientamento della sonda, e la retta n normale alla superficie S in corrispondenza dei punti di misura non deve essere prossima a zero in modo da non provocare artefatti di misura dipendenti da una non ortogonalità della sonda rispetto alla faccia superiore del campione (vedi fig. 12)

Nota la forma della superficie della faccia superiore del campione, mediante insiemi finiti di posizioni ricavate per auto apprendimento o fornite da un eventuale sistema CAD, oppure mediante la conoscenza di equazioni generatrici, il programma per la sintesi del modello di scansione deve individuare un poliedro contenente il campione. In ciascuna faccia del poliedro, definita da quattro punti su un piano, viene inoltre definito il rettangolo di scansione, in modo che per ciascun punto di misura la distanza e l'ortogonalità della sonda rispetto al punto effettivo di misura sulla superficie S non escano dai limiti di tolleranza stabiliti.

La determinazione del poliedro deve essere supervisionata secondo quanto detto al punto a) per evitare che la sonda o le parti meccaniche del braccio entrino in contatto con il campione e con altri ostacoli.

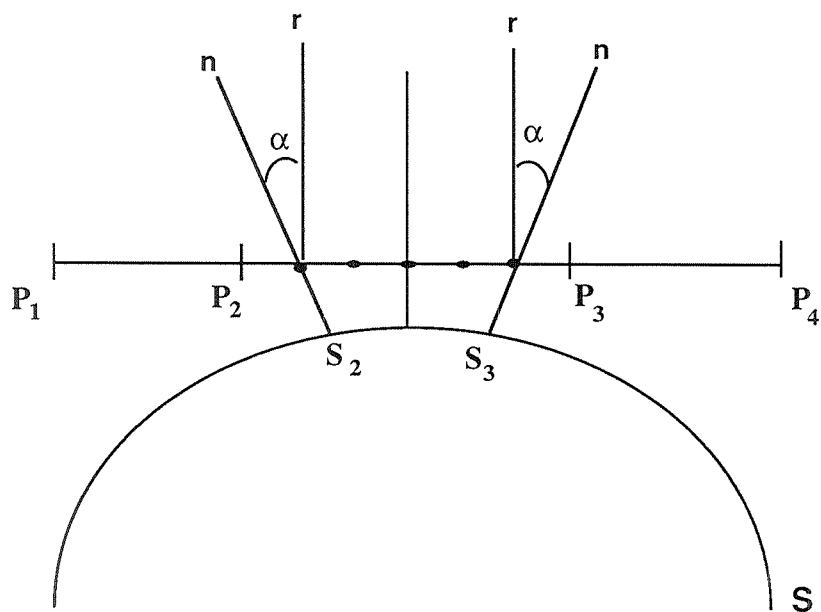


Figura 12

d) *Sviluppo di programmi per scansioni complesse.*

Per eseguire la scansione di un campione la cui faccia superiore non sia assimilabile ad una sola superficie piana, il programma di scansione dovrà:

- 1 - leggere un Position File che definisce le facce piane del poliedro che sviluppa la superficie del campione; il Position File, generato da un programma secondo quanto descritto nel punto b), rispetta i vincoli definiti nel punto a);
- 2 - generare, in base a segnali di sincronismo prodotti all'inizio di ciascuna riga di scansione, gli impulsi di trigger per il generatore di segnali ad ultrasuoni; questi impulsi devono essere prodotti quando la sonda si trova all'interno dell'intervallo utile (P_2, P_3) definito al punto b): se gli impulsi hanno un periodo costante, le distanze fra due successivi punti di misura nell'intervallo (S_2, S_3) individuato dalla proiezione di (P_2, P_3) su S sono tanto più simili quanto più costante è la velocità di traslazione della sonda e quanto più grande è il raggio di curvatura di S lungo la linea di scansione;

- 3 - acquisire i dati ricavati dal segnale prodotto dal dispositivo di ricezione a US in risposta agli impulsi di misura;
- 4 - archiviare e rappresentare in forma grafica e/o pittorica le mappe o i segnali ricavati durante l'operazione di scansione.

5.3 Architettura della struttura di indagine

Le operazioni descritte in precedenza possono essere realizzate mediante la stazione per l'analisi a ultrasuoni mostrata in fig. 13 I componenti principali indicati schematicamente sono:

- un calcolatore di tipo personal, PC;
- un controllore programmabile, S3, per la gestione di un robot a sei assi;
- un generatore-ricevitore di segnali a ultrasuoni, US;
- un oscilloscopio a memoria digitale, CM;
- un calcolatore per l'elaborazione d'immagini, SVP.

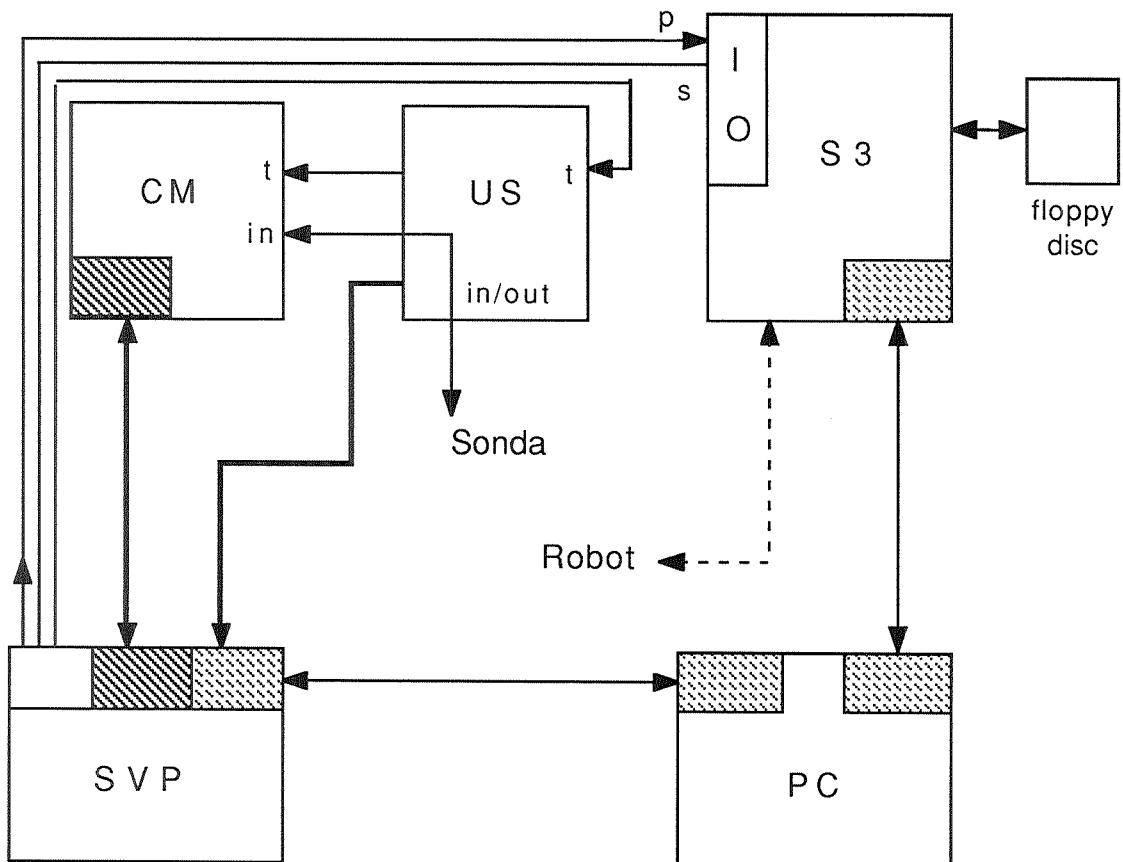
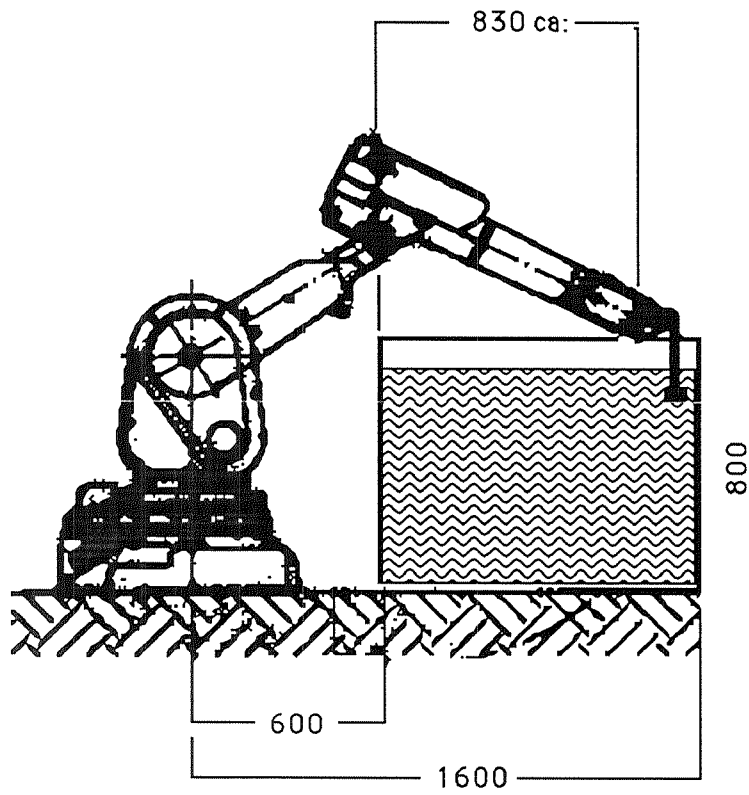
Le operazioni di spostamento della sonda, generazione di segnali, formazione delle immagini digitali associate alle mappe di misura e controllo delle operazioni sono eseguite in parallelo e in maniera interdipendente dai vari componenti della stazione. Al fine di raggiungere sufficienti prestazioni in termini di efficienza complessiva e velocità di esecuzione si sono determinate le seguenti suddivisioni di compiti.

Nel controllore S3 risiede un programma di scansione elementare, che provvede a:

- spostare la sonda alle coordinate attuali di misura;
- dare un pronto per l'esecuzione della misura;
- prepararsi al passo successivo.

Nel calcolatore PC risiede il programma principale di acquisizione, che provvede a:

- caricare o attivare il programma di scansione in S3;
- su richiesta di S3 fornire le coordinate del successivo punto di misura, mediante trasferimenti singoli o a blocchi a seconda della struttura dei programmi in esecuzione;
- sincronizzare le operazioni di inizio e fine programma col sistema SVP.



p = pronto
s = sinc
t = trigger

In SVP risiede il programma per la formazione delle mappe di acquisizione, che provvede a:

- scrivere in memoria i dati ricavati per ciascun punto di misura;
- elaborare i dati grezzi di ingresso;
- comunicare a S3 il pronto per il proseguimento della scansione;
- sincronizzare le operazioni con PC;
- archiviare i dati.

5.3.1 Modi di ispezione dei campioni

Si ritiene che il campione in esame possa essere ispezionato in tre modi.

a) Scansione doppia veloce

In questo caso si esegue dapprima una scansione asincrona per ricavare una mappa di valori a soglia singola: la sonda viene traslata alla massima velocità consentita dal robot e il controllore S3 produce un segnale di sincronismo all'inizio di ciascuna riga di scansione, che si estende al di là dell'intervallo di campionamento (v. fig.12) al fine di consentire al braccio del robot di muoversi a velocità pressoché costante in corrispondenza dei punti di misura. Successivamente e a seguito della eventuale rivelazione automatica o interattiva della zona di difetto, viene avviata una scansione a onda completa su un'area rettangolare che includa le coordinate approssimate del difetto individuato.

b) Scansione doppia sincronizzata

In questo caso si esegue una prima scansione a soglia singola con arresto del robot e sincronizzazione delle operazioni in ciascun punto di misura. Successivamente si esegue l'analisi della mappa e si compiono eventuali scansioni a onda completa su aree rettangolari definite in base alle coordinate esatte ricavate in precedenza.

c) Scansione interlacciata

In questo caso il movimento del robot viene arrestato e sincronizzato in corrispondenza di ciascun punto di misura; il dato a soglia

singola viene memorizzato nella mappa e analizzato: nel caso di rilevamento di difetto viene memorizzata anche la sequenza dei dati ricavati dal campionamento del segnale a onda completa, già presenti nella memoria tampone dell'oscilloscopio digitale.

5.3.2 Scrittura dei dati nella memoria video

L'acquisizione avviene mediante il trasferimento dei dati digitali dalla memoria tampone del dispositivo di digitalizzazione CM al sistema di rappresentazione ed elaborazione SVP 2000. La velocità di trasferimento è limitata dalle caratteristiche tecniche dell'interfaccia di collegamento di CM e dal metodo di scansione adottato: è sufficiente quindi trasferire temporaneamente le stringhe di dati direttamente nella memoria centrale tramite una porta parallela IEEE 488. Successivamente i dati possono essere memorizzati nella memoria video per l'elaborazione ed eventuale rappresentazione, oppure nella memoria di massa per l'archiviazione secondo standard anche diversificati e dipendenti dalla utilizzazione prevista.

Nella memoria video può risultare utile ed efficiente, in termini di velocità di trasferimento ed ottimizzazione della occupazione di memoria, memorizzare i dati secondo uno standard che ne consente una visualizzazione significativa al fine della individuazione anche visiva della struttura ecografica di sezioni ortogonali del campione da analizzare; ovviamente i dati dovranno essere successivamente opportunamente elaborati mediante filtri digitali o operazioni di convoluzione per ricavare dal segnale a radiofrequenza i valori di impedenza acustica all'interno del campione.

Lo standard dovrà essere definito tenendo conto anche delle possibili elaborazioni richieste al fine di ottimizzare il processo di analisi; tuttavia sembra opportuno privilegiare un criterio di scelta legato al metodo di scansione, in modo da ottenere mappe ad esso facilmente correlabili.

Acquisizione a soglia singola

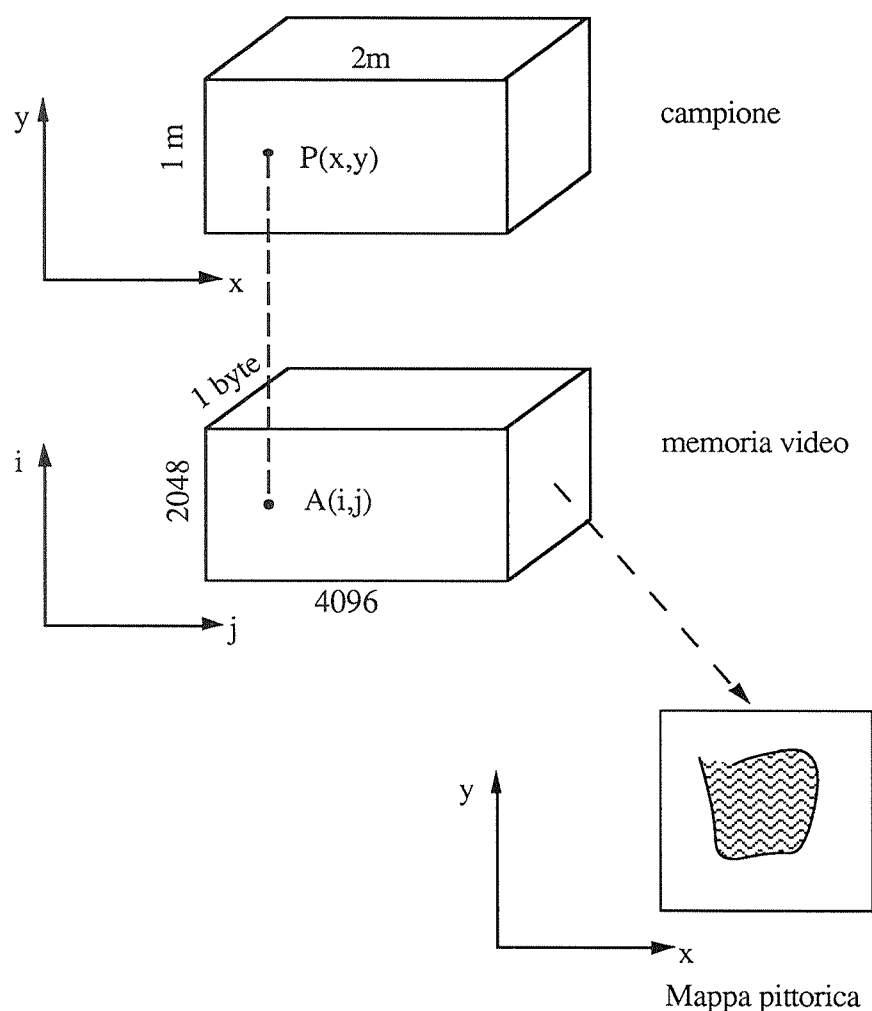


Figura 14

Dalla scansione di un campione con la misura a soglia singola per ciascun punto di misura $P(x,y)$ si ricava un valore $A_{i,j}$ (ad es. attenuazione o tempo di volo) quantizzato su un byte; è necessario quindi un byte di memoria per ciascun dato di misura. La capacità totale della memoria video dell'SVP2000 è di 8 MByte; si può quindi configurare la memoria come un piano di $2048 \cdot 4096 \cdot 1$ byte in cui possono essere immagazzinati i risultati delle misure ottenute da un campione di $1m \cdot 2m$ esplorato con passo di scansione di 0,5 mm.

Inserendo i dati in memoria secondo la corrispondenza mostrata in figura 14 è estremamente agevole ottenere delle mappe che rappresentano

i valori $A_{i,j}$; naturalmente i valori riportati costituiscono il caso limite della applicazione e le mappe effettivamente rappresentabili alla massima risoluzione sul video pittorico hanno dimensioni di $512 \cdot 512$ punti. Le matrici con dimensioni maggiori di queste possono essere rappresentate parzialmente mediante operazioni di panning o integralmente ma con risoluzione ridotta.

Acquisizione ad onda completa

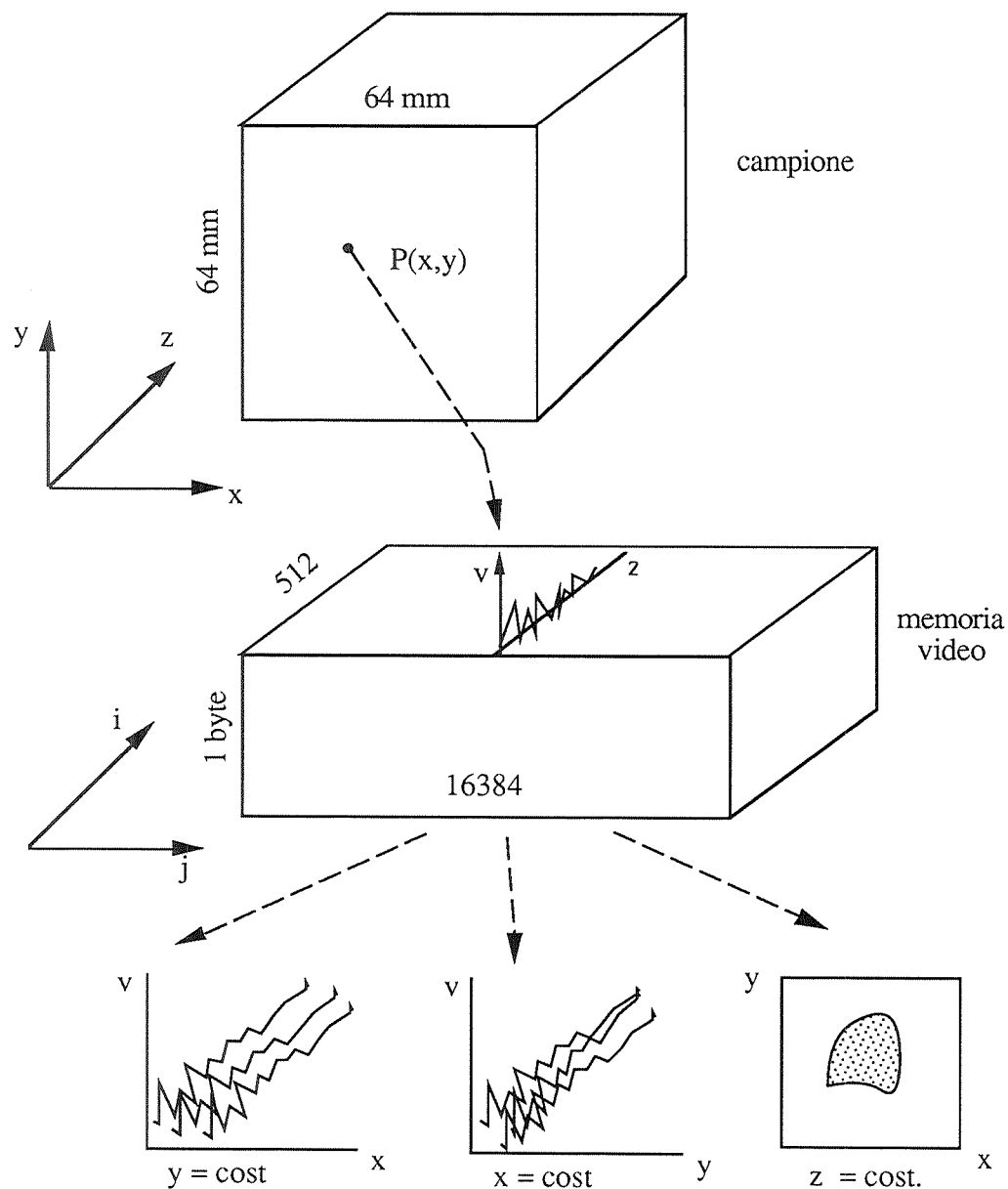


Figura 15

In questo caso per ciascun punto di misura $P(x,y)$ si ricava un segnale v in funzione del tempo e indirettamente della profondità z all'interno del campione. Il segnale analogico v a radiofrequenza viene campionato e quantizzato: si ricavano così 512 campioni v_k definiti ciascuno da un byte.

I dati ricavati per ciascun punto di misura richiedono così 512 byte: si può allora organizzare la memoria video dell'SVP2000 in un piano di $512 \cdot 16384$ celle di 1 byte, e ospitare così i $128 \cdot 128$ vettori di 512 byte ricavati da un campione di $64\text{mm} \cdot 64\text{mm}$ esplorati con passo di campionamento spaziale di 0,5mm.

Con la configurazione di memoria mostrata in figura 15, i dati ricavati da una riga di scansione del campione lungo l'asse x occupano 128 posizioni successive individuate dall'indice di colonna j e contate modulo 128 a partire dalla coordinata iniziale $j=1$; i vettori di 512 punti ricavati dalle misure lungo la direzione di scansione y occupano invece posizioni con indice j incrementato di un multiplo di 128.

Mediante lettura della memoria con gli indirizzamenti sopra citati si possono ottenere agevolmente stereogrammi di $v(x,y,z)$ con x oppure y costanti; in ognuno dei due casi si possono avere 128 stereogrammi definiti da 128 righe di 512 elementi espressi su un byte.

Leggendo la memoria con indirizzo $i=\text{costante}$ e $1 \leq j \leq 16384$ si può ottenere invece una mappa pittorica dei valori $v(x,y,z)$ per $z=\text{costante}$, che fornisce indicazioni sull'impedenza acustica in un piano interno del campione: si possono ottenere 512 mappe di $128 \cdot 128$ elementi a 256 livelli di intensità.

5.4 Requisiti del software di scansione

Secondo i criteri esposti nel paragrafo 5.2 e basandosi sull'architettura e sulle modalità operative illustrate in 5.3 si ritiene indispensabile realizzare un package di programmi basati sull'ABB Communication Tools per operazioni di supervisione dei movimenti, interazione e conversione di coordinate.

I programmi che compongono il package dovranno svolgere le seguenti funzioni:

- a) Manovra del robot mediante assegnazioni da tastiera del PC dei seguenti parametri di movimento:
- terna di incrementi relativi Δx , Δy e Δz , della posizione del robot;
 - terna di angoli, espressa in quaternioni, per orientare l'utensile, nel caso specifico un sensore ultrasonoro, rispetto agli assi coordinati della flangia;
 - variazione relativa di velocità Δv del robot, rispetto ad un valore iniziale assegnato.

I valori assoluti della posizione (coordinate cartesiane e quaternioni) e della velocità dell'utensile, saranno leggibili sul video del PC. Tali valori potranno essere registrati in un file, mediante un comando da tastiera (vedi punto seguente).

- b) Definizione della geometria, rispetto al sistema di riferimento cartesiano del robot, dello spazio di lavoro entro cui si muoverà il robot, ossia
- della vasca d'acqua, in cui vengono immersi i campioni da analizzare,
 - del campione in esame.

Sia la geometria della vasca che quella del campione potranno essere definite analiticamente o mediante un insieme di coordinate spaziali (vedi programma d) o mediante il programma a). Questo, infatti, può essere usato per determinare in modo indiretto le coordinate sia dei punti significativi delle porzioni di piano che descrivono lo spazio di lavoro del robot, sia delle superfici dei campioni che saranno analizzati dal sensore.

Il programma dovrà verificare che eventuali superfici concave del campione siano tali da rendere possibile la scansione di sensori aventi dimensioni e forme prefissate.

- c) Verifica che i comandi di movimento inviati al robot, manualmente od automaticamente, soddisfino i vincoli dello spazio di lavoro, definito con b), allo scopo di prevenire la collisione del robot con ostacoli dell'ambiente circostante.

- d) Cambiamento del riferimento cartesiano dei dati CAD di un campione in quello del robot, e determinazione della sua superficie interpolatrice, note le sezioni trasversali e longitudinali in cui è stato ripartito. Ove necessario, per rappresentare analiticamente particolari forme di campioni, studio ed implementazione di metodi per disassemblarle in solidi definibili analiticamente.
- e) Definizione della superficie di scansione di un campione e dei relativi percorsi di scansione. Essa potrà essere effettuata, con velocità e passo assegnati, secondo percorsi di tipo raster od "a greca". Per la determinazione dei percorsi di scansione, si configurano procedure di calcolo diverse, dipendenti dai seguenti tipi di superficie di scansione:
 - e1) piana, non necessariamente orizzontale, e di forma rettangolare. Il sensore giace o nel piano perpendicolare al piano di scansione, e passante sul percorso di scansione, o nel piano a questo ortogonale; esso può formare, con la normale al piano di scansione, un angolo compreso fra $-\alpha$ e α , dove α è un angolo prefissato;
 - e2) uniformemente convessa, concava oppure con aree convesse e concave. In questo programma vale la seguente condizione: detto P il punto di intersezione dell'asse longitudinale r del sensore con il percorso di scansione, l'angolo che la normale in P forma con r deve essere non maggiore di un angolo prefissato;
- f) Gestione delle porte di I/O del Controllore, per la sincronizzazione dei movimenti del robot e del sensore.
- g) Rappresentazione assonometrica su schermo grafico della vasca e del campione in esame e, in tempo reale, delle posizioni assunte dal sensore, durante il jogging e la scansione, secondo le seguenti opzioni:
 - g1) immagine grafica rappresentata secondo una vista preassegnata,
 - g2) immagine grafica rappresentata in modo variabile,
 - g3) immagine grafica rappresentata con rotazioni rispetto ad uno o più assi,
 - g4) immagine grafica rappresentata con zoom parziale.

6. *DISCUSSIONE*

In questa relazione è stato sinteticamente presentato lo stato attuale delle attività orientate al CND mediante tecniche US che hanno comportato studi, sperimentazioni e sviluppi sia hardware che software, a seguito dei quali si può considerare chiusa la prima fase dei lavori che globalmente prevedevano la definizione particolareggiata della struttura e delle metodologie di indagine.

È stata inoltre avviata l'azione di assemblaggio delle varie componenti provvedendo alla realizzazione o all'acquisto della componentistica necessaria.

Si è ritenuto opportuno dotare l'SVP2000 di una sottostazione grafica Motorola MVME 393, basata sul sistema X-Window; questa stazione servirà per due scopi fondamentali:

- interazione uomo-macchina; mediante la stazione grafica si può realizzare in modo abbastanza semplice un'interfaccia operativa di tipo iconico o con rappresentazione a finestre, che consente operazioni veloci e rende immediata l'iterazione con il calcolatore
- la restituzione grafica; mediante la sottostazione sarà possibile visualizzare i grafici relativi ai segnali acquisito direttamente oppure ai risultati delle elaborazioni.

Sono stati individuate e commissionate alcune interfacce hardware necessarie per il collegamento analogico o digitale delle periferiche adibite alla generazione di ultrasuoni, alla acquisizione dei segnali e alla gestione del processo di indagine (Themis, Tsume 404, Ifet 488, Tsume 405 I/O input-output, Tsume 403 A/D D/A counter): poiché sui sistemi con BUS VME per questi tipi di interfaccia non sono ancora disponibili driver di impiego generale, è stato necessario commissionare la realizzazione di pacchetti in grado di funzionare in PSOS Puls che dovranno essere rielaborati per inserirli nell'ambiente VME Exec operante sul sistema SVP2000 in nostro possesso.

La memoria di massa del sistema SVP2000 all'atto dell'acquisto era di 150 MBy; questa memoria è quasi interamente occupata dai sistemi operativi impiegati (Unix, VME Exec, X-window), si è reso necessario un ampliamento per avere lo spazio sufficiente ad ospitare dati e programmi:

si è quindi provveduto all'acquisto di una estensione di memoria disco di circa 750 MBy.

Si è reso pure necessario ampliare la capacità della memoria video da 2 a 8 MB, per accogliere la grande quantità di dati ricavabili dall'acquisizione dell'onda completa, e la memoria centrale da 8 a 16 MB per soddisfare le esigenze di velocità ed efficienza del processo elaborativo; anche per questi sviluppi si è provveduto alla emissione di opportune commesse.

Gli ampliamenti delle memorie e l'inserimento delle interfacce speciali hanno richiesto alcune modifiche della organizzazione hardware del sistema SVP2000 e l'implementazione di un aggiornamento del software di gestione; tali modifiche sono state definite e quindi commissionate alla società Scriba.

Il sistema IRB 2000 si è dimostrato rispondente alle necessità per quanto riguarda la componente elettro-meccanica e il software di controllo dei movimenti: si deve invece sviluppare il software per l'acquisizione dei segnali, funzionante secondo le modalità esposte in 5.3.

Attualmente è in fase di sviluppo il software di acquisizione, analisi e gestione limitatamente all'indagine di campioni con facce piane e parallele, senza controlli sui vincoli di movimento del braccio meccanico.

I programmi di scansione complessa, lungo traiettorie ricavate da modelli discreti e con controllo sui vincoli di movimento, rispondenti alle indicazioni espresse in 5.4, dovranno essere commissionati dalla Aeritalia a società di sviluppo software specializzate.

Inoltre per quanto riguarda il modo di scansione complesso si è verificato che la possibilità di spostare il braccio meccanico alla velocità di 2 m/s è limitata ad un ristretto numero di casi. Infatti, per evitare di urtare parti del campione qualora siano presenti sporgenze oppure se si devono esplorare delle cavità, la corsa del braccio può essere alquanto limitata; in questo caso il braccio non può raggiungere la massima velocità in quanto la quasi completa escursione della corsa è richiesta per la rampa di accelerazione e decelerazione.

È opportuno osservare che quest'ultima necessità può facilmente determinare l'impossibilità di definire e costruire un poliedro interpolante la superficie del campione in modo che siano soddisfatti i vincoli di tollerabilità prefissati (velocità costante, perpendicolarità dalla sonda al campione).

Se invece è possibile eseguire la scansione lungo le superfici piane corrispondenti alle facce del poliedro, sono necessarie diverse operazioni di ricomposizione del mosaico di acquisizione oltre a controlli accurati sui vincoli di esplorazione: la velocità reale di scansione può essere notevolmente ridotta rispetto al limite teorico di 2 ms e risultare comparabile con quella ottenibile con il metodo di scansione sincronizzato.

Inoltre la scansione alla massima velocità comporta eventualmente anche problemi causati dalla necessità di registrazione geometrica dei dati di misura da ricavarsi da mappe le cui proiezioni sono parzialmente sovrapponibili: queste difficoltà non insorgono invece nel caso di scansione interlacciata in quanto a ciascun dato di misura è associata anche la relativa coordinata spaziale.

7. APPENDICE

Si riportano descrizioni sintetiche e listati di alcuni programmi sviluppati per verificare i modi di programmazione e i tipi di spostamento del robot IRB 2000.

I programmi scritti mediante la tastiera dell'Unità di controllo o in ambiente OLP sul calcolatore ospite sono listati secondo la codifica ARLA e sono identificati con un codice numerico di 4 cifre; i programmi per il controllo in tempo reale sono scritti in C.

Esempi di programmazione con tastiera

Il programma 41 è stato scritto per verificare sia l'istruzione che esegue archi di cerchio sia le istruzioni che consentono tramite un offset di ottenere posizioni nuove posizioni rispetto a quelle già memorizzate.

Per percorrere una traiettoria piana lungo una spirale di Archimede sono inizialmente definite due semicirconferenze con centri distanti 5 mm ed identificati con position 1 e position 2; a partire da queste le successive semicirconferenze vengono definite tramite un'istruzione di offset.

Per definire una semicirconferenza sono necessarie tre istruzioni: la prima e l'ultima contengono rispettivamente il punto iniziale e quello finale, mentre la seconda istruzione definisce un punto qualsiasi appartenente all semicirconferenza; per definire un intero cerchio sono necessarie invece 5 istruzioni.

Il programma 36 fa muovere il TCP lungo il perimetro di un parallelogramma, definito da 5 istruzioni POS.

Il programma 37 fornisce un esempio relativo alle istruzioni di I/O che controllano l'interfaccia digitale; si possono pilotare separatamente le varie linee e si possono ottenere impulsi di durata programmabile, con un minimo di 100 ms, ottenuto con le istruzioni successive SET, RESET; l'istruzione PULSE genera un impulso di ~ 250 ms.

Il programma 30 riporta un esempio di traiettoria ottenuta con istruzioni di lettura-scrittura nei registri di posizione e con le istruzioni di operazioni aritmetiche; la posizione iniziale è definita nell'istruzione 100.

L'insieme dei programmi 2006-2011 costituisce un esempio di scansione sincronizzata: la sonda avanza con passi elementari di 1mm e ad ogni passo viene generato un impulso di sincronizzazione e si attende un consenso per la prosecuzione. Definendo un frame diverso è possibile ottenere scansioni piane su superfici aventi un angolo qualsiasi rispetto alla superficie definita in origine.

In particolare nel programma 2006 sono definiti la velocità di spostamento (1 ms), il tipo di coordinate (RECT COORD), il TCP e il frame. Cambiando l'argomento all'istruzione 60 è possibile iniziare la scansione da un qualunque punto dello spazio.

Il programma 2010 gestisce la scansione partendo dal punto specificato nell'istruzione 60 del programma 2006 e richiamando i programmi 2007, 2008 e 2009 che effettuano spostamenti elementari: modificando il numero delle ripetizioni (argomento REP) è possibile estendere la scansione ad una superficie piana qualsiasi.

I programmi 2007 e 2009 effettuano rispettivamente un incremento e un decremento di 1mm del valore della coordinata y, mentre il programma 2008 incrementa di 1mm il valore della coordinata x; il programma 2011 genera sul bit 1 della porta di uscita un impulso di sincronizzazione che viene usato come trigger del trasmettitore, e legge sul bit 1 della porta d'ingresso il consenso per proseguire la scansione.

Esempio di programmazione fuori linea

Il programma 52 è un esempio di esplorazione di una superficie complessa generata da due segmenti piani e da una curva approssimabile ad un semicilindro.

All'inizio del programma vengono definiti la velocità di spostamento, il tipo di coordinate (MODRECT), il TCP e il frame. Con l'istruzione 60 viene definito il punto iniziale della scansione e cambiando il retpoint è possibile iniziare la scansione da un qualsiasi punto dello spazio. Le istruzioni 60, 70 e 80 fanno spostare il TCP lungo una

traiettoria rettilinea su un piano $z=cost$; con l'istruzione 100 inizia la traiettoria lungo la superficie del semicilindro.

Nel Position File richiamato dalle istruzioni 100-240 sono indicati i valori dei quaternioni che determinano l'orientamento della sonda in modo da mantenerla perpendicolare alla superficie da scandire e i valori x, y, z del TCP che mantengono l'estremità della sonda a distanza costante dalla superficie. Con l'istruzione 260 inizia la seconda traiettoria rettilinea lungo una superficie piana.

Esempi di controllo in tempo reale

Il programma MUOVI è un esempio semplice di programmazione tramite il PC communication tools.

Sullo schermo del calcolatore ospite compaiono i messaggi relativi allo stato del robot e alle istruzioni attivate in modo interattivo. Inizialmente si attiva il collegamento PC-S3 (set_robot_on), si definiscono il frame di lavoro (write_frame), il TCP (write_TCP), e si sincronizza il Robot (cmd_set_mode).

In seguito vengono inserite le coordinate x, y, z per lo spostamento del TCP; nell'istruzione cmd move si fa riferimento ad una tabella di dati in cui sono definiti dei valori costanti di quaternioni, per cui si ha un moto di traslazione della sonda.

Un esempio più complesso di movimento del robot mediante l'assegnazione di parametri di posizione su calcolatore ospite è costituito dal programma MOVEROBO.

I dati specificativi della posizione del robot sono trasmessi al Controllore del Robot, dalla tastiera di un PC, secondo uno schema a menù. Dopo aver collegato il Robot in modo remoto, mediante comandi sulla tastiera dell'Unità di Programmazione, si lancia il programma MOVEROBO dalla tastiera del PC.

Sulla prima riga del video del PC vengono descritti i nomi delle operazioni da svolgere la trasmissione dei parametri che caratterizzano il movimento del robot ed i comandi di attivazione della manovra, oltre a quello di rientro al DOS. Più specificatamente, i nomi delle operazioni sono:

Coordinate	Muovi
Quaternioni	Reset
Rotazioni	Uscita
Opzioni	

Nella riga successiva a quella delle operazioni viene data una breve descrizione dell'operazione selezionata:

Coordinate:	Introduzione dei valori di x, y, z;
Quaternioni:	Introduzione dei valori dei quaternioni q_1, q_2, q_3, q_4 ;
Rotazioni:	Introduzione degli angoli di rotazione degli assi X, Y, Z;
Opzioni:	Introduzione del sistema di riferimento, tipo di coordinate, percentuale velocità;
Muovi:	Movimento del robot;
Reset:	Reset del robot;
Uscita:	Ritorno al DOS.

Le operazioni vengono selezionate tramite i tasti di direzione orizzontale del cursore, o dalla battitura della prima lettera del nome dell'operazione seguito da <CR>.

Per annullare la scelta di un'operazione, si batte il tasto <ESC>.

Ogni qualvolta si seleziona un'operazione di introduzione dati, in corrispondenza del nome dell'operazione, sulla prima riga, si apre una finestra nella quale un menù specifica le opzioni possibili. Ognuna di esse è selezionabile mediante i tasti di direzione verticale del cursore e <CR>. Per ognuno di essi si apre un'altra finestra nella quale sono riportati i dati richiesti. Ogni volta che un dato è stato scritto, la seconda finestra si chiude ed il dato viene trasferito nel riquadro "Dati Input" del menù principale, inizialmente vuoto. Affiancato ad esso, sul lato destro dello schermo, c'è il riquadro "Stato Registri" che contiene il valore dei registri di stato del Controllore.

Per definire i dati "Coordinate", si scrivono, in decimi di mm, le coordinate x, y e z del punto in cui si dovrà spostare il TCP, ovvero di quanto dovrà essere spostato rispetto alla posizione attuale (v. "Sistema di Riferimento").

Quando si definisce l'orientamento della sonda, i relativi parametri sono espressi o in quaternioni od in angoli di rotazione rispetto agli assi coordinati: nel primo caso si sceglie l'opzione "Quaternioni", nel secondo, "Rotazioni".

La scelta dell'opzione Quaternioni, apre una finestra nella quale sono elencati in nomi delle quattro componenti q_1, q_2, q_3, q_4 , ognuna delle quali rimanda ad un'altra finestra, in basso nello schermo, per la scrittura dei corrispondenti valori. Analogamente, per l'opzione "Rotazioni" si apre una finestra nella quale sono elencati i tipi di rotazione da definire, ossia rispetto a quale asse coordinato, più il comando di "Attivazione" che attiva il calcolo dei corrispondenti quaternioni, le cui componenti sono riportate nel riquadro Dati Input.

Anche l'operazione "Opzioni" dà luogo all'apertura di una finestra nella quale sono descritti i tipi di parametro che devono essere assegnati al Controllore del Robot, tramite i tasti di direzione verticale del cursore. Per ciascun tipo, a sua volta, vengono aperte altrettante finestre, come di seguito specificato:

- Sistema di Riferimento
Riferimento Assoluto
Riferimento Relativo
- Tipo di Coordinate,
Coordinate Rettangolari
Coordinate Robot
- Percentuale Velocità

Come si è accennato sopra, se il sistema di riferimento è definito assoluto, il TCP sarà spostato nel punto $P(x,y,z)$, dove le coordinate sono quelle precedentemente definite coll'operazione "Coordinate"; se invece è relativo, il punto avrà le coordinate $P(x_0+x,y_0+y,z_0+z)$, dove $P(x_0,y_0,z_0)$ sono le coordinate attuali del TCP.

Con l'opzione "Tipo di Coordinate" si specifica il tipo di sistema di coordinate del robot per muovere il TCP nella nuova posizione, e con l'opzione "Percentuale Velocità" si definisce, in percentuale rispetto ad un valore massimo prefissato, la velocità di spostamento del robot.

Con il comando "Muovi" si avvia il movimento del Robot nel punto, e con l'orientamento del polso, prima specificati.

Nel caso di un arresto di emergenza impartito manualmente dall'operatore, il comando "Reset" ripristina le condizioni di riavviamento del Robot.

Infine, il comando "Uscita" trasferisce il controllo del PC al sistema operativo DOS.

Il riquadro "Stato Registri"

Come si è accennato sopra, sullo schermo, alla destra del riquadro dei "Dati Input", c'è quello dello "Stato Registri", nel quale sono riassunti ad uso dell'utente i contenuti dei registri attinenti al movimento del Robot. Essi descrivono lo stato del robot dopo l'avviamento o l'ultimo comando eseguito, comunque trasmesso al Controllore, per esempio, manualmente dall'Unità di Programmazione, da un Programma Robot avviato in "local mode" oppure in "remote mode" da PC tramite Communication Tool.

Nel riquadro appaiono nello stesso ordine in cui sono qui elencati, i nomi dei registri ed i loro contenuti logici o numerici:

- Standby: ha il valore True o False secondo che il Robot è in stato di attesa o di esecuzione;
- Operazione: ha il valore True se il Robot è in movimento, False altrimenti;
- Programma: Run se un programma Robot è in esecuzione, altrimenti è Stop;
- Interrupt: Se un programma Robot ha abilitato l'interrupt, viene indicata la dicitura Enable, altrimenti Disable;
- Orientaz.: È specificato il tipo di orientazione abilitata, ossia se è quella del polso è indicato Wrist, se è quella dell'utensile Tool;

Coord. x: Sono indicati i valori delle
Coord. y: coordinate del punto
Coord. z: in cui trovasi il TCP;
Quat. q_1 :
Quat. q_2 : come sopra, per
Quat. q_3 : i quaternioni
Quat. q_4 :

Status: È indicato il codice numerico dell'esito di una operazione. Se è positivo, la funzione del sistema Communication Tool ha avuto esito positivo; se è zero, l'operazione è sospesa, in attesa che venga completata una imprevista richiesta di trasferimento di dati; se è negativo, la funzione ha avuto esito negativo (v. l'elenco dei messaggi nel manuale del Communication Tool);

Emrg. Stop: Se è stato eseguito un arresto di emergenza del Robot, l'indicazione è True, altrimenti è False.

Riferimenti

- 1) E. Bozzi, M. Chimenti
Controllo del movimento del braccio meccanico del robot IRB 2000
Collaborazione Scientifica IEI-Aeritalia.
Nota Interna IEI, B4-05, Febbraio 1991.
- 2) D. Borgioli, M. Chimenti, L. Azzarelli
Un package per l'Image Enhancement sul sistema SVP 2000.
Collaborazione Scientifica IEI-Aeritalia.
Nota Interna IEI, B4-63, Dicembre 1990.
- 3) R. Evangelista, O. Salvetti
La gestione delle look-up table nella stazione SVP200 stand alone.
Collaborazione Scientifica IEI-Aeritalia.
Nota Tecnica B4- , Febbraio 1991
- 4) A. Marchetti, O. Salvetti
Il formato dei dati per il CND
Collaborazione Scientifica IEI-Aeritalia. (In fase di stampa)
- 5) P. Andronico, A. Marchetti, L. Azzarelli, M. Chimenti
Package per l'indagine statistica di immagini monocromatiche.
Collaborazione Scientifica IEI-Aeritalia.
Nota Interna IEI, B4-28, Luglio 1990.
- 6) P. Andronico, R. Guerrini, L. Azzarelli, M. Chimenti
Stereogramma di immagini monocromatiche.
Collaborazione Scientifica IEI-Aeritalia.
Nota Interna IEI, B4-12, Marzo 1990.
- 7) P. Andronico, S. Cerri, E. Fantini, A. Marchetti
Toolkit per interfacce iconografiche di applicativi orientati all'analisi di immagini.
Collaborazione Scientifica IEI-Aeritalia. (In stampa)

PROGRAM 41

UNIT = METRIC

```
$0010 V = 50 MAX = 2000
$0020 RECT COORD
$0030 TCP 0
$0040 FRAME 0
$0050 POS V = 100.0 % REFPOINT OFF #0001
$0060 LET R1 = 0
$0070 LET R2 = 0
$0080 LET R3 = 0
$0090 LET R4 = 10
$0100 LET R5 = 5
$0110 POS V = 100.0 % FINE #0002
$0120 POS V = 100.0 % STORE POSITION 1
$0130 POS V = 100.0 % FINE #0003
$0140 POS V = 100.0 % STORE POSITION 2
$0150 LET R2 = R5
$0190 POS V = 100.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 5.0 Z =
0.0
$0200 POS V = 100.0 % CIRCLE LOCATION 1 OFFSET X = 5.0 Y = 0.0 Z =
0.0
$0210 POS V = 100.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -5.0 Z =
0.0
$0220 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -10.0 Z =
0.0
$0230 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -10.0 Y = 0.0 Z =
0.0
$0240 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 10.0 Z =
0.0
$0250 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 15.0 Z =
0.0
$0260 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 15.0 Y = 0.0 Z =
0.0
$0270 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -15.0 Z =
0.0
$0280 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -20.0 Z =
0.0
$0290 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -20.0 Y = 0.0 Z =
0.0
$0300 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 20.0 Z =
0.0
$0310 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 25.0 Z =
0.0
$0320 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 25.0 Y = 0.0 Z =
0.0
$0330 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -25.0 Z =
0.0
```

File: BOZZ0041.IRB

\$0340 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -30.0 Z =
0.0

\$0350 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -30.0 Y = 0.0 Z =
= 0.0

\$0360 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 30.0 Z =
0.0

\$0370 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 35.0 Z =
0.0

\$0380 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 35.0 Y = 0.0 Z =
= 0.0

\$0390 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -35.0 Z =
0.0

\$0400 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -40.0 Z =
0.0

\$0410 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -40.0 Y = 0.0 Z =
= 0.0

\$0420 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 40.0 Z =
0.0

\$0430 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 45.0 Z =
0.0

\$0440 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 45.0 Y = 0.0 Z =
= 0.0

\$0450 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -45.0 Z =
0.0

\$0460 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -50.0 Z =
0.0

\$0470 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -50.0 Y = 0.0 Z =
= 0.0

\$0480 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 50.0 Z =
0.0

\$0490 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 55.0 Z =
0.0

\$0500 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 55.0 Y = 0.0 Z =
= 0.0

\$0510 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -55.0 Z =
0.0

\$0520 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = -60.0 Z =
0.0

\$0530 POS V = 50.0 % CIRCLE LOCATION 2 OFFSET X = -60.0 Y = 0.0 Z =
= 0.0

\$0540 POS V = 50.0 % FINE LOCATION 2 OFFSET X = 0.0 Y = 60.0 Z =
0.0

\$0550 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = 65.0 Z =
0.0

\$0560 POS V = 50.0 % CIRCLE LOCATION 1 OFFSET X = 65.0 Y = 0.0 Z =
= 0.0

\$0570 POS V = 50.0 % FINE LOCATION 1 OFFSET X = 0.0 Y = -65.0 Z =
0.0

File: BOZZ0036.IRB

PROGRAM 36

```
UNIT = METRIC
$0010 V = 1000 MAX = 2000
$0020 RECT COORD
$0030 TCP 0
$0040 FRAME 0
$0050 POS V = 100.0 % REFPOINT OFF #0001
$0060 POS V = 100.0 % FINE #0002
$0070 CALL PROG 37
$0080 POS V = 100.0 % FINE #0003
$0090 CALL PROG 37
$0100 POS V = 100.0 % FINE #0004
$0110 CALL PROG 37
$0120 POS V = 100.0 % FINE #0005
$0130 CALL PROG 37
$0140 POS V = 100.0 % FINE #0006
$0150 RETURN
```

File: BOZZ0037.IRB

PROGRAM 37

```
UNIT = METRIC
$0010 SET OUTP 1
$0020 PULSE OUTP 8
$0021 SET OUTP 3
$0022 SET OUTP 4
$0030 WAIT UNTIL INP 1 = 1 MAX = 3
$0040 RESET OUTP 1
$0050 RETURN
```

PROGRAM 30

```
UNIT = METRIC
$0010 V = 1000 MAX = 2000
$0020 RECT COORD
$0030 TCP 0
$0040 FRAME 0
$0050 LET R1 = 0
$0060 LET R2 = 0
$0070 LET R3 = 0
$0080 POS V = 100.0 % REFPOINT OFF #0001
$0090 POS V = 100.0 % FINE #0002
$0100 POS V = 100.0 % STORE POSITION 1
$0110 POS V = 100.0 % POSITION 1 OFFSET X = 0.0 Y = -48.6 Z = 6.3

$0120 POS V = 100.0 % POSITION 1 OFFSET X = 80.6 Y = 184.8 Z =
-5.4
$0130 POS V = 100.0 % POSITION 1 OFFSET X = 90.0 Y = 300.0 Z =
6.0
$0140 POS V = 100.0 % FINE POSITION 1 OFFSET SC = 1.0
X = (R1) Y = (R2) Z = (R3)
$0150 LET R1 = 200
$0160 LET R2 = 200
$0170 LET R4 = R1 -194
$0180 LET R3 = R4
$0190 POS V = 100.0 % FINE POSITION 1 OFFSET SC = 1.0
X = (R1) Y = (R2) Z = (R3)
$0200 POS V = 100.0 % FINE POSITION 1 OFFSET X = 200.0 Y = 152.0
Z = 5.0
$0210 POS V = 100.0 % POSITION 1 OFFSET X = 0.0 Y = 48.0 Z = 7.0
$0220 POS V = 100.0 % POSITION 1 OFFSET X = 0.0 Y = 48.0 Z = 15.0

$0222 LET R1 = 0
$0224 LET R2 = 0
$0226 LET R3 = 7
$0230 POS V = 100.0 % POSITION 1 OFFSET SC = 1.0
X = (R1) Y = (R2) Z = (R3)
$0232 POS V = 100.0 % FINE POSITION 1 OFFSET X = 50.0 Y = 50.0 Z
= 7.0
$0233 POS V = 100.0 % POSITION 1 OFFSET X = -45.4 Y = 111.5 Z =
-1.3
$0234 POS V = 100.0 % POSITION 1 OFFSET SC = 1.0
X = (R1) Y = (R2) Z = (R3)
$0240 RETURN
```

File: BOZZ2006.IRB

PROGRAM 2006

UNIT = METRIC
\$0010 V = 1000 MAX = 2000
\$0020 RECT COORD
\$0030 TCP 0
\$0040 FRAME 0
\$0050 POS V = 0.0 % REFPOINT OFF #0001
\$0060 POS V = 100.0 % FINE #0002
\$0070 CALL PROG 2010 REP 45
\$0080 RETURN

File: BOZZ2010.IRB

PROGRAM 2010

UNIT = METRIC
\$0010 POS V = 100.0 % REFPOINT ON #0001
\$0020 CALL PROG 2007 REP 99
\$0030 CALL PROG 2008
\$0040 CALL PROG 2009 REP 99
\$0050 CALL PROG 2008
\$0060 RETURN

PROGRAM 2007

UNIT = METRIC
\$0010 POS V = 100.0 % REFPOINT DN #0001
\$0020 POS V = 100.0 % FINE #0002
\$0030 POS V = 100.0 % FINE #0003
\$0035 CALL PROG 2011
\$0040 RETURN

PROGRAM 2008

UNIT = METRIC
\$0010 POS V = 100.0 % REFPOINT DN #0001
\$0020 POS V = 100.0 % FINE #0002
\$0030 POS V = 100.0 % FINE #0003
\$0035 CALL PROG 2011
\$0040 RETURN

PROGRAM 2009

UNIT = METRIC
\$0010 POS V = 100.0 % REFPOINT DN #0001
\$0020 POS V = 100.0 % FINE #0002
\$0030 POS V = 100.0 % FINE #0003
\$0035 CALL PROG 2011
\$0040 RETURN

PROGRAM 2011
COMMENT invio start portal attesa ok portal;
SET OUTP1
RESET OUTP1
WAIT UNTIL INP1=1 MAX=1
RETURN

PROGRAM 52

```
UNIT = METRIC
$0010 COMMENT QUATER.PIANO XZ ;
$0020 V = 1000 MAX = 2000
$0030 MODRECT COORD
$0035 TCP 0
$0040 TCP 2
$0050 FRAME 0
$0055 POS V = 100.0 % FINE #0001
$0060 POS V = 100.0 % REFPOINT ON #0002
$0070 POS V = 100.0 % FINE #0003
$0080 POS V = 100.0 % FINE #0004
$0090 TCP 2
$0100 POS V = 100.0 % FINE #0005
$0110 POS V = 100.0 % FINE #0006
$0120 POS V = 100.0 % FINE #0007
$0130 POS V = 100.0 % FINE #0008
$0140 POS V = 100.0 % FINE #0009
$0150 POS V = 100.0 % FINE #0010
$0160 POS V = 100.0 % FINE #0011
$0170 POS V = 100.0 % FINE #0012
$0180 POS V = 100.0 % FINE #0013
$0190 POS V = 100.0 % FINE #0014
$0200 POS V = 100.0 % FINE #0015
$0210 POS V = 100.0 % FINE #0016
$0220 POS V = 100.0 % FINE #0017
$0230 POS V = 100.0 % FINE #0018
$0240 POS V = 100.0 % FINE #0019
$0250 TCP 2
$0260 POS V = 100.0 % FINE #0020
$0270 POS V = 100.0 % FINE #0021
$0280 POS V = 100.0 % FINE #0022
$0290 POS V = 100.0 % FINE #0023
$0300 POS V = 100.0 % FINE #0024
$0310 RETURN
```

Name	Source	X mm Q1	Y mm Q2	Z mm Q3	Q4
0013	D	950.0 0.70416	41.8 -0.06158	1086.4 0.70441	-0.06158
0014	D	950.0 0.69635	82.1 -0.12280	1075.5 0.69635	-0.12280
0015	D	950.0 0.68298	120.0 -0.18298	1057.9 0.68298	-0.18298
0016	D	950.0 0.66449	154.3 -0.24182	1033.9 0.66449	-0.24182
0017	D	950.0 0.64087	183.9 -0.29883	1004.3 0.64087	-0.29883
0018	D	950.0 0.61237	207.0 -0.35358	970.0 0.61237	-0.35352
0019	D	950.0 0.57922	225.0 -0.40558	932.1 0.57922	-0.40558
0020	D	950.0 0.70709	100.0 0.00000	1090.0 0.70709	0.00000
0021	D	950.0 0.70709	50.0 0.00000	1040.0 0.70709	0.00000
0022	D	950.0 0.70709	150.0 0.00000	1040.0 0.70709	0.00000
0023	D	950.0 0.70709	150.0 0.00000	1140.0 0.70709	0.00000
0024	D	950.0 0.70709	-150.0 0.00000	1140.0 0.70709	0.00000

Name	Source	X mm 01	Y mm 02	Z mm 03	04
0001	D	950.0 0.70709	-150.0 0.00000	880.4 0.70709	0.00000
0002	D	950.0 0.70709	-150.0 0.00000	880.4 0.70709	0.00000
0003	D	950.0 0.70709	-50.0 0.00000	1040.0 0.70709	0.00000
0004	D	950.0 0.70709	-100.0 0.00000	1100.0 0.70709	0.00000
0005	D	950.0 0.57922	-225.0 0.40558	932.1 0.57922	0.40558
0006	D	950.0 0.61237	-207.9 0.35358	970.0 0.61237	0.35358
0007	D	950.0 0.64087	-183.9 0.29883	1004.3 0.64087	0.29883
0008	D	950.0 0.66449	-154.3 0.24182	1033.9 0.66449	0.24182
0009	D	950.0 0.68298	-120.0 0.18298	1057.9 0.68298	0.18298
0010	D	950.0 0.69635	-82.1 0.12280	1075.5 0.69635	0.12280
0011	D	950.0 0.70416	-41.6 0.06165	1086.4 0.70441	0.06165
0012	D	950.0 0.70709	0.0 0.00000	1090.0 0.70709	0.00000

```
{
    int status;

    read_rob_stat(stat_dat,&status);
    cntrl_status(status,99);

    printf("\n\nProgramma selezionato : %d",stat_dat[0]);
    printf("\nIstruzione selezionata : %d",stat_dat[1]);

    if(stat_dat[2] == 0)
        printf("\nStandby : NO");
    else
        printf("\nStandby : SI");

    if(stat_dat[3] == 0)
        printf("\nOperation : NO");
    else
        printf("\nOperation : SI");

    if(stat_dat[4] == 0)
        printf("\nProgram exec : NO");
    else
        printf("\nProgram exec : SI");

    if(stat_dat[7] == 0)
        printf("\nInterrupt : ENABLED");
    else
        printf("\nInterrupt : DISABLED");

    printf("\nX, Y, Z : %d %d %d",stat_dat[8],stat_dat[9],stat_dat[10]);
    printf("\nQ1, Q2, Q3, Q4 : %d %d %d %d",stat_dat[11],stat_dat[12],stat_dat[13],stat_dat[14]);

    if(stat_dat[29] == 0)
        printf("\nOrientation : WRIST");
    else
        printf("\nOrientation : TOOL");

    printf("\nTCP corrente : %d",stat_dat[30]);

    printf("");
    printf("\n\nPREMERE UN TASTO PER CONTINUARE");
    getch();
    return;
}
```

```
move_dat[0]=0;
move_dat[1]=0;
move_dat[2]=0;
move_dat[3]=0;
move_dat[4]=1;
move_dat[5]=0;
move_dat[6]=5800;
move_dat[7]=8192;
move_dat[8]=7200;
move_dat[9]=1000;
move_dat[10]=12700;
move_dat[11]=16393;
move_dat[12]=0;
move_dat[13]=0;
move_dat[14]=0;
move_dat[15]=0;
move_dat[16]=0;
move_dat[17]=0;
move_dat[18]=0;
move_dat[19]=0;
move_dat[20]=0;
move_dat[21]=0;
move_dat[22]=0;
move_dat[23]=0;
move_dat[24]=0;
move_dat[25]=0;
move_dat[26]=0;
move_dat[27]=0;
move_dat[28]=0;
move_dat[29]=0;
```

```
/****** FRAME *****/
```

```
frame_dat[0]=0;
frame_dat[1]=0;
frame_dat[2]=0;
frame_dat[3]=16393;
frame_dat[4]=0;
frame_dat[5]=0;
frame_dat[6]=0;
```

```
/****** TCP *****/
```

```
tcp_dat[0]=0;
tcp_dat[1]=0;
tcp_dat[2]=0;
tcp_dat[3]=0;
tcp_dat[4]=0;
tcp_dat[5]=0;
return;
}
```

```
/******
```

```
cntrl_status(status, i)
    int status;
    int i;
{
    printf("\nStato : %d", status);
    if(status <= 0 )
    {
        printf("\n\nERRORE stato di ritorno (%d) : %d", i, status);
        exit(-1);
    }
}
return;
}
```

```
/****** leggi stampa stato *****/
```

```
leggi_stampa_stato()
```

```
#include <stdio.h>
#include "c:\robinc\ctsttype.h"
#include "c:\robinc\ctstfunc.h"

tcp_array tcp_dat;
move_array move_dat;
frame_array frame_dat;
stat_array stat_dat;

main ()
{
    int enbl_port;
    int enbl_rob;
    int robot_mode;
    int frame_id;
    int tcp_id;
    int status, i;

    enbl_rob =0;
    enbl_port =0;
    robot_mode=3;
    frame_id=1;
    tcp_id=1;

    init_array();

    /* leggi_stampa_stato(); */

    printf("\n\nPremere un tasto per eseguire -->set_robot_on<--");
    getch();

    i=1;
    set_robot_on(enbl_rob,enbl_port,&status);
    cntrl_status(status,i);

    leggi_stampa_stato();

    /*
    printf("\n\nPremere un tasto per eseguire -->write_frame<--");
    getch();

    i=8;
    write_frame(frame_id,frame_dat,&status);
    cntrl_status(status,i);

    leggi_stampa_stato();

    */
    printf("\n\nPremere un tasto per eseguire -->write_tcp<--");
    getch();

    i=9;
    write_tcp(tcp_id,tcp_dat,&status);
    cntrl_status(status,i);

    leggi_stampa_stato();

    printf("\n\nPremere un tasto per eseguire -->cmd_set_mode<--");
    getch();

    i=2;
    cmd_set_mode(robot_mode,&status);
    cntrl_status(status,i);

    leggi_stampa_stato();

    /* START POINT */
    printf("");
    printf("\n\nATTENDERE CHE SI SPENGA LA SPIA DI SINCRONIZZAZIONE POI");
    printf("\nPremere un tasto per eseguire -->start_point<--");
```

```
getch();
```

```
i=3;
move_dat[2]=2;
move_dat[8]=7600;
move_dat[9]=0;
move_dat[10]=12700;
cmd_move(move_dat,&status);
cntrl_status(status,i);
```

```
leggi_stampa_stato();
```

```
/* MOVE ABS POINT */
printf("\n\nPremere un tasto per eseguire -->move<--");
getch();
```

```
i=4;
move_dat[2]=0;
move_dat[8]=7200;
move_dat[9]=1000;
move_dat[10]=12700;
cmd_move(move_dat,&status);
cntrl_status(status,i);
```

```
leggi_stampa_stato();
```

```
printf("\n\nPremere un tasto per eseguire -->move<--");
getch();
```

```
i=5;
move_dat[2]=0;
move_dat[8]=8000;
move_dat[9]=-1000;
move_dat[10]=10000;
cmd_move(move_dat,&status);
cntrl_status(status,i);
```

```
leggi_stampa_stato();
```

```
/* END POINT */
printf("\n\nPremere un tasto per eseguire -->end_point<--");
getch();
```

```
i=6;
move_dat[2]=3;
cmd_move(move_dat,&status);
cntrl_status(status,i);
```

```
leggi_stampa_stato();
```

```
/* ROBOT OFF */
printf("\n\nPremere un tasto per eseguire -->set_robot_off<--");
getch();
```

```
i=7;
set_robot_off(enbl_rob,&status);
cntrl_status(status,i);
```

```
printf("\n\n-----> T U T T O O. K. <-----\n\n");
}
```

```
/******
```

```
init_array()
{
```

```
/****** MOVE *****/
```